LT4CPR 2026

First Workshop on Language Technologies for Crisis Preparedness and Response (LT4CPR)

Proceedings of the Workshop

November 9, 2026

The LT4CPR organizers gratefully acknowledge the support from the following sponsors.

Awards CNS-2346334 and CNS-2346335



©2026 Association for Computational Linguistics

Order copies of this and other ACL proceedings from:

Association for Computational Linguistics (ACL)
317 Sidney Baker St. S
Suite 400 - 134
Kerrville, TX 78028
USA
Tel: +1-855-225-1962
acl@aclweb.org

ISBN 979-8-89176-450-7

Preface

We are delighted to welcome you to the First Workshop on Language Technologies for Crisis Preparedness and Response (LT4CPR), held on November 9, 2026, in Hengqin, China, in conjunction with AACL-IJCNLP 2026!

During a crisis, timely information is useful only if people can access, understand, and act on it. Warnings, requests for assistance, and accounts of changing conditions travel across languages, communities, and communication channels. Responders must make sense of fragmented and sometimes contradictory reports, while affected communities may face language barriers, limited connectivity, or information that does not reflect their circumstances. These challenges make language and communication central to both preparing for crises and responding to them.

Language technologies offer ways to translate warnings, organize incoming reports, identify urgent needs, and support communication at scale. Yet strong performance on a benchmark does not by itself establish that a system will be useful in a crisis. Errors, missing context, and unsupported claims can undermine the information on which people depend. Research must therefore consider whose languages and needs are represented, how uncertainty is communicated, how systems can be evaluated with the practitioners and communities they are meant to serve, and whether such systems can realistically be deployed in crisis settings. Connecting technical advances with these practical questions is the central motivation for LT4CPR.

Building on discussions at the birds-of-a-feather gatherings at COLING 2025 and ACL 2025, this first edition of LT4CPR brings together researchers in speech and language technologies with crisis-management practitioners, humanitarian organizations, and other stakeholders. The workshop spans multilingual and multimodal communication, information extraction, summarization, translation, and the evaluation of language technologies for crisis settings. We hope it will establish a forum for sustained exchange between methodological research and operational experience, with attention to preparedness as well as response.

The workshop includes 18 accepted contributions: eight archival research papers, seven archival shared-task papers, and three non-archival contributions. These proceedings collect the 15 archival papers. The shared- task papers, including the overview, examine the generation of situation reports from social media messages. The non-archival contributions broaden the workshop discussion while remaining outside the formal proceedings.

We thank all authors and presenters for contributing their work, and the program committee and reviewers for their time, expertise, and thoughtful feedback. We also thank our keynote speaker, Federico Marco Federici, for contributing his perspective on intercultural crisis communication. Finally, we are grateful to the AACL-IJCNLP 2026 workshop and publication chairs and conference organizers for their guidance and administrative support in bringing this first edition together.

This workshop is supported in part by the National Science Foundation under award CNS-2346334 to George Mason University and award CNS-2346335 to the University of Washington.

Antonios Anastasopoulos, Fei Xia, Will Lewis
Belu Ticona, and Haotian Zhu
LT4CPR 2026 Organizers

Organizing Committee

Workshop Co-Chairs

Antonios Anastasopoulos, George Mason University
Fei Xia, University of Washington
William D. Lewis, University of Washington

Workshop Organizers

Belu Ticona, George Mason University
Haotian Zhu, University of Washington

Program Committee

Program Committee

Angelina Aquino, Charles Darwin University
Vipasha Bansal, UW
Patrick Cadwell, Dublin City University
Federico Marco Federici, University College London
Amanda Hughes, Brigham Young University
Marina Kogan, University of Utah
Gina-Anne Levow, University of Washington
Chi-kiu Lo, National Research Council Canada
Helena Moniz, University of Lisbon
Sharon O'Brien, Dublin City University
Steven Richardson, Brigham Young University
Arif Sadri, University of Oklahoma
Giuseppe Samo, Beijing Language and Culture University/University of Geneva/ IDIAP
Yang Song, University of Science and Technology of China
Maria Carmen Staiano, Università degli Studi di Macerata / Dublin
Alp Öktem, CLEAR Global
Philipp Koehn, Johns Hopkins University

Keynote

Multilingual crisis communication: the human-centred pipeline for timely communication

Federico Marco Federici
University College London



Bio: Professor Federico M. Federici is Director of the Centre for Translation Studies at University College London and a Professor of Intercultural Crisis Communication at the university. Dr. Federici has published extensively on language translation in crises, with his current research centered on multilingual communication in crises, in particular, translated communication in medical and healthcare contexts. He was a Principal Investigator on the INTERACT project, a large multi-year EU-funded project, partnering with colleagues at Dublin City University and other European institutions, with a research focus on crisis translation. Dr. Federici's work also extends beyond crises, looking at the use of translation in doctor-patient interactions, in particular, in scenarios where patients have limited English proficiency, and the impact that translation (both human-mediated and automated) has on patient care and outcomes.

Table of Contents

Towards a Benchmark for Fire Safety Code Interpretation in the Hong Kong AEC Industry
Le Qiu, Zhuan Anne Zhang, Hok Ming Adeline Chan and Emmanuele Chersoni 1

CrisiLangs: A Multilingual Resource for Crisis Warning Generation
Giacomo Gonella, Stefano Menini and Marco Guerini 20

SIREN: Situation-aware Index for Response and Emergency ActioNability in Crisis Translation
Maria Carmen Staiano, Lifeng Han, Johanna Monti and Francesca Chiusaroli 32

Communicating with the Entire Population: Machine Translation of Emergency Alerts and Warnings
Jen Wilson, Vipasha Bansal and William D. Lewis 46

Looking Complete Is Not Being Grounded: Evaluating LLM Sitreps Against Human-Authored Situation Reports
Sachin Kumar 65

Is Machine Translation Ready for the Next Crisis or Pandemic?
Vipasha Bansal, William D. Lewis, Nicole Mihaylova, Elizabeth Brown and Benjamin Pong . . 78

Cross-Modal Inconsistency in Multimodal LLMs during Crisis Communication
Anuridhi Gupta and Hemant Purohit 115

Critical evaluation of Knowledge Graph-powered Question Answering Systems to Enable Responsible AI for Disaster Management
Gayatri Ramchandra Vaidya, Yasas Senarath and Hemant Purohit 126

Storylines, Not Snapshots: Error-Driven Prompting for Crisis Situation Reports at LT4CPR 2026
Youcef Daoud 134

Structured Drafting and Evidence-Grounded Refinement for the LT4CPR 2026 Shared Task
Wataru Sakata and Wataru Hirota 141

Extractive Corroboration Pipeline for Crisis Situation Report Generation from Social Media
Sreeja Guha Majumdar 156

Amrood at LT4CPR 2026: A Unified Generate-and-Judge LLM Pipeline for Automatic Situation Report Generation
Muhammad Ahsan Atiq, Mohib Ahmad Butt and Nusrat Shaheen 166

R01G-SR: Provenance-Preserving Selection over Complementary Evidence Views for Crisis Situation Report Generation
Yuqiu Gao, Yuanhe Tian, Xiaofei Chen, Zhentao Yang and Yan Song 177

Frozen-Response Recovery for Evidence-Linked Crisis Situation Reports
Quan Zheng, Ming Wang, Yuanhe Tian and Yan Song 188

Overview of the LT4CPR 2026 Shared Task: Generating Situation Reports from Social Media Messages
Fei Xia, Haotian Zhu, Yun Woo Lee, Eitan Grenev, Allison K Burton, Yingzhe Fan, William D. Lewis, Antonios Anastasopoulos and Belu Ticona 203

Towards a Benchmark for Fire Safety Code Interpretation in the Hong Kong AEC Industry

Le Qiu[1], Zhuan Anne Zhang[2,3], Hok Ming Adeline Chan[2,3], Emmanuele Chersoni[1,4]
[1]Language Science and Technology, Hong Hong Polytechnic University, Hong Kong
[2]AAL INNOVATION Limited, Hong Kong
[3]Journalism and Communication, Hong Kong Shue Yan University, Hong Kong
[4]Artificial Intelligence and the Humanities, Hong Hong Polytechnic University, Hong Kong
lani.qiu@connect.polyu.hk, anne@aalinnovationai.com
adeline@aalinnovationai.com, emmanuele.chersoni@polyu.edu.hk

Abstract

Large Language Models (LLMs) are increasingly explored in the AEC (Architecture, Engineering, and Construction) domain, yet their reliability in regulatory tasks remains uncertain.

In this study, we focused on *Hong Kong fire safety codes*: together with domain experts, we create a localized dataset of multiple-choice (MCQs) and open-ended questions (OEQs) to evaluate 12 widely used LLMs, including both proprietary and open-source models.

Our findings show that proprietary models may achieve 80–90% accuracy on MCQs, while open-source models generally lag behind. On OEQ, humans rated the answers at 50-70% accuracy, suggesting that LLMs in specialized compliance tasks still have large margins of improvement.

1 Introduction

The AEC industry is one of the many domains in which Generative AI and LLMs are rapidly gaining momentum: the most tech-savvy architects have started to adopt LLM-based tools in their everyday design practice, while clients increasingly see the usage of AI as a guarantee for their return of investment (Del Campo and Leach, 2022), as well as a valuable tool for workflows such as cost prediction and quality control (Kampelopoulos et al., 2025). An open question is whether current state-of-the-art LLMs can already be reliable support tools for specialized and safety-critical AEC contexts.

In Tai Po, in the New Territories of the Hong Kong Special Administrative Region, a major and tragic fire incident has recently brought renewed public attention to the importance of fire safety in urban environments[1]. For us, as NLP practitioners, it should be a natural question to ask, whether modern technologies can help us to prevent such type of incident from occurring again in the future.

Motivated by this context, we take fire safety as a critical lens to evaluate the current capabilities of state-of-the-art AI models within the AEC sector. Although prior research has examined LLMs in AEC-related tasks — such as code interpretation and design assistance (e.g. Guo et al., 2025; Liang et al., 2026) — systematic evaluations specifically targeting fire safety remain scarce. Fire safety represents a high-stakes AEC subdomain that relies heavily on precise interpretation of regulatory codes, making it an ideal testbed for assessing whether AI can reliably support professional decision-making in AEC. Building upon this motivation, we have collaborated with domain experts and constructed a Q&A (Question Answering) dataset relevant to local fire safety practice in Hong Kong. Using this dataset, we have evaluated both proprietary LLMs and open-source models to understand their strengths, limitations, and failure modes in this high-stakes domain.

In summary, our work primarily make two contributions: i) We present the first dataset grounded in the Hong Kong fire safety regulatory context, offering a structured benchmark for evaluating AI systems in one of the most safety-critical AEC subdomains; ii) we provide a systematic evaluation of LLMs on code compliance checking tasks in Hong Kong fire safety, analyzing their capabilities, reliability gaps and resource consumptions when interpreting real-world regulatory codes.[2]

2 Related Work

2.1 LLMs in the AEC Industry

AEC workflows are inherently document-intensive, relying on large volumes of unstructured data such as design briefs and safety regulations. This makes the industry particularly receptive to automation technologies. Evidence shows that recent LLMs

[1] https://en.wikipedia.org/wiki/Wang_Fuk_Court_fire

[2] Our dataset and code are available at https://github.com/Laniqiu/HK-FSCode-LLM-Eval.

Proceedings of the First Workshop on Language Technologies for Crisis Preparedness and Response, pages 1–19
November 9, 2026 ©2026 Association for Computational Linguistics

have been swiftly integrated into various AEC tasks, ranging from basic document processing — such as information extraction and summarization — to more advanced applications like regulatory compliance inspection (Kampelopoulos et al., 2025). Commercial general-purpose LLMs—such as ChatGPT, Gemini, and Claude — are among the most widely adopted, due to their strong reasoning capabilities and ready-to-use features.

However, a more widespread application of LLMs in the AEC industry faces several obstacles. The most prominent issue is domain knowledge: AEC tasks often require extensive expertise spanning multiple disciplines, including architecture, structural engineering, mechanical and electrical engineering (MEP), and project management (Ghimire et al., 2024), and even for general-purpose LLMs it is difficult to reliably cover such specialized and multidisciplinary knowledge. Another widely discussed challenge are hallucinations (Huang et al., 2023), where AI systems generate plausible-sounding but incorrect information. In safety-critical AEC subdomains—such as fire safety engineering, structural safety assessment, or code compliance checking—hallucinations pose significant risks (Xiong et al., 2026), as incorrect interpretations may lead to unsafe decisions.

2.2 LLMs for Compliance Checking

In response to these limitations, researchers have developed domain-specific LLMs tailored for AEC, mainly for compliance checking (Zheng et al., 2023, 2025). For example, CivilGPT and SiKong [3] are two models developed for the Chinese AEC. CivilGPT specializes in civil engineering knowledge, while SiKong is a fine-tuned model based on LLaMA and Alpaca, designed to support architecture-related question answering tasks. Both focus on the knowledge systems specific to Mainland China.

Notably, AEC regulations exhibit strong regional specificity, as codes and regulations vary across jurisdictions: for example, a system like SiKong may perform well on the Chinese regulatory frameworks, but its applicability is limited when dealing with the codes of other regions. While this issue implies the need for localization in AEC-oriented LLMs, AEC-specific datasets remain scarce and qualified human experts are extremely difficult to recruit for dataset construction.

[3]CivilGPT: https://civilgpt.tongji.edu.cn/; SiKong: https://github.com/SikongSphere/sikong

Compliance checking on fire safety codes in NLP is not a common research topic, although some benchmarks exist, such as the CODE-ACCORD corpus, containing regulatory codes from the United Kingdom and Finland and also including fire safety ones (Hettiarachchi et al., 2026); and SearchFireSafety, a dataset for statute-centric legal question-answering centered on South Korean fire-safety regulations (Chae et al., 2026).

To our knowledge, no specific NLP research has been conducted dedicated to Hong Kong fire safety codes. Since developing a dataset of instructions to fine-tune a model specifically for the Hong Kong fire safety regulation is not practical due to the costs and the limited applicability, we focus our efforts on constructing an evaluation-only dataset that can be used to test the performance of state-of-the-art LLMs, to see if their exposure to massive scales of data and advanced reasoning capabilities can compensate for the lack of localization.

3 Dataset Creation

We constructed a Q&A dataset sourced from *Code of Practice for Fire Safety in Buildings 2011* (FS Code) [4], published by the Hong Kong Buildings Department. The FS Code is the central regulatory document governing fire safety provisions in Hong Kong and serves as the official reference for both professional practice and local qualification examinations, making it a reliable foundation for assessing LLM understanding of domain-specific fire safety knowledge. Our samples comprehensively cover both core components of the code: an **architectural** component that includes means of escape, means of access, and passive fire protection measures (e.g., fire-resisting construction, fire barriers, etc.), and a **fire engineering** component that specifies the performance and design of active fire protection systems and fire service installations (e.g., automatic sprinklers, fire detection and alarm systems, etc.). Following the style of the existing HKIA/ ARB Professional Assessment (PA)[5], we designed two major categories of ques-

[4]While the FS Code is available in both English and Cantonese, we used the 2024 English version to align with local qualification examinations, which are conducted in English. Source: https://www.bd.gov.hk/doc/en/resources/codes-and-references/code-and-design-manuals/fs_code2011.pdf

[5]The PA is the official professional qualification examination administered by The Hong Kong Institute of Architects (HKIA) and the Architects Registration Board (ARB). It is the primary pathway to obtain professional membership in HKIA

tions: multiple-choice questions and open-ended questions.

We employed a hybrid AI-human workflow for Q&A generation. We first used the recent, state-of-the-art GPT-5 (Singh et al., 2025) model to generate candidate questions — either by prompting the model with specific requirements or by supplying relevant excerpts from the FS Code as source material. Near-identical or overlapping questions were manually removed. Following this stage, each remaining question underwent comprehensive human verification, including review and correction of the question, answer and references. More specifically, the expert review was conducted by two architects and one fire engineer, all of whom are Hong Kong-registered, experienced, and actively practicing professionals, with responsibilities divided according to domain expertise: specialists in fire safety regulations examined the architectural components, while professionals with relevant technical backgrounds reviewed the engineering components. Each question would be reviewed by two experts and retained only if both consider it valid; items on which there was disagreement were discarded from the dataset.

The final version contains a total of 500 Q&A pairs, comprising 400 MCQs and 100 OEQs. Examples are provided in Appendix Table 4.

4 Evaluation Settings

4.1 Models

We evaluated 12 distinct LLMs, as outlined in Table 1: proprietary models (top) include Claude Opus (Anthropic, 2024), DeepSeek (DeepSeek-AI, 2024), Gemini 3 (DeepMind, 2025), GPT-4 (Hurst et al., 2024), Grok-4 (xAI, 2024) and Qwen-Plus (Qwen Team, 2026); open-sourced models (bottom) include Qwen2.5-7B (Qwen Team, 2024; Yang et al., 2024), Llama3-8B, Mistral-7B, DeepSeek-7B, Llama3-70B and Qwen3-80B (Qwen Team, 2025; Yang et al., 2025). For DeepSeek and Gemini 3, both *thinking* and *non-thinking* modes are evaluated and each is counted as a separate evaluation, bringing the total to 14.

The selection was based on several considerations. First, it is impractical to include a large amount of available LLMs, given the time and financial constraints. Still, we aimed to cover the popular and representative LLMs that the AEC practitioners are likely to use in real-world applications. Second, we intentionally included smaller-scale models to examine whether model size affects performance. Third, we included both *thinking* and *non-thinking* models for DeepSeek and Gemini 3, trying to investigate whether explicit reasoning-mode capabilities lead to measurable differences.

Proprietary models are accessed via APIs and open-source models are run through Hugging Face Transformers. We use default inference parameters for all models, except that we set the maximum number of newly generated tokens to 512 to ensure comparable answer lengths.[6]

4.2 Metrics

For MCQs, we use **accuracy** as a standard metric, i.e. the proportion of questions for which the model selects the correct option. For OEQs, which involve free-form text, we use both automatic and human evaluation.

Generally, automatic evaluation operates by quantifying the textual similarity between the model-generated output and the reference answer. We employ standard metrics widely used in natural language generation tasks, including BLEU (Papineni et al., 2002), METEOR (Banerjee and Lavie, 2005), ROUGE-L (Lin, 2004) and BERTScore (Zhang et al., 2020). These metrics capture different aspects of similarity: BLEU emphasizes n-gram precision, METEOR incorporates synonym matching, ROUGE-L focuses on longest-common subsequence recall, and BERTScore measures semantic similarity using contextual embeddings.[8]

For human evaluation, we designed two rating dimensions that potentially mirror criteria commonly used in professional examinations: **accuracy** and **clarity**. Accuracy assess whether the response appropriately addresses the question and adheres to the facts; clarity measures the linguistic quality of the response — whether it is presented in a clear,

and to become registered architects in Hong Kong.

[6] This constraint only applies to the final answer returned to the user, as the internal reasoning tokens in the thinking mode are not counted in the generation limit. Note that the specific parameter names may vary across providers.

[7] Prices are quoted per 1 million tokens in USD and provided for reference only (actual costs may vary depending on usage time, mode, cache status and membership tier); The thinking and non-thinking modes for DeepSeek and Gemini 3 are implemented within the same model architecture, with the mode selection controlled by API parameters.

[8] Scores are computed using *pycocoevalcap* for BLEU (n=4), METEOR and ROUGE; and *bert_score* for BERTScore (F1 scores from RoBERTa-large are reported) .

Table 1: A summary of model identifiers and pricing[7]

	Identifier	In/ Out Price
Claude Opus	claude-4.5-opus	5/ 25
DeepSeek	deepseek-chat	0.007∼ 1.32 / 0.66∼ 3.96
Gemini 3	gemini-3-flash-preview	0.05 ∼ 1 / 3
GPT-4	gpt-4.1	0.5 ∼ 2 / 8
Grok-4	grok-beta	0.2 / 0.5
Qwen Plus	qwen-plus	0.4 ∼ 2 / 1.6 ∼ 6
Qwen2.5-7B	Qwen/Qwen2.5-7B-Instruct	N.A.
Llama3-8B	meta-llama/Llama-3.1-8B-Instruct	N.A.
Mistral-7B	mistralai/Mistral-7B-Instruct-v0.3	N.A.
DeepSeek-7B	deepseek-ai/deepseek-llm-7b-chat	N.A.
Llama3-70B	meta-llama/llama-3.3-70b-instruct	N.A.
Qwen3-80B	Qwen/Qwen3-Next-80B-A3B-Instruct	N.A.

Table 2: Human evaluation rubric: dimensions, descriptors and scoring ranges.

Accuracy	
Fully accurate	The response is correct, complete, and aligned with the Hong Kong Fire Safety Code (100%).
Minor inaccuracy	The response is generally correct but contains small inaccuracies or omissions that do not fundamentally affect compliance (90%).
Major issues	The response contains substantial problems. This level is further divided into four sub-categories:
	Irrelevant but real clause cited (80%). Omission of critical information (60%). Hallucination or fabricated content (30%) Fundamental misunderstanding of the regulation (0%).
Clarity	
Clear and professional	Fluent, logical language with precise domain terminology (100%).
Minor clarity issues	Slightly awkward phrasing or redundancy, but still comprehensible (90%).
Inaccurate or inconsistent terminology	Uses incorrect, imprecise, or non-standard Hong Kong fire safety terms (80%).
Logical gaps or weak reasoning	Disconnected arguments, logical jumps, or unsupported conclusions (50%).
Insufficient explanation / justification	Conclusions lack specific clause references, rationale, or step-by-step reasoning (50%).
Poor readability / disorganized structure	Difficult to follow due to lack of structure, paragraphing, or logical flow (0%).

fluent, coherent and professional manner. Each dimension was defined through multiple descriptors that specify different aspects of performance, as shown in Table 2. Each descriptor carried a fixed penalty level. The selected descriptor determined the final score for that dimension. For example, a response classified as fully accurate would receive the full score (100%) for accuracy, whereas a response categorized as having major issues might incur the maximum penalty and end up with a score of zero (0%).

Due to time constraints and the high cost associated with human assessment, we did not evaluate OEQ outputs for all models. Instead, we selected a subset of five models that performed strongly in the automatic evaluation stage and conducted human evaluation on their OEQ outputs. By this design, for each question, we have 5 Q&A pairs (one for each of the 5 evaluated models), and each Q&A pair would be independently and randomly evaluated by at least two raters. Rather than assigning numerical scores, raters are invited to classify each model response into the descriptors that best characterized its accuracy and clarity, without knowing the model identity. The judgment should be based on their professional knowledge and the pro-

vided reference answers. All raters are AEC professionals registered and practiced in Hong Kong. Architectural-related questions are evaluated by architects, and engineering-related questions were rated by fire engineers.

4.3 Prompts

The evaluation prompts follow a fixed format as shown in Table 5. The prompts explicitly indicate that the applicable jurisdiction is Hong Kong (Table 5a). For MCQs, however, we prepare a second set of prompts that does not specify the jurisdiction (Table 5b). By this design, we intend to examine whether the absence of regional awareness — likely in real expert-LLM interactions — affects model performance.

5 Results

The scores are presented in Table 3, along with the average token cost (including both input and output) and the processing time (in seconds).

For the human evaluation of OEQs, we reported their average ratings in Table 3d. As noted earlier, we evaluated the outputs from five models: four proprietary models and one open-source model — Gemini 3 (Thinking and Non-Thinking variants), GPT-4, Claude Opus, and Qwen2.5-7B. The selection was based on multiple considerations, including their automatic evaluation scores on both MCQs and OEQs, resource consumption, and availability in Hong Kong, among other practical factors. Qwen2.5 was included to provide a term of comparison of what can be obtained with a smaller-size, open model.

Overall, the proprietary models outperform smaller open-source models, which is expected given their substantial differences in parameter counts.[9] The bigger open-source models (i.e. Llama3-70B and Qwen3-80B) are the closest to the level of performance of proprietary models, although they generally still lag behind in all metrics. Interestingly, for most automatic generation metrics the larger open models do not improve over the scores of their smaller versions.

On MCQs (see Table 3a and 3b), the best scores were achieved by Grok-4 with prompts specifying the Hong Kong regulatory context (w/ HK) and Gemini 3 without such specification (w/o HK), and the lowest by DeepSeek-7B under both prompting conditions. The top proprietary LLMs — Gemini 3 and Claude Opus, etc. — have been primarily trained on English corpora. Grok-4's real-time web searching may have further contributed to its top ranking under the w/ HK context, while simultaneously leading to substantial token consumption. In contrast, DeepSeek and Qwen Plus, adapted to Chinese context, were less impressive. This might be explained by the differences in the language distribution of the training data. Our dataset was constructed based on common Hong Kong industry practices, using the English version of the FS Code as a reference. Although both Chinese and English are official languages in Hong Kong, professional examinations (e.g. the PA) are conducted in English, which likely favors models with stronger English training coverage. The results between two prompting conditions aligned with commonsense expectations — specifying the jurisdiction could in general improve accuracy. When jurisdiction was not specified in the prompts, the models might produce longer responses with higher token usage and increased latency, probably in an attempt to reconcile multiple legal systems.

In contrast, the OEQ results showed a narrower gap between model types across metrics. BLEU scores clustered around 2-3, METEOR and ROUGE mostly fell in the 10-20 range — all relatively low. This is expected, as these metrics emphasize lexical overlap. For OEQs, valid responses may diverge substantially from the reference, leading to lower overlap. Meanwhile, BERTScore values of semantic similarity with the reference were consistently high, which, however, does not help discriminating between models.

When running the human evaluation, we had to make sure to analyze items for which the annotators were able to reach an agreement, avoiding questions for which even experts gave polarized answers. This led to the exclusion of 32 out of 100, on which the experts showed large disagreements. This filtering step left us with a subset of 68 questions, each one with 5 Q&A pairs, for a total of 340 pairs. In turn, each of them received two ratings from the experts.

The human evaluation revealed narrow accuracy scores around 57 to 65 and clarity scores around 62 to 78. For the scores assigned by humans, we computed the intraclass correlation coefficient (Koo and

[9] Most proprietary LLMs do not disclose their parameter size, but it is generally assumed that they are very large, often in the hundreds of billions or even trillions.

[10] Under this prompting condition, 5 invalid outputs are identified. Values in parentheses are computed on the valid instances

Table 3: A summary of evaluation results. The (*) marker indicates the model operates in a thinking mode, otherwise non-thinking mode by default, time cost is in seconds. The top scores are underlined.

	Accuracy	Time Cost	Token Cost
Qwen Plus	79.5	6	372
Grok-4	90	12	2728
GPT-4	80.5	3	265
Gemini 3	88.5	2	257
Gemini 3*	89.25	17	2254
DeepSeek	80.75	7	266
DeepSeek*	83.5	68	1718
Claude Opus	86	6	2562
Mistral-7B	69.25	N.A.	N.A.
Qwen2.5-7B	75.5	N.A.	N.A.
Llama3-8B	77	N.A.	N.A.
DeepSeek-7B	66	N.A.	N.A.
Llama3-70B	79	6	274
Qwen3-80B	79	1	292

(a) Accuracy for MCQs (w/ HK)

	Accuracy	Time Cost	Token Cost
Qwen Plus	81.75	5	366
Grok-4	83.75	15	2881
GPT-4	78.75	2	2454
Gemini 3	85.75	33	2221
Gemini 3*	86	24	3082
DeepSeek	80.75	15	334
DeepSeek*	81	76	1770
Claude Opus	81.25	6	366
Mistral-7B	67.25	N.A.	N.A.
Qwen2.5-7B	73	N.A.	N.A.
Llama3-8B	73	N.A.	N.A.
DeepSeek-7B	60.75	N.A.	N.A.
Llama3-70B	77.75 (78.73)[10]	8	N.A.
Qwen3-80B	78.25	2	N.A.

(b) Accuracy for MCQs (w/o HK)

	BERTScore	BLEU	METEOR	ROUGE	Time Cost	Token Cost
Qwen Plus	84.6	3.42	16.23	15.35	6	356
Grok-4	84.41	3.19	19.78	15.97	16	3280
GPT-4	85.8	4.66	17.1	17.74	4	298
Gemini 3	85.41	3.43	14.36	16.8	2	281
Gemini 3*	85.55	4.03	15.16	17.53	15	1346
DeepSeek	85.58	3.64	13.75	16.33	9	270
DeepSeek*	85.56	4.34	15.6	16.4	56	1375
Claude Opus	85	3.14	19.25	15.86	10	2472
Mistral-7B	85.2	3.97	15.08	16.15	N.A.	N.A.
Qwen2.5-7B	85.63	3.21	12.93	16.52	N.A.	N.A.
Llama3-8B	84.9	3.68	14.23	16.69	N.A.	N.A.
DeepSeek-7B	84.91	2.76	10.89	15.63	N.A.	N.A.
Llama3-70B	84.46	0.73	7.32	12.41	8	N.A.
Qwen3-80B	84.90	1.63	9.55	13.45	2	N.A.

(c) Automatic evaluation results for OEQs

	Accuracy	Clarity
Gemini 3	61.88	69.13
Gemini 3*	64.11	73.76
GPT-4	57.61	70.22
Claude Opus	65.89	79.43
Qwen2.5-7B	57.36	67.21

(d) Human ratings for OEQs

Accuracy	0.54
Clarity	0.54

(e) ICC rater agreement (6 classes)

Li, 2016), obtaining values around 0.54 for both dimensions (see also Table 3e), which indicates a moderate level of agreement. Considering that both accuracy and clarity are rated on ordinal scales of six values, we deemed it as a sufficiently reliable agreement between our experts.

Regarding the reasoning modes, enabling thinking did not necessarily improve performance. For

instance, DeepSeek* (thinking) performed slightly worse than DeepSeek (non-thinking) on OEQs. In other cases, the differences were less pronounced and almost negligible. Yet the resource consumption of the thinking mode was significantly higher: on MCQs, the token usage and response time of DeepSeek* (thinking) were more than five times of DeepSeek (non-thinking). Under the w/ HK context, Gemini 3 (non-thinking) ranked third with an accuracy of 88.5%, with the lowest cost in terms of both time and token usage. Grok-4, while achieving the highest accuracy, had consumed the largest amount of tokens.

To assess the validity of automatic evaluation metrics, we measured the Spearman correlation between each metric and the human expert scores. Unfortunately, we obtained a negative result: none of the automated metrics showed a significant correlation with expert-rated accuracy, while moderate correlations were observed between clarity and METEOR ($p = 0.38$), ROUGE ($p = 0.36$) and BERTScore ($p = 0.34$).

Compared to industry-standard expectations and requirements, these findings suggest that current general-purpose LLMs cannot yet serve as reliable final recommendations for fire safety design and engineering scenario analysis in the Hong Kong region. Their outputs should only be considered as reference information, subject to supervision and, whenever possible, validation by human experts. It is interesting, in this sense, that experts generally rate LLMs much higher for clarity than for accuracy: in generating an answer, LLMs might be very clear and professional-sounding, but the expressed factual knowledge is not necessarily correct. Moreover, as our correlation analysis shows, traditional automated metrics are not reliable when it comes to the assessment of factual accuracy.

Finally, beyond the observations above, one further implication emerged from our human evaluation. Although there remain many areas for improvement, the evaluation revealed practical challenges. First, interpretations of regulatory provisions can vary — even the official Buildings Department may provide different interpretations — so expert scoring may remain highly subjective despite the provision of references and instructions. Second, experts' attitudes toward AI outputs differed: some raters, once aware that the text was AI-generated (even without knowing which model), adopted a more skeptical stance, which often resulted in lower scores. Future initiatives — whether academic projects or industry application — should account for the inherent subjectivity of human experts and their attitudes toward AI outputs.

6 Conclusion

This work presents the first systematic evaluation of 12 LLMs, both proprietary and open-source, on their ability to interpret the Hong Kong fire safety code. To support this study, we collaborated with fire safety experts in Hong Kong to construct a Q&A dataset comprising both MCQs and OEQs, emulating the structure of official professional examination.

Overall, proprietary models outperform open-source ones. On MCQs, proprietary models achieved about 80–90% accuracy on MCQs, while open-source models performed substantially lower. On OEQs, model performance remains limited. Ratings from human experts on OEQs reveal that accuracy of model outputs remain only around 50–70%, with most responses falling into the major issues category rather than reaching fully accurate or minor inaccuracy levels. This indicates that current LLMs cannot yet reliably capture the precision required for regulatory compliance. It is also found that enabling thinking did not yield significant gains on either setting, but incurred high resource consumption, underscoring the need to balance efficiency with effectiveness. Moreover, enabling thinking modes did not yield significant accuracy gains in either MCQs or OEQs, but increased resource consumption. Finally, all automated metrics were not correlated with accuracy and only moderately correlated with clarity, thus confirming the challenges in evaluation when the feedback of human experts is not easily available.

While constrained by dataset scarcity and limited expert availability, this study contributes a localized benchmark for evaluating LLMs against the Hong Kong fire safety code.

Although our focus is on fire safety, the underlying methodology is transferable to other AEC subdomains. For future work, we will prioritize improving evaluation frameworks — especially human evaluation, and expand the benchmark beyond fire safety to include broader regulatory tasks in AEC.

Limitations

We acknowledged the limitations of the current work are primarily: First, since fire safety experts

are difficult to recruit and our dataset requires manual review, screening and revision to ensure quality and accuracy, we are aware of the fact that the dataset size is small. This issue, together with the specific focus on the Hong Kong regulation, is an important limitation to the the generalizability of the findings. The same challenge of expert recruitment also limited the coverage of human evaluation.

Ethics Statement

Potential risk: The potential risk of this work lies in misinterpretation of compliance-checking outputs, which could lead to incorrect conclusions about safety compliance. To mitigate this, we emphasize that LLM predictions are advisory only and must be validated by certified fire safety professionals.

License: All artifacts (LLMs, data, and software packages) were used in accordance with their intended use and licensing terms in our dataset construction and evaluation experiments. Proprietary models (Claude Opus, Qwen Plus, GPT-5, GPT-4, Gemini 3, DeepSeek) were accessed via licensed APIs in compliance with providers' terms of service and are not freely re-distributable. Open-source models were used under their respective permissive licenses (e.g., Apache 2.0, MIT), ensuring reproducibility. Other open-source tools (such as `pycocoevalcap`) are released under permissive licenses.

Data Safety: Our dataset is derived from publicly available regulatory documents and has been manually reviewed to ensure that it does not contain personally identifying information or offensive content. Human evaluation data was also collected under voluntary participation. Personal identifying information such as names is removed to protect privacy.

Expert participation: Human experts were provided with clear instructions outlining the evaluation task, including definitions, examples, and scoring rubrics (an example survey is presented in Appendix B). They were informed of the study purpose and how the data would be collected and used. The participation was voluntary and all participants retained their rights to withdraw or remove their data at any point. As compensation, domain-specific workshops were offered. All participants gave informed consent before contributing.

AI Assistance: Copilot was used to support writing in language polishing.

Acknowledgment

EC acknowledges support from the project "Analyzing the semantics of Transformers representations for financial natural language processing" (ZVYU), funded by the Faculty of Humanities of the Hong Kong Polytechnic University (PolyU-UGC). We would also like to thank our reviewers for the constructive feedback.

Special thanks are also due to all contributors who assisted with dataset creation and evaluation.

Dataset Creation:

- Ar. Chau Lok Kan, Director, RA(HK) HKIA — Co-creator
- Ir. Zhao Yueci, Associate, BEng CPEng NPER MIEAust MHKIE — Contributor

Human Evaluation (all as Contributors):

- Ar Dr. Adrian Lo, RA(HK) HKIA RIBA
- Ar. Amy Chan Nga Yin, Senior Project Manager, RA(HK) HKIA Authorized Person
- Ar. Emilie Wong, Founder, RA(HK) HKIA RIBA EDAC
- Ar. Jessica Chong, Assistant Manager, RA(HK) HKIA
- Ar. Jessica Kong, Director, RA(HK) HKIA
- Ar. Jiang Hejia, Co-founder, RA(HK) HKIA
- Ar. Lily Yu, Design Manager, RA(HK) HKIA
- Ar. Lu Ruixing, Senior Associate, RA(PRC)
- Ar. Nam Wu, RA(HK) HKIA
- Ir. Pin Wai Kei, Technical Director, MHKIE MIFireE
- Ar. Simon Chan, Lecturer, RA(HK) HKIA
- Ar. Song Huiqing, Architect, RA(HK) HKIA
- Ar. Sun Bochao, Architect, RA(HK) HKIA
- Ir. Sunny Wong Ho Ching, MHKIE RPE(Fire) CEng MIMEchE MCIBSE BEAM Pro CESGA

- Ar. Tian Yi, Senior Associate, RA(PRC) RA(HK) HKIA
- Ir. Victor Tung Kok Keung, Retired Fire Officer, RPE(Fire)
- Ar. Vincent Choi Kei Shun, Design Director, RA(HK) HKIA
- Ar. Xia Meng Yuan, Senior Associate, RA(HK) HKIA
- Ar. Xu Wei, Associate, RA(HK) HKIA
- Ar. Zeng Jingyao, Architect, RA(HK) HKIA
- Ar. Zhang Shuang, Architect, RA(HK) HKIA

References

Anthropic. 2024. Claude opus. https://www.anthropic.com. Large language model.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Kyubyung Chae, Jewon Yeom, Jeongjae Park, Seunghyun Bae, Ijun Jang, Hyunbin Jin, Jinkwan Jang, and Taesup Kim. 2026. Evaluating Structure-Aware Retrieval and Safety in Statute-Centric Legal QA. In *Proceedings of ACL*.

Google DeepMind. 2025. Gemini 3. https://deepmind.google/technologies/gemini. Large language model.

DeepSeek-AI. 2024. Deepseek-v3 technical report.

Matias Del Campo and Neil Leach. 2022. *Machine Hallucinations: Architecture and Artificial Intelligence*. John Wiley & Sons.

Prashnna Ghimire, Kyungki Kim, and Manoj Acharya. 2024. Opportunities and Challenges of Generative AI in Construction Industry: Focusing on Adoption of Text-Based Models. *Buildings*, 14(1).

Xingang Guo, Yaxin Li, Xiangyi Kong, Yilan Jiang, Xiayu Zhao, Zhihua Gong, Yufan Zhang, Daixuan Li, Tianle Sang, Beixiao Zhu, and 1 others. 2025. Toward Engineering AGI: Benchmarking the Engineering Design Capabilities of LLMs. *arXiv preprint arXiv:2509.16204*.

Hansi Hettiarachchi, Mohamed Medhat Gaber, Pouyan Parsafard, and Edlira Vakaj. 2026. SNOWTEC: Synthetic Natural language Oversampling With Transformer-based Information ExtraCtion for Automated Compliance Checking. *Machine Learning with Applications*, 24:100911.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and 1 others. 2023. A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions. *ACM Transactions on Information Systems*.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, and 1 others. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Dimitrios Kampelopoulos, Athina Tsanousa, Stefanos Vrochidis, and Ioannis Kompatsiaris. 2025. A Review of LLMs and their Applications in the Architecture, Engineering and Construction Industry. *Artificial Intelligence Review*, 58(8):250.

Terry K Koo and Mae Y Li. 2016. A guideline of selecting and reporting intraclass correlation coefficients for reliability research. *Journal of Chiropractic Medicine*, 15(2):155–163.

Chen Liang, Zhaoqi Huang, Haofen Wang, Fu Chai, Chunying Yu, Huanhuan Wei, Zhengjie Liu, Yanpeng Li, Hongjun Wang, Ruifeng Luo, and 1 others. 2026. AECBench: A Hierarchical Benchmark for Knowledge Evaluation of Large Language Models in the AEC Field. *Advanced Engineering Informatics*, 71:104314.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Qwen Team. 2024. Qwen2.5: A party of foundation models.

Qwen Team. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Qwen Team. 2026. Qwen3.6-Plus: Towards real world agents.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, and 1 others. 2025. OpenAI GPT-5 System Card. *arXiv preprint arXiv:2601.03267*.

xAI. 2024. Grok-4. https://x.ai. Large language model.

Ruoxin Xiong, Yael Netser, Pingbo Tang, Beibei Li, and Joonsun Hwang. 2026. Socio-technical Assessment of Generative AI Integration in Architecture, Engineering, and Construction (AEC) Workflows: An Empirical Study Using O* NET Occupational Taxonomy. *Advanced Engineering Informatics*, 71:104392.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, and 40 others. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

An Yang, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoyan Huang, Jiandong Jiang, Jianhong Tu, Jianwei Zhang, Jingren Zhou, Junyang Lin, Kai Dang, Kexin Yang, Le Yu, Mei Li, Minmin Sun, Qin Zhu, Rui Men, Tao He, and 9 others. 2025. Qwen2.5-1m technical report. *arXiv preprint arXiv:2501.15383*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Chunmo Zheng, Saika Wong, Xing Su, Yinqiu Tang, Ahsan Nawaz, and Mohamad Kassem. 2025. Automating Construction Contract Review Using Knowledge Graph-enhanced Large Language Models. *Automation in Construction*, 175:106179.

Zhe Zheng, Ke-Yin Chen, Xin-Yu Cao, Xin-Zheng Lu, and Jia-Rui Lin. 2023. LLM-funcmapper: Function Identification for Interpreting Complex Clauses in Building Codes via LLM. *arXiv preprint arXiv:2308.08728*.

A Supplementary Tables

Table 4: Example instances from the dataset.

Question:	What is the main purpose of a Qualitative Design Review (QDR) in a performance-based fire safety design? A) To confirm building materials meet combustibility limits B) To set fire safety objectives, identify design scenarios, choose acceptance criteria, and outline analysis methods C) To run computational fluid dynamics (CFD) simulations D) To commission fire service installations
Reference Answer:	B
Explanation:	A qualitative design review establishes the fire safety objectives, scenarios, acceptance criteria, and the analysis approach before any detailed quantitative work is carried out.
Question:	An office tenant proposes to shorten the escape path by routing occupants through a storage room and staff pantry to reach the exit stair. The storage is used for paper and small solvent supplies and is kept locked during normal hours. Which of the following design options best meets the means of escape requirement? A) Keep the route but add illuminated exit signs and emergency lighting through the storage to replace the visual fire alarm inside the office B) Reroute the escape path so it avoids rooms with higher fire load and provide a protected corridor where necessary to connect to the exit stair C) Allow the route but restrict it to staff via access control during evacuation D) Keep the route but install a wider door from the storage to the stair, keep the door from the office to the storage opaque, additional visual fire alarms not required
Reference Answer:	B
Explanation:	Escape routes should not pass through rooms with elevated fire risk (e.g., storerooms, kitchens/pantries). The intent is that occupants move within protected routes separated from fire hazards until reaching a place of safety. Signage, door width, or administrative controls cannot compensate for the inherent hazard of routing through high-risk rooms; a protected corridor or lobby is used to maintain separation where needed.
Question:	A restaurant with an estimated 80 patrons uses a frameless glass pivot door on its main exit that swings inward into the dining area to avoid obstructing the sidewalk. Is this acceptable, and what modification is required?
Reference Answer:	This is not acceptable because doors serving a floor or storey for more than 30 persons must swing in the direction of egress. In case the doors are swinging in both ways upper view panels shall be provided. Inward-swinging doors impede escape under crowd pressure and can trap occupants and are not allowed for rooms or storeys accommodating more than 30 persons. The door should be re-hung to swing outward (toward the place of safety) and be readily openable without a key or special knowledge. If sidewalk clearance is a concern, provide a small vestibule with an outward-swinging exit door while maintaining the required clear exit width.

(Table continues on the next page)

(Table continues from previous page)

Explanation:	The intent is to prevent crowd surge from blocking egress and to ensure doors can be opened quickly during an emergency. The outward swing requirement for exit doors is triggered by higher occupant loads, and doors must be openable without delay or special action. Any redesign must preserve the minimum clear exit width and avoid hardware or stops that reduce it.
Engineering-Related:	No
Question:	A performance-based smoke control design is proposed for a tall atrium to reduce the smoke exhaust rates. What must be demonstrated for the design to be acceptable, and how is occupant safety evaluated?
Reference Answer:	Define credible design fire scenarios and use validated models to predict smoke production, movement, and layer heights. Show that the available safe egress time exceeds the required safe egress time by an appropriate margin, considering pre-movement and evacuation durations. Demonstrate tenability along evacuation paths (visibility, temperature, and toxicity) remains within accepted limits at head height for the evacuation period. Provide sensitivity analyses for key parameters, document system reliability and management provisions, and demonstrate equivalence to prescriptive intent.
Explanation:	Performance-based acceptance hinges on meeting fire safety objectives through comparison of ASET and RSET and confirming tenability criteria under plausible worst cases. Validated modeling and sensitivity analyses ensure robustness against uncertainty. Management and reliability measures uphold the performance assumptions throughout the building's life.
Engineering-Related:	Yes

Table 5: Templates of Evaluation Prompts

`You are answering a question about the Hong Kong fire safety code (Code of Practice for Fire Safety in Buildings 2011).` [PLACEHOLDER] `Return the result strictly as a JSON object in the following format: {"answer": "xx", "reason": "xx"}` [QUESTION]

(a) Template for OEQs and MCQs with Hong Kong context specified[11]

`You are given a multiple-choice question. Choose ONE most appropriate answer from the options. Return the result strictly as a JSON object in the following format: {"answer": "xx", "reason": "xx"}` [QUESTION]

(b) Template for MCQs without Hong Kong context (w/o HK)

B Survey Template for Human Evaluation

For demonstration purposes, we created a simple survey template using Google Form. Any sections that could potentially reveal author information have been concealed. All expert raters (architects and fire engineers) were recruited, contacted, and invited to participate by two of the authors. Information such as evaluation purpose and instructions were personally communicated to participants (either face-to-face, via email, or through other means of contact). To protect participant privacy, identifying information has been hidden.

[11]The [PLACEHOLDER] is determined by the question type. For OEQs, it is filled with `You are given a short-answer question. Respond in a concise, accurate, and professional manner.` Otherwise, it is `You are given a multiple-choice question. Choose ONE most appropriate answer from the options.`

** Evaluation Form (Demo)

** is currently evaluating the performance of Large Language Models on the Code of Practice for Fire Safety in Buildings 2011. We greatly appreciate your support in assessing the quality of candidate answers.

Please find the evaluation details below:

1. You will be presented with 10 questions, their reference answers and the candidate answers from various AI (the questions vary in difficulty) .

2. Based on the reference answer and your domain expertise, please rate each candidate answer along two dimensions:
- Accuracy (Accuracy & Factual Correctness)
- Clarity (Clarity, Coherence & Professional Communication)

3. For each dimension, please complete the following two steps:
Step1: Choose one of the three levels.
Step2: If the candidate answer has a major issue, select one or all options that best apply.
*Place a "✅ " in the corresponding cell to indicate your selection.

If you have any questions, please feel free to contact **:
WhatsApp: **
Email: **

Thank you very much for your time and expertise!

* Indicates required question

1. Email *

QID: 01

Question:
In a high-occupancy commercial tower, what ongoing management practices are required to keep means-of-access features (fireman's lifts, firefighting lobbies, EVA gates) ready for use, and how should compliance be evidenced?

Reference Answer:
Establish a maintenance regime with periodic inspection and testing of fireman's lift emergency functions, lobby fire doors and pressurization, and any EVA gates or barriers, carried out by competent persons and registered contractors. Keep all access routes, lobbies, and stair approaches free from obstruction with documented housekeeping checks, and train staff on keeping these areas clear and on assisting responders during an incident. Conduct regular drills and audits, maintain a fire safety logbook with test results, defect rectification records, and certifications, and make keys and operation instructions readily available to the Fire Services.

Candidate Answer 1

Required ongoing management practices include: (1) Regular inspection and testing of fireman's lifts (typically monthly functional tests and annual comprehensive inspections), (2) Keeping firefighting lobbies clear of obstructions, properly ventilated, and fire doors maintained, (3) Ensuring EVA gates remain accessible, unobstructed, clearly marked, and keys/access available to fire services, (4) Maintaining emergency lighting, signage, and communication systems, (5) Staff training on emergency procedures, and (6) Coordinating with fire services for familiarization visits.

2. Which options best describe the candidate answer above? *

Mark only one oval.

- Fully accurate: The answer contains no factual errors and correctly addresses all major aspects with practical and code-compliant recommendations.
- Minor factual inaccuracy: The answer includes small, non-critical errors that do not materially affect the overall compliance judgment.
- Irrelevant but correct clause cited : The answer references a real COP FS provision, but one that is unrelated to the question or does not support the required judgment.
- Omission of critical information: The answer fails to include essential clauses, limits, or requirements necessary for a complete and accurate response.
- Hallucination / Fabricated content: The answer introduces plausible but entirely false information that does not exist in COP FS.
- Fundamental misunderstanding: The answer severely misinterprets the question, the regulatory intent, or the core compliance requirement.

3. Which options best describe the candidate answer? *

Mark only one oval.

- Clear and professional: The answer is fluent, well-structured, and logically presented, using precise and appropriate domain terminology.
- Minor clarity issues: The answer contains slightly awkward phrasing or minor redundancy, but the overall meaning remains clear and comprehensible.
- Inaccurate or inconsistent terminology: The answer uses incorrect, imprecise, or non-standard Hong Kong fire safety terminology.
- Logical gaps or weak reasoning: The answer shows disconnected arguments, unjustified logical jumps, or conclusions that are not adequately supported by the explanation.
- Insufficient explanation / justification: The answer provides conclusions without necessary clause references, rationale, or step-by-step reasoning.
- Poor readability / disorganized structure: The answer is difficult to follow due to unclear structure, lack of paragraphing, or poor logical flow.

Candidate Answer 2

Required ongoing management practices include: (1) Regular inspection and testing of fireman's lifts (typically monthly functional tests and annual comprehensive inspections), (2) Keeping firefighting lobbies clear of obstructions, properly ventilated, and fire doors maintained, (3) Ensuring EVA gates remain accessible, unobstructed, clearly marked, and keys/access available to fire services, (4) Maintaining emergency lighting, signage, and communication systems, (5) Staff training on emergency procedures, and (6) Coordinating with fire services for familiarization visits.

4. Which options best describe the candidate answer above? *

Mark only one oval.

- Fully accurate: The answer contains no factual errors and correctly addresses all major aspects with practical and code-compliant recommendations.
- Minor factual inaccuracy: The answer includes small, non-critical errors that do not materially affect the overall compliance judgment.
- Irrelevant but correct clause cited : The answer references a real COP FS provision, but one that is unrelated to the question or does not support the required judgment.
- Omission of critical information: The answer fails to include essential clauses, limits, or requirements necessary for a complete and accurate response.
- Hallucination / Fabricated content: The answer introduces plausible but entirely false information that does not exist in COP FS.
- Fundamental misunderstanding: The answer severely misinterprets the question, the regulatory intent, or the core compliance requirement.

5. Which options best describe the candidate answer? *

Mark only one oval.

- Clear and professional: The answer is fluent, well-structured, and logically presented, using precise and appropriate domain terminology.
- Minor clarity issues: The answer contains slightly awkward phrasing or minor redundancy, but the overall meaning remains clear and comprehensible.
- Inaccurate or inconsistent terminology: The answer uses incorrect, imprecise, or non-standard Hong Kong fire safety terminology.
- Logical gaps or weak reasoning: The answer shows disconnected arguments, unjustified logical jumps, or conclusions that are not adequately supported by the explanation.
- Insufficient explanation / justification: The answer provides conclusions without necessary clause references, rationale, or step-by-step reasoning.
- Poor readability / disorganized structure: The answer is difficult to follow due to unclear structure, lack of paragraphing, or poor logical flow.

6. (Optional) If you have any questions or comments for this Q-A pair, you may leave them here.

Skip to section 3 (Thank you for your time. Please click "Submit" to finalize your response.)

Thank you for your time. Please click "Submit" to finalize your response.

This content is neither created nor endorsed by Google.

Google Forms

CrisiLangs: A Multilingual Resource for Crisis Warning Generation

Giacomo Gonella[1, 2], **Stefano Menini**[1], **Marco Guerini**[1]
[1]Fondazione Bruno Kessler, Italy
[2]University of Trento, Italy
{ggonella, menini, guerini}@fbk.eu

Abstract

Communicating with the public during an emergency entails issuing warning messages promptly. In Europe, it often requires producing them in more than one language. Recent work has shown that language models can be trained to generate warning messages as a crisis unfolds, but the available data covers only English. To fill this gap, we present *CrisiLangs*, a multilingual extension of *CrisiText* (an English dataset of crisis events paired with warning messages written according to expert communication guidelines) to Italian, Bulgarian, German, and Greek. Expert native speakers translated the warning messages of 1,000 events drawn from 411 scenarios, for a total of 5,000 event-message pairs across five languages. We test the resource to fine-tune a multilingual model on all five languages jointly, with and without the emergency guidelines in the prompt, and compare it with a zero-shot baseline. The fine-tuned models outperform the baseline on ROUGE-L, BLEU and BERTScore in every language and configuration. In a human preference study, low decoding temperatures receive the most preferences in all five languages and in both configurations, while the effect of the guidelines varies by language.

1 Introduction

Timely and effective communication is a critical aspect of crisis management: during an emergency, authorities must inform the population about what is happening and about what to do, and they must do it repeatedly as the situation evolves. This has to be done under time pressure and the volume of information handled during an event is a recognized source of stress for emergency operators (Misra et al., 2020). The issue is even worse in Europe, where nations are linguistically diverse (and many have multiple official languages) and there is a high cross-border mobility. Therefore, warnings issued in a single language can fail to reach part of the population they are meant to protect. Issuing the same alert in several languages, at the speed an emergency requires, complicates the work of operators. Recent advances in Artificial Intelligence, and in particular Large Language Models (LLMs), which can interpret and produce text, offer a way to support this task (Foubert et al., 2026).

Crisis NLP has largely worked on analyzing text produced by users, with tasks such as detecting events in social media feeds and extracting informative content (Olteanu et al., 2014; Alam et al., 2021), but recent work has also studied how to use LLMs to assist operators in writing warning messages for ongoing crises. For instance, the *CrisiText* dataset (Gonella et al., 2026a) contains crisis scenarios as chronological chains of events, each paired with a warning message.

Existing resources and related models such as *CrisiText* focus only on English. Multilingual coverage in crisis Natural Language Generation (NLG) has been addressed separately as a translation problem, by applying domain-adapted machine translation to existing crisis text (Staiano et al., 2025; Castaldo et al., 2026). Crisis content, however, is sensitive to automatic translation. Ticona and Anastasopoulos (2026) report lower quality for low-resource languages and when translating out of English. One way to avoid sending automatically translated content to civilians would be to generate messages directly in the target language. However, to the best of our knowledge, no comparable training resource is available for languages other than English.

To address this gap, we present *CrisiLangs*[1], a multilingual extension of *CrisiText* including Italian, Bulgarian, German, and Greek, chosen to cover different writing systems and alphabets. In collaboration with crisis communication experts we manu-

[1]The dataset can be found at https://huggingface.co/datasets/LanD-FBK/CrisiLangs

Proceedings of the First Workshop on Language Technologies for Crisis Preparedness and Response, pages 20–31
November 9, 2026 ©2026 Association for Computational Linguistics

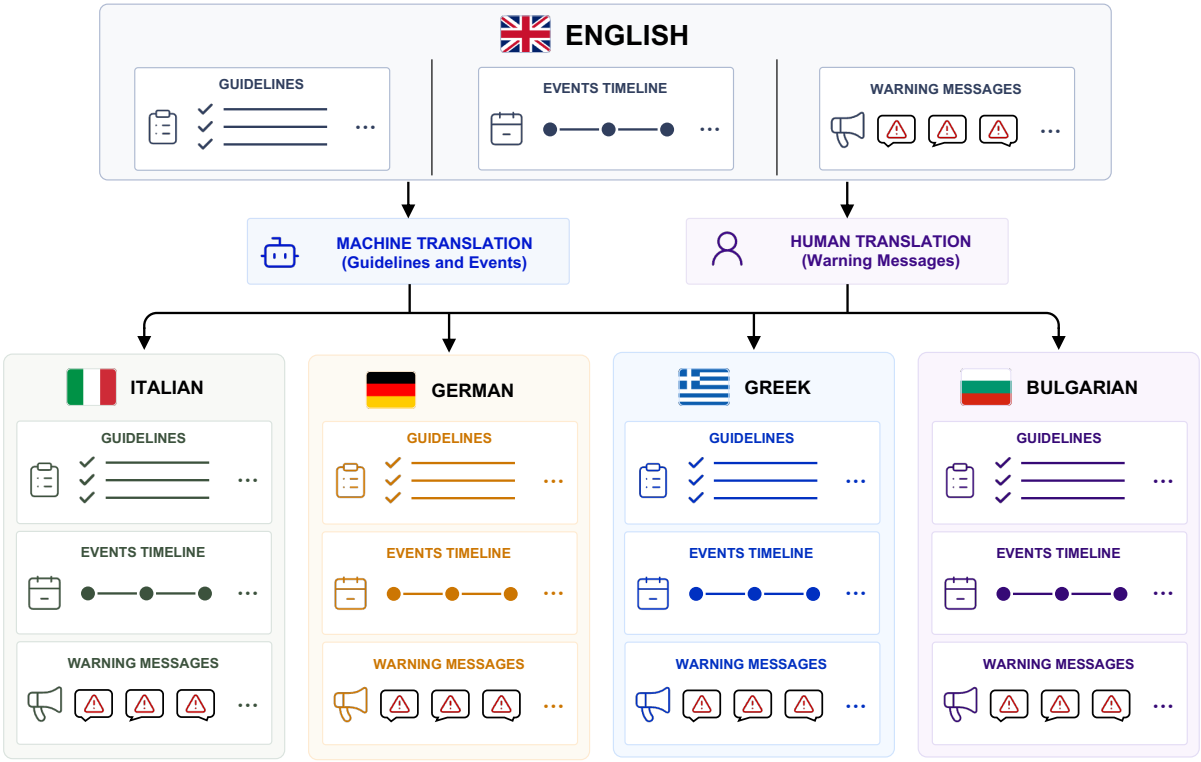


Figure 1: Workflow for the creation of *CrisiLangs*. Event descriptions and FEMA guidelines are machine-translated into each target language, while the warning messages are translated and adapted by a native speaker.

ally translated a subset of the *CrisiText* dataset into their native languages. The subsample covers 1,000 events from 411 crisis scenarios for a total of 5,000 event-message pairs across the five languages.

We then use the resource to jointly fine-tune a multilingual model on all five languages in two configurations: with and without the emergency guidelines in the prompt. We tested the two configurations in two ways. First, we evaluate the generations with automatic metrics at five decoding temperatures, with a zero-shot model used as the baseline. Since warning messages have to follow specific communication and behavioral guidelines, we varied the temperature to assess which level of lexical variation is best suited to the task. Second, to assess the quality of the generations, we asked native speakers to evaluate them in all five languages.

2 Related Work

Most NLP work on crises relies on social media data, and the resulting resources are designed for classification and information extraction rather than automatic generation (Imran et al., 2015; D. Lewis et al., 2025). Datasets of user-created content collected during floods, earthquakes or tornadoes (Olteanu et al., 2014; Imran et al., 2016) are annotated for informativeness, crisis detection or crisis type recognition, and are used to train classifiers (Alam et al., 2021; Liu et al., 2021). Fewer resources cover languages other than English: HumSet (Fekih et al., 2022) provides humanitarian response documents in English, French and Spanish, and Kawarith (Alharbi and Lee, 2021) collects Arabic crisis tweets. Relatedly, Sánchez et al. (2023) study the cross-lingual transfer of classifiers trained on English crisis data to Spanish and Italian, and Vitiugin and Purohit (2024) detect and rank help requests posted in English, Spanish and Turkish during natural disasters.

Crisis NLP does, however, include generation, mainly in the form of summarization. The CrisisFACTS track (McCreadie and Buntain, 2023) recasts disaster response as the production of fact-based daily summaries from social media and news, and recent systems address it by combining retrieval with instruction-following LLMs (Seeberger and Riedhammer, 2024). A related line works at the level of the individual emergency call, extracting the relevant information and presenting it to the call-taker (Otal et al., 2024). In both cases the output supports the response effort and is derived from

text that already exists. Warning messages, on the other hand, are directly addressed to the population, and they are not produced by recapping existing textual reports.

How to reach speakers of different languages during an emergency has been studied in translation research under the name of crisis translation (O'Brien and Federici, 2019), which looks at language needs in disaster settings and at how translation can be included in emergency planning. In NLP, the same problem has been approached as machine translation. Staiano et al. (2025) evaluate general-purpose LLM and NMT systems on Italian civil protection texts, while Castaldo et al. (2026) fine-tune a small model for crisis-domain translation to produce simplified English output, proposed as a "lingua franca" when full multilingual coverage is not feasible. This transfer is not uniformly reliable: Ticona and Anastasopoulos (2026) evaluate MT systems and LLMs on crisis-domain data in 30 languages and report substantial quality degradation, particularly for low-resource languages and in the direction from English into other languages.

Finally, a different line of work addresses the production of the message itself. The setting differs from the translation work described above, which starts from existing crisis text and moves it into another language. *CrisiText* (Gonella et al., 2026a) pairs chronologically ordered crisis events with English warning messages written according to crisis guidelines, and has been used to fine-tune LLMs for message generation and post-editing. Other work addresses different communication settings: Hong et al. (2025) generate replies to individual social media posts in which affected people express their needs. Guidance can also be grounded in the spatial layout of the affected area: Gonella et al. (2026b) benchmark vision-language models on the generation of narrowcast guidance, i.e. guidance tailored to the individual civilian, in a simulated evacuation environment. These works produce messages in English, leaving the task in other languages unexplored.

3 Multilingual Dataset

Training an LLM to produce warning messages requires event-message data pairs. In order to address the task in a multilingual setting, we built *CrisiLangs* by extending the English *CrisiText* data. Specifically, we used a manually curated subset of the original dataset and had its warning messages translated into Italian, Bulgarian, German, and Greek by four expert native speakers, one for each language. The four languages were selected to ensure diversity in writing systems and language families, including three scripts (Latin, Cyrillic and Greek) and four European language families (Romance, Slavic, Germanic and Hellenic). Figure 1 shows the workflow used to create the dataset.

Source Data. *CrisiText* (Gonella et al., 2026a) contains more than 100,000 event-message pairs, where each event is labeled with one or more crisis categories (e.g., attack, flood or tornado). The messages were synthetically generated with an LLM following expert guidelines on communicative tone (Sutton and Kuligowski, 2019) and the behavioral guidance of the corresponding crisis categories, extracted from the official FEMA website.[2] Events are also grouped into *scenarios*, in the form of chronological chains of events describing the unfolding of a single crisis, and each scenario is entirely contained in one of the original training, development, and test splits.

Category	Scenarios	Events
Human-caused	**253**	**500**
explosion	105	201
fire	76	153
attack	72	146
Natural	**158**	**500**
tornado	33	111
hurricane	26	95
flood	21	59
winter weather	14	49
heat	17	48
thunderstorm	16	48
avalanche	10	35
earthquake	7	12
landslide	4	12
volcano	3	12
wildfire	4	10
tsunami	3	9
Total	**411**	**1,000**

Table 1: Distribution of the event and message main categories over the 1,000 selected elements, grouped into human-caused and natural crises.

Selected Data. As part of the data quality assessment of *CrisiText*, two crisis communication experts reviewed a sample of 1,000 event-message pairs, balanced between human-caused and natural disasters. We built our multilingual extension on this expert-reviewed sample. The pairs were drawn

[2] https://www.ready.gov/

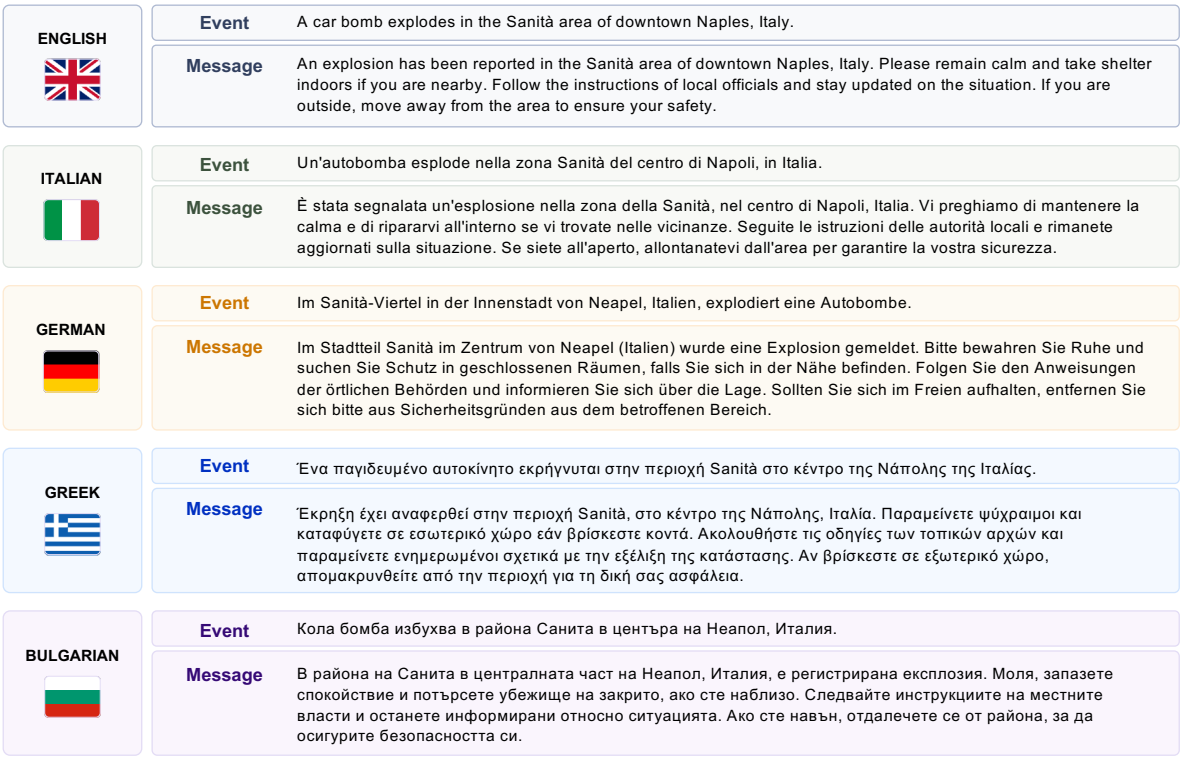


Figure 2: Example of event and warning message from *CrisiLangs* in the five languages. The English event and message are the originals from *CrisiText*. In the other four languages the messages are human translations produced by native speakers, while the events are machine-translated.

All	EN	IT	DE	EL	BG
306.9	268.6	316.4	324.1	331.1	294.4

Table 2: Mean warning message length in characters per language in *CrisiLangs*.

from the original training, development, and test splits in an 80/10/10 proportion. Table 1 reports the distribution of message and event categories. Events can carry multiple labels: the table reports only the main category of each event, i.e. the first label listed for its scenario, and omits the remaining labels assigned to the same scenario. The sample is balanced by construction, with 500 human-caused and 500 natural events.

Multilingual Translation. The subsample was given to four native speakers, one per target language (details about them are reported in Appendix C). Each translator received the original description of the event (both in English and automatically translated into their own language) and its associated warning message in English. The FEMA guidelines used to generate the warning messages in *CrisiText* were also automatically translated.[3] Translators were asked to render the English message in their own language and to adapt it where necessary so that it would sound more natural and well formed.[4] In both the messages and the guidelines we replaced the US emergency number "9-1-1" with the European one, "112", since the target audience is European. This localized version of the guidelines is also the one used at training and generation time (Section 4).

Human translation was limited to the warning messages. This was a deliberate design choice: the message is what the model has to produce and what the human evaluation judges, whereas the event description and the guidelines only enter as input. Translation noise on the input side is therefore less critical, and it reflects a realistic deployment setting, where the input is processed automatically. Translators could rely on the English source in case of doubt. In total, we obtained 5,000 event-message pairs (1,000 per language, including English), from 411 unique scenarios. Figure 2 shows one such pair in all five languages, with

[3]The translation was done with GOOGLETRANSLATE API.

[4]Translators were allowed to use machine translation tools as support, provided that they revised the output.

the machine-translated event alongside the human-translated message. These parallel versions differ slightly in length, with per-language mean lengths reported in Table 2.

4 Experimental Design

We used the dataset to fine-tune a model in two configurations and to establish a zero-shot baseline for comparison. In order to assess which settings produce the most appropriate warning messages and how generation quality changes across the different languages, the generations were evaluated with automatic metrics and with a human preference study. The aim is to give future work on the resource a benchmark to compare against.

Model and Training Data Construction. Following the original *CrisiText* work, we fine-tuned a single model, `Llama-3.1-8B-Instruct`, with LoRA (Hu et al., 2021), keeping the same hyperparameters. To allow comparison with it we also kept the original prompt template (Figure 3), in which the instructions are in English while the events description are in the target language. In a preliminary zero-shot check we observed that the model consistently produced the warning message in the language of the input, so we did not add any explicit indication of the output language to the template. In particular, each training instance contains an event description to be used as the basis for the warning message, plus the preceding events description to be used as context. The message is generated only for the last event of the sequence. Training was performed jointly across all five languages using a single multilingual model, rather than separate models for each language, with the aim of simplifying deployment in real-world settings.

Training Configurations. We trained the model in two configurations. In the *plain* setup, the user prompt contains only the event descriptions. On the other hand, in the *+guidelines* setup, the prompt additionally contains the translated FEMA guidelines of the crisis categories of the event. When an event belongs to more than one category, the corresponding guidelines are concatenated in the order in which the categories are listed. The comparison tests whether the guidelines have to be given explicitly at generation time, or whether fine-tuning on the data is enough for the model to infer them. Further details on the settings and on the hyperparameters are given in Appendix A.

USER

†`{Based on the provided guidelines,} Create a warning message informing on the current happening, providing a suggestion, for the last line in the following chain of events (be short, max 300 characters). No other output other than the message.`

†`Guidelines: [GUIDELINES]`

`Chain of events: [EVENTS]`

Figure 3: Prompt example. The instruction is always in English, while `[EVENTS]` is replaced by the chain of events of the instance, in its language, with the target event as its last line. The segments marked with † appear only in the *+guidelines* configuration, where `[GUIDELINES]` is replaced by the translated FEMA guidelines of the crisis categories of the event.

Generation and Automatic Metrics. After training, we generated warning messages for the 100 test events in all five languages, in both configurations and at five decoding temperatures (1.0, 0.8, 0.6, 0.4 and 0.2), to assess the trade-off between output variety and conformity to the crisis guidelines. As a baseline, we also ran the same generations with the off-the-shelf model in a zero-shot setting, using the same prompt as in training. In order to measure the effect of fine-tuning and of the decoding temperature, all generations were evaluated against the reference messages with two overlap metrics, ROUGE-L and BLEU, and with BERTScore, which is a semantic metric based on contextual embeddings. These scores were then used to select the data included in the human study. More information on the metrics is given in Appendix B.

Human Preference Study. Since automatic metrics are only a proxy for the quality of a warning message, we complemented them with a human preference study. On the basis of the automatic evaluation we restricted the study to the three best-performing decoding temperatures. For each of the two training configurations, one native speaker per language was presented with 50 test event-message pairs in their own language, each accompanied by the three corresponding generations. The two configurations were annotated separately, and the three candidates were shown anonymized, without any indication of the temperature that distinguished them. Annotators were asked to choose the message they considered grammatically better

			All			English			Italian			German			Greek			Bulgarian		
		T	R-L	B	BS	R-L	B	BS	R-L	B	BS	R-L	B	BS	R-L	B	BS	R-L	B	BS
Zero-shot	*plain*	1.0	.168	.037	.878	.209	.072	.890	.175	.041	.882	.185	.038	.884	.130	.019	.865	.143	.016	.871
		0.8	.184	.043	.884	.232	.083	.893	.196	.045	.886	.202	.047	.888	.145	.021	.874	.148	.021	.878
		0.6	.201	.048	.888	.234	.084	**.895**	.215	.049	.891	.218	.055	.892	.176	.032	.883	.166	.021	.883
		0.4	.205	.051	.890	.232	.083	.894	.214	.053	.890	.226	.050	.894	.174	.036	.887	.181	.035	.887
		0.2	**.211**	**.056**	**.892**	**.240**	**.090**	.894	**.221**	**.056**	**.892**	**.233**	**.059**	**.896**	**.182**	**.039**	**.888**	**.182**	**.035**	**.889**
	+guid.	1.0	.187	.049	.881	.231	.080	.892	.175	.047	.881	.211	.056	.884	.146	.032	.872	.170	.031	.878
		0.8	.196	.054	.885	.226	.068	.890	.191	.056	.886	.236	.072	.892	.154	.040	.878	.171	.031	.882
		0.6	.207	.056	.889	.247	.088	.895	.196	.051	.888	.229	.066	.890	.174	.035	.883	.191	.038	.888
		0.4	.218	.067	.890	.248	.094	.894	.213	.067	.890	.247	.076	.893	.178	**.048**	**.885**	**.202**	**.051**	**.890**
		0.2	**.220**	**.069**	**.891**	**.259**	**.104**	**.896**	**.216**	**.070**	**.891**	**.247**	**.077**	**.894**	**.179**	.044	.884	.198	.050	**.890**
Fine-tuned	*plain*	1.0	.259	.111	.902	.311	.159	.915	.245	.106	.901	.278	.110	.903	.236	.089	.893	.224	.079	.897
		0.8	.284	.134	.907	.350	.189	.920	.269	.125	.906	.308	.137	.909	.256	.111	.900	.235	.093	.902
		0.6	.307	.158	.912	.366	.211	.922	.297	.145	.912	.328	.161	.913	.282	.138	.905	.258	.122	.908
		0.4	.320	.169	.914	.387	.226	.924	.308	.159	.912	.340	.176	.915	.291	.136	.910	.271	.123	.910
		0.2	**.331**	**.177**	**.916**	**.389**	**.227**	**.925**	**.331**	**.171**	**.915**	**.345**	**.177**	**.916**	**.306**	**.150**	**.911**	**.281**	**.138**	**.912**
	+guid.	1.0	.258	.116	.901	.342	.190	.917	.240	.101	.901	.269	.112	.900	.215	.081	.892	.220	.082	.896
		0.8	.287	.135	.907	.370	.214	.921	.258	.114	.905	.303	.132	.907	.260	.121	.901	.243	.089	.902
		0.6	.308	.154	.911	.378	.221	.922	.299	.145	.910	.329	.163	.913	.286	.136	.907	.247	.099	.905
		0.4	.322	.165	**.914**	**.396**	**.233**	**.924**	.297	.149	.911	**.341**	**.172**	**.915**	.299	.144	.909	.277	.120	.910
		0.2	**.330**	**.171**	**.914**	.393	.232	.922	**.319**	**.162**	**.913**	**.341**	.167	.914	**.318**	**.159**	**.910**	**.283**	**.128**	**.911**

Table 3: ROUGE-L F1 (R-L), BLEU (B) and BERTScore F1 (BS) on the test split, per configuration, temperature and language. Bold marks the best temperature within each block. Zero-shot uses the un-finetuned `Llama-3.1-8B-Instruct` with identical configuration.

and more compliant with the task, and could leave free-text comments to motivate the choice or report problems the selection alone could not express. At the end of the annotation, we also asked which of the two configurations they preferred overall.

5 Results

This section reports what the experiments described in Section 4 indicate about the resource. The automatic evaluation serves two purposes: it establishes whether fine-tuning on the resource is necessary, and it narrows the set of decoding temperatures presented to the annotators. The human preference study then addresses what the metrics cannot capture, namely whether the generations are appropriate in each language and which of the two training configurations should be preferred.

5.1 Automatic Evaluation

Table 3 reports ROUGE-L, BLEU and BERTScore on the test split for each language and decoding temperature, in the two training configurations and in the corresponding baselines. The comparison is intended as intra-language, across temperatures and configurations. All three metrics are sensitive to language-specific factors such as tokenization and morphology, so inter-language differences do not indicate a quality ranking.

The fine-tuned models score higher than the baseline on all three metrics in every language, temperature and prompt configuration. ROUGE-L and BLEU reward two aspects of a generation at once, its length and its wording. BLEU keeps them separate, as the product of the brevity penalty and the n-gram precision term. As Table 4 shows, at $T = 0.2$ the fine-tuned models match the length of the human references, while the baseline with the plain prompt produces little more than half of the 300 characters. The table also reports BLEU without the brevity penalty: the n-gram precision term rises from .114 to .177, so fine-tuning also improves the wording. Adding the guidelines to the baseline prompt recovers part of the length gap, but the gain is small compared to fine-tuning and the n-gram precision term does not improve (.114 to .104). BERTScore shows the same pattern: at T=0.2 fine-tuning raises it from .892 to .916, while adding the guidelines to the baseline prompt leaves it virtually unchanged (.891). Therefore, these results indicate that fine-tuning on this data contributes substantially to both the length and the wording of the generations, and we restrict the rest of the evaluation to the fine-tuned models.

Across all setups, zero-shot and fine-tuned alike, scores rise as temperature decreases. This pattern reflects the fact that lower temperatures sharpen the probability around the most likely sequence, yielding wording that is less likely to include rare or un-

	Length	BLEU w/o BP
Gold	304	–
ZS, plain	173	.114
ZS, +guid.	205	.104
FT, plain	295	.177
FT, +guid.	299	.171

Table 4: Mean generation length in characters and BLEU without the brevity penalty (BP), at T=0.2, over all five languages, for the zero-shot (ZS) and fine-tuned (FT) models. Per-language values are in Appendix B.

expected choices, and thus closer to the reference text (which these metrics reward by construction). Whether such messages are also better is assessed in the human study (Section 5.2). When results are pooled across languages, both ROUGE-L and BLEU increase in a strictly monotonic way from temperature 1.0 down to 0.2. BERTScore does the same, except for a tie between 0.4 and 0.2 in the fine-tuned +guidelines configuration (.914). At the language level, however, the trend holds but is not always monotonic: in the +guidelines configuration, for example, ROUGE-L for English and BLEU for German are marginally lower at 0.2 than at 0.4. In fine-tuned models these deviations are less relevant, never exceeding .005.

The two training configurations are close to each other. In the 0.6-0.2 range, ROUGE-L and BLEU favor the +guidelines configuration for English and Greek, while the plain configuration is better on Italian. On German the two are close, with the plain configuration slightly ahead at temperature 0.2. Bulgarian is the only language where the two overlap metrics disagree: at the two lowest temperatures the plain configuration is ahead on BLEU and the +guidelines one on ROUGE-L. BERTScore separates the two configurations even less, with differences of at most .003 in every language in this range, and does not reproduce the advantage of the guidelines on English and Greek. Considering the aggregated metrics over all languages, the two configurations are virtually identical. No configuration is consistently better, so the automatic metrics do not settle which configuration to prefer, motivating the human evaluation described in Section 5.2.

Based on the automatic scores, we focused on temperatures 0.6, 0.4 and 0.2 for the human evaluation.

	plain				*+guidelines*			
	0.6	0.4	0.2	N	0.6	0.4	0.2	N
English	8	15	27	50	12	8	30	50
Italian	6	15	29	50	7	15	27	49
German	12	13	24	49	7	17	26	50
Greek	11	2	37	50	4	22	24	50
Bulgarian	4	17	29	50	3	9	25	37

Table 5: Preference counts per configuration and language. N is the number of annotated events out of 50.

5.2 Human Preference Study

The automatic scores rank the temperatures consistently, but they do not indicate which generation a reader would prefer, and they do not separate the two training configurations. We therefore asked one native speaker per language, with expertise in crisis communication, to compare the three generations and to select the one they considered most natural, grammatical and consistent with the task. The annotation covered 50 test events, with the candidates generated at temperatures 0.6, 0.4 and 0.2, and was repeated for both training configurations. When no candidate was acceptable, the annotator could leave the item unannotated.

Temperature. Table 5 shows that temperature 0.2 receives the most preferences in all five languages and in both training configurations, although the trend is not uniformly evident in all cases (e.g., for the +guidelines model). In Bulgarian, the generations using the guidelines led to the rejection of all three candidates in 13 items because of errors in the generated text. In the plain configuration a choice was always made, although the comments indicate that some candidates were still flawed. Lastly, Greek in the +guidelines configuration shows the smallest margin in the table, with only two preferences separating temperature 0.2 and 0.4, so the annotation does not indicate a clear preference between them.

Training Configuration. At the end of the task, each annotator also indicated which of the two models they preferred overall. All chose the plain model except the Greek annotator, who preferred the +guidelines one. The Italian annotator reported the plain model to have a better use of terminology and more specific warnings about the event, and the German annotator a better quality of the German text and slightly more accurate warning messages. Similarly, the English annotator motivated the preference with the accuracy of the messages, although

reporting a narrower difference between the two models. The Bulgarian annotator's justification was negative, referring to the amount of non-compliant content produced by the +guidelines model. On the other hand, the Greek annotator described the +guidelines generations as clearer, better structured and more effective in conveying urgency. ROUGE-L and BLEU favor +guidelines in the 0.6-0.2 range only for Greek and English, matching the Greek annotator but not the English one, while BERTScore does not separate the two configurations in either language.

Qualitative Feedback. In addition to the preference judgment, annotators could leave free-text comments. These were optional and used to different extents, so they are reported as qualitative evidence rather than as a systematic annotation. They point to a difference between the two configurations that the automatic metrics do not capture. The Greek, Italian and English annotators motivated their choices based on the wording of the warning and the appropriateness of the recommended actions. In Greek these reasons differ between the two configurations: the plain model was chosen for its simpler wording and calmer tone, the +guidelines model for the specificity of its information and of the recommended actions. In Bulgarian and German the comments instead report defects in the generations, most frequently grammatical errors and unfinished text. These are concentrated on the +guidelines model, consistent with the overall preference for the plain configuration reported above. None of these problems is visible in the automatic scores, and this is where the human study contributes most.

6 Discussion

Our experiments show that fine-tuning on *CrisiLangs* yields a single model that produces warning messages in all five languages, above the baseline in the automatic evaluation, but the human study suggests that quality is not uniform across languages. For English, Italian, German and Greek, the annotators expressed a preference for almost every item, and their comments mostly point to specific flaws rather than to unusable generations. By contrast, with the Bulgarian plain model a choice could always be made, even when candidates were non-compliant or contained errors. With the +guidelines model, 13 items were left unannotated because none of the three candidates was acceptable. The resource therefore supports warning generation in all five languages, with the Bulgarian +guidelines output being less reliable, mostly for limitations of the LLM.

Across the five languages we evaluated, low decoding temperatures are consistently preferred. The automatic metrics increase as the temperature decreases, as they favor wording closer to the reference by construction, so the more informative evidence comes from the human study: among the three temperatures compared, 0.2 receives the most preferences in all languages and in both training configurations (Table 5). Since annotators were asked to judge grammaticality and compliance with the task, this indicates that they found lower-temperature messages better on these criteria. The pattern holds in both the plain and the +guidelines configuration, so it does not depend on having the guidelines in the prompt. For practical use of this resource, low temperatures appear to be the safer choice.

The effect of the guidelines in the prompt is less uniform. Four of the five annotators preferred the plain model overall, citing better wording and more relevant content, and only the Greek annotator preferred the +guidelines one, for its clarity and structure. These preferences, together with the automatic metrics (where neither configuration is consistently ahead), suggest that in-prompt guidelines are not consistently beneficial. One possible explanation is that fine-tuning on the gold messages already induces much of the intended compliance, making the guidelines largely redundant; moreover, adding a long block of instructions increases prompt length and can dilute attention to the event-specific details, which may ultimately hurt generation quality.

A final observation concerns the relation between the two evaluations. The automatic metrics rank the temperatures consistently, but none of them, BERTScore included, anticipates the Bulgarian result: in Table 3 the +guidelines scores for Bulgarian are close to the plain ones at every temperature, yet the annotator found none of the three candidates acceptable in 13 of the 50 items in that configuration and in none of the plain ones. Human validation therefore remains necessary when crisis warning generation is extended to new languages.

7 Conclusion

We presented *CrisiLangs*, a multilingual extension of *CrisiText* to Italian, Bulgarian, German, and Greek. The resource contains 1,000 events from 411 scenarios, with human-translated warning messages in four languages alongside the original English ones, for a total of 5,000 event-message pairs.

To benchmark this resource we fine-tuned `Llama-3.1-8B-Instruct` on the five languages and compared it with a zero-shot baseline. We evaluated the generations in the five languages both with automatic metrics and with a human preference study. We observed that, compared to the baseline, fine-tuning improved ROUGE-L, BLEU and BERTScore in every language and prompt configuration (with or without guidelines). Lower decoding temperatures lead to higher automatic scores, which can be expected from reference-based metrics, this is then confirmed by the annotators where the temperature 0.2 received the most preferences in all five languages and in both configurations.

The human evaluation suggests that providing the emergency guidelines to the model during generation does not reliably improve the output: four of the five annotators preferred the plain configuration overall, and only the Greek annotator preferred the configuration with guidelines. In our evaluation, the automatic metrics did not reveal the problems found by the annotators: the Bulgarian +guidelines generations score close to the plain ones, yet in 13 of the 50 items none of the candidates was acceptable. This points to the importance of expert native-speaker assessment when benchmarking warning message generation.

Limitations

In the current setup the model receives the event in the target language and generates the warning message in the same language, so automatic translation is used only on the input side. The resource also supports feeding the model the English event directly and generating the message in the target language in a single pass, removing automatic translation from the pipeline entirely, but we do not evaluate this setting, as it would require a further round of automatic and human evaluation.

A second limitation is that we focus on fine-tuning a single multilingual model. Complementary comparisons (e.g., comparisons with language-specific pretrained LLMs, particularly for Greek and Bulgarian, which use non-Latin scripts, and with a model fine-tuned exclusively on English data to assess the added value of target-language supervision) are natural next steps that we leave to future work. Translation and preference annotation were carried out by a single native speaker per language, reflecting the specialized expertise required for crisis communication. Future work with additional annotators would enable measuring inter-annotator agreement; nonetheless, the preference for temperature 0.2 is shared by all five annotators.

Finally, the resource provides 1,000 expert-reviewed event-message pairs per language; by comparison, *CrisiText* contains more than 100,000 English pairs, as our extension focuses on the expert-reviewed subset. Human translation was also limited to the warning messages: the messages are what the model produces and what the annotators judge, whereas the events and guidelines only enter as input. Translation noise on the input side therefore remains in the data.

References

Firoj Alam, Hassan Sajjad, Muhammad Imran, and Ferda Ofli. 2021. Crisisbench: Benchmarking crisis-related social media datasets for humanitarian information processing. *Proceedings of the International AAAI Conference on Web and Social Media*, 15(1):923–932.

Alaa Alharbi and Mark Lee. 2021. Kawarith: an Arabic Twitter corpus for crisis events. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 42–52, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.

Antonio Castaldo, Maria Carmen Staiano, Johanna Monti, Sheila Castilho, and Francesca Chiusaroli. 2026. Translating under pressure: Domain-aware llms for crisis communication. *Preprint*, arXiv:2604.26597.

William D. Lewis, Haotian Zhu, Keaton Strawn, and Fei Xia. 2025. Tapping into social media in crisis: A survey. In *Proceedings of the Fourth Workshop on NLP for Positive Impact (NLP4PI)*, pages 306–331, Vienna, Austria. Association for Computational Linguistics.

Selim Fekih, Nicolo' Tamagnone, Benjamin Minixhofer, Ranjan Shrestha, Ximena Contla, Ewan Oglethorpe, and Navid Rekabsaz. 2022. HumSet: Dataset of multilingual information extraction and classification for humanitarian crises response. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4379–4389, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Samsara Foubert, Minjung Park, Kanad Pardeshi, Thomas Manzini, Robert McDaniel, David Merrick,

Robin Murphy, Aarti Singh, and Hoda Heidari. 2026. Investigating the role of ai in emergency management: Use cases, challenges, and opportunities. In *Proceedings of the 2026 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '26, page 2055–2091, New York, NY, USA. Association for Computing Machinery.

Giacomo Gonella, Gian Maria Campedelli, Stefano Menini, and Marco Guerini. 2026a. CrisiText: A dataset of warning messages for LLM training in emergency communication. In *Findings of the Association for Computational Linguistics: EACL 2026*, pages 6657–6677, Rabat, Morocco. Association for Computational Linguistics.

Giacomo Gonella, Stefano Menini, and Marco Guerini. 2026b. Guide me out: A framework to benchmark vlm operators communication in crisis scenarios. *Preprint*, arXiv:2606.09428.

Lingzi Hong, Xiaoying Song, Anirban Saha Anik, and Vanessa Frias-Martinez. 2025. Dynamic fusion of large language models for crisis communication. *Proceedings of the International ISCRAM Conference*, 22.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *Preprint*, arXiv:2106.09685.

Muhammad Imran, Carlos Castillo, Fernando Diaz, and Sarah Vieweg. 2015. Processing social media messages in mass emergency: A survey. *ACM computing surveys (CSUR)*, 47(4):1–38.

Muhammad Imran, Prasenjit Mitra, and Carlos Castillo. 2016. Twitter as a lifeline: Human-annotated Twitter corpora for NLP of crisis-related messages. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1638–1643, Portorož, Slovenia. European Language Resources Association (ELRA).

Junhua Liu, Trisha Singhal, Lucienne T.M. Blessing, Kristin L. Wood, and Kwan Hui Lim. 2021. Crisisbert: A robust transformer for crisis classification and contextual crisis embedding. In *Proceedings of the 32nd ACM Conference on Hypertext and Social Media*, HT '21, page 133–141, New York, NY, USA. Association for Computing Machinery.

Richard McCreadie and Cody Buntain. 2023. Crisisfacts: building and evaluating crisis timelines. *Proceedings of the 20th ISCRAM Conference*.

Shalini Misra, Patrick Roberts, and Matthew Rhodes. 2020. Information overload, stress, and emergency managerial thinking. *International Journal of Disaster Risk Reduction*, 51:101762.

Alexandra Olteanu, Carlos Castillo, Fernando Diaz, and Sarah Vieweg. 2014. Crisislex: A lexicon for collecting and filtering microblogged communications in crises. In *Proceedings of the international AAAI conference on web and social media*, volume 8, pages 376–385.

Hakan T. Otal, Eric Stern, and M. Abdullah Canbaz. 2024. Llm-assisted crisis management: Building advanced llm platforms for effective emergency response and public collaboration. In *2024 IEEE Conference on Artificial Intelligence (CAI)*, pages 851–859.

Sharon O'Brien and Federico Marco Federici. 2019. Crisis translation: considering language needs in multilingual disaster settings. *Disaster Prevention and Management*, 29(2):129–143.

Cinthia Sánchez, Hernan Sarmiento, Andres Abeliuk, Jorge Pérez, and Barbara Poblete. 2023. Cross-lingual and cross-domain crisis classification for low-resource scenarios. In *Proceedings of the international AAAI conference on web and social media*, volume 17, pages 754–765.

Philipp Seeberger and Korbinian Riedhammer. 2024. Multi-query focused disaster summarization via instruction-based prompting. *arXiv preprint arXiv:2402.09008*.

Maria Carmen Staiano, Lifeng Han, Johanna Monti, and Francesca Chiusaroli. 2025. ITALERT: Assessing the quality of LLMs and NMT in translating Italian emergency response text. In *Proceedings of Machine Translation Summit XX: Volume 1*, pages 566–577, Geneva, Switzerland. European Association for Machine Translation.

Jeannette Sutton and Erica D. Kuligowski. 2019. Alerts and warnings on short messaging channels: Guidance from an expert panel process. *Natural Hazards Review*, 20(2).

Belu Ticona and Antonios Anastasopoulos. 2026. Llm-powered automatic translation and urgency in crisis scenarios. *Proceedings of the International ISCRAM Conference*, 23.

Fedor Vitiugin and Hemant Purohit. 2024. Multilingual serviceability model for detecting and ranking help requests on social media during disasters. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 18, pages 1571–1584.

A Hyperparameters and Generation Settings

LoRA Adapter Configuration. We fine-tuned `meta-llama/Llama-3.1-8B-Instruct` with a `peft` LoRA adapter: rank 16, alpha 16, dropout 0, no bias adaptation, task type `CAUSAL_LM`. The adapter targets `q_proj`, `k_proj`, `v_proj`, `o_proj`, `gate_proj`, `up_proj` and `down_proj`. The base model was loaded in `bfloat16` without quantization, and the adapter was merged back into the base weights before generation.

SFT. Training used `trl`'s `SFTTrainer`. Sequences were capped at 1024 tokens; the learning rate was $3e{-}4$ with a cosine schedule, weight decay 0.01 and warmup ratio 0.03. We used a batch size of 2 for training and 4 for evaluation, with gradients accumulated over 4 steps, for an effective batch size of 8. We ran for up to 10 epochs, evaluating and checkpointing every 50 steps with early stopping patience 5, and kept the checkpoint with the lowest validation loss. Data were shuffled with seed 42 and the trainer ran with seed 0. The two configurations differ only in the prompt: PLAIN receives the instruction and the chain of events, +GUIDELINES additionally receives the localized guideline blocks. Both were trained with the same hyperparameters on the same 4,000 training examples.

Chat Formatting. Padding used Llama-3.1's `<|finetune_right_pad_id|>` on the right, and turns were formatted with the Llama-3.1 chat template.

Llama Generation. Decoding used sampling with a top-p of 1 and a cap of 128 new tokens, sufficient for a warning message. Each test prompt was decoded once per temperature (1, 0.8, 0.6, 0.4, 0.2), giving 2,500 generations per model configuration (100 test records $\times$ 5 languages $\times$ 5 temperatures).

Zero-shot Baseline. The zero-shot baseline was the same `meta-llama/Llama-3.1-8B-Instruct` without the LoRA adapter, and was run over the same test split with the same prompts, the same five temperatures and the same decoding arguments and post-processing.

Compute. All fine-tuning and generation runs were performed on a single NVIDIA A40 GPU.

B Automatic Metrics

Evaluation Packages. For automatic metrics we used Hugging Face's `evaluate` library (`evaluate` 0.4.6, `rouge-score` 0.1.2, `nltk` 3.10.2) for ROUGE, `sacrebleu` 2.6.0 for BLEU, and `bert-score` 0.3.13 (`transformers` 4.57.6) for BERTScore, all computed per language, configuration and temperature.

ROUGE-L is the F1 of the longest common subsequence, computed over the message as a single token sequence, and aggregated as the median of 1,000 bootstrap resamples. `rouge_score`'s default tokenizer lowercases and keeps only Latin letters and digits, discarding all Cyrillic and Greek characters: with this configuration only about 2% of Bulgarian and 5% of Greek reference tokens survive, and what remains is digits and Latin-script names. We therefore passed a Unicode-aware tokenizer, so that all five languages are scored on their own scripts.

BLEU was computed at corpus level with SacreBLEU's default settings[5], including the script-agnostic `13a` tokenizer. Scores were divided by 100 to match the 0–1 scale of the other metrics.

BERTScore was computed with `xlm-roberta-large` for all five languages, without idf weighting and without baseline rescaling, as no rescaling baseline is available for Bulgarian and Greek. Each reported score is the mean of the per-item F1 scores. Without rescaling, all scores in Table 3 fall between .865 and .925, so BERTScore differences are small in absolute terms and should only be compared within its own columns.

All reported scores were computed on CPU directly from the released generation files.

Per-Language Metrics. Table 6 reports the per-language values behind the aggregated numbers of Table 4. Both components move in the same direction in all five languages: the fine-tuned models are closer to the reference length than the corresponding baseline, and the n-gram precision term is higher in every language and in both prompt configurations.

[5] Signature: `nrefs:1|case:mixed|eff:no|tok:13a|smooth:exp|version:2.6.0`

	All	EN	IT	BG	DE	EL
mean length (characters)						
Gold	304	267	315	289	317	330
ZS, plain	173	188	163	163	182	168
ZS, +guid.	205	204	214	192	221	195
FT, plain	295	269	297	272	326	309
FT, +guid.	299	270	312	284	317	314
BLEU without brevity penalty						
ZS, plain	.114	.137	.129	.076	.117	.101
ZS, +guid.	.104	.139	.102	.080	.112	.080
FT, plain	.177	.227	.184	.138	.177	.156
FT, +guid.	.171	.232	.162	.128	.167	.162

Table 6: Per-language mean generation length in characters and BLEU without the brevity penalty, at $T{=}0.2$, for the zero-shot (ZS) and fine-tuned (FT) models with the plain and +guidelines (+guid.) prompts. As in Table 3, the comparison is intra-language. Languages are English (EN), Italian (IT), Bulgarian (BG), German (DE) and Greek (EL).

C Human Annotations

The annotators were members of organizations engaged in crisis management, selected if native speakers in the target language and according to their professional experience. For the translation and annotation work, annotators were compensated fairly by their respective organizations in compliance with applicable national laws.

For each language and each of the two model configurations, annotators compared the outputs of the three best-scoring temperatures (0.6, 0.4, 0.2) on 50 test events. The three outputs were presented as blind candidate A/B/C, and annotators selected one candidate per event and could optionally leave a free-text comment.

Figure 4 reproduces verbatim the instructions given to the annotators for the translation and for the preference task.

TRANSLATION TASK

The task consists in creating alert messages in your language based on the English alert messages we provide. We ask you to translate the messages and, when necessary, adapt them to sound more natural and appropriate as alert messages in your language.

PREFERENCE TASK

Your task is to:

– For each row, select your preferred message using the dropdown menu, based on grammaticality and consistency with the task.

– At the top of each sheet, indicate which configuration (column) you preferred overall, and briefly explain why.

– Let us know which of the two models (sheets) you preferred overall, and why.

(Note: there is no need to correct the text of the messages.)

Figure 4: Instructions given to the annotators, reproduced verbatim. The translation instructions were provided together with the English source; the preference instructions accompanied one spreadsheet per training configuration.

SIREN: Situation-aware Index for Response and Emergency ActioNability in Crisis Translation

Maria Carmen Staiano[1], Lifeng Han[2], Johanna Monti[3], Francesca Chiusaroli[1]

[1]University of Macerata [2]LUMC & Leiden University [3]University of Naples "L'Orientale"

m.staiano@unimc.it, l.han@lumc.nl, jmonti@unior.it, f.chiusaroli@unimc.it

Abstract

Machine translation (MT) is increasingly utilized in multilingual crisis communication, yet conventional evaluation does not directly assess whether translated emergency messages preserve their actionability. We introduce SIREN (Situation-aware Index for Response and Emergency ActioNability), a human evaluation framework assessing whether translation choices compromise protective action, directive force, and urgency of the source text. We evaluate SIREN in two complementary settings. At the error level, we apply it to HOPE-annotated Italian–English translations produced by ChatGPT-4o and Google Translate: 31.63% and 48.73% of errors, respectively, compromise actionability, predominantly by affecting the protective action. At the message level, we evaluate English–French translations produced by TranslateGemma-4B and Google Translate, with actionability preserved in 85.34% and 92.67% of translations, respectively. TranslateGemma-4B failures predominantly weaken directive force, whereas Google Translate failures more often compromise protective actions and receive Major impact ratings. These findings show that translation systems can differ not only in how often actionability is lost, but also in the nature and functional impact of these failures. Alongside SIREN, we contribute an English–French crisis-translation resource with human actionability annotations and post-edited reference translations, supporting further research on function-oriented evaluation for crisis communication. Our experimental data is available at https://github.com/mcstaiano/SIREN.

1 Introduction

Artificial intelligence (AI) is shaping how warnings are delivered, making them faster and easier to access. Current AI systems can translate emergency alerts into multiple languages, tailor messages to different audiences and contexts, and adapt dissemination strategies in real time across communication channels (International Telecommunication Union et al., 2026). International frameworks such as the Sendai Framework for Disaster Risk Reduction (UNISDR, 2015) and the United Nations Sustainable Development Goals (SDGs) (United Nations, 2015) call for multilingual early warning systems that can reach entire populations, including linguistic minorities, remote communities, displaced populations and tourists who do not speak the local language.

Meeting this multilingual communication need at scale increasingly depends on the use of automatic translation. Yet, when disseminating crisis messages, linguistic adequacy on its own does not ensure effective reception by target users. Emergency warnings may be multimodal, time-sensitive or anchored to specific cultural contexts (O'Brien and Federici, 2020). Their evaluation should therefore extend beyond the linguistic surface to consider the **actionability** of the translated message, that is, *whether it enables recipients to identify and perform the intended protective action* (Sengupta et al., 2024).

This perspective resonates with Nida's functional equivalence theory, according to which "translation should also aim to produce the same effect on the reader as the original text did" (Nida, 1964). In crisis communication, it invites us to reconsider how Machine Translation (MT) quality is assessed. The question is not only whether a translation preserves the meaning of the source, but also *whether it supports the behavioural response that the message is intended to elicit*. This motivates the central question of our study: **Does a translated emergency message preserve the actionability of its source?**

To address this question, we introduce **SIREN** (**S**ituation-aware **I**ndex for **R**esponse and **E**mergency Actio**N**ability), a human evaluation framework designed to assess *actionability preservation* in emergency-message translation. We

Proceedings of the First Workshop on Language Technologies for Crisis Preparedness and Response, pages 32–45
November 9, 2026 ©2026 Association for Computational Linguistics

define actionability preservation as the extent to which a translated message retains the capacity of the source to enable recipients to identify and perform the intended protective action. Rather than assessing general translation quality, SIREN focuses on whether translation choices may affect the protective behaviour expected from recipients. It thus complements semantic and error-based MT evaluation with a functional, pragmatically-oriented perspective.

2 Related Work

MT evaluation in the Crisis Domain. Current approaches to MT evaluation in the crisis domain revolve around two main paradigms: (1) an output-focused approach, which assesses translation quality through textual and linguistic properties of the translated output, and (2) a reception-oriented approach, which focuses on how end users understand, interpret, and act upon translated information (Hao, 2026). Within the output-focused paradigm, translations can be evaluated using automatic metrics such as lexicon-based BLEU (Papineni et al., 2002), METEOR (Lavie and Agarwal, 2007), LEPOR (Han et al., 2012), chrF (Popović, 2015), embedding-space COMET (Rei et al., 2022) and BERTScore (Zhang et al., 2020), as well as human evaluation frameworks such as Multidimensional Quality Metrics (MQM) (Lommel and Melby, 2018; Lommel et al., 2024).

Previous research adopting this perspective in crisis MT has primarily examined translation quality through automatic metrics (Anastasopoulos et al., 2020; Lankford and Way, 2024), human error analysis, or a combination of the two (Castaldo et al., 2026; Robinson, 2025; Staiano et al., 2025). For instance, Staiano et al. (2025) investigated the performance of neural machine translation and LLM-powered automatic translation using ITALERT, a parallel Italian–English corpus covering eight crisis scenarios. Combining automatic metrics with human error analysis based on an MQM-derived framework, the study highlighted the limitations of relying exclusively on automatic evaluation scores. Nevertheless, Ticona and Anastasopoulos (2026) noted that research evaluating MT in the crisis domain using LLMs remains limited, underscoring the insufficient coverage of low-resource languages.

Severity-aware human evaluation further makes it possible to account for the potential consequences of translation errors that may not be reflected in aggregate automatic scores, thereby supporting more informed assessments of whether MT is suitable for deployment in crisis scenarios (Cadwell et al., 2019).

Recent work has also broadened this perspective by considering MT *usability* and *readiness*, shifting attention from relative system performance towards whether translation quality is sufficient for practical use in crisis response (Bansal et al., 2026). Other work has begun to examine the preservation of crisis-relevant communicative properties. Investigating MT alongside LLM-based *urgency* assessment, Ticona and Anastasopoulos (2026) show that translation and prompting choices can systematically alter perceived urgency across languages. Altogether, these approaches reflect a growing concern with evaluating MT not only in terms of translation quality, but also in relation to its role in downstream crisis-response tasks, such as disaster situation assessment and automatic triage.

Actionability in Crisis Translation. Reception-oriented approaches offer a complementary perspective by shifting the focus from linguistic properties of the translated output to its pragmatic use by intended recipients. In crisis communication, this perspective has been associated with dimensions such as *timeliness*, *understandability*, *trustworthiness*, and *actionability* (Hao, 2026). Among these, actionability is particularly relevant to emergency messages containing protective instructions, as their communicative purpose is not only to convey information but also to guide recipients towards an appropriate response.

Actionability has already been investigated empirically in the reception of machine-translated emergency messages. Hajek et al. (2024) examined the viability of MT for emergency communication from a reception-oriented perspective, comparing raw MT, human post-edited MT, and MT of pre-edited source texts with speakers of Spanish, Greek, Chinese and Dari. Their study investigated how recipients understood and responded to machine-translated emergency information, including whether translation inaccuracies affected their decisions about the actions to take.

Building on this line of research, we approach actionability from a translation-evaluation perspective. While Hajek et al. (2024) investigate actionability through recipients' responses to different MT workflows, we shift the unit of analysis to

the source–translation relationship, asking whether, and to what extent, the behavioural function encoded in the source emergency message is preserved in translation. We further broaden the technological scope of previous work by investigating actionability preservation across both commercial MT systems and open-weight LLMs.

3 Methodology

Overview. SIREN is applied in two variants that differ in their unit of analysis: the error-span level (SIREN-span; Figure 1) and the full-message level (SIREN-msg; Figure 2). In both, annotators first give a binary judgement (YES/NO). If actionability is compromised, they select one of four predefined rationale categories and assign an impact level on a three-point scale: *Minor* (1), *Medium* (2), or *Major* (3).

Rationale categories. The four rationale categories are defined as follows: **Directive force weakened** indicates that the intended instruction becomes less direct, imperative, or insistent; **Directive force strengthened** indicates that it becomes more direct, imperative, or insistent than in the source; **Protective action compromised** indicates that the required action is changed, its scope is narrowed or broadened, or it becomes unclear or non-executable; and **Urgency marker omitted** indicates that an explicit or implicit urgency marker is omitted or attenuated, reducing the perceived immediacy of the required action.

The taxonomy was designed by drawing on complementary perspectives from pragmatics and applied frameworks for action-oriented communication. First, the distinction between *weakened and strengthened directive force* is informed by speech-act theory. Austin (1975) distinguishes between locutionary, illocutionary and perlocutionary acts, while Searle (1976) further develops the theory, by classifying illocutionary acts into five categories, including directives, which aim to get the addressee to perform a specific action. This distinction allows SIREN to capture translations in which the intended protective action remains identifiable, but the strength with which the directive is communicated changes.

Second, the *protective action compromised* category draws on the notion of actionability operationalised in the Patient Education Materials Assessment Tool (PEMAT) (Shoemaker et al., 2014), according to which materials are actionable when users of diverse backgrounds can identify what they can do on the basis of the information presented. Finally, *urgency marker omitted* is motivated by the Common Alerting Protocol (CAP) (OASIS, 2010), which treats urgency as a distinct property of an emergency alert, indicating the timeframe in which a responsive action should be taken.

Taken together, these sources motivate three functional dimensions of actionability: the protective action itself, the directive force with which it is communicated, and the urgency associated with the alert message, which SIREN operationalises through the four rationale categories.

SIREN-span. At the error-span level, we build on error annotations produced with HOPE (Gladkoff and Han, 2022), an MQM-derived human evaluation framework that assigns translation errors a category and severity level. SIREN-span is applied as an additional annotation layer by asking, for each error identified by HOPE: *Does the error compromise the actionability of the emergency message?* Since a single sentence can contain multiple errors, the number of error spans can be higher than the original number of sentences. If the answer is YES, annotators select a rationale category and an impact level as described above.

SIREN-msg. For SIREN at the full-message level, annotators directly assess the translated emergency message by answering the binary question: *Does the translation preserve the actionability of the source message?* If actionability is not preserved, annotators follow the same decision procedure as in SIREN-span: they select the rationale that best characterises the functional change and assign an actionability impact level.

The two settings provide complementary operationalisations of the same underlying construct. In the **error-level setting** (SIREN), annotators judge whether each individual translation error already compromises actionability. In the **message-level setting**, annotators judge whether the translated message as a whole preserves actionability, answer the same questions, and post-edit the message to restore it, producing a gold standard. The first operationalises actionability as a metric; the second provides an independent, holistic validation of that metric and a reusable, actionability-preserving resource.

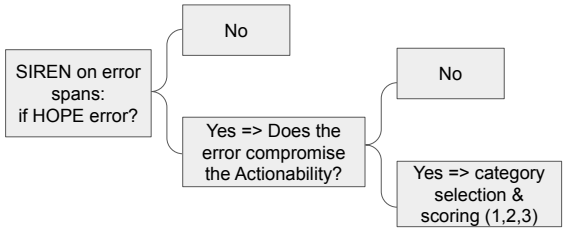


Figure 1: SIREN on Error Span Level.

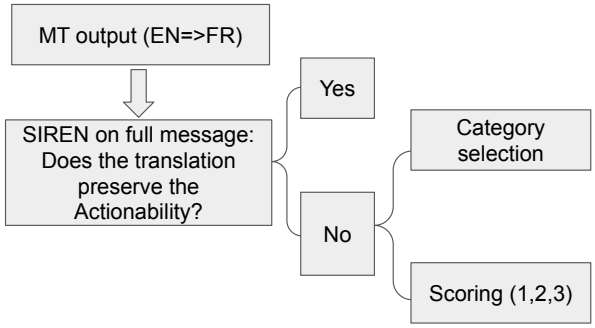


Figure 2: SIREN on Full Message Level.

4 Experimental Work

4.1 Corpus

We use data from the ITALERT corpus (Staiano et al., 2025), a bilingual Italian–English collection of authentic emergency-response messages drawn from the Italian Civil Protection campaign *Io non rischio*[1]. The corpus comprises 440 segments across eight crisis subdomains: flooding, earthquake, forest fire, volcanic eruption, tsunami, industrial accident, nuclear risk, and dam failure. Each source segment is paired with a professional human reference translation, and the corpus is open-sourced for reusability[2].

4.2 Study 1: Error-Level Actionability

Study 1 evaluates actionability preservation at the level of individual translation errors. We use the Italian→English translations from the ITALERT corpus produced by the two MT systems evaluated in the original study: ChatGPT-4o and Google Translate. First, we conduct pilot experiments with two annotators and calculate the inter-annotator agreement. Then, during the main annotation phase (rollout), the annotators divided the dataset and applied the SIREN protocol to 393 previously identified HOPE error instances: 196 from ChatGPT-4o and 197 from Google Translate. Detailed annotator profiles are provided in Appendix A.

4.2.1 Layer-I: Existing HOPE Annotations

Each translation already carries error annotations previously produced with the HOPE framework (Gladkoff and Han, 2022) by three professional linguists. Each error is assigned one of seven categories: Impact (IMP), Terminology (TRM), Ungrammatical (UGR), Mistranslation (MIS), Style (STL), Proofreading (PRF), Proper Name (PRN), along with a severity score (minor 1, medium 2, major 4, severe 8, critical 16). The original HOPE evaluation identified 611 errors for ChatGPT-4o and 728 for Google Translate. SIREN builds on these pre-existing annotations, without re-detecting translation errors or modifying their original HOPE severity.

4.2.2 Layer-II: SIREN-span Annotation

The second annotation layer applies SIREN-span to the pre-existing HOPE errors described in Layer I. The individual HOPE error constitutes the unit of analysis: each annotated error span is treated as a separate SIREN instance, even when multiple errors occur within the same sentence. For each error, annotators answer a binary question:

> *Does this error compromise the actionability of the emergency message?*

A NO judgement indicates that, despite the presence of a translation error, the intended protective action remains actionable. A YES judgement triggers two additional annotations following the SIREN-span procedure described in Section 3: annotators identify the functional rationale for the loss of actionability and assign an actionability impact level (MINOR, MEDIUM, or MAJOR).

SIREN judgements are made with respect to the functional consequences of the single error rather than its linguistic quality. The presence of a HOPE error therefore does not automatically entail a loss of actionability, nor does its HOPE severity determine its SIREN impact. This separation allows us to examine which translation errors have consequences for the protective function of the emergency message and how substantial those consequences are.

The complete SIREN-span annotation procedure is illustrated in the decision tree provided in Appendix B.

[1] Source: "Dipartimento della Protezione Civile-Presidenza del Consiglio dei Ministri".

[2] https://github.com/mcstaiano/ITALERT

4.2.3 Pilot Experiments on Italian–English

To validate the protocol before full-scale annotation, we ran three pilot rounds, adopting the iterative batch-based methodology of Temnikova et al. (2024) in which inter-annotator agreement is computed after each batch and disagreements are reconciled before proceeding to the next round.

In the first pilot round, two annotators independently applied the SIREN layer to 14 errors identified in the ChatGPT-4o outputs and 9 errors in the Google Translate outputs. Agreement on the binary actionability decision varied substantially across the two MT systems. For ChatGPT-4o, the annotators agreed on 8 of 14 errors (57.14%), yielding Cohen's $\kappa = -0.024$. For Google Translate, they agreed on 8 of 9 errors (88.89%), with substantially higher chance-corrected agreement ($\kappa = 0.769$).

Following the first pilot, a reconciliation meeting was held with both the annotators and a meta-reviewer, who did not participate in the annotation process. The disagreements were discussed qualitatively to identify recurring sources of ambiguity in the annotation protocol. Most involved cases in which an existing translation error affected lexical or terminological precision without necessarily altering the protective action conveyed by the message. Examples included uncommon lexical choices such as *semi-basement* and *cell phone* found in the English translations. These cases highlighted the difficulty of distinguishing general translation-quality issues from errors that have a functional impact on actionability in a real-world context. Based on the reconciliation process, the annotation guidelines were revised to make this distinction more explicit and to simplify the rationale taxonomy. In particular, the original categories *Protective action distorted* and *Required action unclear*, which annotators found difficult to distinguish consistently, were merged into a single category, *Protective action compromised*. The revised guidelines were then applied in the second pilot round.

In the second pilot, agreement improved to 27/30 judgements (90.00%; Cohen's $\kappa = 0.615$) for ChatGPT-4o. For Google Translate, the annotators agreed on 23/30 judgements (76.67%), with Cohen's $\kappa = 0.516$. These results suggested that the revisions introduced after the first pilot had improved the consistency of the binary actionability judgement, particularly for ChatGPT-4o. However, the lower agreement observed for Google Translate indicated that some borderline cases remained difficult to classify consistently. The disagreements concerned non-idiomatic expressions, such as *fix* instead of *secure*, *prudent attitude*, for which the intended protective action remained understandable despite potentially confusing wording. The second pilot therefore highlighted a difference in the decision threshold for actionability: specifically, whether additional cognitive effort or potential hesitation should be sufficient to classify an error as compromising actionability. This distinction was further clarified during reconciliation before proceeding to the third pilot. A third pilot was therefore conducted to assess whether the annotation protocol would remain stable when applied to a new batch of errors. For ChatGPT-4o, the annotators agreed on 30 of 35 judgements (85.71%), yielding Cohen's $\kappa = 0.684$. For Google Translate, the annotators agreed on 30 of 33 judgements (90.91%), yielding Cohen's $\kappa = 0.812$.

4.2.4 Rollout Annotation Italian–English

Following the three pilot rounds, the revised SIREN-span protocol was applied to the remaining Italian–English data selected for annotation. The refinements introduced during the pilot phase clarified the distinction between general translation-quality issues and errors affecting actionability, as well as the decision threshold for borderline cases.

Actionability Preservation. In Study 1, the preservation of actionability is assessed at the error-level using SIREN-span, which asks whether an identified translation error compromises the actionability of the emergency message. Accordingly, a YES judgement indicates that actionability is compromised, whereas a NO judgement indicates that the error does not affect actionability. The rollout comprised 196 SIREN-annotated HOPE error instances for ChatGPT-4o and 197 for Google Translate. As shown in Table 1, 62 ChatGPT-4o errors (31.63%) and 96 Google Translate errors (48.73%) were judged to compromise actionability. Here, N denotes the number of previously identified HOPE error instances evaluated with SIREN rather than the number of source sentences. The results should therefore be interpreted as the proportion of existing HOPE errors that have a functional impact on actionability, rather than as overall system-level error rates.

Rationale Distribution. Among errors judged to compromise actionability, we examine the distribu-

Table 1: SIREN-span results for the Italian–English experiment. A YES judgement indicates that the identified translation error compromises the actionability of the emergency message.

System	N	YES	NO	YES (%)	NO (%)
ChatGPT-4o	196	62	134	31.63	68.37
Google Translate	197	96	101	48.73	51.27

tion of the four SIREN rationale categories. Since a YES judgement in SIREN-span indicates that actionability is compromised, rationale distributions are calculated conditionally on YES cases. The appropriate conditional probability is therefore:

$$P(\text{category} \mid \text{YES}) = \frac{\#\ \text{YES errors in category}}{\#\ \text{YES errors with a valid category}} \times 100\%.$$

All actionability-compromising errors received a rationale annotation, resulting in 62 categorised YES cases for ChatGPT-4o and 96 for Google Translate.

As shown in Table 2, *protective action compromised* was by far the most frequent rationale for both systems, accounting for 47 of 62 ChatGPT-4o cases (75.81%) and 70 of 96 Google Translate cases (72.92%). *Directive force weakened* was the second most frequent rationale, observed in 14.52% of ChatGPT-4o cases and 19.79% of Google Translate cases. *Directive force strengthened* occurred only for ChatGPT-4o (4/62; 6.45%), whereas *urgency marker omitted* was relatively infrequent for both systems, accounting for 3.23% of ChatGPT-4o and 7.29% of Google Translate cases.

Actionability Impact. For errors judged to compromise actionability, we additionally examine their SIREN actionability impact. Impact is assessed independently of the original HOPE severity: an error assigned a low severity under HOPE may nevertheless have a substantial functional impact on actionability, and vice versa. As with the rationale distribution, impact is calculated conditionally on YES cases.

As shown in Table 3, the two systems exhibit different actionability-impact profiles. For ChatGPT-4o, the majority of actionability-compromising errors were rated MINOR (38/62; 61.29%), while 18 (29.03%) were rated MEDIUM and six (9.68%) MAJOR. Google Translate showed a larger proportion of higher-impact cases: 39 of 96 errors (40.63%) were rated MEDIUM and 19 (19.79%) MAJOR. Thus, approximately one fifth of Google Translate's actionability-compromising errors were rated MAJOR, compared with approximately one tenth of ChatGPT-4o's.

4.3 Study 2: Message-Level Actionability

Study 2 evaluates actionability preservation at the level of the translated message as a whole. We use the same source material to generate new English→French translations with TranslateGemma-4B, an open-weight translation model, and Google Translate. Two annotators independently evaluate these translations using the SIREN protocol. Detailed annotator profiles are provided in Appendix A.

Unlike Study 1, which applies SIREN to individual errors already identified through HOPE, Study 2 does not rely on pre-existing error annotations. Instead, annotators compare each English source message with its French translation and assess actionability by answering the following question:

> *Is the actionability of the emergency message preserved in the translation?*

Annotators provide a binary YES/NO judgement. A YES judgement indicates that the translation preserves the recipient's ability to identify and perform the intended protective action with the appropriate directive force and urgency, whereas a NO judgement indicates that actionability is compromised.

4.3.1 Prompting

For TranslateGemma-4B, we followed the prompt format recommended in the model's technical documentation on Ollama, adapting it to the English–French translation direction and adding a brief domain-specific instruction indicating experience with crisis-related content. No examples or additional translation instructions were provided. The complete prompt is reported in Appendix D.

4.3.2 Pilot Experiments on English–French

Prior to the full-scale evaluation, we conducted a pilot annotation to assess the applicability of SIREN at the message level, following the iterative-batch methodology described in Section 4.2.3. Two annotators independently evaluated 13 English–French translations produced by TranslateGemma-4B and the corresponding 13 translations produced by Google Translate.

For TranslateGemma-4B, the annotators agreed on 10 of 13 translations (76.92%), yielding Cohen's

Table 2: Distribution of SIREN rationale categories among errors judged to compromise actionability (YES) in the Italian–English experiment.

	ChatGPT-4o ($n = 62$)		**Google Translate** ($n = 96$)	
SIREN rationale	n	**%**	n	**%**
Directive force weakened	9	14.52	19	19.79
Directive force strengthened	4	6.45	0	0.00
Protective action compromised	47	75.81	70	72.92
Urgency marker omitted	2	3.23	7	7.29
Total	62	100.00	96	100.00

Table 3: Distribution of SIREN actionability-impact scores among errors judged to compromise actionability (YES) in the Italian–English experiment. Scores 1, 2, and 3 correspond to Minor, Medium, and Major impact, respectively.

		ChatGPT-4o ($n = 62$)		**Google MT** ($n = 96$)	
Impact	**Score**	n	**%**	n	**%**
Minor	1	38	61.29	38	39.58
Medium	2	18	29.03	39	40.63
Major	3	6	9.68	19	19.79
Total	–	62	100.00	96	100.00

Table 4: SIREN-msg results for the English–French experiment. A YES judgement indicates that the translation preserves the actionability of the emergency message.

System	N	**YES**	**NO**	**YES (%)**	**NO (%)**
TranslateGemma-4B	191	163	28	85.34	14.66
Google Translate	191	177	14	92.67	7.33

$\kappa = 0.316$. For Google Translate, they agreed on 12 of 13 translations (92.31%), with Cohen's $\kappa = 0.629$. Following the first pilot, disagreements were reviewed and the guidelines were refined accordingly. In particular, the annotators agreed that overly literal and wordy translations should be classified as instances of directive force weakening when the literal rendering reduced the directness of the instruction compared with the source message.

A second pilot was then conducted on 24 English–French translations per system. Agreement increased for both systems. For TranslateGemma-4B, the annotators agreed on 22 of 24 messages (91.67%), with Cohen's $\kappa = 0.48$. For Google Translate, they reached complete agreement (24/24; 100%), with both annotators judging actionability to be preserved across all translations in the second pilot.

A third pilot was conducted on a larger batch to further assess the stability of the revised annotation protocol. For TranslateGemma-4B the annotators agreed on 47 of 51 judgements (92.16%), yielding Cohen's $\kappa = 0.769$. For Google Translate, agreement reached 49 of 51 judgements (96.08%), with Cohen's $\kappa = 0.811$. Taken together, the three pilot rounds show a progressive stabilisation of inter-annotator agreement for the English–French language pair, particularly for TranslateGemma-4B, for which both observed agreement and Cohen's κ increased across successive rounds.

4.3.3 Rollout Annotation English–French

Following the three pilot rounds, the annotators applied the revised SIREN-msg protocol to the full rollout English–French data. The refinements introduced during the pilot phase clarified the treatment of overly literal or wordy translations, particularly when such renderings preserved the intended protective action but reduced the directive force of the source message.

Actionability Preservation. A total of 191 messages were evaluated for each system. As shown in Table 4, TranslateGemma-4B preserved actionability in 163 cases (85.34%), compared with 177 cases (92.67%) for Google Translate. Conversely, actionability was compromised in 28 TranslateGemma-4B translations (14.66%) and 14 Google Translate translations (7.33%). Thus, the observed rate of actionability-compromising translations was twice as high for TranslateGemma-4B:

$$\frac{28/191}{14/191} = 2.00$$

Rationale Distribution. Among translations for which actionability was not preserved, we examine the distribution of the four SIREN rationale categories. Since a NO judgement in SIREN-msg

indicates that actionability is compromised, rationale distributions are calculated conditionally on NO cases. The appropriate conditional probability is therefore:

$$P(\text{category} \mid \text{NO}) = \frac{\#\,\text{NO messages in category}}{\#\,\text{NO messages}}$$

As shown in Table 5, the two systems exhibited different distributions of actionability-related failures. For TranslateGemma-4B, *directive force weakened* was the predominant rationale, accounting for 23 of 28 actionability-compromising translations (82.14%). By contrast, Google Translate failures were predominantly classified as *protective action compromised*, accounting for 9 of 14 cases (64.29%). *Urgency marker omitted* was infrequent for both systems, while no instances of *directive force strengthened* were observed.

Actionability Impact. For translations in which actionability was not preserved, we additionally examine their SIREN actionability impact. As in Study 1, impact is classified as MINOR, MEDIUM, or MAJOR, corresponding to scores of 1, 2, and 3, respectively.

As shown in Table 6, most TranslateGemma-4B actionability-compromising translations were rated MINOR (15/28; 53.57%) or MEDIUM (10/28; 35.71%), while only three cases (10.71%) were rated MAJOR. In contrast, Google Translate exhibited a different impact profile: 8 of its 14 actionability-compromising translations (57.14%) were rated MAJOR, compared with four MINOR (28.57%) and two MEDIUM cases (14.29%). Thus, although Google Translate had fewer actionability-compromising translations overall, its observed failures were more frequently associated with MAJOR functional impact.

5 Discussion

Error-Level Actionability. The Italian–English experiment reveals differences in both the frequency and functional impact of actionability-compromising errors. Among the HOPE errors evaluated with SIREN, Google Translate errors were more frequently judged to compromise actionability than ChatGPT-4o errors (48.73% vs. 31.63%). A breakdown of actionability-compromising errors by HOPE error category is provided in Appendix F. For both systems, these cases mainly involved the protective action itself. However, their impact profiles differed: ChatGPT-4o's actionability-compromising errors were mostly MINOR, whereas Google Translate showed larger proportions of MEDIUM and MAJOR impacts.

Message-Level Actionability. In the English–French experiment, Google Translate preserved actionability more frequently than TranslateGemma-4B (92.67% vs. 85.34%). The systems nevertheless exhibited distinct failure profiles. TranslateGemma-4B failures predominantly weakened the directive force and were primarily rated MINOR or MEDIUM. In contrast, Google Translate's failures, although less frequent, compromised more often the protective action and were more frequently assigned MAJOR impact.

Comparisons of the Two Studies. In both experimental settings, SIREN identifies functional changes affecting the protective action, the directive force, and the urgency of the emergency message. While compromised protective actions were the most frequently observed rationale for both systems in the Italian–English experiment, the English–French experiment revealed different failure profiles. TranslateGemma-4B's failures mostly involved weakened directive force, whereas those of Google Translate more frequently compromised the protective action itself.

The findings show that the frequency of actionability-compromising errors or translations does not necessarily reflect their functional impact. In the English–French experiment, Google Translate preserved actionability more frequently than TranslateGemma-4B, yet a larger proportion of its compromised translations received MAJOR impact ratings. More broadly, the results indicate that asking whether one MT system is simply "better" than another, based on aggregate scores alone, may be insufficient for high-stakes crisis communication. Furthermore, these results demonstrate that *an error identified through conventional translation-quality assessment does not necessarily constitute an actionability failure*, while also showing that **SIREN can distinguish the functional consequences of errors that do compromise the protective function of a message**.

6 Conclusions

This paper introduced SIREN (Situation-aware Index for Response and Emergency ActioNability),

Table 5: Distribution of SIREN categories among translations for which actionability was not preserved (NO). Percentages are calculated within the NO cases of each system.

	TranslateGemma-4B ($n = 28$)		**Google Translate** ($n = 14$)	
SIREN category	n	**%**	n	**%**
Directive force weakened	23	82.14	4	28.57
Directive force strengthened	0	0.00	0	0.00
Protective action compromised	4	14.29	9	64.29
Urgency marker omitted	1	3.57	1	7.14
Total	28	100.00	14	100.00

Table 6: Distribution of actionability-impact scores among translations for which actionability was not preserved (NO). Scores 1, 2, and 3 correspond to Minor, Medium, and Major impact, respectively.

		TranslateGemma-4B ($n = 28$)		**Google MT** ($n = 14$)	
Impact	**Score**	n	**%**	n	**%**
Minor	1	15	53.57	4	28.57
Medium	2	10	35.71	2	14.29
Major	3	3	10.71	8	57.14
Total	–	28	100.00	14	100.00

a human evaluation framework designed to assess whether machine-translated emergency messages preserve the action-oriented dimensions of their source. SIREN evaluates actionability preservation and, where failures are identified, characterises their functional rationale and impact.

The SIREN framework was applied in two complementary experimental settings. The Italian–English study demonstrated its applicability at error level, augmenting existing HOPE annotations with assessments of their consequences for actionability. The English–French study extended the framework to holistic message-level evaluation, independently of error annotations. The latter setting additionally produced an *English–French crisis-translation resource* containing human actionability judgements and post-edited reference translations, providing a basis for further research on actionability-aware evaluation.

Together, the two studies illustrate how this framework can support the assessment of actionability at different levels of granularity. By focusing on the pragmatic consequences of translation choices, SIREN provides a complementary layer of evaluation alongside established linguistic, semantic, and automatic MT evaluation approaches, rather than replacing them or human expertise.

Limitations

This study has several limitations. First, the experiments are conducted on a relatively limited dataset, with particularly small numbers of actionability-compromising cases in some analyses. Experiments on larger datasets would provide more stable estimates and allow the robustness of the observed patterns to be assessed. Second, SIREN is evaluated on only two language pairs and a limited number of MT systems, warranting further validation across languages, crisis domains, and translation architectures. Extending the evaluation to languages beyond the Indo-European family would help assess the framework's applicability to linguistic systems in which directive force and urgency may be expressed through particles or verbal morphology. Finally, although the pilot experiments showed increasingly consistent annotation, actionability remains a human judgement; future work should adopt a reception-oriented approach with end-user studies to examine whether translated emergency instructions are correctly understood and acted upon.

Acknowledgments

This work was partially supported by the PhD Programme in Humanities and Technologies funded by the University of Macerata under D.R. No 253/2023. The second author acknowledges financial support from the HumanAI Cluster at the Leiden Institute of Advanced Computer Science (LIACS), Leiden University. The authors thank the anonymous reviewers for their feedback and suggestions, which helped improve this article.

Ethics Statement

The authors wish to acknowledge that SIREN is a research framework currently under development

and should not be interpreted as a certification of MT system safety or readiness for operational deployment. The framework is intended to support research on actionability preservation and to complement, rather than replace, human expertise and established evaluation procedures. We acknowledge that the framework and the judgments reported in this study are subject to the limitations of the data, language pairs, and human annotation process, and further validation is required before any operational application.

References

Antonios Anastasopoulos, Alessandro Cattelan, Zi-Yi Dou, Marcello Federico, Christian Federmann, Dmitriy Genzel, Francisco Guzmán, Junjie Hu, Macduff Hughes, Philipp Koehn, Rosie Lazar, Will Lewis, Graham Neubig, Mengmeng Niu, Alp Öktem, Eric Paquin, Grace Tang, and Sylwia Tur. 2020. TICO-19: the translation initiative for COvid-19. In *Proceedings of the 1st Workshop on NLP for COVID-19 (Part 2) at EMNLP 2020*, Online. Association for Computational Linguistics.

John Langshaw Austin. 1975. *How to do things with words*, volume 5. Harvard university press.

Vipasha Bansal, Elizabeth Brown, Chelsea Kendrick, Benjamin Pong, and William D. Lewis. 2026. Is MT ready for the next crisis or pandemic? *arXiv preprint arXiv:2601.10082*.

Patrick Cadwell, Sharon O'Brien, and Eric DeLuca. 2019. More than tweets: A critical reflection on developing and testing crisis machine translation technology. *Translation Spaces*, 8(2):300–333.

Antonio Castaldo, Maria Carmen Staiano, Johanna Monti, Sheila Castilho, and Francesca Chiusaroli. 2026. Translating under pressure: Domain-aware LLMs for crisis communication. In *Proceedings of the 26th Annual Conference of the European Association for Machine Translation*, Tilburg, Netherlands. European Association for Machine Translation.

Serge Gladkoff and Lifeng Han. 2022. HOPE: A task-oriented and human-centric evaluation framework using professional post-editing towards more effective MT evaluation. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 13–21, Marseille, France. European Language Resources Association.

John Hajek, Yu Hao, Ambrin Hasnain, Anila Hasnain, Ke Hu, Maria Karidakis, Rachel Macreadie, Anthony Pym, and Juerong Qiu. 2024. Understanding and improving machine translations for emergency communications.

Aaron L. F. Han, Derek F. Wong, and Lidia S. Chao. 2012. LEPOR: A robust evaluation metric for machine translation with augmented factors. In *Proceedings of COLING 2012: Posters*, pages 441–450, Mumbai, India. The COLING 2012 Organizing Committee.

Yu Hao. 2026. Quality in multilingual crisis communication: A reception-oriented perspective. In *Translation Research Projects 6: Testing Borders - A Tribute to Anthony Pym*, pages 117–134. Universitat Rovira i Virgili, Tarragona.

International Telecommunication Union, United Nations Office for Disaster Risk Reduction, World Meteorological Organization, and International Federation of Red Cross and Red Crescent Societies. 2026. Leveraging AI to enhance multi-hazard early warning systems (MHEWS). Report, International Telecommunication Union, Geneva, Switzerland.

Séamus Lankford and Andy Way. 2024. Leveraging llms for mt in crisis scenarios: A blueprint for low-resource languages. In *Proceedings of the 16th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Track)*, pages 4–13.

Alon Lavie and Abhaya Agarwal. 2007. METEOR: An automatic metric for MT evaluation with high levels of correlation with human judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 228–231, Prague, Czech Republic. Association for Computational Linguistics.

Arle Lommel, Serge Gladkoff, Alan K. Melby, Sue Ellen Wright, Ingemar Strandvik, Katerina Gasova, Angelika Vaasa, Andy Benzo, Romina Marazzato Sparano, Monica Foresi, et al. 2024. The multi-range theory of translation quality measurement: Mqm scoring models and statistical quality control. In *Proceedings of the 16th Conference of the Association for Machine Translation in the Americas (Volume 2: Presentations)*, pages 75–94.

Arle Lommel and Alan Melby. 2018. Tutorial: MQM-DQF: A Good Marriage (Translation Quality for the 21st Century). In *Proceedings of the 13th Conference of the Association for Machine Translation in the Americas: Volume 1*, Boston, MA. Association for Machine Translation in the Americas.

Eugene Albert Nida. 1964. *Toward a science of translating: With special reference to principles and procedures involved in Bible translating*. Brill Archive.

OASIS. 2010. Common alerting protocol version 1.2. OASIS standard, Organization for the Advancement of Structured Information Standards (OASIS).

Sharon O'Brien and Federico Marco Federici. 2020. Crisis translation: Considering language needs in multilingual disaster settings. *Disaster Prevention and Management: An International Journal*, 29(2):129–143.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal. Association for Computational Linguistics.

Ricardo Rei, José G. C. de Souza, Duarte Alves, Chrysoula Zerva, Ana C Farinha, Taisiya Glushkova, Alon Lavie, Luisa Coheur, and André F. T. Martins. 2022. COMET-22: Unbabel-IST 2022 Submission for the Metrics Shared Task. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 578–585, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Nathaniel Romney Robinson. 2025. Take shelter, zanmi: Digitally alerting cyclone victims in their languages. In *Proceedings of the Fourth Workshop on NLP for Positive Impact (NLP4PI)*, pages 70–76, Vienna, Austria. Association for Computational Linguistics.

John R Searle. 1976. Toward a linguistic theory of speech acts.

Medha Sengupta, Anthony Pym, Yu Hao, John Hajek, Maria Karidakis, Robyn Woodward-Kron, and Riccardo Amorati. 2024. On the transcreation, format and actionability of healthcare translations. *Translation & Interpreting: The International Journal of Translation and Interpreting Research*, 16(1):121–141.

Sarah J. Shoemaker, Michael S. Wolf, and Cindy Brach. 2014. Development of the patient education materials assessment tool (pemat): A new measure of understandability and actionability for print and audiovisual patient information. *Patient Education and Counseling*, 96(3):395–403.

Maria Carmen Staiano, Lifeng Han, Johanna Monti, and Francesca Chiusaroli. 2025. ITALERT: Assessing the quality of LLMs and NMT in translating Italian emergency response text. In *Proceedings of Machine Translation Summit XX: Volume 1*, pages 566–577, Geneva, Switzerland. European Association for Machine Translation.

Irina Temnikova, Iva Marinova, Silvia Gargova, Ruslana Margova, Alexander Komarov, Tsvetelina Stefanova, Veneta Kireva, Dimana Vyatrova, Nevena Grigorova, Yordan Mandevski, and Stefan Minkov. 2024. SMFEEL-BG - the first Bulgarian datasets and classifiers for detecting feelings, emotions, and sentiments of Bulgarian social media text. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 14954–14966, Torino, Italia. ELRA and ICCL.

Belu Ticona and Antonios Anastasopoulos. 2026. Llm-powered automatic translation and urgency in crisis scenarios. In *Proceedings of the 23rd International ISCRAM Conference: Building Stronger Futures: Ensuring Public Safety in Times of Crisis*, volume 23, The Hague, Netherlands.

UNISDR. 2015. *Sendai Framework for Disaster Risk Reduction 2015–2030*. United Nations Office for Disaster Risk Reduction, Geneva.

United Nations. 2015. *Transforming Our World: The 2030 Agenda for Sustainable Development*. United Nations, New York, NY.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

A Annotator Profiles

A.1 Study 1: Error-Level Actionability

Two annotators participated in the error-level SIREN annotation described in Section 4.2. Annotator 1 is a native speaker of British English with C1 proficiency in Italian and a background in modern languages and law. Annotator 2 is a native speaker of Italian with C1 proficiency in English and a background in international cooperation.

A.2 Study 2: Message-Level Actionability

A separate pair of annotators participated in the message-level SIREN annotation described in Section 4.3. Annotator 1 is a native speaker of French with a background in modern languages and translation. Annotator 2 is a native speaker of Italian with C1 proficiency in French and a background in translation and crisis communication.

B SIREN Annotation Decision Tree

Figure 3 provides the complete decision procedure used for SIREN-span annotation.

C Qualitative Examples of SIREN Annotations

Tables 7 and 8 provide representative examples from the Italian–English and English–French experiments, respectively, illustrating how different translation choices can affect the functional dimensions captured by SIREN.

D Translation Prompt

The following prompt was used to generate the English–French machine translations evaluated in Study 2. The source emergency message was inserted in the final placeholder.

```
You are a professional English (en) to French (fr) translator with experience translating crisis-related content.
Your goal is to accurately convey the meaning and nuances of the original English text while adhering to French grammar, vocabulary, and cultural sensitivities.
Produce only the French translation, without any additional explanations or commentary.

Please translate the following English text into French:
{source_text}
```

E Pilot Study IAA Scores

Table 9 reports inter-annotator agreement (IAA) scores across the three pilot annotation batches for both the Italian–English and English–French experiments.

F Actionability Loss by HOPE Error Type

Table 10 reports, for each system in the Italian–English rollout, how many errors for each HOPE type were judged to compromise actionability (YES).

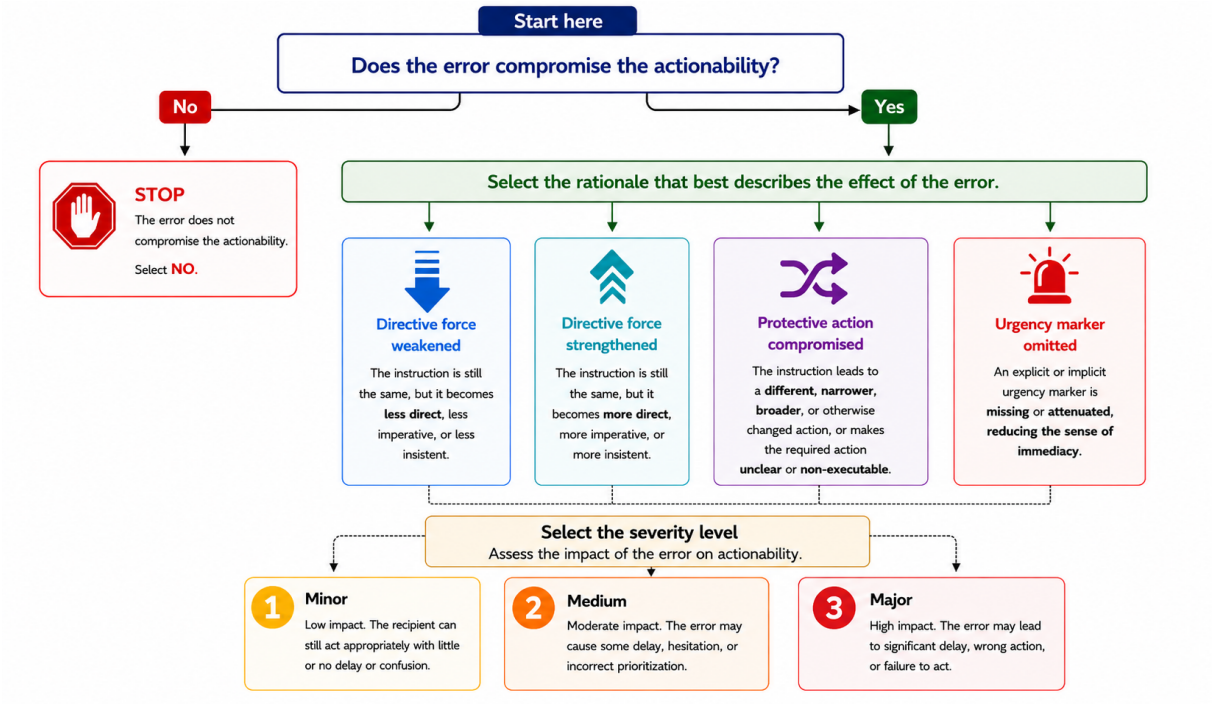


Figure 3: SIREN-span annotation decision tree. Annotators first determine whether an existing translation error compromises actionability. For actionability-compromising errors, they identify the functional rationale and subsequently assign an actionability impact level (MINOR, MEDIUM, or MAJOR).

System	Source	MT Output	SIREN Rationale	Impact
ChatGPT-4o	Non transitare lungo strade allagate: potrebbero esserci voragini, buche, tombini aperti o cavi elettrici tranciati.	**Do not walk along** flooded streets: there may be voids, holes, open manholes, or cut electrical wires.	Protective action compromised	Major
ChatGPT-4o	Prima di uscire di casa chiudi gas, acqua e luce e indossa le scarpe.	Before leaving your home, turn off the gas, water, and electricity, and put on **sturdy shoes**.	Directive force strengthened	Medium
Google Translate	Usa il telefono solo per reale necessità.	Use your phone only for **real needs**.	Directive force weakened	Medium
Google Translate	Informati sulle vie di allontanamento da seguire previste dalle pianificazioni comunale e nazionale.	Find out about the **distancing routes** to follow provided for by municipal and national plans.	Protective action compromised	Medium

Table 7: Representative SIREN-span annotations from the Italian–English experiment. Boldface highlights translation choices relevant to the actionability judgement.

System	Source	MT Output	SIREN Rationale	Impact
TranslateGemma-4B	Inform yourself about your municipality's Civil Protection plan.	**Vous informer** sur le plan de protection civile de votre commune.	Directive force weakened	Minor
TranslateGemma-4B	As a general indication, indoor sheltering may last up to 48 hours.	Pour information générale, **la recherche d'abri en intérieur** peut durer jusqu'à 48 heures.	Protective action compromised	Major
Google Translate	If you live in the intervention area, check that your home is safe and take all necessary steps to eliminate situations that can represent a danger in case of an earthquake.	Si vous habitez dans la zone d'intervention, **vérifiez la sécurité de votre logement** et prenez toutes les dispositions nécessaires pour éliminer les situations pouvant présenter un danger en cas de séisme.	Protective action compromised	Medium
Google Translate	Don't use the car, it could become a trap.	N'utilisez pas votre voiture ; elle pourrait **se transformer en piège**.	Directive force weakened	Minor

Table 8: Representative SIREN-msg annotations from the English–French experiment. Examples are balanced across TranslateGemma-4B and Google Translate. Boldface highlights translation choices relevant to the actionability judgement.

Batch	System	N	Agreement	κ
1	ChatGPT-4o	14	57.14%	−0.024
	Google Translate	9	88.89%	0.769
2	ChatGPT-4o	30	90.00%	0.615
	Google Translate	30	76.67%	0.516
3	ChatGPT-4o	35	85.71%	0.684
	Google Translate	33	90.91%	0.812

Italian–English. N denotes the number of HOPE error instances evaluated with SIREN.

Batch	System	N	Agreement	κ
1	TranslateGemma-4B	13	76.92%	0.316
	Google Translate	13	92.31%	0.629
2	TranslateGemma-4B	24	91.67%	0.480
	Google Translate	24	100.00%	–
3	TranslateGemma-4B	51	92.16%	0.769
	Google Translate	51	96.08%	0.811

English–French. N denotes the number of messages evaluated with SIREN.

Table 9: Inter-annotator agreement across the three pilot annotation batches for the Italian–English and English–French experiments.

		ChatGPT-4o		Google Translate	
Code	HOPE error type	N	YES (%)	N	YES (%)
IMP	Impact	16	5 (31.25)	18	11 (61.11)
TRM	Terminology	40	12 (30.00)	33	17 (51.52)
UGR	Ungrammatical	4	1 (25.00)	12	5 (41.67)
MIS	Mistranslation	23	9 (39.13)	29	14 (48.28)
STL	Style	84	25 (29.76)	87	38 (43.68)
PRF	Proofreading	29	10 (34.48)	18	11 (61.11)
PRN	Proper Name	0	–	0	–
Total		**196**	**62 (31.63)**	**197**	**96 (48.73)**

Table 10: Actionability-compromising errors by HOPE error type in the Italian–English rollout. N indicates the total number of errors in each category. YES indicates the number of errors judged to compromise actionability, with the percentage relative to N in parentheses.

Communicating with the Entire Population: Machine Translation of Emergency Alerts and Warnings

Jen Wilson*, **Vipasha Bansal**, **William D. Lewis**
Computational Linguistics Program
University of Washington

Abstract

It is essential that we communicate with the entirety of an affected population during a crisis, including those who do not speak a region's dominant language(s). Commercial Translation tools offer a tempting solution to broaden access and can provide a lifeline when no other resources are available. However, Emergency response managers often operate in the dark regarding the quality and adequacy of these automated outputs. In this paper, we conduct a nationwide survey of emergency managers to understand their current workflows, perspectives, and operational challenges. We then evaluate commercial translation tools against these needs, shedding light on their utility and offering practical guidance on how and when these tools can be safely used in emergency response.

1 Introduction

Communication during emergencies is a core part of public safety. Access to timely information during a crisis is as essential as access to "food, water, shelter, and medication" (Meier, 2015). Emergency alerts and warnings systems, such as Wireless Emergency Alerts (WEA)[1]—must reach the *entirety* of an affected population, rather than an English-fluent subset. Failing to achieve equitable coverage puts lives directly at risk.

This outreach is difficult in practice. Local emergency managers, public information managers, and communication staff often serve multilingual communities while working with limited time, staffing and budgets, sometimes across dozens or even hundreds of languages (Ciribuco et al., 2025). In regions where alerts are distributed only in the dominant language(s), non-fluent residents are inevitably excluded, which can have devastating consequences in time-sensitive crises (Villarreal et al., 2025; Pabón et al., 2025; O'Brien and Federici, 2019).

Because human translation is often cost-prohibitive or too slow for local jurisdictions, practitioners increasingly turn to automated solutions. Commercial Machine Translation (MT) systems, such as Google Translate[2] and Microsoft Translator[3], are widely available and support a large number of languages. Large Language Models (LLMs) such as ChatGPT[4], Claude[5], and Google Gemini[6] are also increasingly used for translation.

These tools may offer speed and convenience, but emergency messaging demands accuracy and usability. Inaccurate translations risk delivering alerts that are uninterpretable or dangerously misleading. Crucially, strong MT performance in one domain or language pair does not guarantee cross-lingual or cross-domain reliability (Bogoychev and Chen, 2021; Zouhar et al., 2024), requiring comprehensive multi-target evaluation to ensure safety.

Although this paper focuses on emergency information translation in the United States, the challenges we describe are not unique to any one country and are likely familiar to officials in many multilingual settings. As claims regarding Artificial Intelligence (AI)[7] expand across crisis management, practitioners face a rapidly evolving landscape of software tools and vendor claims (Foubert et al., 2026; Whyte and Haupt, 2026). In this paper, we seek to shed light on automated translation in emergency response from two angles.[8] First, we survey emergency managers across the United States to

*Corresponding author: jenwils@uw.edu

[1]https://www.fema.gov/emergency-managers/practitioners/integrated-public-alert-warning-system/public/wireless-emergency-alerts

[2]https://translate.google.com/

[3]https://www.microsoft.com/en-us/translator/

[4]https://chatgpt.com/

[5]https://claude.com/

[6]https://gemini.google.com/app

[7]Machine Translation being one of many forms of AI.

[8]In this work, we chose to focus on the quality and utility of commercially available tools that do translation. We leave the analysis of MT workflows, logistics of their implementation and use in the field, etc., to others.

Proceedings of the First Workshop on Language Technologies for Crisis Preparedness and Response, pages 46–64
November 9, 2026 ©2026 Association for Computational Linguistics

understand current translation practices, needs, and challenges.

Second, we examine the translation tools available to practitioners and review existing research on their evaluation and use in crisis settings. By bringing together the perspectives of emergency management practitioners and research on translation technologies, we hope to help bridge the gap between the development and evaluation of these tools and the contexts in which they are used.

2 Wireless Emergency Alerts

In the United States, the Wireless Emergency Alerts (WEA)[1] system broadcasts alerts to mobile and other WEA-enabled devices, providing actionable information in times of crisis. However, native support is restricted to English, leaving non-English speakers at a disadvantage unless they proactively sign up for translations.

In 2023, the US Federal Communications Commission (FCC, 2023) recognized that "it is essential that the public be able to receive in accessible language and format WEA Messages that are intended for them...to help the millions of people with access and functional needs, including people who primarily speak a language other than English or Spanish...better understand and take protective actions." Despite this, the ability of agencies to meet multilingual and accessible alerting needs varies widely depending on local demographics, staffing, funding, and available translation infrastructure.

3 Establishing the Need: A Survey of Emergency Managers

The role of language technology in emergency management is underexplored and deserves attention. Foubert et al. (2026) surveyed emergency managers to identify general perceptions and potential use cases for emerging AI. The specific adoption and efficacy of automated translation tools in crisis contexts require dedicated study.

To examine current translation practices, needs, and operational hurdles, we surveyed members of the Language & Accessibility for Alert & Warning Workgroup (LAAWW)[9] in December 2025. LAAWW members represent 42 U.S. states, territories, and Tribal Nations who are "working toward greater language and disability access in emergency alert and warning systems."

We used purposive sampling strategy to reach practitioners actively involved in language access.[10] Therefore, this survey is not intended to be statistically representative of all U.S. emergency management agencies; rather, it provides insight into the tools, decision-making processes, and challenges involved in producing multilingual alerts. Survey details are in Appendix F.

The survey asked about respondents' roles and responsibilities, languages spoken in their communities, translation tools and practices, and challenges encountered in multilingual communication. We received 25 responses, approximately 25% of meeting attendees.[11]

More than half of respondents identified themselves as a *Local Emergency Manager* or *Public Information Officer*, while 36% identified as an *Operations, Logistics, or Communications Manager* or *Language Coordinator or Interpreter*.[12]

Participants noted that while many jurisdictions have residents who speak a variety of languages, only a subset of these languages are consistently supported for time-critical alerts. Agencies must prioritize languages while balancing accuracy, usability, speed, and cost. Translations should be usable and accurate, while also affordably and quickly produced.

Although participants prioritized quality translations, for those who shared their challenges, 100% cited money, time, and/or limited resources.

3.1 Translation Needs Across LAAWW Membership

Translation needs vary substantially across jurisdictions represented in the survey. Respondents reported serving communities where between two and more than 200 languages are spoken. English was the primary language across all reported jurisdictions and Spanish was the most consistently supported additional language. However, the remaining language needs varied considerably by region. A partial list of languages that survey participants work with is available in Appendix D.

The scale of these differences creates distinct operational requirements. Smaller jurisdictions may

[9] https://sites.google.com/view/laaww

[10] This approach surfaces qualitative operational insights; future work with randomized sampling would be required for broader statistical generalization.

[11] Attendance at LAAWW meetings is approximately 100 people per month, representing approximately 80 unique entities (agencies, vendors, cities, counties, etc).

[12] The survey allowed respondents to select more than one role.

be able to reach most residents by translating alerts into only a few languages, while larger and more linguistically diverse jurisdictions may need to support many more. For example, one respondent from Maryland supports a community of approximately 5,500 people and three commonly spoken languages. Another works in New York City, where more than 200 languages are spoken across a population of more than 8.5 million. While we can look at average language populations, nearly every area has populations that deviate from that national average. These jurisdictions face fundamentally different translation-planning challenges.[13] Survey respondents support over 36 million people across the United States[14].

Agencies must identify the languages most important for their communities and develop processes for producing translations that are accurate, usable, affordable, and fast enough for time-critical alerts. The central need is not simply translation into more languages, but scalable and locally informed translation capacity. For more details on translation needs and speaker populations, as reported by the LAAWW membership, please see Appendix C.

3.2 Translation Tools and Practices

The majority of respondents (55%) relied exclusively on human translation and did not use automated tools. Another 27% used a mix of human translators and automated tools, while 18% relied exclusively on automated tools (Figure 1). For those using human translation, work was performed by organization staff (45%), community members or volunteers (26%), and contractors or commercial vendors (29%). These results show that jurisdictions use a range of staffing models to produce multilingual alerts.

Among respondents using translation-specific commercial tools, Google Translate[2] was the choice in 90% of cases, followed by DeepL Translator[15] at 20%.[16] A minority (and we suspect growing) number of respondents use LLM–based

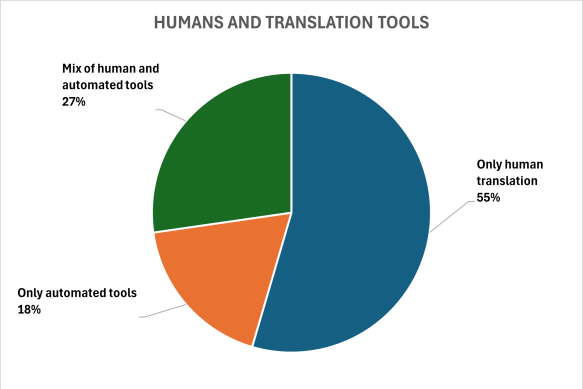


Figure 1: Human Translation and Translation Tools

tools, with ChatGPT[4] used in 30% of cases and Google Gemini[6] used in 10% of cases.[16] Given the small sample size, reported use of LLM-based tools likely reflects exploratory or supplemental use rather than standardized, organization-wide deployment. These results are consistent with broader interest in AI among emergency managers. Foubert et al. (2026) found that 75% of their respondents were interested in using AI for public and stakeholder messaging, including multilingual translations. Together, these findings suggest that automated tools are already part of some emergency communication workflows, while their organizational adoption remains uneven.

3.3 Factors Influencing Tool Selection

Respondents selected translation tools primarily on the basis of practical considerations rather than formal requirements. Availability or ease of access was a criteria for 56% of respondents, and perceived accuracy or quality was important for 52%. Cost or budget constraints influenced 40% of respondents' choices, while ease of use or staff comfort influenced 32%. These results suggest that agencies favor tools that are accessible, affordable, and sufficiently reliable for everyday workflows.

Compliance requirements (16%) and "other" factors (12%) were cited less often, indicating that regulatory or external mandates play a smaller role in shaping tool choices.

Tool adoption therefore appears to be driven more by immediacy and feasibility than by formal risk assessment or regulatory guidance (Figure 2). This pattern is consequential because a tool that is convenient to access may still require systematic evaluation before it is used for high-stakes messages

[13] Appendix D displays this data as percentage off expected.

[14] The population information reported here reflects the total population of respondents' jurisdictions and do not imply that all residents require translated alerts or that language services are uniformly provided across all languages in each jurisdiction. Rather, they illustrate the scale and diversity of contexts in which respondents operate.

[15] https://www.deepl.com/en/translator

[16] Respondents could specify more than one tool and the percentages do not necessarily add to 100%.

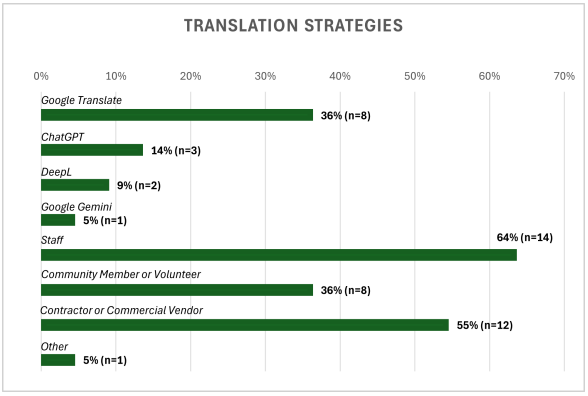


Figure 2: Strategies of respondents to obtain translations.

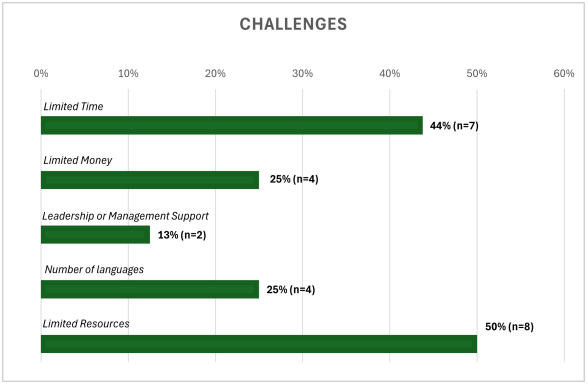


Figure 3: Operational challenges faced by survey respondents.

3.4 Evaluation and Review Practices

Formal evaluation of translation tool output was uncommon. Only 10% of respondents reported having a formal process for assessing accuracy, usability, or effectiveness. Sixty percent used an informal evaluation process, while 30% reported having no evaluation process.

Processes for reviewing or updating tools were similarly inconsistent. Forty percent of respondents reported reviewing or updating tools as needed, 25% reported having no formal review process and another 25% were unsure whether such a process existed. The remaining 10% reported that their tools were not reviewed at all. Thus, many jurisdictions may be using translation tools in high-stakes contexts without systematic procedures for evaluating or monitoring their performance, creating a risk of uneven message quality across languages.

3.5 Operational Challenges

Respondents described multiple barriers to producing multilingual alerts, including limited time, funding, staff, and management support.

Translator time and availability were also recurring concerns. Participant L1 emphasized the need to "minimiz[e] time delays in alerts and warnings...especially with translations into various languages." Participant P1 similarly described the "challenge [with] human translation is always time and availability" of translators.

Participant H1 reported that their state maintained a registry of approved interpreters, but very few were available in their rural county. As linguistic diversity increases, agencies may need to support additional languages even when human translators and external vendors are slow or unavailable.

Respondents also reported that the usefulness of translation tools varied by language. Participant R1 explained "multiple languages in the state are very unique thus singular translation services for only a few languages [have] proven ineffective in [the] past." This aligns with our CHRF++ findings (Section 5), which showed significant variation across languages and systems

Beyond operational barriers, participants also noted that language access does not guarantee message comprehension, particularly for populations with limited literacy or differing communication preferences. These constraints show a deeper challenge: language access alone is insufficient. The findings point to the need for complementary accessibility strategies, such as plain-language messaging, audio or pictorial alerts, and community-based dissemination, especially in high-risk populations (Villarreal et al., 2025).

4 Differences between MT and LLM use by Emergency Managers

Our survey demonstrated that respondents in emergency management and communications use both MT and LLM-based tools. There are many differences between the two technologies, but for the purposes of this paper, we will discuss the expected outputs and the types of data used in the training of Google Translate and OpenAI's ChatGPT.[17]

MT is traditionally trained on a corpus of aligned sentence pairs (Bapna et al., 2022), where the sentences corresponding to one another in each language have the same meaning. The MT models learn how text in one language corresponds

[17] Discussion of other similarities and differences, such as model architecture, training objectives, training process, inference mechanisms, etc. is beyond the scope of this paper.

to equivalent text in another language. Commercial tools, such as Google Translate and Microsoft Translator, are trained in this manner and publish lists of languages that are supported by their services.[18]

While LLMs such as ChatGPT are pre-trained on publicly available data (websites, news, etc.), multimodal data, non-English data, and data licensed for training (OpenAI, 2024), the specific data sources are rarely disclosed. The data may also include aligned sentence pairs and text in multiple languages (Chua et al., 2025). However, unlike the MT services discussed in the previous paragraph, LLMs generally do not list what languages they support for MT. In fact, MT may arise as an emergent property rather than as a training goal (Wei et al., 2022). LLMs are trained to generate plausible output, and will provide output even when lacking sufficient data to provide one that is correct. This can lead "models to convert uncertainty into content that is often plausible but wrong" (Kalai et al., 2026).

5 Quality of Commercial MT Tools in Crisis Response

The survey of LAAWW membership shows a clear interest in automated tools for translation. Several studies have evaluated the use and performance of MT in crisis contexts, such as for emergencies (Turner et al., 2019), multilingual crisis-related communication (Rossetti et al., 2020), or pandemic-related communication across multiple languages (Anastasopoulos et al., 2020). More recent work has evaluated both MT systems and LLMs on emergency-response texts and multilingual crisis data, finding limitations in areas including domain-specific terminology, contextual ambiguity, register, and the preservation of urgency (Staiano et al., 2025; Ticona and Anastasopoulos, 2026; Court et al., 2026). Less attention has been given to systematically comparing currently available commercial translation tools on emergency alert content across multiple languages. This distinction is important for emergency managers, who are more likely to encounter translation tools as commercially available services than as models requiring specialized technical expertise to use or adapt.

[18]For a list of languages supported by Google Translate, see: https://docs.cloud.google.com/translate/docs/languages. For Microsoft Translator, see: https://www.microsoft.com/en-us/translator/languages/.

5.1 Quantitative Evaluation

Bansal et al. (2026) evaluated seven commercially available MT systems on the TICO-19 pandemic-related benchmark dataset (Anastasopoulos et al., 2020), which contains English-language COVID-19 material translated into 36 languages, and assessed translation usability based on whether the output was sufficiently understandable and adequately conveyed the relevant information. Their results demonstrate substantial variation in translation performance across languages and systems, particularly for lower-resource languages.

Pang et al. (2025) find that LLMs trained across multiple domains perform consistently across a range of tasks but can fall short of domain-specific models. To investigate whether this pattern extends to crisis communication, we compare scores on the full TICO-19 dataset with scores on its News and Announcements (N+A) subset, whose short, information-rich texts more closely resemble Wireless Emergency Alerts (WEAs). Because WEAs are issued in English, we focus on English-to-target-language (EX) translation scores.

LAAWW members identified 24 supported languages that overlap with the languages represented in TICO-19. This overlap enables a unique domain-specific evaluation of emergency-relevant text for those languages, allowing a comparison with general-purpose content. Following Anastasopoulos et al.'s (2020) categorization, we group the languages into three tiers: *Pivot* (high-resource), *Important* (mid-resource), and *Priority* (low-resource).

We examine CHRF++ scores for both the overall and N+A subset of TICO-19. CHRF++ is a quality metric widely used in the MT field for measuring quality (Popović, 2017). The differences between these scores for some language-system combinations, suggests that performance on a broader evaluation set does not necessarily predict performance on text with characteristics more similar to emergency alerts.

Pivot languages generally achieved the strongest Overall scores, while Important and Priority languages showed lower and more variable performance. No system was consistently strongest across languages, and the best-performing system sometimes changed between the Overall and N+A evaluations. These scores suggest that overall benchmark scores are insufficient for predicting performance on emergency-relevant text and that

evaluation should account for both language and domain.[19]

For example, Microsoft's Hindi N+A CHRF++ score was 13.16 points *lower* than its Overall score, Claude was 12.24 points lower, and GPT-5.4-mini was 10.33 points lower. In fact, all of the Pivot languages showed lower scores for the N+A subset, except for Swahili, for which all services were between 1.54 and 3.13 *higher* (Appendix G, Table 3). For most language–system pairs, CHRF++ scores were higher on Overall than on the N+A subset, with particularly large differences for Hindi and Arabic. However, this pattern was not universal: Bengali, Khmer, Kinyarwanda, Swahili, Somali, Tamil, and Zulu had lower Overall than N+A scores for at least one system.

5.2 Qualitative Evaluation

Qualitative analyses by Hajek et al. (2024) identify errors that may sound fluent, but substantially alter the intended meaning of emergency instructions. For example, the English instruction "Try to position the car *towards* the approaching fire" was translated to Chinese and then back into English as "Try to park your car *near* the fire," reversing the intended instruction and potentially creating a very serious safety risk.[20]

Hajek et al. (2024) also describe problems with lexical ambiguity. The instruction "shelter in a room" was translated into Spanish as "refugio en una habitación," which when backtranslated, means "a refuge in a room." The word *shelter* in this case was translated as a noun (*refugio*), when actually it was a verb (a command to take shelter). It may be tempting to assume that *something* is better than *nothing* when it comes to emergency messaging. However, these examples demonstrate that that is not always the case.

These examples illustrate a type of MT failure that is dangerous in emergency contexts: mistranslations that are fluent, possibly plausible-sounding are therefore unlikely to be flagged by non-expert reviewers, but convey meaningfully different instructions (Šorak, 2024).

Additionally, Ticona and Anastasopoulos (2026) demonstrate that LLMs inconsistently translate urgency, such as "a gas leak scenario classified as Critical in English is downgraded to Low in Marathi and Not Urgent in Amharic."

Qualitative evaluations provide examples that show where MT has gone wrong and can be useful in providing actionable content that can help guide how translations are done, used, or evaluated. However, they do not give an overall picture of MT output but only specific instances identified by human translators and may be, or may not be, representative of broader issues. This limitation underscores the need for domain-specific and systematic evaluation frameworks for MT in emergency settings.

6 What to Do and What Not to Do with Commercial MT Tools

The gold standard for translations of emergency alerts and messaging is to translate all anticipated messages by a human before they are needed (Federici, 2022; Villarreal et al., 2025). When an organization has limited resources, or an unanticipated crisis occurs, or where professional translators are not available, this process may not be possible. Commercial machine translation can bridge the gap and get information out to people quickly, while buying time to have a more thorough translation prepared.

Based on work by Ticona and Anastasopoulos (2026) and Bansal et al. (2026), we can see that there are gaps in the quality of commercial and open-source MT tools in the crisis domain cross-lingually, and potentially undesirable errors in output, such as those noted by Hajek et al. (2024). Knowing the quality gaps can help emergency managers integrate translation tools into their workflows appropriately. It is important to know how one can improve the quality of MT and how much a human should be in the loop. In this section, we will look at ideas and rules of thumb that emergency managers can adopt for using such tools.

6.1 The Epic Highs and Lows of Raw Machine Translation

As MT quality has improved, especially for high resource languages, the use of raw MT has become more prevalent, especially in contexts where human translators are not available (Nurminen, 2021; Nurminen and Papula, 2018; Kasperė et al., 2021; Vieira et al., 2022; Kenny et al., 2022)—a constraint reflecting the standard operating environment of many LAAWW practitioners.

[19] See Appendix G, Table 4 for full scores.

[20] They did not note which system produced this translation. Hajek et al. (2024) gathered their translations in 2023. Upon subsequently examining Google Translate (captured on 14 Jan, 2026), the Chinese translation is 着将汽向逼近的火焰。

However, there is a significant literature on the use of MT in crises, with the general recommendation is that humans should be in the loop to prevent severe errors with potentially catastrophic consequences[21] (Cadwell et al., 2019; O'Brien and Federici, 2019; Federici and Cadwell, 2018; Federici, 2022). Reflecting these risks, many Language Access Policies discourage the use of raw MT, *i.e.,* using the direct output of MT engines without any post-editing or human-in-the-loop (City of LA, 2022; U.S. GSA, 2026; US DOJ LEP Cmte, 2021).

Strictly avoiding raw MT may not be feasible when contending with budget limits, as highlighted by our survey respondents. But there are things that can be done to help mitigate some of these dangers, either through simplifying text, preparing messages in advance, use of templates, or doing as much as possible in advance of crises.

6.2 Translated Templates

The Federal Communication Commission (FCC) provides templates for WEAs, translated into the 13 most commonly spoken languages in the United States.[22] By using templates like these, one can sidestep the use of MT altogether in certain crisis scenarios.

We recommend creating or repuposing templates in advance of crises that may occur regularly. Use the following steps to create templates that can be repurposed:

1. **Pre-approval:** Draft templates, using short, plain-language messages for hazards or actions.

2. **Placeholder management:** Use clearly labeled and unambiguous placeholders in the templates for event specific details (*e.g.,* location and place names, dates, times, etc.). Ensure the placeholders are untranslated, and have the correct placement for the target languages.

3. **Rapid deployment:** When an incident occurs and a message template has already been prepared, it can be filled in and translated to additional languages through automated tools.

6.3 Text Simplification

Text simplification has long been used to improve MT output, and is discussed in the literature under various guises: text simplification (Štajner and Popovic, 2016), controlled language (Aikawa et al., 2007), and pre-editing (Sánchez-Gijón, 2022), among others. In the distributed messages setting, such as with WEAs, all messages should be as clear and direct as possible to increase the chance of information being translated accurately.

Štajner and Popovic (2016) find that text simplification can improve the fluency of the MT output. While their work focuses on automated simplification, the same principle can be extended to include manual simplification techniques, *i.e.,* where the authors of the messages simplify the text themselves. Hajek et al. (2024), for example, recommend avoiding culture-specific terms, avoiding two-part verbs (*e.g.,* look up, get along with, etc.), and avoiding elements with multiple grammatical functions. Recommendations to simplify crisis communications before translation have also been made by local governments (King County Executive, 2010), policy reports (Fordham, 2023), and government reports (Hajek et al., 2024).

LLMs can also be used in text simplification (Guidroz et al., 2025). For example, the following text could be used as a prompt to an LLM to simplify text supplied by the user:

Rewrite the following text in plain language at approximately a 6th-grade reading level. Use short sentences and everyday vocabulary. Keep all safety instructions, locations, and times accurate.

The text simplification process should be treated as a separate step from translation. Evaluating the simplified text before translation can help confirm that the source message remains accurate, retains all essential information, and is adequately simplified and ready for the next stage. Separating these steps makes it possible to identify and correct problems with the source message before it is disseminated to the public.

6.4 Using LLMs for Translations

The majority of LAAWW members use human translation (Figure 1), but respondents also indicated interest in translating with LLMs. The use of general-purpose LLMs, such as GPT and Gemini, in high-stakes communication contexts introduces risks that go beyond translation quality as measured by evaluation metrics. LLMs are known to produce fluent but semantically incorrect outputs — a phenomenon called "hallucination" — at rates that are difficult to predict and vary by language, domain,

[21] We discuss notable examples in Section 5.2.

[22] https://www.fcc.gov/multilingual-wireless-emergency-alerts hosts example templates.

and prompt formulation (Ji et al., 2023). In crisis communication, this unpredictability is a serious concern.

To reduce the risk of hallucinations, we recommend using a prompt such as the one below, which instructs the LLM to leave text untranslated when it is uncertain. Although this prompt may lower the likelihood of hallucinations, it is not guaranteed. For languages not well supported by the model, LLM-based translation should be avoided. See Appendix G for additional guidance.

Translate this message into [TARGET LANGUAGE]. Be literal. Do not add, omit or invent information. If a phrase is ambiguous, mark it as [UNCERTAIN] and leave it untranslated. Preserve names, dates, and numbers exactly. After the translation, provide a one-line confidence: High / Medium / Low and list any parts you were unsure about.

The authors acknowledge the mixed messaging this section provides. While we suggest that LLMs not be used for translations, we also give guidance on how to use LLMs for translations. As researchers, we are hesitant to recommend the use of LLMs for the reasons stated above. As people who have some understanding of the community that uses these tools, we know that LLMs may be used for translations regardless of what an academic paper may recommend. To that end, we also recommend strategies to use LLMs as effectively as possible.

The models used in commercial LLMs can change with little notice to end users, making it difficult to track how system behavior evolves over time. Because these system updates may be undocumented or only partially disclosed, a prompt or output that is usable at one point in time may produce different results later, even when the input remains unchanged. For emergency management practitioners who use LLMs for translation, these changes can be especially consequential. Small shifts in wording, terminology, or tone may affect whether a message is interpreted accurately and in a way that supports timely response. This potential inconsistency in output should be considered if choosing to use an LLM instead of a commerial MT tool, such as Google Translate or DeepL.

6.5 Post-Editing and Usage

It is challenging for organizations to determine how to best allocate limited resources. One option for translation, beyond the methods discussed previously (*e.g.,* using templates, simplification, etc.) is to spend resources on post-editing rather than on message creation and translation. In this scenario, human translators rely on MT to generate a translated output, and then subsequently correct any errors in the output. This allows translators to spend their time reviewing and editing translated messages rather than translating messages from scratch. The use of MT in workflows has been shown to reduce costs.

To make messages broadly consumable, in addition to the steps outlined in Sections 6.2 and 6.3, we recommend that the messages are shared on available public and social platforms (Purohit et al., 2025). Some platforms provide built-in translation features, allowing users to translate messages without copying the text into a separate program or application. However, this approach removes the ability to monitor quality of the translations and increases the risk that people will get misleading, incorrect, or harmful information. It will be up to the user to determine whether this will work for their situation.

MT engines can produce errors, some of which can have catastrophic consequences, such as examples from Hajek et al. (2024). Although quality evaluation is certainly possible, it is useful to determine *adequacy* for the specific need versus absolute *quality*. In other words, a translation that has some limited errors or is ungrammatical may still be of sufficient quality and good enough to be used. For suggestions on how to determine the adequacy of translations, see Appendix E.

Ad hoc translations likely occur regularly during crisis situations, given the ubiquity of translation tools and people's familiarity with them. In urgent situations, messages are often issued first in the dominant language(s) and translated for dissemination in other languages later. For those who do not speak the dominant language, urgency will drive them to quickly obtain their own translations, using some of the same commercial translation tools described in this paper. In some cases, a translated message may not be disseminated in a specific language at all, forcing recipients to translate the message themselves. Organizations should therefore consider not only the quality of translations *they* produce, but also the translations that recipients are likely to obtain independently when official translations are unavailable.

7 Conclusion

We recommend having humans in the loop in translating, editing, and evaluation whenever resources permit. In practice, however, jurisdictions that communicate in dozens of languages may not be able to review each message by hand. Automated workflows can expand translation capacity, but should be designed around the risks of the message, language, and tool rather than treated as a universal replacement for human review.

An important point to make is that MT engines will make errors, and not all errors are created equally. The types and severity of errors can vary greatly depending on the content, *i.e.,* the complexity of the original text or the amount of resources for the target language. As we have shown, MT tends to do better with higher-resource languages such as Indonesian or Spanish than for lower-resource languages such as Amharic or Pashto. Performance on the N+A subset did not always follow performance on the broader evaluation. These differences make language- and domain-specific evaluation essential.

In a world where communities are becoming ever more multilingual, translation will remain an important component of emergency and crisis communication. Fortunately, the quality and utility of MT is improving, but its use should be paired with realistic expectations, along with human oversight. In crisis settings, the goal is not simply to produce a translation, but rather to deliver information that affected communities can understand and act on safely.

Limitations

The survey done for this paper captures the languages that US-based agencies *report* that they need to support. It does not systematically compare those responses against actual distribution of non-English speakers in the respondents' jurisdictions as measured by US Census data or available American Community Survey data.

Ethical Considerations

Considerable care was taken in designing and administering the survey to ensure that the questions were appropriate to the study objectives, respectful of participants, and attentive to the needs and perspectives of the communities being studied. The survey was developed after careful consideration of the roles and responsibilities of LAAWW members. The methodologies and goals of the study were presented during a LAAWW meeting and members were able to ask questions. This process helped ensure that the survey questions, framing, and proposed use of the data were clearly understood and acceptable to the participating community.

The study was also submitted for review through the University of Washington (Seattle) Human Subjects Division Institutional Review Board (IRB). The IRB determined that the proposed activity constituted human subjects research and qualified for exempt status. Data collection proceeded in accordance with the requirements and conditions of that determination.

Participation in the survey was designed to protect participants' privacy. Participants were able to respond anonymously and were not required to provide identifying information. Participants were given the option to voluntarily provide contact information and geographic location if they wished to do so. These fields were optional and were not required for participation. The study therefore collected only information that participants chose to provide, with identifying information separated from the anonymous survey responses as appropriate for the purposes of analysis and reporting.

Taken together, these steps were intended to ensure that the survey process was conducted thoughtfully and responsibly, from initial community consultation through IRB review, data collection, privacy protection, and subsequent analysis.

Acknowledgments

This work was partially supported by the National Science Foundation under the award CIRC 2346335. We would like to thank the two LT4CPR workshop reviewers for their very helpful comments and suggestions, all of which helped improve this paper. We also extend our warmest THANK YOU to the members of LAWWW, whose contributions to the survey, including their suggestions, questions, and support, made this paper what it is today. Without LAWWW, this paper could not have been written.

References

Takako Aikawa, Lee Schwartz, Ronit King, Mo Corson-Oliver, and Carmen Lozano. 2007. Impact of Controlled Language on Translation Quality and Post-editing in a Statistical Machine Translation Environ-

ment. In *Proceedings of the Machine Translation Summit XI*.

Antonios Anastasopoulos, Alessandro Cattelan, Zi-Yi Dou, Marcello Federico, Christian Federmann, Dmitriy Genzel, Franscisco Guzmán, Junjie Hu, Macduff Hughes, Philipp Koehn, Rosie Lazar, Will Lewis, Graham Neubig, Mengmeng Niu, Alp Öktem, Eric Paquin, Grace Tang, and Sylwia Tur. 2020. TICO-19: the translation initiative for COvid-19. In *Proceedings of the 1st Workshop on NLP for COVID-19 (Part 2) at EMNLP 2020*, Online. Association for Computational Linguistics.

Vipasha Bansal, William D. Lewis, Nicole Mihaylova, Elizabeth Brown, and Benjamin Pong. 2026. Is MT Ready for the Next Crisis or Pandemic? In *Proceedings of the Language Technologies for Crisis Preparedness and Response (LT4CPR Workshop)*, Hengqin, Zhuhai, China. Association for Computational Linguistics.

Ankur Bapna, Isaac Caswell, Julia Kreutzer, Orhan Firat, Daan van Esch, Aditya Siddhant, Mengmeng Niu, Pallavi Baljekar, Xavier Garcia, Wolfgang Macherey, Theresa Breiner, Vera Axelrod, Jason Riesa, Yuan Cao, Mia Xu Chen, Klaus Macherey, Maxim Krikun, Pidong Wang, Alexander Gutkin, and 5 others. 2022. Building machine translation systems for the next thousand languages. *Preprint*, arXiv:2205.03983.

Nikolay Bogoychev and Pinzhen Chen. 2021. The highs and lows of simple lexical domain adaptation approaches for neural machine translation. In *Proceedings of the Second Workshop on Insights from Negative Results in NLP*, pages 74–80, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Patrick Cadwell, Sharon O'Brien, and Eric DeLuca. 2019. More than tweets: A critical reflection on developing and testing crisis machine translation technology. *Translation Spaces*, 8(2):300–333.

Lynn Chua, Badih Ghazi, Yangsibo Huang, Pritish Kamath, Ravi Kumar, Pasin Manurangsi, Amer Sinha, Chulin Xie, and Chiyuan Zhang. 2025. Crosslingual capabilities and knowledge barriers in multilingual large language models. *Preprint*, arXiv:2406.16135.

Andrea Ciribuco, Federico M. Federici, Lorenzo Guadagno, Brigid Maher, Loredana Polezzi, and Rita Wilson. 2025. *Migrants, multilingual communication and cascading crises*, volume 1, pages 87–102. Routledge, Abingdon, Oxfordshire, UK.

City of LA. 2022. Citywide language access plan fy 22-24. https://communityinvestment.lacity.gov/sites/g/files/wph2001/files/2025-04/Citywide%20Language%20Access%20Plan%20FY%2022-24%2C%20City%20of%20Los%20Angeles%20%281%29.pdf. [Accessed 16 Jul 2026].

Sara Court, Lara Downing, and Micha Elsner. 2026. LLMs in the real world: Evaluating "AI" in emergency contexts. In *Findings of the Association for Computational Linguistics: ACL 2026*, pages 35748–35760, San Diego, California, United States. Association for Computational Linguistics.

FCC. 2023. Emergency Alert System; Wireless Emergency Alerts. Final Rule.

Federico Federici and Patrick Cadwell. 2018. Training citizen translators: Design and delivery of bespoke training on the fundamentals of translation for New Zealand Red Cross. *Translation Spaces*, 7(1):20–43.

Federico M. Federici. 2022. Translating hazards: multilingual concerns in risk and emergency communication. *The Translator*, 28(4):375–398.

Anni Fordham. 2023. Beyond barriers. https://churchilltrust.my.salesforce-sites.com/api/services/apexrest/v1/image/?Id=069RF00000NxDTtYAN. [Accessed 01 Jun 2026].

Samsara Foubert, Minjung Park, Kanad Pardeshi, Thomas Manzini, Robert McDaniel, David Merrick, Robin Murphy, Aarti Singh, and Hoda Heidari. 2026. Investigating the role of AI in emergency management: Use cases, challenges, and opportunities. In *Proceedings of the 2026 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '26, page 2055–2091, New York, NY, USA. Association for Computing Machinery.

Theo Guidroz, Diego Ardila, Jimmy Li, Adam Mansour, Paul Jhun, Nina Gonzalez, Xiang Ji, Mike Sanchez, Sujay Kakarmath, Mathias MJ Bellaiche, Miguel Ángel Garrido, Faruk Ahmed, Divyansh Choudhary, Jay Hartford, Chenwei Xu, Henry Javier Serrano Echeverria, Yifan Wang, Jeff Shaffer, Eric (Yifan) Cao, and 7 others. 2025. LLM-based Text Simplification and its Effect on user Comprehension and Cognitive Load. arXiv:2505.01980.

John Hajek, Anthony Pym, Yu Hao, Maria Karidakis, Ambrin Hasnain, Anila Hasnain, Juerong Qiu, Ke Hu, and Rachel Macreadie. 2024. Understanding and improving machine translations for emergency communications. https://works.hcommons.org/records/j3fba-nsn06. [Accessed 01 Jun 2026].

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Comput. Surv.*, 55(12).

Adam Tauman Kalai, Ofir Nachum, Santosh S. Vempala, and 1 others. 2026. Evaluating large language models for accuracy incentivizes hallucinations. *Nature*, 653:1047–1051.

Ramunė Kasperė, Jolita Horbačauskienė, Jurgita Motiejūnienė, Vilmantė Liubinienė, Irena Patašienė, and Martynas Patašius. 2021. Towards Sustainable Use of Machine Translation: Usability and Perceived Quality from the End-User Perspective. *Sustainability*, 13(23):13430.

Dorothy Kenny, Olga Torres-Hostench, Caroline Rossi, Alice Carré, Pilar Sánchez-Gijón, Sharon O'Brien, Joss Moorkens, Juan Antonio Pérez-Ortiz, Mikel L. Forcada, Felipe Sánchez-Martínez, and Gema Ramírez-Sánchez. 2022. *Machine Translation for Everyone*. Language Science Press.

King County Executive. 2010. Written language translation process king county written language translation process manual guidelines, effective practices, maps, resources. https://kingcounty.gov/exec/styleguide/~/media/cbf01a3bf9644c69b9ca5af88abb1202.ashx?la=en. [Accessed 01 Jul 2026].

Arle Lommel, Serge Gladkoff, Alan Melby, Sue Ellen Wright, Ingemar Strandvik, Katerina Gasova, Angelika Vaasa, Andy Benzo, Romina Marazzato Sparano, Monica Foresi, Johani Innis, Lifeng Han, and Goran Nenadic. 2024. The multi-range theory of translation quality measurement: Mqm scoring models and statistical quality control. *Preprint*, arXiv:2405.16969.

Patrick Meier. 2015. *Digital Humanitarians*. CRC Press, Boca Raton.

Mary Nurminen. 2021. *Investigating the influence of context in the use and reception of raw machine translation*. **PhD**, Tampere University.

Mary Nurminen and Niko Papula. 2018. Gist MT Users: A Snapshot of the Use and Users of One Online MT Tool. In *Proceedings of the 21st Annual Conference of the European Association for Machine Translation*, pages 219–228, Alicante, Spain.

OpenAI. 2024. Gpt-4o system card.

Sharon O'Brien and Federico Federici. 2019. Crisis translation: Considering language needs in multilingual disaster settings. *Disaster Prevention and Management*, 29(2):129–143.

América R. Gaviria Pabón, Makenzie J. Krocak, Joseph T. Ripberger, and Joseph E. Trujillo-Falcón. 2025. Linguistic challenges in severe weather communication: Comparing english- and spanish-speaking populations. *Bulletin of the American Meteorological Society*, 106(9):E1940 – E1954.

Jianhui Pang, Fanghua Ye, Derek Fai Wong, Dian Yu, Shuming Shi, Zhaopeng Tu, and Longyue Wang. 2025. Salute the classic: Revisiting challenges of machine translation in the age of large language models. *Transactions of the Association for Computational Linguistics*, 13:73–95.

Maja Popović. 2017. chrF++: Words helping character n-grams. In *Proceedings of the Second Conference on Machine Translation*, pages 612–618, Copenhagen, Denmark.

Hemant Purohit, Cody Buntain, Amanda Lee Hughes, Steve Peterson, Valerio Lorini, and Carlos Castillo. 2025. Engage and mobilize! understanding evolving patterns of social media usage in emergency management. *Proc. ACM Hum.-Comput. Interact.*, 9(2).

Alessandra Rossetti, Sharon O'Brien, and Patrick Cadwell. 2020. Comprehension and trust in crises: Investigating the impact of machine translation and post-editing. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 9–18, Lisboa, Portugal. European Association for Machine Translation.

Pilar Sánchez-Gijón. 2022. Selecting and preparing texts for machine translation: Pre-editing and writing for a global audience.

Vanessa Šorak. 2024. From terminological inconsistencies to major mistranslations: A qualitative analysis of nmt errors in public health communication. *AI-Linguistica. Linguistic Studies on AI-Generated Texts and Discourses*, 1(1).

Maria Carmen Staiano, Lifeng Han, Johanna Monti, and Francesca Chiusaroli. 2025. ITALERT: Assessing the quality of LLMs and NMT in translating Italian emergency response text. In *Proceedings of Machine Translation Summit XX: Volume 1*, pages 566–577, Geneva, Switzerland. European Association for Machine Translation.

Sanja Štajner and Maja Popovic. 2016. Can text simplification help machine translation? In *Proceedings of the 19th Annual Conference of the European Association for Machine Translation*, pages 230–242.

Belu Ticona and Antonios Anastasopoulos. 2026. Llm-powered automatic translation and urgency in crisis scenarios. *Proceedings of the International ISCRAM Conference*, 23.

Anne M Turner, Yong K Choi, Kristin Dew, Ming-Tse Tsai, Alyssa L Bosold, Shuyang Wu, Donahue Smith, and Hendrika Meischke. 2019. Evaluating the usefulness of translation technologies for emergency response communication: A scenario-based study. *JMIR Public Health Surveill*, 5(1):e11171.

U.S. Census Bureau. 2024. Language spoken at home by ability to speak english for the population 5 years and over. U.S. Census Bureau. Accessed 27 August 2026.

US DOJ LEP Cmte. 2021. Improving access to public websites and digital services for limited english proficient (lep) persons. https://niwaplibrary.wcl.american.edu/wp-content/uploads/LEP_Website_Language_Access_Guide_508.pdf. [Accessed 16 Jul 2026].

U.S. GSA. 2026. Introduction to translation technology — digital.gov. https://web.archive.org/web/20260625210818/https://digital.gov/resources/introduction-to-translation-technology. [Accessed 16 Jul 2026].

Lucas Nunes Vieira, Carol O'Sullivan, Xiaochun Zhang, and Minako O'Hagan. 2022. Machine translation in society: Insights from UK users. *Language Resources and Evaluation*.

Melissa Villarreal, Carson MacPherson-Krutsky, and Mary Angelica Painter. 2025. Barriers and best practices for inclusive emergency alerts and warnings. *International Journal of Disaster Risk Reduction*, 125:105581.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022. Emergent abilities of large language models. *Preprint*, arXiv:2206.07682.

Christopher Whyte and Brittany 'Brie' Haupt. 2026. Building centaur responders: is emergency management ready for artificial intelligence? *Disasters*, 50(3):e70054.

Vilém Zouhar, Shuoyang Ding, Anna Currey, Tatyana Badeka, Jenyuan Wang, and Brian Thompson. 2024. Fine-tuned machine translation metrics struggle in unseen domains. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 488–500, Bangkok, Thailand. Association for Computational Linguistics.

APPENDIX

A Translation Dataset

As part of our collaboration with LAAWW, we are compiling a collection of pre-translated emergency alerts from multiple agencies across the United States. The alerts span various emergency types, including (*e.g.,* weather, public health, hazmat, etc.). Each alert includes the original English message, professional human translations into dozens of languages, and metadata such as agency, date issued, language pair, alert type, etc.

We intend to make this dataset available to all those in the emergency management community who wish to use these pre-translated alerts. We also intend to make the dataset available to the research community to support future work on translation quality evaluation, multilingual emergency communication, etc. We are actively growing this data set. If you have data to share, please contact the authors.

B Participant Information

Table 1 is intended to give the reader further insight into participant comments. Only survey participants who are quoted in this paper are included in this table.

Type of EM	ID	Juridisction Scope	Location
Local Emergency Manager	L1	County	West Coast
Public Health Coordinator	H1	County	West Coast
Public Information Officer	P1	City	East Coast
Advocate (Retired Researcher)	R1	State	West Coast

Table 1: Demographic information of quoted survey participants.

C Translation Needs and Speaker Populations

The language needs of each geographic area vary because populations are not evenly distributed across the nation. Figure 4 shows the difference between the national and NY/NJ populations who speak a language other than English as their primary language. The population of NY/NJ is approximately 5.85% of the total population of the United States. The trend line is the expected result if the percentages in NY/NJ were the same nationally. Languages that appear above the line are more heavily represented in NY/NJ while languages that appear below the line are more heavily represented nationally.

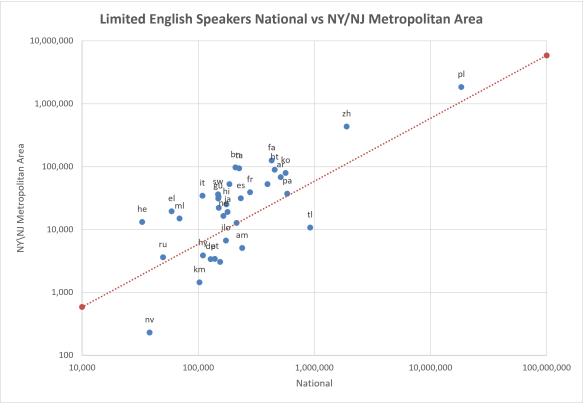


Figure 4: Count of Language Speakers in the NY/NJ Metropolitan area vs Nationally. Trend line represents the expected population if the percentages were identical in the NY/NJ Metropolitan area vs Nationally. (U.S. Census Bureau, 2024)

This data allows us to see some striking outliers in the populations. There are approximately 435,000 people who speak Chinese (zh) and "Speak English Less than Well" in NY/NJ accounting for nearly 23% of the approximately 1,895,000 people nationally, much higher than the expected 5.85%.

Table 2 reports a similar analysis across major metropolitan areas in the U.S, where the top row lists expected demographic share based on national population proportions. For example, Seattle-Tacoma-Bellevue, WA Metropolitan Area represents 1.22% of the U.S. population, but contains substantial demographic skew: eighteen language communities are over-represented (>2 x baseline) and eight are under-represented (<0.5 x baseline).

We can use Haitian Creole[23] as a specific language example. In Table 2 we see that it is underrepresented in nearly every area. This is because Haitian Creole speakers are disproportionally represented in Florida, were approximately 50% of Haitian Creole speakers live.

The data also shows that there is no language group within these areas that is spoken by the expected proportion of speakers. Each county, metropolitan area, or state has its own unique mix of languages, thus each area must to do their own language analysis to determine their community's language and communication needs.

D Languages and Language Varieties

If they chose, survey participants were able to provide the languages spoken by the communities they serve. The list below is an example of the

[23]The US Census uses the language name Haitian.

	National	Atlanta	Chicago	Dallas	Los Angeles	Minneapolis	San Francisco	Seattle
Total	321,745,943	1.88%	2.77%	2.43%	3.82%	1.10%	1.37%	1.22%
Amharic, Somali, or other Afro-Asiatic languages	238,986	3.73%	3.56%	4.96%	2.26%	14.53%	1.84%	6.31%
Arabic	513,293	0.48%	4.20%	3.23%	5.14%	0.65%	1.86%	1.11%
Armenian	109,630	0.00%	0.54%	0.00%	83.24%	0.00%	0.40%	0.65%
Bengali	209,543	1.66%	1.74%	2.17%	3.35%	0.32%	0.17%	0.71%
Chinese	1,895,001	1.36%	2.76%	1.33%	14.23%	0.54%	10.76%	3.22%
French (incl. Cajun)	279,230	2.76%	2.01%	3.11%	2.36%	1.22%	1.38%	1.61%
German	127,650	0.83%	1.98%	0.96%	1.86%	2.07%	0.79%	1.31%
Greek	59,054	0.73%	8.86%	0.10%	2.49%	0.00%	0.73%	0.28%
Gujarati	148,495	5.28%	12.67%	3.60%	2.88%	0.48%	1.35%	0.45%
Haitian	454,970	1.73%	0.18%	0.21%	0.13%	0.26%	0.03%	0.10%
Hebrew	32,924	1.66%	1.64%	0.60%	9.32%	0.29%	1.25%	2.37%
Hindi	174,630	3.09%	4.63%	4.39%	4.79%	0.45%	7.12%	2.66%
Hmong	91,284	0.69%	0.09%	0.38%	0.69%	28.45%	0.00%	0.19%
Ilocano, Samoan, Hawaiian, or other Austronesian languages	173,239	1.06%	2.40%	1.09%	6.79%	0.25%	3.78%	4.88%
Italian	108,644	0.52%	5.52%	1.42%	2.43%	0.35%	1.55%	0.08%
Japanese	179,098	1.20%	2.39%	1.51%	13.42%	0.54%	4.20%	3.72%
Khmer	102,600	2.19%	0.75%	2.36%	14.97%	2.36%	3.74%	7.49%
Korean	565,989	4.23%	3.60%	3.16%	22.66%	0.32%	2.30%	4.41%
Malayalam, Kannada, or other Dravidian languages	69,100	1.23%	6.26%	12.09%	1.32%	0.73%	1.99%	2.43%
Navajo	38,127	0.00%	0.00%	0.30%	0.00%	0.03%	0.00%	0.12%
Nepali, Marathi, or other Indic languages	164,869	2.94%	1.51%	7.76%	3.21%	0.64%	4.47%	1.40%
Persian (incl. Farsi, Dari)	213,566	2.43%	1.45%	2.27%	20.99%	0.37%	5.89%	5.02%
Polish	185,755	0.37%	35.58%	0.28%	0.99%	0.21%	0.37%	0.48%
Portuguese	394,010	1.63%	0.53%	1.20%	1.26%	0.18%	1.05%	1.22%
Punjabi	150,447	0.31%	2.01%	1.11%	2.46%	0.22%	5.77%	4.43%
Russian	429,378	1.01%	4.26%	0.57%	7.49%	1.32%	2.25%	3.48%
Serbo-Croatian	79,111	1.93%	13.69%	1.77%	2.22%	0.85%	0.72%	2.13%
Spanish	18,432,221	1.41%	3.46%	3.73%	9.51%	0.39%	1.71%	0.62%
Swahili or other languages of Central, Eastern, and Southern Africa	138,888	7.02%	0.97%	8.13%	0.84%	2.94%	0.25%	3.17%
Tagalog (incl. Filipino)	584,724	0.49%	3.09%	1.40%	14.52%	0.58%	9.23%	2.85%
Tamil	49,660	3.16%	3.91%	5.99%	2.76%	1.25%	5.10%	3.43%
Telugu	104,199	3.69%	2.32%	12.31%	0.84%	0.82%	3.73%	3.69%
Thai, Lao, or other Tai-Kadai languages	154,016	1.17%	2.80%	3.08%	9.62%	1.73%	3.13%	4.25%
Ukrainian or other Slavic languages	232,585	0.67%	15.75%	0.42%	2.69%	1.28%	1.03%	7.71%
Urdu	148,178	1.97%	9.39%	6.34%	3.96%	0.09%	1.92%	1.04%
Vietnamese	918,210	3.23%	1.37%	4.87%	18.03%	0.84%	3.11%	3.93%
Yiddish, Pennsylvania Dutch or other West Germanic languages	225,109	0.05%	0.21%	0.12%	0.17%	0.10%	0.00%	0.09%
Yoruba, Twi, Igbo, or other languages of Western Africa	149,225	2.09%	4.83%	2.93%	2.08%	4.24%	0.78%	2.53%

Table 2: Language Variation of actual vs. expected representation for select Metropolitan areas in the United States (U.S. Census Bureau, 2024). underrepresented = expected/2 overrepresented = expected*2
Languages spoken by fewer than 10,000 people are grouped with other languages by the U.S. Census Bureau. More information can be found here: https://usa.ipums.org/usa/volii/00language.shtml.

data they shared. If participants shared a language variety (as they defined it), it is included here. The data is formatted showing the principal language, followed by varieties (if provided), *e.g.,* **Language**: variety.

Ahtna; **Alutiiq / Sugpiaq**; **Amdo Tibetan**; **Amharic**; **Arabic**: Algerian, Chadian, Classical, Egyptian, Gulf, North Levantine, South Levantine, Yemeni; **Armenian (Western)**; **Aromanian**; **Ashanti**; **Avestan**; **Azeri**;

Bahamian Creole; **Bambara**; **Barbadian Cre-**

ole; **Bavarian**; **Belarusian**; **Bengali**; **Bislama**; **Brazilian**; **Bukhori**; **Burmese (Intha)**;

Calabrese; **Cambodian / Khmer**; **Cebuano**; **Chinese**: Cantonese, Mandarin, Simplified, Fujianese, Hainanese (Southern Min), Hokkien, Hunanese, Han, Shanghainese, Taishanese, Taiwanese, Wenzhounese; **Coptic**; **Crimean Tatar**; **Croatian**;

Dari; **Deg Xinag**; **Dena'ina**; **Dutch**; **Ecuadorian Kichwa**; **Efik**; **English**: African-American, Australian, Canadian, Ghanaian, Hawaiian, Indian, Irish, Italian, Jewish, Liberian, Native American, New Zealand, Newfoundland, Nigerian, Pakistani, Philippine; **Ewe**; **Eyak**; **Fante**; **Farsi**; **Finnish**; **French**; **Fulani**;

Galician; **Garifuna**; **Geg Albanian**; **Georgian**; **German**: Austrian, Gottscheerisch, Swabian; **Greek**: Cretan, Cypriot, Koine; **Grenadian Creole**; **Gujarati**; **Guyanese Creole**; **Gwich'in**; **Haida**; **Haitian Creole**; **Hakha Chin**; **Hamgyŏng Korean**: Joseonjok; **Hebrew**: Biblical; **Hindi**; **Hmong**; **Holikachuk**; **Hungarian**;

Icelandic; **Igbo**; **Ilocano**; **Indonesian**; **Inupiaq**; **Irpino**; **Italian**; **Iu Mien**; **Jamaican Patois**; **Japanese**; **Jewish Babylonian Aramaic**; **Judeo-Shirazi**; **Judeo-Spanish**; **Juhuri**; **Kalderash Romani**; **Kannada**; **Karen**; **Kinyarwanda**; **Kirundi**; **Konkanoi**; **Korean**; **Koyukon**;

Lahu; **Lao**; **Lemko**; **Lenape**; **Lithuanian**; **Lukumí**; **Maithili**; **Malayalam**; **Maltese**; **Mandinka**; **Maranao**; **Marathi**; **Marshallese**; **Marwari**; **Matinecock**; **Mixtec**; **Mohican**; **Mon**; **Montaukett**; **Montserrat Creole**;

Nahuatl; **Nauruan**; **Navajo**; **Ndebele**; **Neapolitan**; **Nepali**; **Newari**; **Noakhailla**; **North Frisian**; **Nuer**; **Pashto**; **Polish**; **Portuguese**: Brazilian, Portuguese; **Pugliese**; **Punjabi**; **Quiripi**; **Ramaluk**; **Romagnol**; **Romanian**; **Russian**;

Sandwippa; **Sanskrit**; **Serer**; **Sherpa**; **Sicilian**; **Sinhalese**; **Somali**; **Soninke**; **Spanish**: Argentine, Bolivian, Castilian, Central American, Colombian, Cuban, Dominican, Ecuadorian, Mexican, Puerto Rican, Venezuelan; **Spanglish**; **St. Lawrence Island Yupik**; **St. Lucian Creole**; **Swahili**; **Sylheti**;

Tagalog; **Tajik**; **Tamang**; **Tamil**; **Tanacross**; **Tanana**; **Telugu**; **Tetum**; **Thai**; **Tibetan**: Classical, Tö; **Tlingit**; **Tosk Albanian**; **Trinidadian Creole**; **Tsimshian**; **Turkish**; **Twi**; **Ukrainian**; **Unangam Tunuu / Aleut**; **Upper Kuskokwim**; **Upper Tanana**; **Urdu**; **Uzbek**; **Yiddish**: Hasidic, Litvish; **Yup'ik**; **Zulu**.

E More on Post-editing and Evaluating Translations

Here are some rules of thumb for determining the adequacy of the translations provided by a commercial translation tool (or of any translations):

- Use Multidimensional Quality Metric (MQM) (Lommel et al., 2024) to define what *major errors* might exist. Ultimately, there should be no *major errors*, but minor errors may be acceptable.[24]
- Critical data, such as place names, should be kept in the original language because geographic place names tend to stay consistent and do not change based on language spoken. This information can be evaluated even by those who do not read the target language.

Post-editing is often done using Computer Assisted Translation (CAT) tools such as Trados or MemoQ[25], although it can be done in tools such as Word or Excel (with a little more difficulty). One of the advantages of using CAT tools is that they track translations and can reuse material from earlier translations (through *Translation Memories*). Many professional translators are familiar with these tools.

In scenarios where time is limited or it is not possible to do a formal evaluation, one can do round-trip translation and cross checking. For round-trip translation, use an MT tool to perform the initial translation into the target language. Once the translation output is generated, copy the translated text into a different MT tool and translate it back into the original language. If the new translation preserves the meaning, then the translation is likely usable. To use cross checking, use two different MT tools to produce a translation for the same text and compare the outputs. If the outputs are identical and likewise pass the back translation test, then the translation is likely usable.

There are many other aspects of translation evaluation, including fluency, tone, meaning, terminology, additions/omissions, cultural appropriateness, localization, accessibility, etc. These are beyond the scope of this paper.

[24] Examples of major errors may include incorrectly translated safety instructions, incorrect locations, or incorrect times.

[25] https://www.trados.com/, https://www.memoq.com/

F Survey Questions

The survey was presented to LAAWW membership during a monthly LAAWW meeting. Survey goals and methodologies were explained to the group and time was given for questions.
The survey questions below are reproduced as administered. Response choices and formats are provided. If *Other* was a Response Choice, participants were able to use a Free Response to answer.

1. **Which of the following best describes your current role(s) in emergency management?**
   Response format: Multiple Choice; Text
   Response choices: Local emergency manager; State emergency manager; Public information officer; Language coordinator / interpreter; Operations, logistics, or communications manager; Researcher; Other

2. **Please list the city, county, or region your agency or institution primarily serves.**
   Response format: Free response

3. **Approximately how many different languages are spoken by the communities you serve?**
   Response format: Free response

4. **If you have a convenient list of the languages, please paste it here or share a URL.**
   Response format: Free response

5. **Do you translate FROM languages other than English?**
   Response format: Yes-no style response
   Response choices: Yes; No

6. **What languages do you translate FROM other than English?**
   Response format: Free response

7. **Feel free to share more information about any of your answers from this section.**
   Response format: Free response

8. **How does your agency/institution identify the languages to prioritize for translation and outreach?**
   Response format: Multiple Choice; Text
   Response choices: US Census or American Community Survey data; Your State's Department of Education Data; Local community or needs assessments; Input from schools, health, or social service agencies; Feedback from community organizations; Staff or volunteer knowledge; Other

9. **What are the translation tools used by your agency/institution?**
   Response format: Multiple Choice; Text
   Response choices: Google Translate; Microsoft Translator; DeepL; ChatGPT; DeepSeek; Google Gemini; Staff member; Contractor; Community member or volunteer; Other

10. **What are the main reasons your agency/institution selected these tools or methods?**
    Response format: Multiple Choice; Text
    Response choices: Cost or budget constraints; Availability or ease of access; Accuracy or quality of translation; Easy to use or staff comfort; Compliance or policy requirements; Other

11. **Has your agency/institution evaluated how well these tools work for accuracy, usability, or effectiveness?**
    Response format: Yes-no style response
    Response choices: Yes - formal evaluation; Yes - informal evaluation; No

12. **Does your agency/institution review or update its translation tools or methods?**
    Response format: Yes-no style response
    Response choices: Yes - annually; Yes - as needed; No formal review process; Not sure; No

13. **Who usually handles translations or multilingual communication within your agency/institution?**
    Response format: Multiple Choice; Text
    Response choices: Dedicated translation team; Individual staff member; Shared among staff; External vendor or contractor; Volunteers / community partners; Other

14. **Feel free to share more information about tools, processes, personnel, and capacity here.**
    Response format: Free response

15. **What additional tools, training, or partnerships would most help your agency improve multilingual communication and translation capacity?**
    Response format: Free response

16. **What are some challenges?**
    Response format: Free response

17. **May we contact you if we have follow up questions? If so, please provide your preferred contact information.**
    Response format: Free response

G Evaluation Results

Language	Microsoft	Google	GPT-5.4-mini	Gemini 2026	Claude
Pivot					
Arabic	7.34	7.42	7.66	8.34	7.67
Chinese	11.97	10.66	13.02	11.10	12.67
French	5.67	4.85	6.03	4.89	5.47
Hindi	13.16	9.28	10.33	8.34	12.24
Indonesian	1.98	3.33	2.97	3.03	2.44
Portuguese (BR)	4.41	4.41	4.42	4.69	4.93
Russian	6.66	8.06	6.68	6.97	6.70
Spanish (LA)	4.04	6.14	3.94	4.03	3.75
Swahili	-2.96	-3.13	-3.04	-1.54	-2.30
Important					
Bengali	1.65	1.05	0.04	-0.66	-0.13
Farsi	1.99	3.53	4.78	4.17	3.96
Marathi	2.07	1.72	1.20	0.90	1.70
Myanmar	1.95	3.69	2.22	2.72	3.88
Tagalog	1.47	2.52	2.13	2.40	0.75
Tamil	0.03	0.28	-0.78	0.98	0.25
Urdu	1.34	2.96	2.56	2.59	2.04
Priority					
Amharic	4.61	4.47	1.96	3.73	1.41
Dari	1.25	–	1.18	0.80	2.29
Khmer	-6.54	-3.96	-3.40	-1.36	-2.54
Kinyarwanda	1.32	0.03	-0.39	-1.08	0.60
Nepali	3.45	3.91	4.65	4.35	3.76
Pashto	5.94	10.11	5.34	2.98	4.96
Somali	-0.21	-0.23	-1.01	0.07	-0.25
Zulu	-2.19	-2.93	-1.38	-2.17	-0.59

Table 3: Differences between Overall CHRF++ Score and News+Announcements CHRF++. Blue indicates Overall score is higher than N+A, orange indicates Overall is lower than N+A. Google Translate results were unavailable for Dari. Khmer, Myanmar, and Chinese do not use whitespace to mark word boundaries and CHRF was used instead. (Bansal et al., 2026)

	Microsoft		Google		GPT-5.4-mini		Gemini 2026		Claude	
Language	**Overall**	**N+A**	**Overall**	**N+A**	**Overall**	**N+A**	**Overall**	**N+A**	**Overall**	**N+A**
Pivot										
Arabic	55.08	47.73	**59.23**	**51.82**	57.86	50.20	57.27	48.93	55.64	47.97
Chinese	45.74	33.77	**50.48**	**39.82**	48.10	35.08	42.66	31.56	45.89	33.21
French	63.64	57.97	63.13	58.28	**65.52**	**59.49**	63.05	58.16	63.02	57.55
Hindi	**65.62**	52.46	61.76	52.48	62.81	52.48	61.75	**53.41**	62.42	50.18
Indonesian	74.72	**72.74**	74.88	71.54	**74.93**	71.95	73.92	70.88	73.74	71.30
Portuguese (BR)	72.06	67.65	72.05	67.64	**72.99**	**68.56**	**72.99**	68.30	72.62	67.69
Russian	56.13	49.47	**59.57**	**51.51**	58.11	51.43	58.19	51.23	57.14	50.44
Spanish (LA)	71.88	67.84	**74.17**	68.03	74.08	**70.14**	73.64	69.60	72.84	69.09
Swahili	59.96	62.92	**61.59**	**64.72**	60.44	63.48	60.16	61.70	55.86	58.16
Important										
Bengali	**51.40**	**49.75**	46.69	45.64	47.25	47.21	44.38	45.04	47.55	47.68
Farsi	52.32	50.33	**56.44**	**52.92**	52.71	47.93	51.80	47.63	51.38	47.42
Marathi	**46.80**	**44.73**	46.05	44.33	44.97	43.76	44.33	43.43	41.57	39.87
Myanmar	48.16	46.21	**54.56**	50.87	53.99	**51.77**	52.62	49.90	42.84	38.95
Tagalog	64.72	63.25	**68.27**	**65.75**	66.61	64.48	63.42	61.02	63.83	63.08
Tamil	**56.87**	**56.85**	53.11	52.84	48.75	49.53	50.68	49.70	46.48	46.23
Urdu	50.77	49.42	**52.80**	**49.84**	50.67	48.12	49.17	46.58	49.37	47.33
Priority										
Amharic	**35.84**	31.23	35.78	**31.31**	31.23	29.27	32.62	28.89	27.60	26.18
Dari	**48.15**	**46.90**	–	–	44.23	43.05	44.56	43.76	43.63	41.34
Khmer	60.53	**67.07**	**61.60**	65.66	57.52	60.92	56.86	58.22	46.66	49.20
Kinyarwanda	45.88	44.56	**47.87**	47.84	47.74	48.13	47.50	**48.59**	33.23	32.64
Nepali	53.42	49.98	**56.19**	**52.27**	52.39	47.75	49.35	45.00	50.25	46.49
Pashto	39.47	33.53	**48.52**	**38.40**	42.25	36.91	40.10	37.12	33.66	28.70
Somali	44.92	45.12	**46.62**	**46.85**	45.44	46.46	44.53	44.46	37.18	37.44
Zulu	52.76	54.95	**53.61**	**56.54**	53.34	54.73	50.31	52.49	42.54	43.13

Table 4: Overall and News+Announcements (N+A) CHRF++ scores for English→Target Language across translation systems and languages. Highest scoring language result for Overall and N+A is in bold. Shading indicates relative score ranges within the evaluation: blue represents higher scores (>60), pale green represents mid-range scores (45-60), and orange represents lower scores (<45). CHRF++ does not have a universal safety or usability threshold, so these colors indicate relative performance only. Google Translate results were unavailable for Dari. Khmer, Myanmar, and Chinese do not use whitespace to mark word boundaries and CHRF was used instead. (Bansal et al., 2026)

Looking Complete Is Not Being Grounded: Evaluating LLM Sitreps Against Human-Authored Situation Reports

Sachin Kumar[*]
LexisNexis, USA
sachinkumar.ait@live.com

Abstract

Evaluation for automatic crisis reporting should measure what practitioners need, not what is convenient to compare. Through LLM-assisted, author-approved content analysis, we derive a 15-slot information schema from professional humanitarian situation reports using qualitative content analysis, then evaluate 121 human and 261 LLM-generated reports across five disasters and three hazard types. Under `google/gemini-3.7-flash` on Cyclone Idai, the best machine arm is nominally above the empirical human ceiling (0.827 vs. 0.821), but the difference is statistically indistinguishable from zero (95% bootstrap CI $[-.049, .059]$; permutation $p = .852$), and the ordering does not replicate under `openai/gpt-5.6-luna`. On a blind 100-item Idai sample, the author agrees with the Gemini judge at 0.910 accuracy and unweighted Cohen's $\kappa = .799$. All four judges clear the plan's raw-agreement criterion of .7; only Gemini reaches .7 on chance-corrected κ. Across the three primary events, machine reports reproduce 5.4–11.0% of same-day official figures. Across those same events, raw exact-number matching links 14.3–19.6% of official figures and 85.4–90.1% of machine figures to the social stream; manual audit confirms 45/50 declared absences, 4/25 official matches, and 24/25 machine matches semantically. The issuing-provenance gap is positive in all 12 observed event–judge cells: report date, organisation, series, and authority were not supplied at professional-report coverage in any cell. On the three primary events, ROUGE-L is uninformative, BERTScore is negatively associated with completeness after multiplicity correction, and a generic LLM judge changes sign.

1 Introduction

A situation report (sitrep) is not simply a summary. It is a dated, attributed operational document that supports the coordination of a response by reporting current needs, actions, gaps, and key figures (United Nations Office for the Coordination of Humanitarian Affairs, 2023). The distinction matters for the humanitarian teams who would act on such reports and for the LT4CPR shared task: a fluent report can mention every expected topic while omitting the quantitative and provenance-bearing information on which operational use depends.

Most automatic evaluation begins with available instruments: overlap with a reference, embedding similarity, or a general-purpose model asked whether an output is good. We instead begin with practice. We ask what information professional humanitarian organisations actually place in sitreps, how much of it a system can recover from social media, and whether common metrics recognise that recovery.

We make three contributions. First, we induce a 15-slot schema from professional reports through LLM-assisted qualitative content analysis, and report what each computational step contributed and what it excluded. Second, we define *operational completeness* relative to an empirical human ceiling and locate a *visibility ceiling*: information that is present in official reports but absent from the source stream. Third, we meta-evaluate ROUGE-L, BERTScore, and a generic LLM judge against operational completeness. The apparent paradox is the paper's central result: machine sitreps look complete by slot presence, but their quantitative picture is bounded by an input stream containing only a small fraction of the official record. We close with implications for humanitarian practice (Section 7) and for shared-task evaluation (Section 8).

2 Related Work

Crisis informatics has long treated social media as a rapid source for situational awareness, while also documenting its noise, brevity, and uneven cover-

[*] This research was conducted independently and does not reflect the views or represent work done at LexisNexis.

Proceedings of the First Workshop on Language Technologies for Crisis Preparedness and Response, pages 65–77
November 9, 2026 ©2026 Association for Computational Linguistics

age (Imran et al., 2016; Alam et al., 2021). Recent work surveys the resulting language-technology ecosystem and its operational challenges (Lewis et al., 2025). Our question differs from message classification: we compare a generated operational product against reporting practice and explicitly measure what the input medium did not supply in this corpus.

Crisis summarisation is the nearest task family. CrisisLTLSum extracts and abstractively summarises 1,000 local Twitter timelines (Rajaby Faghihi et al., 2022); CrisisFACTS produces event-day timelines from social and news streams and evaluates pooled facts for comprehensiveness and redundancy (Buntain et al., 2023); prompt-based LLM systems have also aggregated geolocated social feedback into disaster reports (Cantini et al., 2025). The current LT4CPR task maps crisis posts to a structured official template (LT4CPR Shared Task Organisers, 2026).We add a combination these benchmarks do not provide: a slot-level comparison of model reports with independently published humanitarian sitreps, a practice-derived schema, and an audit of which official information was visible in the input. CrisisFACTS's earlier events used official ICS-209 incident forms as an alternative ground truth (Buntain et al., 2023), but without a practice-derived schema or an input-visibility audit.

Our schema also differs from NLP event-schema induction, which learns entity and event-role structures from corpora (Chambers, 2013), and from methods that optimise prompt tokens for a downstream objective (Shin et al., 2020). It is an LLM-assisted, author-approved content-analysis description of recurrent document information types. Operational completeness inherits the content-unit intuition of Pyramid and later Summary Content Unit evaluation (Nenkova and Passonneau, 2004; Nawrath et al., 2024), but its units are practice expectations induced across reports, its categories allow partial credit, and its reference point is an empirical human ceiling.

Summarisation evaluation commonly uses lexical overlap such as ROUGE (Lin, 2004) and contextual similarity such as BERTScore (Zhang et al., 2020). Prior analyses show that such metrics can track topical similarity rather than information quality and that realistic system-level ROUGE correlations may approach zero (Deutsch and Roth, 2021; Deutsch et al., 2022). The latter critique is system-level, whereas our main Table 3 correlations are between reports; we therefore also report a descriptive ordering of the four generator–prompt arms. LLM judges offer flexible reference-free evaluation (Liu et al., 2023), but can favour their own generations (Panickssery et al., 2024). We test all three metric families against a domain-derived criterion rather than general ratings.

3 Data

Professional reports. We manually collected public ReliefWeb products from OCHA, IFRC, UNICEF, WHO, CDEMA, and other humanitarian organisations. ReliefWeb's API required a pre-approved `appname` that had not been granted, while scripted site requests returned an HTTP 202 bot challenge. Manual download followed by automated scanning and bookkeeping was therefore the compliant collection route. The scanner extracted dates, organisations, and titles; content hashes removed duplicates. Table 1 distinguishes collection from evaluation: three Idai rows are exact SHA-256 duplicates. Puebla's fifteenth PDF ingested to `_2.json`; the manifest reader uniformly reserves leading-underscore files for helpers, so it was excluded, reconciling 15 collected versus 14 evaluated. Relative to $15J(H_{\text{eval}} + M_{\text{eval}})$, unusable judgements number 178/0/0/4/2 by table row; subtracting them produces every last-column cell. The supplement records titles, sources, verified dates, exact-title ReliefWeb locators, hashes, and the exclusion rule for all 125 products. We do not assume every ReliefWeb product is the same genre: most of Puebla's evaluated products are infographics or snapshots (10 of 14), so its score is not used as a sitrep ceiling, and Idai's 56 include 19 flash updates, three snapshots, and four IFRC DREF or emergency-appeal documents.

Social streams and machine reports. We use the English-language HumAID corpus (Alam et al., 2021), which contains tweet identifiers and text across multiple disasters. We derive UTC dates from identifier snowflakes, bucket posts by event-day, and generate on days with at least 30 posts (for Kerala, only the first five such days, to stay within our cost limit); each generation reads a seed-17 sample of at most 400 of that day's posts, stratified by HumAID label, while source availability is measured against the complete day. We cross `api-strong=deepseek-v4-pro` and `api-fast=deepseek-v4-flash` with generic and schema-guided prompts. Calls ran 30–31 Au-

Event	Human coll./eval.	Machine gen./eval.	Judges	Judged pairs
Cyclone Idai 2019	59 / 56	104 / 102	4	9,302
Hurricane Irma 2017	19 / 19	48 / 48	2	2,010
Hurricane Maria 2017	25 / 25	44 / 43	2	2,040
Puebla earthquake 2017	15 / 14	48 / 48	2	1,856
Kerala floods 2018	7 / 7	20 / 20	2	808
Total	125 / 121	264 / 261	–	16,016

Table 1: Manifest-reconciled corpus. A "judged pair" is one parsable verdict for one (judge, physical document, slot); repeated judges multiply the count. Each cell is $\sum_j \sum_d |\mathcal{O}_j(d)|$, and the total equals the column sum. Idai uses DeepSeek, Luna, Gemini, and Qwen; other events use Luna and Gemini. Human and machine exclusions are shown uniformly as collected/generated versus evaluated.

gust 2026 UTC. The public aliases were then the Pro GA update dated 13 August and the Flash-0731 update dated 31 July; DeepSeek exposed no immutable snapshot suffix (DeepSeek, 2026). The schema-guided prompt supplies the 15 slot names and definitions and asks for the names as section headings. Two empty Idai generations and one empty Maria generation are excluded from every judge and metric. Social-media text is neither redistributed nor quoted in this paper.

4 Inducing a Practice-Derived Schema

Interdisciplinary methods note. We borrow the phases of inductive content analysis—preparing material, organising codes into categories, and reporting an auditable structure (Elo and Kyngäs, 2008; Krippendorff, 2018)—but a model, not the researcher, did the open coding. Qualitative researchers interviewed by Schroeder et al. (2025) worried that model-led coding imposes the model's interpretations and changes what researchers notice; Schroeder et al. recommend validating each LLM task, for example against human coding, as Wang et al. (2025) did by comparing LLM-generated codes and themes with a prior human analysis of the same data. We have no human coding of our reports to compare against. We therefore separate computation that organised material (embedding and clustering near-synonymous codes) from steps that structured the conceptual space: our coding instructions and the model's codes set the vocabulary, a 25-cluster cap chose what reached review, and model-written definitions and an AI-drafted merge, which the author approved, set category boundaries. We treat the schema as one defensible description of Idai reporting practice, not the only one.

Open coding. The induction set is all 56 evaluated Idai reports (OCHA 44, IFRC 4, UNICEF 3, WFP 2, WHO 2, IOM 1): a sampler stratified by organisation and response phase (thirds of the collection's date range) requested 60, so every available report was included, and most are early-response products (45 early, eight middle, two late, one undated when sampled). `deepseek-v4-pro` open-coded each report in a separate call, instructed to record only information types present in it, each with an internal evidence span and location. The instructions also named five reporting conventions to record (attribution, as-of dates, hedged figures, revision language, and figures with sources) and gave four example code names, three of which (casualty figures, access constraints, funding-appeal status) became the most frequent code names of three candidates, along with four example section headings for locating codes. The conventions and example names echo our project plan, so these categories' prominence reflects our instructions as well as the reports. Reports longer than 30,000 characters (12 of 56, including all four from IFRC) were coded from their first 24,000 and last 6,000 characters, so 13% of the induction text was never coded. The pass produced 1,360 codes; evidence spans remain in the private audit trail and are not reproduced here.

Consolidation. We embedded code names and definitions with MiniLM sentence embeddings and clustered them (average linkage, cosine distance, threshold .35), yielding 382 clusters. Clusters were ranked by the number of reports they covered, then by code count and name, and the top 25 became candidate slots, each named, defined, and given two synthetic examples by `deepseek-v4-pro`. The other 357 clusters, holding 780 of the 1,360 codes (57%), never reached review. Most occur in a single report (209 clusters) or fragment concepts that the final slots cover. But ranking each cluster by its own report count also penalised concepts that the

coder named inconsistently across separately coded reports, a redundancy Wang et al. (2025) also observed: protection and the hedging of unconfirmed figures were split across several smaller clusters, and the largest logistics cluster, like a revision-language cluster, reached 12 reports but lost an alphabetical tie-break with the last two candidates. The sector-needs, sector-response, and access slots can absorb some logistics and protection content, but no slot records hedging.

Human adjudication and audit trail. The author reviewed every candidate, merged overlapping concepts, renamed categories, rewrote definitions, and supplied two synthetic examples per slot. The frozen schema records the approver and approval time. The audit trail retains sampled report identifiers, per-report codes, and consolidation decisions. The final slots are: data currency and report metadata; sourcing and attribution; casualties and affected population; displacement and accommodation sites; health, disease outbreak and nutrition; physical, economic and agricultural damage; funding and donor contributions; education; WASH; food security and assistance; access constraints; coordination arrangements; sector needs and gaps; sector response actions; and shelter and non-food items.

5 Measurement

Operational completeness. Let j index the judge, d a physical document, $z \in \mathcal{Z}$ one of 15 slots, and $s_j(d,z) \in \{0, .5, 1\}$ the score for ABSENT, PARTIAL, or PRESENT. Let $\mathcal{O}_j(d) \subseteq \mathcal{Z}$ be the slots with a parsable judgement and $m_j(d) = |\mathcal{O}_j(d)|$. Operational completeness is

$$C_j(d) = \frac{1}{m_j(d)} \sum_{z \in \mathcal{O}_j(d)} s_j(d,z). \qquad (1)$$

Thus a missing judgement is not scored zero or imputed: it is dropped and reduces the denominator; a document with $m_j(d) = 0$ is excluded. Each decision must include an evidence span. For event e, exact generator/prompt arm a, and $D_{j,a,e,z}$, the unique documents in that arm with a usable judgement for z, slot coverage is

$$\text{coverage}_j(z; a, e) = \frac{1}{|D_{j,a,e,z}|} \sum_{d \in D_{j,a,e,z}} s_j(d,z).$$

Humans form one arm; the four machine arms remain separate. We do not pool models or prompts when locating a gap: $M_j(z,e) = \max_{a \in \mathcal{A}_{\text{machine}}} \text{coverage}_j(z; a, e)$ and $G_j(z,e) = \text{coverage}_j(z; \text{human}, e) - M_j(z,e)$. The thresholded binary flag invoked in Recommendation 4 is explicitly

$$B_j(z,e) = \mathbf{1}\left[\begin{array}{l} \text{coverage}_j(z; \text{human}, e) \geq .8, \\ M_j(z,e) \leq .6 \end{array} \right].$$

The *human ceiling* is the mean of $C_j(d)$ over an event's evaluated professional reports, not an assumed 1.0. For Idai these are the 56 reports from which the schema was induced, so the Idai ceiling is in-sample and may favour human reports; Irma, Maria, Puebla, and Kerala are held out from schema induction. Per-judge, per-arm tables release every numerator, denominator, and score sum.

Two judges, `openai/gpt-5.6-luna` and `google/gemini-3.7-flash`, score all five events. Neither shares a model family with the DeepSeek generator models. Idai additionally retains `deepseek-v4-pro`, which does share the generator family, and `qwen/qwen3.8-flash`, which does not, yielding 12 event–judge cells. Figure 2 uses 1,000 within-arm resamples of unique document scores, a percentile 95% interval, and `random.Random(17)`. The arm-minus-human contrast uses 10,000 independent two-sample bootstrap resamples and a 100,000-draw, two-sided label-permutation test (seed 17); reports are not naturally paired across arms.

Content gate and judge reliability. The original prompt requires substantive content, but we additionally re-score schema-guided PRESENT cells deterministically. A matched heading with an empty or explicitly no-information body is capped at ABSENT; a non-negated body with fewer than eight alphabetic tokens is capped at PARTIAL; other scores are unchanged.

For human validation, a seed-17 slot-stratified Idai sample contains 100 unique document–slot items, balanced between professional and machine reports and spanning all 15 slots. The author applied the same three-category rubric while blind to the withheld verdicts of all four Idai judges. For each judge, we report exact-match accuracy, unweighted nominal Cohen's κ, and linear-weighted κ as an ordinal sensitivity check; for primary Gemini we additionally report one-vs-rest precision, re-

call, and F1 for each label, treating the human label as ground truth. A missing model verdict is dropped only for that judge and reduces its stated denominator. Our project plan, written before annotation, set the acceptance rule as judge agreement $\geq .7$ on a 100-item hand-scored sample without naming the statistic; our pre-annotation scoring code applied it to raw agreement, but some pre-annotation notes restated it as Cohen's $\kappa \geq .7$, so we report both. The Luna–Gemini Fleiss values are a separate inter-judge consistency check outside the acceptance rule, computed on unique complete-case document–slot items with nominal categories and two fixed raters per event.

Figure grounding and source availability. A regex extractor identifies numeric values in machine reports, same-day human reports, and the complete same-day social stream. Figure precision is the share of machine values also in the human report; recall reverses the denominator; casualty recall restricts the comparison to deaths, injuries, and missing people. A source-availability test asks both whether machine figures are traceable to the stream and whether official figures occur anywhere in that stream. Official-in-stream is averaged over unique valid event-days because its value repeats across machine arms; other cells average valid same-day machine–official pairs. Undefined denominators are dropped. Percentile 95% CIs use 10,000 seed-17 resamples of event-days for official-in-stream and pairs for the other cells. A seed-17 manual audit samples 50 official figures marked stream-absent plus 25 official–stream and 25 machine–stream exact matches; two-sided Clopper–Pearson intervals bound each error rate. This analysis contains no model judge.

Metric meta-evaluation. We score each non-empty machine report against same-day or nearest same-window human references using ROUGE-L, BERTScore, and a schema-blind LLM quality score produced by `google/gemini-3.7-flash`, the same model used as the Gemini slot judge. Meta-evaluation includes only the three primary events, Idai, Irma, and Maria, because each has a substantial reference corpus; Puebla and Kerala each have only eight same-day pairs and weaker or genre-contaminated references. We compute per-event Spearman correlations with operational completeness. Reported p-values are SciPy 1.18's two-sided asymptotic `spearmanr` values; Table 3 applies Holm–Bonferroni correction jointly across

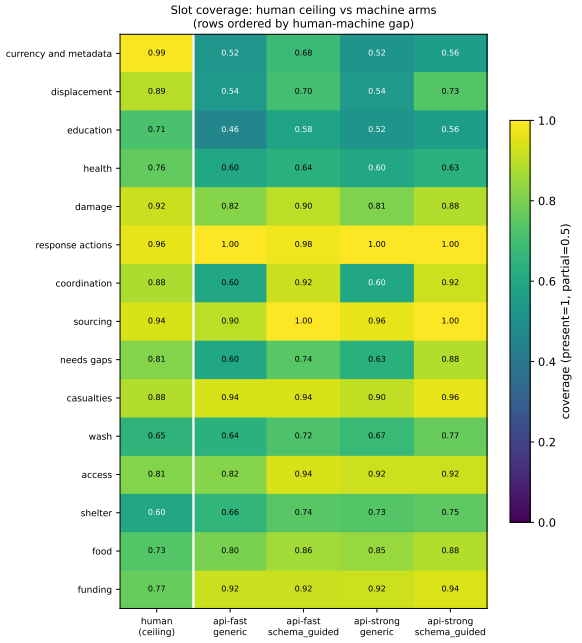


Figure 1: Idai slot coverage under judge `google/gemini-3.7-flash`. Rows are ordered by $G_j(z, e)$, human coverage minus the best exact machine arm.

its nine tests. The quality judge sees only a request to rate usefulness, clarity, organisation, and credibility on a 1–10 scale. Appendix A specifies the exact implementations and prompts.

6 Results

6.1 Presence produces a nominal inversion

Figure 1 shows the central mismatch. Under `google/gemini-3.7-flash` on Idai, machine arms address most schema slots as often as or more often than human reports. In Figure 2, `api-strong/schema-guided` is nominally above the empirical human ceiling: 0.827 [.783,.862] versus 0.821 [.781,.854]. This $+.0061$ difference is statistically indistinguishable from zero (95% difference CI $[-.0490, .0591]$; permutation $p = .852$).

The content-gated score remains nominally above under Gemini (0.824 vs. 0.821), so heading-only credit does not fully explain that judge's ordering; it remains statistically unsupported. The ordering is single-judge: under `openai/gpt-5.6-luna`, the best Idai machine arm scores 0.819 versus a human ceiling of 0.845, and the DeepSeek and Qwen judges also place the human ceiling above every machine arm. It is also genre-dependent: without Idai's three snapshots, the Gemini human ceiling

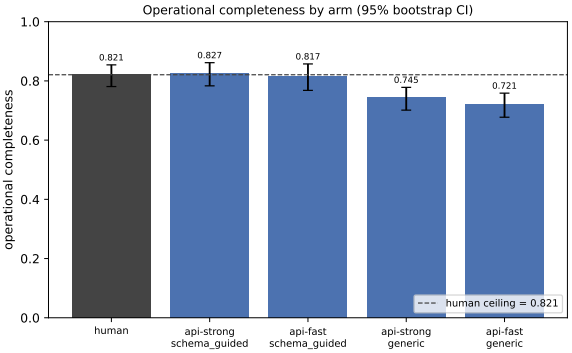


Figure 2: Idai $C_j(d)$ under judge `google/gemini-3.7-flash`; percentile 95% CIs resample unique documents within each arm (1,000 draws, seed 17). The nominal machine–human difference is not significant.

rises to 0.834, above every machine arm. As a supplementary small-corpus observation rather than replication of Idai, the descriptive mean over Kerala's 20 machine reports is 0.800 versus 0.743 for seven humans under Luna; its four arms are pooled only for this observation, not for gap or arm comparisons.

6.2 One gap is structural

Data currency and report metadata has $G_j(z, e) > 0$ in all 12 observed event–judge cells, spanning five events, three hazard types (tropical cyclone, earthquake, flood), and four judge families. Applying Table 1's manifest-keyed document rule gives a range of +0.200 to +0.458. The conclusion is unchanged when the 159-loss judge is excluded: removing Qwen's entire cell leaves every remaining cell positive (11/11). Under Qwen's observed-case analysis, 28/56 human versus 3/102 machine judgements are missing for this slot, so we do not use that cell to claim robustness to missingness.

The result separates medium-driven from event-driven gaps. Under `openai/gpt-5.6-luna`, Idai displacement looks like a second ceiling slot ($G_j = +0.288$), but the same defined quantity changes across Idai, Irma, and Maria (+0.288, -0.035, +0.070). Our a priori medium argument was that a social post is not the issuing instrument for a sitrep's organisation, report number, publication date, or stated authority. Empirically, stream-derived arms supplied that provenance less often than humans in all 12 observed cells. Displacement instead depends on what an emergency makes salient in public communication. A single-event study would have published it as a ceiling slot. The absolute flag B_j is less stable: for Idai metadata, $M_j = .520/.540/.615/.680$ across DeepSeek/Luna/Qwen/Gemini, so the .6 boundary splits the judges even though all gaps are positive.

6.3 The ceiling is in the source

Table 2 resolves the paradox without a model in the loop, while also exposing the regex's limits. Across Idai, Irma, and Maria, exact-number recall is 11.0%, 5.4%, and 8.9%; casualty recall is 33.0%, 52.1%, and 12.5%. Raw matching links 14.3–19.6% of official figures and 85.4–90.1% of machine figures to the stream. Manual review confirms 45/50 declared official absences, but only 4/25 official matches are semantic, versus 24/25 machine matches. Thus the official column is an ungated exact-match rate, inflated by common-number collisions, while sampled machine traceability is substantially more reliable. Puebla and Kerala are not averaged into the headline: each has only eight same-day pairs, Puebla's references are figure-sparse short products, and Kerala has the densest stream in the corpus.

6.4 Standard metrics fail differently

Table 3 and Figure 3 show three distinct failures on the three primary events. ROUGE-L is not significant on any of them and changes sign. BERTScore is significantly negative on all three primary events after Holm correction, with a stable magnitude of -0.43 to -0.47. This association is partly length-mediated: completeness correlates +0.532 with length and BERTScore correlates -0.446 on Idai; the partial correlation controlling for length remains -0.248. The metric penalises some of the elaboration that slot completeness rewards.

The generic LLM judge is less stable still: its correlation is positive on Idai and negative on Irma and Maria; neither formerly nominally significant cell survives correction. It is also circular within the pilot. When the generic judge and completeness scorer share a model family, $\rho = +0.250$ (raw $p = .011$); across families, $\rho = +0.174$ (raw $p = .080$). An evaluator selected on one disaster or model family can therefore transfer poorly.

To match prior system-level critiques at their level of analysis, we also average reports within each of the four generator–prompt arms and correlate the four arm means, descriptively only. Across Idai/Irma/Maria, BERTScore's arm-level ρ is $-1.0/-1.0/-.8$, ROUGE-L's is $-.8/.0/-.8$, and the Gemini quality judge's is $+.4/-.6/-.6$.

Event	Pairs	Official in stream	Machine in stream	Precision	Recall	Casualty recall
Idai	83	14.3 [9.5,19.7]	85.4 [83.7,87.3]	14.9 [12.9,17.0]	11.0 [8.7,13.5]	33.0 [25.7,40.7]
Irma	28	14.3 [7.0,23.1]	90.1 [87.4,92.8]	16.3 [12.4,20.1]	5.4 [4.0,6.8]	52.1 [40.6,62.5]
Maria	20	19.6 [11.6,28.3]	85.8 [81.8,90.0]	15.3 [10.2,21.0]	8.9 [6.4,11.8]	12.5 [0.0,25.0]
Puebla	*8*	*36.2 [35.3,37.0]*	*93.2 [90.4,96.1]*	*18.9 [16.4,21.9]*	*21.0 [17.3,24.3]*	*36.9 [29.3,44.5]*
Kerala	*8*	*39.0 [25.0,52.9]*	*90.7 [89.1,92.4]*	*5.9 [3.9,7.4]*	*21.9 [14.3,28.1]*	*22.9 [14.6,31.2]*

Table 2: Regex grounding percentages with percentile-bootstrap 95% CIs (10,000 draws, seed 17); no model judge. "Official" and "machine" mean exact-number matches to the day's stream. Official-in-stream averages and resamples unique valid event-days ($n = 21/7/4/2/2$); all other cells average and resample valid same-day machine–official pairs. Undefined denominators are dropped: casualty-recall $n = 79/8/12$ for Idai/Irma/Maria and Maria recall $n = 16$; other non-official cells use Pairs. Italicised events have only eight pairs. In manual audit, 5/50 declared absences were errors (10.0%, exact 95% CI [3.3,21.8]), as were 21/25 official matches (84.0% [63.9,95.5]) and 1/25 machine matches (4.0% [0.1,20.4]).

Metric	Idai	Irma	Maria
ROUGE-L	+.070 (.486)	+.178 (.483)	−.272 (.310)
BERTScore	$-.426^{***}$ (.0001)	$-.469^{**}$ (.006)	$-.439^{*}$ (.023)
Generic judge	+.250 (.067)	−.205 (.483)	−.375 (.067)

Table 3: Spearman ρ with $C_j(d)$ ($n = 102/48/43$); parentheses are Holm-adjusted p over all nine tests. $^{*}p < .05$, $^{**}p < .01$, $^{***}p < .001$. Completeness judge: `google/gemini-3.7-flash` for Idai and `openai/gpt-5.6-luna` for Irma/Maria. The schema-blind quality judge in every "Generic judge" cell is `google/gemini-3.7-flash`.

Judge	n	Accuracy	κ	κ_{lin}
DeepSeek	100	.790	.592	.627
Luna	99	.828	.617	.668
Qwen	90	.856	.681	.715
Gemini	100	.910	.799	.806
Primary Gemini by human label				
Label	Support	Precision	Recall	F1
ABSENT	9	.875	.778	.824
PARTIAL	19	.773	.895	.829
PRESENT	72	.957	.931	.944

Table 4: Blind author agreement on 100 seed-17, slot-stratified Idai document–slot items (50 professional, 50 machine). Accuracy is exact label agreement; κ is unweighted nominal Cohen's κ; κ_{lin} is a sensitivity check. The project plan set agreement $\geq .7$ without naming the statistic (Section 5). Gemini class metrics treat human labels as ground truth. Missing verdicts reduce n (Luna: 1; Qwen: 10). Judge models are `deepseek-v4-pro`, `openai/gpt-5.6-luna`, `qwen/qwen3.8-flash`, and `google/gemini-3.7-flash`.

With only four systems per event these are order summaries, not inferential evidence, but the BERTScore reversal is not solely a report-level phenomenon.

6.5 Human validation is judge-specific

All four judges reach the plan's .7 threshold on raw agreement, the statistic our pre-annotation scoring code applied. Only Gemini also reaches .7 on the separately reported chance-corrected κ; it is the judge behind the Idai nominal inversion and Figures 1–2. Its agreement is high on both professional-report items ($\kappa = .861$, raw .940) and machine-report items ($\kappa = .730$, raw .880). Luna, which supplies the principal cross-event check, falls from $\kappa = .762$ on professional-report items to .470 on machine-report items. The labels therefore support the Gemini-based pilot measurement more strongly than the Luna-based replication, rather than validating LLM judging in general.

7 Implications for Humanitarian Practice

We ran no deployment or practitioner study, and our drafts came from two DeepSeek models reading English-language posts, so what follows is what the measurements imply for that setting, not an observed effect; nor did we measure whether drafts add value by arriving before official reports.

For information managers and report authors. On Idai, under our best-validated judge, drafts prompted with the schema covered it about as fully as professional reports (Figure 2), although truncating long professional reports for judging understates their coverage. Their figures and provenance were the weak point: on Idai, Irma, and Maria, exact matching linked only 5–11% of same-day official figures to them (Table 2), and on every event they stated their own as-of date, issuing organisation, and update number less often than professional reports. A draft can therefore look finished without being sourced: reviewers should check its figures, dates, and sources, not its headings. It is best treated as an unverified situational signal whose figures need re-sourcing from official or assessment channels and whose provenance must come from the issuing office; it should not circulate

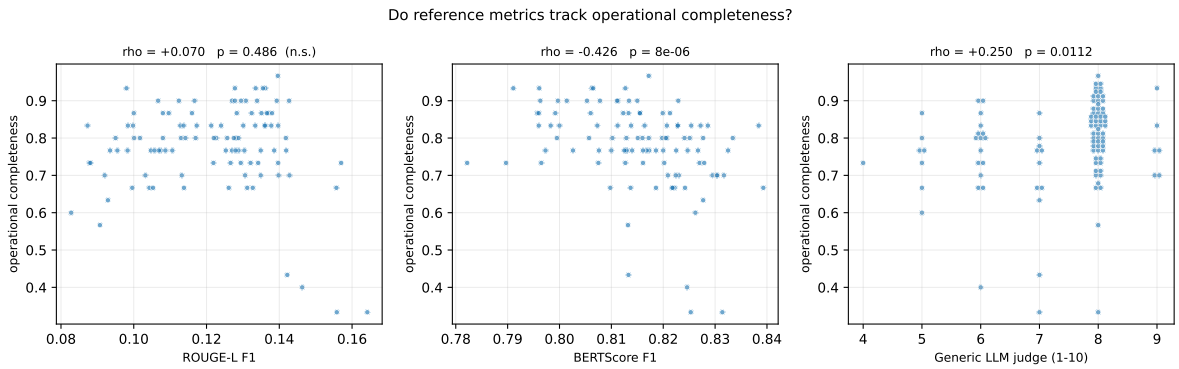


Figure 3: Idai reference metrics against $C_j(d)$ scored by `google/gemini-3.7-flash`; the schema-blind quality score is from the same model. Panel p-values are raw; Table 3 gives the nine-test Holm correction. In the generic-judge panel, exactly coincident points are spread slightly sideways and upward for visibility; statistics use the unmodified values.

Event	n_H	Luna human	Gemini human
Idai	56	.845 [.808,.875]	.821 [.781,.854]
Irma	19	.739 [.695,.777]	.709 [.653,.758]
Maria	25	.763 [.696,.825]	.717 [.636,.789]
Puebla†	14	.567 [.493,.640]	.500 [.431,.564]
Kerala	7	.743 [.695,.781]	.779 [.700,.839]

Table 5: Per-event empirical human ceiling, the mean of $C_j(d)$ with 95% document-bootstrap CIs. Judges are `openai/gpt-5.6-luna` and `google/gemini-3.7-flash`. †Puebla is the collected-product score, not used as a sitrep ceiling because 10/14 products are short infographics or snapshots.

under an agency's name. Because generations can repeat phone numbers and email addresses from crisis posts (Appendix A), drafts also need screening for personal data.

For organisations piloting or funding reporting tools. Similarity to professional reports and generic LLM quality ratings did not track completeness here (Table 3). Pilots and procurement can instead test a tool on past emergencies: check its figures against same-day official reports, compare it with professional reports of the same type rather than with a perfect score, and ask which inputs it used (Section 8).

For tool builders. Most official figures never appeared in the day's HumAID posts, a labelled sample of English-language tweets, so a stronger model could not have recovered them from that input. Drafting is therefore likely to be more useful when connected to the channels official figures come from, with social media flagging emerging needs and locations for verification; we did not test this. Gaps also vary by event: drafts covered displacement much less often than professional reports on Idai, but not on Irma or Maria, so a gap seen in one emergency needs re-checking in the next.

For professional reporting itself. Professional reports also scored below 1.0 (Table 5), partly because shorter products such as snapshots and infographics are not meant to cover every slot. A practitioner-reviewed slot list, adapted to each product type and extended with categories ours lacks or only partly covers, such as protection, could help authors check for unintended omissions; whether that improves operational use needs evaluation with practitioners.

8 What the Shared-Task Community Should Adopt

As of 31 August 2026, the official LT4CPR guidelines score report sections 3–11 using text similarity, bullet-level alignment, and confidence-label agreement, then average replicates within windows, windows within crises, and crises; the public task page also named ROUGE, BLEU, and BERTScore as anticipated automatic metrics. The official results, released since, rank systems by the mean of BERTScore F1 and BLEURT (LT4CPR Shared Task Organisers, 2026); because BERTScore was negatively associated with completeness on all three primary events (Table 3), that ranking need not track operational completeness, although we did not test BLEURT or the task's own reference reports. The results suggest seven explicit deltas to that protocol rather than another single metric.

1. **Use an empirical human ceiling.** Across

the three primary events and two judges in Table 5, human reports score 0.71–0.85, so a system score is not a percentage of perfection. *Delta:* add human calibration, which the current protocol does not define.

2. **Do not rank systems by presence alone.** Slot coverage can place machine reports nominally above professional ones without a significant difference or cross-judge replication. *Delta:* show bullet-level alignment (now only a secondary workbook metric) and confidence-label agreement (specified in the guidelines but not computed by the official evaluator) beside the similarity-based primary score on the leaderboard, and replace any presence-only leaderboard interpretation.

3. **Require figure-level grounding.** Report precision, recall, and casualty recall against the official record alongside slot coverage; Table 2 gives primary-event casualty recall of 33.0/52.1/12.5%. *Delta:* add number-specific exactness beyond text similarity and bullet alignment.

4. **Report G_j with intervals, not B_j.** The 0.6 inequality in Section 5 lies inside the observed .520/.540/.615/.680 spread for one slot; two judges flag it and two do not. *Delta:* add uncertainty to the current point-score aggregation.

5. **Report per-judge missingness.** Unparsable outputs were 0/19/159/0 across four judges, with the largest loss concentrated in the slot carrying the largest gap. *Delta:* keep the evaluator's coverage and metric-status checks, and extend them to per-judge parse-failure rates whenever an LLM judge is used.

6. **Use a human-validated cross-family judge, or disclose that this was not done.** Same-family agreement can masquerade as validity, and Table 4 shows that cross-family judges are not interchangeable. *Delta:* add blind human validation and cross-model robustness when an LLM judge supplements official metrics.

7. **State the source ceiling.** When inputs are social media, evaluation must quantify what share of reference facts the input actually contains. *Delta:* report the task's existing input-to-reference audit with the official results, and extend it from the tweets each reference cites to the whole input, distinct from output-to-reference scoring.

Together these measures separate three questions that a shared task should not collapse: whether a report names the right topics, whether its claims match an operational reference, and whether the supplied evidence could support those claims.

9 Conclusion

Professional reporting practice reveals a sharp distinction between looking complete and being operationally grounded. Under one judge, an LLM arm is nominally above but statistically indistinguishable from the empirical human ceiling; the ordering does not replicate under Luna and is smaller than the truncation bias measured among sampled affected reports. Across the measured primary events, systems reproduced little of the official quantitative record, and across all 12 observed event–judge cells they supplied issuing provenance less often than professionals. Crisis-report evaluation must measure source availability, figure grounding, and judge-indexed human-relative completeness together. For practitioners, a draft built from social media largely restates figures already in that day's posts; it cannot stand in for the official record or carry an agency's authority.

Limitations

Judge validity. The blind validation in Table 4 uses one annotator, who also approved the schema, and only Idai; it may therefore overstate agreement with independent practitioners and does not validate transfer to other events. All four judges reach the plan's .7 threshold on raw agreement, but only Gemini reaches .7 on the separately reported chance-corrected κ. On unique document–slot complete cases, unweighted nominal Fleiss' κ between Luna and Gemini is 0.653 ($n = 2{,}351$), 0.646 ($n = 1{,}005$), and 0.707 ($n = 1{,}020$) on Idai, Irma, and Maria. These values lie outside the acceptance rule and measure consistency, not correctness; Qwen's missingness is non-random. All LLM-judged conclusions should therefore be read alongside the model-independent source-availability result.

Schema and corpus scope. The schema was induced from all 56 Idai reports, 44 of them OCHA's, so it chiefly reflects OCHA reporting practice, and

its Idai ceiling is in-sample; held-out professional reports on Irma and Maria score 0.71–0.76 (Table 5), below Idai's 0.82–0.85. Its conceptual space was shaped before human review (Section 4): our instructions seeded five reporting conventions, one model proposed the codes from truncated text for the 12 longest reports, a 25-candidate cap left clusters holding 57% of the codes unreviewed, with drafted consolidation, capped in advance at 15 slots. The seeded provenance conventions are nonetheless pervasive in practice: an LLM-free lexicon released with our code counts 395 as-of date markers and 413 attribution phrases in the same reports. The model that proposed the codes and named the candidates, `deepseek-v4-pro`, is also the `api-strong` generator, whose schema-guided arm receives the slot names and definitions; slots shaped by its salience could favour its outputs. Completeness therefore describes the 15 measured slots, not all of reporting practice, and the omissions cut both ways: agency-reported categories such as protection and logistics would likely favour professional reports, whereas public-facing ones such as search and rescue or hazard forecasts could favour stream-derived drafts. No second researcher or practitioner independently coded the reports, so schema construction has no inter-coder reliability estimate; the released audit trail allows others to run that check. These limits bear on the completeness-based results (the inversion, the provenance gap, the human ceilings, and the metric correlations), not on figure grounding or source availability, which use neither the schema nor a model judge. HumAID and the generated reports are English-language, all drafts come from one generator family, and HumAID is itself a labelled sample of about 77K tweets drawn from about 24M (Alam et al., 2021), so the source ceiling describes this sample rather than social media at large; the ReliefWeb collection includes a small number of non-English products, limiting multilingual claims. Irma and Maria have 19 and 25 evaluated human reports, and Kerala has seven; their ceiling estimates are noisier than Idai's. Puebla is genre-contaminated: 10 of 14 judged documents are infographics or snapshots, lowering its Luna collected-product score from 0.725 for sitreps alone to 0.567. We therefore treat Puebla and Kerala as supplementary.

Measurement scope. Regex figure matching can count coincidental numeric matches and miss semantically equivalent expressions. The manual audit quantifies both modes: 5/50 declared absences and 21/25 official exact matches were erroneous, while only 1/25 machine exact matches was erroneous. A machine figure absent from an official report is unverified, not necessarily false. Original judge inputs truncated reports after 20,000 characters; 56/100 evaluated primary-event professional reports exceeded that cap (30/56 Idai, 11/19 Irma, 15/25 Maria). We re-judged a seed-17 random sample of 12 affected reports (four per primary event) at full length with Gemini: the paired ceiling shift was +0.056 [+.031,+.081] (10,000 document-bootstrap draws). This quantifies bias among long reports, not the whole corpus, but is larger than the nominal Idai machine advantage. Presence also risks rubric leakage; our deterministic body gate lowered the nominal Gemini machine score from .827 to .824 but did not reverse it.

Ethical Considerations

Data protection: ReliefWeb reports are public. Social-media text is not redistributed, no user identifiers appear, and no post is quoted verbatim.

Dual use and deployment hazards: The system is intended as evaluation infrastructure, not an autonomous reporting tool. An operationally incomplete report that appears complete could misdirect scarce resources or falsely convey authority; the statistically unsupported nominal ordering still illustrates this deployment hazard.

References

Firoj Alam, Umair Qazi, Muhammad Imran, and Ferda Ofli. 2021. HumAID: Human-annotated disaster incidents data from Twitter with deep learning benchmarks. *Proceedings of the International AAAI Conference on Web and Social Media*, 15(1):933–942.

Cody Buntain, Amanda Lee Hughes, Richard McCreadie, Benjamin D. Horne, Muhammad Imran, and Hemant Purohit. 2023. CrisisFACTS 2023 – overview paper. In *The Thirty-Second Text REtrieval Conference Proceedings (TREC 2023)*, volume 1328 of *NIST Special Publication*. National Institute of Standards and Technology (NIST).

Riccardo Cantini, Cristian Cosentino, Fabrizio Marozzo, Domenico Talia, and Paolo Trunfio. 2025. Harnessing prompt-based large language models for disaster monitoring and automated reporting from social media feedback. *Online Social Networks and Media*, 45:100295.

Nathanael Chambers. 2013. Event schema induction with a probabilistic entity-driven model. In *Proceed-*

ings of the 2013 Conference on Empirical Methods in Natural Language Processing, pages 1797–1807, Seattle, Washington, USA. Association for Computational Linguistics.

DeepSeek. 2026. DeepSeek API change log. Official API documentation: https://api-docs.deepseek.com/updates/.

Daniel Deutsch, Rotem Dror, and Dan Roth. 2022. Re-examining system-level correlations of automatic summarization evaluation metrics. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 6038–6052, Seattle, United States. Association for Computational Linguistics.

Daniel Deutsch and Dan Roth. 2021. Understanding the extent to which content quality metrics measure the information quality of summaries. In *Proceedings of the 25th Conference on Computational Natural Language Learning*, pages 300–309, Online. Association for Computational Linguistics.

Satu Elo and Helvi Kyngäs. 2008. The qualitative content analysis process. *Journal of Advanced Nursing*, 62(1):107–115.

Muhammad Imran, Prasenjit Mitra, and Carlos Castillo. 2016. Twitter as a lifeline: Human-annotated Twitter corpora for NLP of crisis-related messages. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1638–1643, Portorož, Slovenia. European Language Resources Association (ELRA).

Klaus Krippendorff. 2018. *Content Analysis: An Introduction to Its Methodology*, fourth edition. SAGE Publications, Thousand Oaks, CA.

William D. Lewis, Haotian Zhu, Keaton Strawn, and Fei Xia. 2025. Tapping into social media in crisis: A survey. In *Proceedings of the Fourth Workshop on NLP for Positive Impact (NLP4PI)*, pages 306–331, Vienna, Austria. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-Eval: NLG evaluation using GPT-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2511–2522, Singapore. Association for Computational Linguistics.

LT4CPR Shared Task Organisers. 2026. LT4CPR shared task on automatic situation report generation. Official task page: https://lt4cpr.github.io/aacl2026-workshop-LT4CPR/shared-task.html. Guidelines: https://github.com/LT4CPR/2026_aacl_workshop;.

Marcel Nawrath, Agnieszka Nowak, Tristan Ratz, Danilo C. Walenta, Juri Opitz, Leonardo F. R. Ribeiro, João Sedoc, Daniel Deutsch, Simon Mille, Yixin Liu, Lining Zhang, Sebastian Gehrmann, Saad Mahamood, Miruna Clinciu, Khyathi Chandu, and Yufang Hou. 2024. On the role of summary content units in text summarization evaluation. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 272–281, Mexico City, Mexico. Association for Computational Linguistics.

Ani Nenkova and Rebecca Passonneau. 2004. Evaluating content selection in summarization: The pyramid method. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, pages 145–152, Boston, Massachusetts, USA. Association for Computational Linguistics.

Arjun Panickssery, Samuel R. Bowman, and Shi Feng. 2024. LLM evaluators recognize and favor their own generations. In *Advances in Neural Information Processing Systems*, volume 37, pages 68772–68802.

Hossein Rajaby Faghihi, Bashar Alhafni, Ke Zhang, Shihao Ran, Joel Tetreault, and Alejandro Jaimes. 2022. CrisisLTLSum: A benchmark for local crisis event timeline extraction and summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 5455–5477, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Hope Schroeder, Marianne Aubin Le Quéré, Casey Randazzo, David Mimno, and Sarita Schoenebeck. 2025. Large language models in qualitative research: Uses, tensions, and intentions. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, pages 481:1–481:17, Yokohama, Japan. Association for Computing Machinery.

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. 2020. AutoPrompt: Eliciting knowledge from language models with automatically generated prompts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4222–4235, Online. Association for Computational Linguistics.

United Nations Office for the Coordination of Humanitarian Affairs. 2023. Situation report (sitrep). OCHA Knowledge Base: https://knowledge.base.unocha.org/wiki/spaces/imtoolbox/pages/42047193/Situation+Report+Sitrep.

Qile Wang, Moath Erqsous, Kenneth E. Barner, and Matthew Louis Mauriello. 2025. LATA: A pilot study on LLM-assisted thematic analysis of online social network data generation experiences. *Proceedings of the ACM on Human-Computer Interaction*, 9(2):CSCW124:1–CSCW124:28.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with BERT. In *International Conference on Learning Representations*.

A Reproducibility and Audit Trail

All LLM calls pass through one client with a disk cache, a fail-closed cost gate, and a JSONL log containing request hashes, token counts, costs, and lengths but never text. Seeds are fixed at 17. The recorded spend is $31.35, including $1.72 for the untruncated sensitivity run. The code and data release is available at `https://github.com/techsachinkr/llm-sitrep-evaluation/`. It includes `supplementary/reproducibility/`: every prompt verbatim with sentinel payloads, the frozen 15-slot schema with definitions and two synthetic examples per slot, public model identifiers and call dates, the 125-report ReliefWeb manifest, and text-free matcher-audit labels with context hashes. The repository also releases the schema audit trail (`data/processed/schema/`): the induction sample, every open code's name and definition by report, all 382 clusters, the 25 candidates, and the merge rationales, without evidence spans. The manifest specifies first-20,000-Python-character truncation followed by `[...truncated...]`, as well as titles, sources, verified dates, retrieval locators, hashes, and exclusions.

The publication tables are generated by `python -m src.paper_analysis`. `corpus_manifest.csv` records document and judgement denominators; `slot_coverage_by_judge_arm.csv` releases every coverage_j numerator and denominator; the document, inversion, content-gate, truncation, agreement, Holm, and four-arm tables reproduce the remaining claims. `human_validation_pairs.csv` releases the 100 text-free human and withheld judge labels, and `human_validation_summary.csv` recomputes Table 4, including per-judge exclusions. `per_judgement.csv` releases the text-free raw rows, including failures and duplicate keys, so the last-row/observed-case rules are independently checkable. Figure 2 reads the corrected Gemini table directly.

Measurement specification. The artifact publishes the exact number and date-exclusion regexes, scale dictionary, unit handling, and matching rule: remove comma, ASCII space, and non-breaking space; apply thousand/million/billion/lakh/crore scales; keep percentages separate; deduplicate within documents; drop recognised dates and bare years 1900–2100; and match `(unit, round(value,6))` exactly with no tolerance. It pins `rouge-score==0.1.2` with `RougeScorer(['rougeL'], use_stemmer=True)` and maximum-F1 multi-reference selection; `bert-score==0.3.13`, `roberta-large` revision `722cf37b`, English, no baseline rescaling, and multi-reference maximum; and `sentence-transformers/all-MiniLM-L6-v2` revision `1110a243`, normalised embeddings, average-linkage precomputed cosine distance, and threshold .35. The full revisions, tokenisation, packages, runtime, and 76,484-row non-text HumAID ID/hash manifest are machine-readable.

Public machine-output text is deliberately omitted after a spot check found that generations can repeat phone numbers and email addresses from crisis posts. Each of 264 outputs instead has a request hash, output hash, call time, and public model ID; licensed HumAID records have IDs and text hashes. Thus the released numerical tables and measurement functions are bit-for-bit auditable, and acquired inputs can be hash-verified, but fresh generation from mutable aliases is not promised to be deterministic. This is a privacy and provider-version boundary, not an unreported omission.

Four silent implementation defects were found and corrected. First, the DeepSeek client sent `max_completion_tokens`, which that endpoint ignored; 75% of 3,532 calls exceeded the configured cap. Second, hyphenated ReliefWeb keys and underscored HumAID keys prevented reference pairing; canonical normalisation repaired it. Third, a 32,000-token generation probe truncated 25% of schema-guided versus 8% of generic Irma map calls; the reported replication uses 48,000, leaving 1.0% truncation overall and 0.0% for schema-guided Irma. Fourth, reference-metric CSVs recorded the default judge alias even when `–judge-model judge_gemini` was passed. Request logs establish `google/gemini-3.7-flash`; labels and code are corrected, with no numeric score changed. Regression tests cover all four repairs.

Role	Public model identifier(s)	Temp./top-p	Before fix	Corrected cap
Open coding	`deepseek-v4-pro`	default / unsent	6,000 requested, unsent	6,000
Cluster naming	`deepseek-v4-pro`	default / unsent	900 requested, unsent	900
Generation	`deepseek-v4-pro`, `deepseek-v4-flash`	default / unsent	2,000 requested, unsent	48,000
Idai slot judging	`deepseek-v4-pro`, `openai/gpt-5.6-luna`	default / unsent	DS 400 unsent; Luna 400	400
Idai slot judging	`qwen/qwen3.8-flash`, `google/gemini-3.7-flash`	default / unsent	3,500; 1,200	unchanged
Replication slot judging	`openai/gpt-5.6-luna`, `google/gemini-3.7-flash`	default / unsent	1,200 or 3,500	unchanged
Generic quality	`google/gemini-3.7-flash`	default / unsent	512	unchanged
Full-text sensitivity	`google/gemini-3.7-flash`	default / unsent	1,200	unchanged

Table 6: Decoding parameters by LLM role. Temperature was omitted (provider default); the wrapper does not implement or transmit top-p. "Unsent" is the first defect listed at the end of this appendix: the DeepSeek endpoint ignored the wrong parameter name. The Idai generation, coding, naming, and DeepSeek-judge calls are the pre-fix reported calls; four-event replication generation used the corrected 48,000 cap. The artifact disaggregates every event–judge cap and records effort.

Is Machine Translation Ready for the Next Crisis or Pandemic?

Vipasha Bansal William D. Lewis Nicole Mihaylova Elizabeth Brown Benjamin Pong
Computational Linguistics Program
University of Washington
{vipashab, wlewis2, nicolemi, egollhof, benpong}@uw.edu

Abstract

Communication in times of crisis is essential. However, there is often a mismatch between the language of governments, aid providers, doctors, and those to whom they are providing aid. It is reasonable to turn to commercial MT tools in these scenarios. But how effective are these tools for translating to and from low resource languages, particularly in the crisis or medical domain? In this study, we evaluate 5 commercial systems used for MT (Google Translate, Microsoft Translator, GPT, Gemini and Claude) against the TICO-19 dataset, which contains pandemic-related sentences from a large set of high priority languages spoken by communities most likely to be adversely affected in the next pandemic. We assess the current degree of "readiness" for this next pandemic based on the usability of the output translations.

1 Introduction

COVID-19 was the worst pandemic in over a century. In order to slow the spread of the virus and to reduce hospitalizations and death, it was vital to communicate information to vulnerable populations. However, official communications from health agencies such the World Health Organization were only available in high resource languages.

The TICO-19 dataset (Anastasopoulos et al., 2020) was created during the pandemic to encourage MT research in under-resourced languages, specifically languages whose communities might be impacted by the pandemic but had not yet been touched by it. [1,2] Translators without Borders (TwB) was the driving force behind the choice of languages in TICO-19; TwB was, and still is, engaged in translating varied content for numerous languages spoken across Africa, Southeast Asia and other parts of the world.

The dataset contains n-way parallel test and dev sets across 36 languages. At the time of TICO-19's development, more than half of its languages had no commercial translation systems available (*e.g.,* Dari, Kurdish, Zulu). This situation has changed markedly in the ensuing years: 36 of the 37 languages in the TICO-19 test set are now supported by at least one commercial system. This change in MT availability raises the following questions: How good are commercial MT systems at translating into and out of these languages? In particular, how good are they for translating pandemic-related content? Would they be adequate for aid providers, medical professionals, or governments to communicate with speakers of these languages on the ground during a future public health crisis?

Our study measures the quality of five existing commercial systems in the context of pandemic response. We use the TICO-19 dataset to systematically evaluate two major MT-specific providers, Google Translate and Microsoft Translator. Given the global popularity of LLMs, including in MT research (Kocmi et al., 2024), and their potential competitiveness with existing translation services (Hendy et al., 2023; Robinson et al., 2023), we also evaluate three popular LLMs: GPT, Gemini and Claude. For the translation providers, we captured three snapshots over time, in 2023, 2025, and 2026, giving us a sense not only of their current performance, but also of how this performance has changed over the past three years.

We recognize the value of the output generated in our study to various fields, including Machine Translation, crisis response, medicine, and public health. We see this paper as the beginning and want to facilitate further research in this area. To this end, we are releasing all data collected in our study here: https://mt-crisis-ready.github.io.

[1] TICO-19 was created in mid-2020, early in the pandemic.

[2] A secondary goal of TICO-19 was to provide translation memories and glossaries for translators. These, along with the MT benchmark data used in this paper, are available for download from the TICO site: https://tico-19.github.io/.

Proceedings of the First Workshop on Language Technologies for Crisis Preparedness and Response, pages 78–114
November 9, 2026 ©2026 Association for Computational Linguistics

2 Background and Related Work

2.1 The Use of MT and Other Language Technologies in Crisis Scenarios

There has been a large amount of research on the use of language technologies in crisis response over the past several years. Lewis et al., 2025, for instance, reviews 355 papers published over a five-year timeframe ending in 2025. These papers focus on various language tasks, albeit specific to social media use in crises. Most concentrate on classification, with a smattering of work on other technologies including NER, geo-location, summarization, and topic modeling. Oddly, given the prevalence of large-scale crises that occur in areas where English is not spoken (see Tin et al., 2024, and discussion in Lewis et al., 2025), almost none of the papers in the five-year study show work on translation or MT as it relates to crises and social media.

That is not to say that there has been no work on crises and translation. The INTERACT project[3], an EU-funded research consortium, produced a number of publications on translation in crises, focused on various aspects of *crisis communication* including translation workflows and training translators (Federici and Cadwell, 2018), translation needs assessments in times of crisis (O'Brien and Federici, 2019), and ethics and crisis translation (Hunt et al., 2019). Similarly, Bird, 2022 looks at crisis communication holistically, considering cross-cultural communication and the use of "trade" languages (*i.e.,* regional *linguae francae*). Further, Lankford and Way, 2024 examine the use of LLMs for low-resource MT in a pandemic, and Roussis, 2022 walks through the data collection and training of an end-to-end neural model dedicated to pandemic translations. And of course, Lewis, 2010 established the crucial need for commercial MT for low-resource languages in a crisis scenario.

Both commercial MT tools and LLMs have seen broad general-purpose usage and acceptance, and commercial MT systems are often used in medical settings, including for doctor-patient communication (Khoong et al., 2019; Khoong and Rodriguez, 2022). It is likely that MT is used similarly in crises, with some work showing MT being used by crisis managers in the United States (Wilson et al., 2026; Foubert et al., 2026).

[3]https://cordis.europa.eu/project/id/734211

2.2 Quality, Usability and Readiness

The end goal of much MT research is to design systems that produce high-quality translations across domains, *e.g.,* in bake-offs and competitions (Haddow et al., 2025; Salesky et al., 2026). Quality is often measured for competing systems using automated metrics, such as CHRF++ (Popović, 2017) and BLEU (Papineni et al., 2002), or human evaluation (Graham et al., 2013).

Our focus in this paper however, is to look at the readiness of competing systems in the context of pandemic response. Our goal is therefore to assess whether a translation system is good enough to be used in the field during an active crisis. The key to this is usability: it is essential that the translated information be *understood*, whether the translations are *to* a population from, say, aid workers or government bodies (O'Brien and Federici, 2019), or *from* a population, say, as posted to social media (Lewis et al., 2025; Imran et al., 2015).

So how does one measure usability, especially in the context of MT usage in a crisis scenario? One could ideally identify tasks that the MT would be used for in the field, *e.g.,* communicating directly with those affected, or sending emergency alerts to multilingual populations (Wilson et al., 2026; Foubert et al., 2026). Although usability is not the same as quality, there is a minimum threshold for quality that translations must meet, or any task dependent on those translations will fail. But how do we determine if output translations meet such a threshold? Further, how do we set this threshold?

We use four evaluation metrics against our translated output: BLEU (Papineni et al., 2002), CHRF++ (Popović, 2017), BERTScore (Zhang et al., 2020), and COMET (Rei et al., 2020). Of these, BLEU and CHRF++ are n-gram based; despite their dependence on matching surface output with reference(s),[4] they have been shown to correlate well with human evaluation, the gold standard (Papineni et al., 2002; Popović, 2017).

BERTScore and COMET are "semantic" based, and instead focus on the holistic "meaning" of the output (Zhang et al., 2020). Unfortunately, these metrics do not adequately support many of the low resource languages in the TICO-19 dataset. This lack of coverage, along with issues with reliable performance even on supported low-resource lan-

[4]One of the main weaknesses of n-gram metrics is that an output could preserve meaning when compared with a reference but be assigned a low score because there is low n-gram overlap between the two (GoogleCloud, 2023).

guages and on specialized (or unseen) domains, has been noted in various studies (NLLB-Team et al., 2022; Tang et al., 2024; Falcão et al., 2024; Zouhar et al., 2024a,b). In particular, Zouhar et al. (2024b) notes the poor performance of COMET in the biomedical domain—a domain close to that of TICO-19.

Given the limited language coverage of BERTScore and COMET across the TICO-19 dataset, we turned to BLEU and CHRF++ to build our threshold. We follow a traditional rule-of-thumb to interpret BLEU scores, captured by Google's MT API documentation [5] (Google Cloud, 2025) and shown in Table 1. We see 30 as our usability threshold, as this indicates "understandable to good" quality translations (per the table). With CHRF++, we follow NLLB-Team et al. (2022), who identified a CHRF++ score of 45 as a minimum threshold. Although their task was focused on toxicity detection, we see this as a reasonable starting point. Section 4.3 provides further details on how we use these metrics for thresholding, and Section 5.2.3 outlines the results of a preliminary human evaluation to verify our proposal. [6]

BLEU Score	Interpretation
< 10	Almost useless
10 - 19	Hard to get the gist
20 - 29	Clear gist, major grammatical errors
30 - 40	Understandable to good translations
40 - 50	High quality translations
50 - 60	Very high quality, fluent translations
> 60	Quality often better than human

Table 1: BLEU Score Interpretation

3 The TICO-19 Dataset

The dataset consists of 3,071 sentences (971 dev, 2,100 test) drawn from Wikipedia, PubMed, and existing corpora such as the CMU English-Haitian Creole dataset (Anastasopoulos et al., 2020). The sentences were translated into 37 languages, making TICO uniquely n-ways parallel. There are 9 *pivot* languages, representing global or regional *linguae francae*; 20 *priority* languages identified by TwB based on translation requests during the pandemic or their importance to partner aid organizations such as the Red Cross; and 8 additional languages labeled *important*, included because they have large populations of speakers across Asia. Although these labels were not explicitly designed to map to language resource level, they align remarkably well: Pivot languages tend to be high resource, Priority languages low resource, and Important languages somewhere in between.

The dataset's structure allows for bidirectional evaluation that enables the measurement of MT quality in scenarios where English could be either the source, *e.g.,* of public health content, or the destination, *e.g.,* for translated reports or social media. Domain-source correspondence is shown in Table 2. We utilize these subdomains to examine potential training data contamination (see Appendix F).

Two languages were excluded from our study: the data for Congolese Swahili was unavailable for download, and the data file for Nuer was corrupted.

Data Source	Domain	# Sentences
CMU	Conversational, medical	104
PubMed	Scientific, medical	676
Wikipedia	General	1,153
Wikivoyage	Travel	dev only
Wikinews	News	66
Wikisource	Announcements	99

Table 2: TICO-19 Test Data Statistics by Domain

4 Methods

4.1 Dedicated Machine Translation Systems

We evaluated Google and Microsoft Translation services for three time frames: April-May 2023, April-July 2025, and May-June 2026. We used Microsoft Azure Cognitive Services Translator in 2023 and Azure Translator in 2025 and 2026[7] and Google Cloud's API throughout to access each model. The test sentences were identical across all three snapshots.[8]

4.2 LLM systems

We evaluated three GPT models (GPT-4o, GPT-5.4, and GPT-5.4-mini), Gemini-3.1-Flash-Lite, (for which translation is listed as a recommended use case),[9] and Claude-Haiku-4.5. The models were ac-

[5] https://cloud.google.com/translate/docs/advanced/automl-evaluate

[6] We realize that there is significant debate about using n-gram based metrics, particularly BLEU, for thresholding. Please refer to Appendix K for additional discussion on this topic and related issues.

[7] Although there were changes to the Azure API between 2023 and 2026, likely indicating reorganizations within Microsoft, the underlying translation service remained the same.

[8] The TICO-19 benchmark dataset has remained stable since its release in 2020, with no changes made to its test data.

[9] https://ai.google.dev/gemini-api/docs/models/gemini-3.1-flash-lite

cessed through Azure's OpenAI Service, Google's Cloud API, and Anthropic's Claude Platform. [10]

Unlike the dedicated MT systems, which provide specific and documented language translation support, LLMs can generate text in multiple languages but are not explicitly optimized for translation. While an LLM can, in theory, be prompted to translate to and from *any* language, the lack of training data for low-resource languages means that, in reality, a limited number of languages are supported (Chua et al., 2025). We determined language support based on the inverse proportion of sentences where the model failed to produce a translation for a given language (see Appendix E). A failure consisted of blank output, a statement that the model could not do the task, output translations in the wrong language, or string repetitions.

For the three GPT models, some translations incorrectly triggered OpenAI's content filters (see Appendix G). While problematic for usability, unlike the failure types listed above, this was not a failure to produce a translation, but rather a failure to output it to the user. Therefore, filtered outputs were not used to determine language support.

When evaluating supported languages, failed outputs were not excluded from the final scoring; this ensured consistency and comparability of results across languages and systems. Instead, filtered sentences were treated as blank output, while the remaining failure types were retained and evaluated as is. Error handling was consistent across metrics.

4.3 Evaluation Metrics

BLEU and CHRF++ were calculated using the SacreBleu package (Post, 2018).[11] For Khmer, Myanmar, and Chinese, which do not use whitespace to mark word boundaries, we used the character-based CHRF metric (Popović, 2015) instead of CHRF++.[12,13] For BERTScore (Zhang et al., 2020), we used the `bert_score` Python package with default settings. Baseline rescaling—which normalizes scores to fall between 0–1—was applied to translations into English.[14] For COMET, we used the XLM-RoBERTa-based COMET model released as part of the WMT22 Quality Estimation shared task (Zerva et al., 2022). The model was loaded using the `comet` Python package.

To ensure consistency in evaluating longitudinal data, BLEU and CHRF++ scores for all 2023 and 2025 outputs were recalculated alongside 2026 outputs; therefore the same version of the SacreBleu toolkit (and the associated tokenizers) was used throughout. For the neural metrics (BERTScore and COMET), we used identical models and hyperparameters for all snapshots.

Kumar et al. (2026) recommend evaluating translations against both BLEU *and* CHRF++ together, particularly for low resource languages, as the metrics provide complementary insights. Following this, we use both in determining whether translations are usable. As outlined in Section 2.2, we use 30 and 45 as individual thresholds for BLEU and CHRF++, respectively. We consider translations that score above *both* thresholds to be usable, and translations that meet only one threshold *borderline usable*. Translations that do not pass either threshold are considered unusable. See Section 5.2.3 for a preliminary human evaluation of usability, and preliminary support for these thresholds.

5 Results

Figures 1 (a)-(f) show the CHRF++ scores for each system and language, separated by TICO classification (Pivot, Important, Priority) and translation direction. Example scores showing overall trends are in Table 3, and full results across all four metrics are in Tables 15–22, Appendix A.

5.1 System Comparison

5.1.1 Coverage

All 36 languages that we evaluated are supported by GPT-5.4-mini in both translation directions. The full size version (GPT-5.4) provides partial support for Kurdish Kurmanji (in the EX direction, it produced all output in Kurdish Sorani instead, as did Gemini), and full support for the remainder. Similarly, Google Translate provides full support for all languages but Dari, which is partially supported.[15] Dinka, Nigerian Fulfulde, and Kanuri were problematic throughout; none are supported by Microsoft or Claude, Gemini provides only XE support for each, while GPT-4o does not support Dinka and provides only XE support for the others.

[10] See Appendix C for details on model configurations.

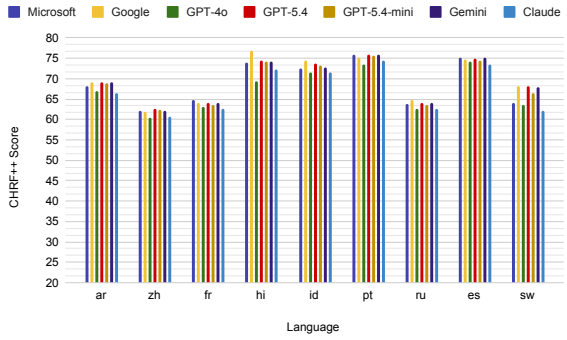
[11] https://github.com/mjpost/sacrebleu

[12] CHRF++ considers both word and character n-grams together (Popović, 2017)

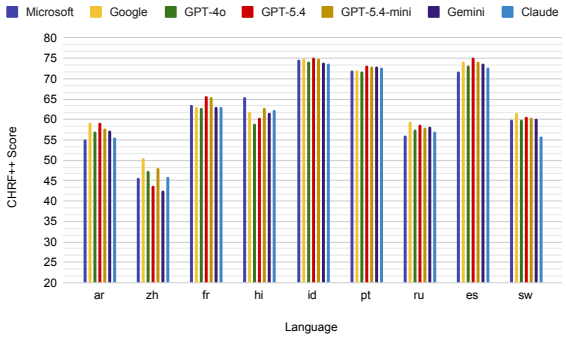
[13] See Appendix D for metric signatures.

[14] As rescaling is not supported for all languages, it was not applied in the EX direction, as it would prevent cross-lingual comparison of results.

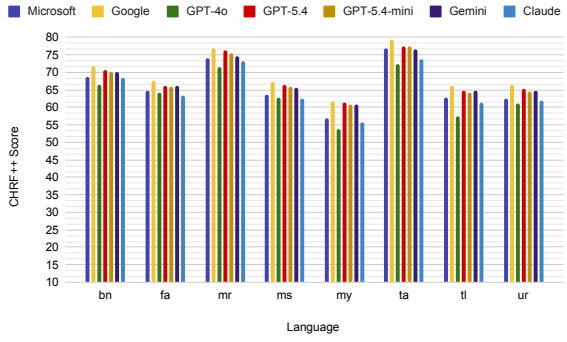
[15] While the Google Translate app supports Dari in both directions, the API does not show support for the language. However, we found that translations in the API *from* Dari were supported, but they were not supported *to* Dari.



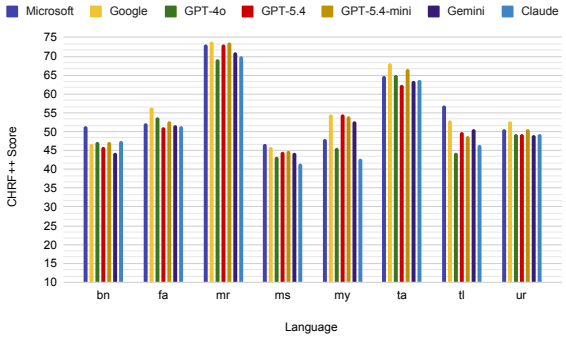
(a) X to E CHRF++ Pivot Languages



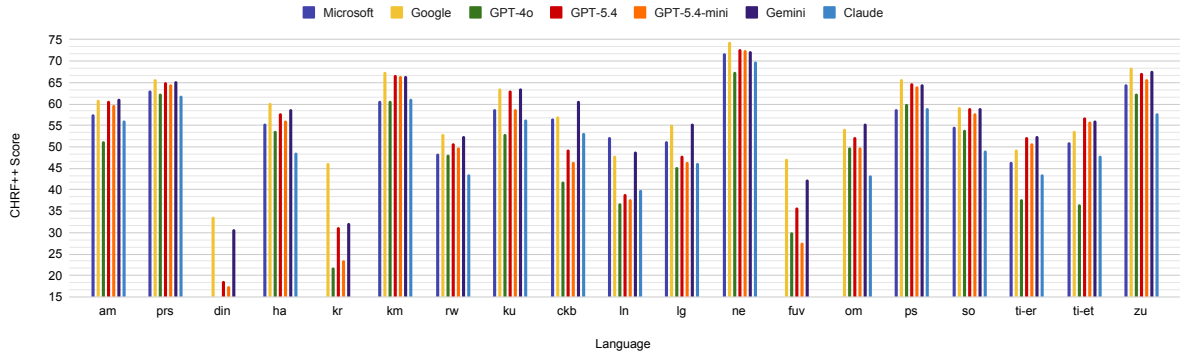
(b) E to X CHRF++ Pivot Languages



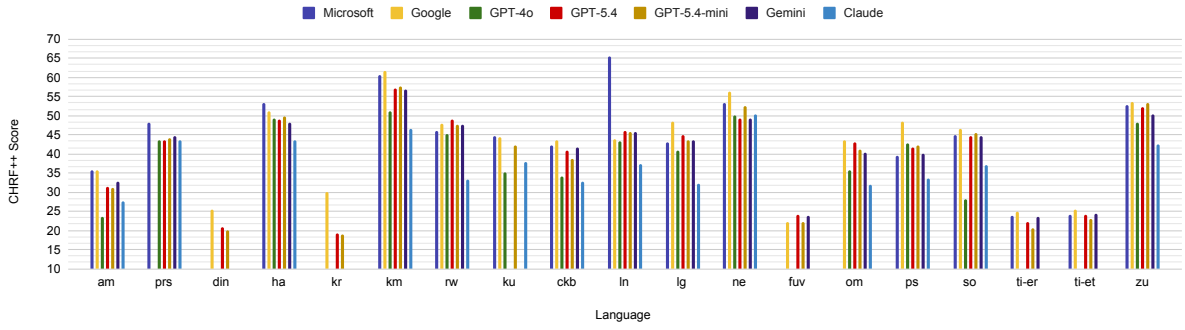
(c) X to E CHRF++ Important Languages



(d) E to X CHRF++ Important Languages



(e) X to E CHRF++ Priority Languages



(f) E to X CHRF++ Priority Languages

Figure 1: Graphs showing comparative 2026 CHRF++ scores across language pairs (E to X and X to E)

Further, both dialects of Tigrinya were supported in only the XE direction by Claude and Gemini, and Microsoft does not support Oromo.

5.1.2 Quality

Google produced translations with the highest scores (in both BLEU and CHRF++) for most languages, in both translation directions. When translating *from* Pivot languages *to* English or *from* English *to* Important languages, there was an even split with Microsoft for best system. Otherwise, Microsoft generally had comparable or slightly lower scores than Google. GPT-5.4 was competitive when translating *from* English *to* Pivot languages, and produced the highest scoring translations for about half of these languages. Surprisingly, Microsoft's scores were closer to those of the lowest performing LLMs here (in contrast to its high performance on Pivot languages in the opposite (X to E) translation direction).

GPT-4o and Claude produced the lowest quality translations throughout (Claude did particularly poorly on Priority languages). Gemini achieved comparably low EX scores, but was more competitive with dedicated MT systems in the XE direction, often matching CHRF++ (but not BLEU) scores.

Exceptions to these trends (see Table 4) include Hindi and Hausa, where Microsoft outperformed Google (and GPT-5.4) in the EX direction, and Lingala, where Microsoft scored the highest in both directions (Microsoft's EX scores were 28 BLEU and 19 CHRF++ points above the next best system). Additionally, when translating into Myanmar and Kinyarwanda, GPT-5.4 produced the highest scoring translations, while in the XE direction, both GPT-5.4 models and Gemini outperformed both Microsoft and Google on Ethiopian Tigrinya. Gemini also achieved the highest scores for Arabic, Kurdish Sorani, Oromo, and Eritrean Tigrinya.

Table 5 shows the best MT system for each language, based on the highest scores on *both* metrics, alongside the usability of that system. For Pivot and Important languages, both BLEU and CHRF++ indicated good quality translations for all models. While LLMs can provide adequate translations for high resource languages, they were rarely the best system, and overall not competitive with dedicated MT systems for low-resource languages.

Some systems produced translations with extremely high scores (BLEU above 55) for a number of high resource languages. These scores are problematic, and suggest potential data contamination (see Appendix F). Likewise, some systems showed dismally low EX scores (BLEU below 5) for a subset of low-resource languages (*e.g.,* Dinka). For LLMs this indicates the presence of hallucinations (see Appendix J).

5.1.3 Translation Direction

BLEU and CHRF++ scores were significantly higher for translations in the XE direction across all systems and languages, which is unsurprising given the vast quantity of English pretraining data available. This was especially true for low resource languages. Arabic had particularly large differences in scores between translation directions compared to other Pivot languages. Exceptions to this trend include Indonesian, where all systems (except Google) had lower XE scores, and Lingala, where Microsoft had significantly lower XE scores. [16]

5.2 Google & Microsoft Changes Over Time

5.2.1 Coverage over Time

In 2023, many of the languages were not supported by one or both of Microsoft and Google. Coverage has improved since then: by 2025, all TICO-19 languages (except Congolese Swahili) were supported by at least one service. In 2026, the main change is Google's addition of partial support for Dari.

5.2.2 Quality over Time

Table 6 shows notable changes in CHRF++ scores since 2023. Full results for 2023 and 2025 across the four metrics are shown in Tables 23–26, Appendix B. Microsoft's performance remained mostly stable when translating to English, with consistent improvements in Important and Priority languages. In particular, Myanmar, Tamil, and Khmer scores showed a strong increase in both metrics in 2025, as did Kurdish Kurmanji, Dari, and Ethiopian Tigrinya in 2026. In the EX direction, however, there was a significant decrease in quality for almost every language in 2025—especially Arabic, Indonesian, Malay, and Pashto—followed by near universal improvement in 2026. Hausa and Khmer were exceptions; both showed improvements throughout. It is also worth noting 2026 scores for Lingala, which improved by 16/7 BLEU/CHRF++ points, and Tagalog and Somali, which both dropped significantly.

[16] Chinese, Myanmar and Khmer appear to buck this trend in BLEU; they each do better in EX vs. XE. However, the EX direction for these languages was scored using character-based tokenizers provided by SacreBleu. Character-based measures tend to have inflated scores, and are not directly comparable.

Language	EX/XE	Microsoft	Google	GPT-4o	GPT-5.4	GPT-5.4-mini	Gemini	Claude
es-la	EX	71.88	74.17	73.17	**75.05**	74.08	73.64	72.84
	XE	**75.22**	74.70	74.09	74.87	74.47	**75.08**	73.57
sw	EX	59.96	**61.59**	59.97	60.55	60.44	60.16	55.86
	XE	63.97	**68.14**	63.51	68.10	66.35	67.91	62.19
fa	EX	52.32	**56.44**	53.76	51.16	52.71	51.80	51.38
	XE	64.64	**67.55**	64.25	66.21	65.77	66.14	63.45
bn	EX	**51.40**	46.69	47.28	45.97	47.25	44.38	47.55
	XE	68.76	**71.94**	66.46	70.77	70.21	70.19	68.40
am	EX	**35.84**	**35.78**	23.51	31.47	31.23	32.62	27.60
	XE	57.54	**60.98**	51.41	60.71	59.82	61.11	56.14
ne	EX	53.42	**56.19**	50.06	49.15	52.39	49.35	50.25
	XE	71.96	**74.38**	67.49	72.92	72.50	72.42	69.83

Table 3: 2026 CHRF++ Scores showing example comparisons between providers for Pivot, Important, and Priority languages. See Tables 15–22 for complete scores.

Language	EX/XE	Microsoft	Google	GPT-4o	GPT-5.4	GPT-5.4-mini	Gemini	Claude
id	EX	74.72	74.88	74.15	**75.08**	74.93	73.92	73.74
	XE	72.53	**74.33**	71.53	73.77	73.27	72.70	71.61
hi	EX	**65.62**	61.76	59.06	60.34	62.81	61.75	62.42
ha	EX	**53.44**	51.26	49.12	48.91	49.77	48.21	43.49
rw	EX	45.88	47.87	45.14	**49.10**	47.74	47.50	33.23
ckb	XE	56.53	57.09	41.84	49.45	46.52	**60.66**	53.37
ln	EX	**65.59**	43.85	43.42	46.07	45.62	45.68	37.34
	XE	**52.21**	47.85	36.95	39.07	37.69	48.86	40.07
ti-et	XE	51.17	53.70	36.56	**56.83**	55.97	**56.17**	47.83

Table 4: 2026 CHRF++ Scores showing notable exceptions to overall trends across languages and MT providers. See Tables 15–22 for complete scores.

E to X	
Microsoft	**hi**, bn, mr, ta, am, ha, ku, **ln**, prs
Google	**ar**, **ru**, **sw**, **zh**, fa, **ms**, **tl**, ur, ckb, din, **km**, kr, lg, ne, om, ps, so, ti-er, ti-et, zu
GPT-5.4	**es-la**, **fr**, **id**, **pt-br**, **my**, rw
GPT-5.4 -mini	**fr**
X to E	
Microsoft	**es-LA**, **fr**, **pt-br**, **zh**, **ln**
Google	**hi**, **id**, **ru**, **sw**, **bn**, **fa**, **mr**, **ms**, **my**, **ta**, **tl**, **ur**, din, fuv, **ha**, **km**, kr, **ku**, **lg**, **ne**, **prs**, **ps**, rw, **so**, **zu**
GPT-5.4	**ti-et**
Gemini	**ar**, **pt-br**, **zh**, **ckb**, **om**, rw, ti-er, **ti-et**

Table 5: Best System and Usability. Bold indicates a usable system, and underline a borderline one.

System		2023	2025	2026
Microsoft (XE)	km	53.70	58.69	60.67
	prs	57.44	57.45	63.11
Microsoft (EX)	ps	46.64	37.84	39.47
	ln	59.00	58.19	65.59
Google (XE)	ar	69.61	70.87	69.17
	so	57.53	37.13	59.39
Google (EX)	pt-br	72.48	74.04	72.05
	ta	54.18	56.13	53.11

Table 6: Notable Changes in CHRF++ Scores between 2023, 2025, and 2026 (red indicates drops is quality).

Google showed similar instability in both directions for Pivot and Important languages, which saw small improvements in 2025, but universal drops in 2026. For Priority languages, Khmer and Kurdish Kurmanji showed good (XE) improvements in 2025 and 2026, respectively, while Somali (XE) was particularly unstable: in 2025 BLEU/CHRF++ dropped by 17/20 points, then recovered by 19/22 points in 2026. Scores for most remaining Priority languages were stable in both directions.

5.2.3 Utility of the CHRF++ and BLEU Thresholds

We recognize that the CHRF++ and BLEU thresholds used in this paper, although based on existing literature or rules of thumb, are still somewhat arbitrary. In that vein, we did a limited human evaluation following the scale and methodology described by Picinini and Castilho (2025), with the Likert scale descriptors changed to meet the needs of the crisis scenario (see Appendix I). The evaluation was conducted over a small subset of XE language pairs to determine if the thresholds were reasonable, and fall within the ballpark range. Based on this human evaluation and subsequent statistical analysis, we determined that the com-

bined thresholds were reasonably accurate for the languages we evaluated, although we might lower the threshold for BLEU in the future. The details of the human evaluation, subsequent analysis, and discussion of measuring usability can be found in Appendix H. We save more detailed investigation for future work.

6 Discussion

6.1 MT Usability by Language

Table 5 shows the highest level of usability (and the best system) for each language. For all Pivot languages, in addition to Malay, Myanmar, and Tagalog, there was at least one usable system to translate both into and out of the language, making bidirectional translation possible. Additionally, for all Important languages and most Priority languages there was at least one system that could translate X to E. In the EX direction, all Important languages had at least a borderline usable system. Priority languages, however, are where we see gaps. With the exception of Lingala and Khmer, no Priority languages had usable EX systems, though many had at least one borderline system. Languages without usable systems in either direction included Dinka, Eritrean Tigrinya, Kinyarwanda, Nigerian Fulfulde, and Kanuri. Dinka was the only language without even a borderline usable XE system.

Claude and GPT-4o were particularly unreliable: for XE translations, they were usable for only 6 and 7 Priority languages respectively (as opposed to 12, 12, 10, 10 and 13 for Microsoft, Google, GPT-5.4, GPT-5.4-mini, and Gemini). Claude was often the only system without usable translations. Despite the quality variations, for high resource languages, all models produced usable, often high quality, translations. However for Priority languages, LLMs were not competitive with the traditional MT systems. Factors beyond quality—such as cost, model features, and content filters—that also impact usability are discussed in Appendix G.

6.2 MT Usability by Region

Tin et al., 2024 examined natural disasters globally between 1995 and 2022. In that period, 11,360 natural disasters occurred (on average 350 disasters per year). They further broke the disasters down by type (*e.g.,* hydrological, biological, meteorological, etc.) and by continent. Biological disasters are epidemic-related, and 10.6% of the disasters were biological. However, when we look at the prevalence of biological disasters by continent we see much higher numbers: in Africa, 32.2% of disasters were biological—far exceeding all other continents. Asia was 5.6%, followed by the Americas at 4.6%, then Europe at 2.9%.

If we treat Africa as a priority for pandemic response—the bulk of TwB's Priority languages in TICO-19 are African, and based on Tin et al., 2024, epidemics are far more common on that continent—MT is not ready. Only Swahili, Lingala, and Arabic are truly usable by our metric of usability, assuming the need for bidirectionality. If XE is all that is needed for a particular use case, then about half the African languages are usable. If borderline cases are included, that number is closer to 80%.

For other continents, readiness is much greater. Europe is well covered by the pivot languages (and arguably English in many contexts). Asia is a mixed bag: none of the Priority languages (except Khmer) are usable in the EX direction. However the Pivot and Important languages of Asia (*e.g.,* Hindi, Malay, etc.) are well-covered.

6.3 Overall Assessment of Readiness

In crisis response, when lives are often on the line, we must be opportunistic. It does not matter what provider produces a particular system, it only matters whether it will be usable in the context of our need. Our results indicate that Pivot languages are well supported by *all* translation systems, including the LLMs. The quality of these systems is high, and using them to translate content will likely result in understandable results.

For other languages, the task that the MT system would be used for on the ground——specifically whether that task requires bidirectionality—is relevant. For Important languages, the quality of XE translations seems high enough across systems to consider them ready, but in the EX direction, only Malay, Myanmar, and Tagalog have usable translations. All other languages are at best borderline.

Priority languages, for the most part, fall through the cracks. Other than Khmer and Lingala, translations to these languages were *at best* borderline. For the lowest resource languages, even XE translation was problematic (Dinka, Nigerian Fulfulde, Kanuri, Tigrinya)—in fact, only around 50% of Priority languages had fully usable XE translations. For Priority languages, model choice has a stronger impact—dedicated MT systems (particularly Google) have a higher likelihood of producing usable (or at least borderline) output, though

this not a guarantee, *especially* in the EX direction. Given that Priority languages are where translation need is likely to be the greatest—they were identified by TwB based on the situation on the ground in 2020—it would seem that MT is not yet ready to be used in this way.

For regions likely to be hardest hit by biological disasters, whether those be epidemics or pandemics, most notably Africa and Asia, a reliance on MT for higher resource trade or colonial languages may be necessary. However, this reliance may come at a cost in regions where these languages, even if official, are not spoken by a sufficiently large percentage of the population.

Another issue that can affect readiness is changes made by providers over time. These changes are not always positive (as we noted, there were drops from 2023 to 2025, and some drops from 2023 to 2026). It is good to be aware that this kind of instability is possible and can have both positive and (unfortunately) negative impacts on readiness.

7 Conclusion

In March 2020, at the start of the pandemic and beginning of the TICO-19 project, most priority languages were not covered by commercial translation engines. By 2025, the coverage was nearly complete. But language support is not enough—it is also important that the translations have sufficient quality to be usable in crises. Further, determining usability in the general domain is not enough; we need usable translations within the crisis (or pandemic) domain. That was the focus of our study.

Is MT ready for the next pandemic? For regions where high-resource languages are spoken, the answer is mostly yes. But there are significant gaps in usability when we look at lower resource languages, especially those spoken in Africa and Asia, regions that arguably are most likely to be hit hardest in the next pandemic (following Tin et al., 2024). There is still much work to be done for the lower-resource languages used in these parts of the world before we can consider MT truly ready.

Limitations

One major limitation of our work is the absence of human evaluations across most of the languages and directions in our study. Alas, they are expensive to implement, and outside of the budgetary considerations for our project. It would also be difficult to source evaluators for some of the languages in TICO-19.

We also did not use mBERT or equivalent models in our evaluations. We felt that these models are still exploratory, and were concerned about whether the signal we might receive through these models could be relied on.

In doing this study, we translated to and from English. By assuming that we should only be measuring translation quality into and out of English, we have introduced an unfortunate bias: we are assuming that *all* aid providers, governing bodies, NGOs, etc., will be English-speaking. This is not at all true (Doctors without Borders is a prime example of a primarily non-English speaking aid provider), and marks a significant limitation of this work.

Nuer and Congolese Swahili were regrettably excluded from our study.

There were some mismatches between the locales represented in TICO-19 and in the translation services we used. Where locales were listed in the relevant API documentation that coincided with the locales in the TICO-19 dataset, *e.g.,* Brazilian Portuguese, Latin American Spanish, this locale information was passed to the API, *e.g.,* pt-BR, es-LA. For Tigrinya and Fulfulde, neither Microsoft nor Google indicated the locale that was supported by their translation engines, so no locale information was passed to the API. TICO-19 has data for two dialects of Tigrinya (Eritrean and Ethiopian) as well as the Nigerian dialect of Fulfulde. We also could not pass locales to the API when processing these languages.

Ethical Considerations

All of the TICO-19 data, including the benchmark data we used, is publicly available. The APIs for each of the translation providers, Microsoft and Google, are publicly accessible for a fee. Likewise, for the three LLMs,Claude, GPT and Gemini. All code for processing the TICO-19 data and calling the various APIs, was done by members of our team. We are not aware of any ethical issues that arose while conducting this work.

Acknowledgements

This work was partially supported by the National Science Foundation under the award CIRC 2346335. We want to thank the two anonymous reviewers for their thorough reviews and many help-

ful comments. Clearly both spent an enormous amount of time reviewing our paper, and we incorporated most of the changes they suggested. We would also like to thank the students of the Spring 2023, 2025 and 2026 LT4CPR graduate seminars at the University of Washington (Priyam Basu, Melody Bechler, Allison Burton, Alexandra Cassell, Jose Cols, Richard Forrester, Chelsea Kendrick, Lily Kurek, Vanesa Marar, Ijeoma Osakwe, Yongsin Park, Anantha Rao, Helen Salgi, and Natasha Schimka) for their helpful input and suggestions. Special thanks also goes to Jen Wilson, whose very closely related paper presented at the same LT4CPR workshop, along with her work with emergency managers across the United States, spurred many changes to this paper, making it more consumable by the very communities who would benefit most from its conclusions and suggestions. It is these communities—emergency managers, aid providers of all stripes, public information managers, communication staff officers, etc.—who were always in our thoughts throughout the work we put into this paper, ultimately spread over several years. It is for you, and the work you do every day saving lives, that we wrote this paper.

References

Antonios Anastasopoulos, Alessandro Cattelan, Zi-Yi Dou, Marcello Federico, Christian Federman, Dmitriy Genzel, Francisco Guzmán, Junjie Hu, Macduff Hughes, Philipp Koehn, Rosie Lazar, Will Lewis, Graham Neubig, Mengmeng Niu, Alp Öktem, Eric Paquin, Grace Tang, and Sylwia Tur. 2020. Proceedings of the 1st workshop on nlp for covid-19 (part 2) at emnlp 2020. In *NLP COVID-19 Workshop*, Online.

Ron Artstein. 2017. Inter-annotator agreement. In Nancy Ide and James Pustejovsky, editors, *Handbook of Linguistic Annotation*, pages 297–313. Springer, Dordrecht.

Steven Bird. 2022. Local languages, third spaces, and other high-resource scenarios. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7817–7829, Online. Association for Computational Linguistics.

Emanuele Bugliarello, Sabrina J. Mielke, Antonios Anastasopoulos, Ryan Cotterell, and Naoaki Okazaki. 2020. It's easier to translate out of English than into it: Measuring neural translation difficulty by cross-mutual information. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1640–1649, Online. Association for Computational Linguistics.

Lynn Chua, Badih Ghazi, Yangsibo Huang, Pritish Kamath, Ravi Kumar, Pasin Manurangsi, Amer Sinha, Chulin Xie, and Chiyuan Zhang. 2025. Crosslingual capabilities and knowledge barriers in multilingual large language models. *Preprint*, arXiv:2406.16135.

Júlia Falcão, Claudia Borg, Nora Aranberri, and Kurt Abela. 2024. COMET for low-resource machine translation evaluation: A case study of English-Maltese and Spanish-Basque. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 3553–3565, Torino, Italia. ELRA and ICCL.

Federico Federici and Patrick Cadwell. 2018. Training citizen translators: Design and delivery of bespoke training on the fundamentals of translation for New Zealand Red Cross. *Translation Spaces*, 7(1):20–43.

Samsara Foubert, Minjung Park, Kanad Pardeshi, Thomas Manzini, Robert McDaniel, David Merrick, Robin Murphy, Aarti Singh, and Hoda Heidari. 2026. Investigating the role of ai in emergency management: Use cases, challenges, and opportunities. In *Proceedings of the 2026 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '26, page 2055–2091, New York, NY, USA. Association for Computing Machinery.

Shahriar Golchin and Mihai Surdeanu. 2024. Time travel in llms: Tracing data contamination in large language models. *Preprint*, arXiv:2308.08493.

Google Cloud. 2025. Evaluate models | cloud translation - google cloud. Accessed: 2025-05-25.

GoogleCloud. 2023. Evaluate models. Accessed on May 11, 2023.

Yvette Graham, Timothy Baldwin, Alistair Moffat, and Justin Zobel. 2013. Continuous measurement scales in human evaluation of machine translation. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 33–41, Sofia, Bulgaria. Association for Computational Linguistics.

Barry Haddow, Tom Kocmi, Philipp Koehn, and Christof Monz, editors. 2025. *Proceedings of the Tenth Conference on Machine Translation*. Association for Computational Linguistics, Suzhou, China.

Amr Hendy, Vikas Raunak Mohamed Gabr Hitokazu Matsushita Young Jin Kim Mohamed Afify Mohamed Abdelrehim, Amr Sharaf, and Hany Hassan Awadalla. 2023. How Good are GPT Models at Machine Translation? a Comprehensive Evaluation. arXiv:2302.09210.

Matthew Hunt, Sharon O'Brien, Patrick Cadwell, and Dónal P O'Mathúna. 2019. Ethics at the intersection of crisis translation and humanitarian innovation. *Journal of Humanitarian Affairs*, 1(3):23–32.

Muhammad Imran, Carlos Castillo, Fernando Diaz, and Sarah Vieweg. 2015. Processing social media messages in mass emergency: A survey. *ACM Comput. Surv.*, 47(4).

Elaine C. Khoong and Jorge A. Rodriguez. 2022. A Research Agenda for Using Machine Translation in Clinical Medicine. *Journal of General Internal Medicine*, 37(5):1275–1277.

Elaine C. Khoong, Eric Steinbrook, Cortlyn Brown, and Alicia Fernandez. 2019. Assessing the Use of Google Translate for Spanish and Chinese Translations of Emergency Department Discharge Instructions. *JAMA Internal Medicine*, 179(4):580–582.

Tom Kocmi, Eleftherios Avramidis, Rachel Bawden, Ondřej Bojar, Anton Dvorkovich, Christian Federmann, Mark Fishel, Markus Freitag, Thamme Gowda, Roman Grundkiewicz, Barry Haddow, Marzena Karpinska, Philipp Koehn, Benjamin Marie, Christof Monz, Kenton Murray, Masaaki Nagata, Marin Popel, Maja Popović, and 3 others. 2024. Findings of the WMT24 General Machine Translation Shared Task: The LLM Era is Here but MT is Not Solved Yet. In *Proceedings of the Ninth Conference on Machine Translation*, pages 1—46, Miami, Florida. Association for Computational Linguistics.

Sanjeev Kumar, Preethi Jyothi, and Pushpak Bhattacharyya. 2026. Evaluating extremely low-resource machine translation: A comparative study of ChrF++ and BLEU metrics. *Preprint*, arXiv:2602.17425.

Séamus Lankford and Andy Way. 2024. Leveraging LLMs for MT in crisis scenarios: a blueprint for low-resource languages. arXiv:2410.23890.

William Lewis. 2010. Haitian Creole: How to build and ship an MT engine from scratch in 4 days, 17 hours, & 30 minutes. In *Proceedings of the 14th Annual Conference of the European Association for Machine Translation*, Saint Raphaël, France. European Association for Machine Translation.

William Lewis, Haotian Zhu, Keaton Strawn, and Fei Xia. 2025. Tapping into Social Media in Crisis: A Survey. In *Proceedings of the 4th Workshop on NLP for Positive Impact, EMNLP 2025*, vienna, Austria.

Benjamin Marie, Atsushi Fujita, and Raphael Rubino. 2021. Scientific credibility of machine translation research: A meta-evaluation of 769 papers. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7297–7306, Online. Association for Computational Linguistics.

NLLB-Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, and 20 others. 2022. No language left behind: Scaling human-centered machine translation. *Preprint*, arXiv:2207.04672.

Atul Kr. Ojha, Chao-hong Liu, Ekaterina Vylomova, Flammie Pirinen, Jonathan Washington, Nathaniel Oco, and Xiaobing Zhao, editors. 2026. *Proceedings for the Ninth Workshop on Technologies for Machine Translation of Low Resource Languages (LoResMT 2026)*. Association for Computational Linguistics, Rabat, Morocco.

Sharon O'Brien and Federico Federici. 2019. Crisis translation: Considering language needs in multilingual disaster settings. *Disaster Prevention and Management*, 29(2):129–143.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Silvio Picinini and Sheila Castilho. 2025. Context-aware monolingual evaluation of machine translation. In *Proceedings of Machine Translation Summit XX: Volume 1*, pages 138–149, Geneva, Switzerland. European Association for Machine Translation.

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the 10th Workshop on Machine Translation*.

Maja Popović. 2017. chrF++: Words helping character n-grams. In *Proceedings of the Second Conference on Machine Translation*, pages 612–618, Copenhagen, Denmark.

Matt Post. 2018. A Call for Clarity in Reporting BLEU Scores. In *Proceedings of the Third Conference on Machine Translation*, Brussells, Belgium.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online. Association for Computational Linguistics.

Nathaniel Robinson, Perez Ogayo, David R. Mortensen, and Graham Neubig. 2023. ChatGPT MT: Competitive for high- (but not low-) resource languages. In *Proceedings of the Eighth Conference on Machine Translation*, pages 392–418, Singapore. Association for Computational Linguistics.

Dimitrios Roussis. 2022. Building End-to-End Neural Machine Translation Systems for Crisis Scenarios: The Case of COVID-19.

Elizabeth Salesky, Antonios Anastasopoulos, Matteo Negri, and Marcello Federico, editors. 2026. *Proceedings of the 23rd International Conference on Spoken Language Translation (IWSLT 2026)*. Association for Computational Linguistics, San Diego, USA (in-person and online).

Gongbo Tang, Oreen Yousuf, and Zeying Jin. 2024. Improving bertscore for machine translation evaluation through contrastive learning. *IEEE Access*.

D. Tin, L. Cheng, D. Le, R. Hata, and G.Ciottone. 2024. Natural disasters: a comprehensive study using EM-DAT database 1995–2022. *Public Health*, 226:255–260.

Jen Wilson, Vipasha Bansal, and William D. Lewis. 2026. Communicating with the Entire Population: Machine Translation of Emergency Alerts and Warnings. In *Proceedings of the Language Technologies for Crisis Preparedness and Response (LT4CPR Workshop)*, Hengqin, Zhuhai, China. Association for Computational Linguistics.

Chrysoula Zerva, Frédéric Blain, Ricardo Rei, Piyawat Lertvittayakumjorn, José G. C. de Souza, Steffen Eger, Diptesh Kanojia, Duarte Alves, Constantin Orăsan, Marina Fomicheva, André F. T. Martins, and Lucia Specia. 2022. Findings of the WMT 2022 Shared Task on Quality Estimation. In *Proceedings of the Seventh Conference on Machine Translation*, Abu Dhabi. Association for Computational Linguistics.

Andy K. Zhang, Kevin Klyman, Yifan Mai, Yoav Levine, Yian Zhang, Rishi Bommasani, and Percy Liang. 2025. Language model developers should report train-test overlap. In *Proceedings of the 42nd International Conference on Machine Learning*, Vancouver, Canada.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with BERT. In *International Conference on Learning Representations*.

Vilém Zouhar, Pinzhen Chen, Tsz Kin Lam, Nikita Moghe, and Barry Haddow. 2024a. Pitfalls and outlooks in using COMET. In *Proceedings of the Ninth Conference on Machine Translation*, pages 1272–1288, Miami, Florida, USA. Association for Computational Linguistics.

Vilém Zouhar, Shuoyang Ding, Anna Currey, Tatyana Badeka, Jenyuan Wang, and Brian Thompson. 2024b. Fine-tuned machine translation metrics struggle in unseen domains. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 488–500, Bangkok, Thailand. Association for Computational Linguistics.

Appendices

A 2026 Full BLEU, CHRF++, BERTScore, and COMET Results

Tables 15-22 show the full BLEU, CHRF++, BERTScore, and COMET[17] results for all seven systems evaluated in 2026. Figures 2 and 3 plot all 2026 BLEU and CHRF++ scores against one another, with each threshold marked. These tables and figures can be found at the end of the appendices.

B BLEU, CHRF++, BERTScore, and COMET Results for 2023 and 2025 for Microsoft and Google

Tables 23, 24, 25, and 26 show the full BLEU, CHRF++, BERTScore, and COMET results for Microsoft and Google from 2023 and 2025, in both translation directions. These illustrate the change in the performance of these two systems during this period. These tables can be found at the end of the appendices.

C Model Configurations

Due to time and funding constraints, reasoning_effort was set to 'none' for both GPT-5.4 checkpoints. GPT-4o, Gemini, and Claude were run with temperature=0. All systems were given the following, identical prompt: "Translate the following text from {source_language} to {target_language} in one line without any transliterations or explanations".

D SacreBleu Signatures

Table 7 shows the metric signatures for the BLEU and CHRF++ (and CHRF) metrics, run using SacreBleu. (1) shows the BLEU signature for Khmer and Myanmar in the EX direction, (2) shows the BLEU signature for Chinese in the EX direction, (3) shows the CHRF signature for Khmer, Myanmar, and Chinese in the EX direction, while (4) and (5) show the BLEU and CHRF++ signatures respectively for all remaining languages in the EX direction and all languages in the XE direction. These are the same for all longitudinal data as well.

[17] Only supported languages are included in COMET results. The list of supported languages can be found here: github.com/Unbabel/COMET

(1)	nrefs:1/case:mixed/eff:no/tok:char/smooth:exp/version:2.3.1
(2)	nrefs:1/case:mixed/eff:no/tok:zh/smooth:exp/version:2.3.1
(3)	nrefs:1/case:mixed/eff:yes/nc:6/nw:0/space:no/version:2.3.1
(4)	nrefs:1/case:mixed/eff:no/tok:13a/smooth:exp/version:2.3.1
(5)	nrefs:1/case:mixed/eff:yes/nc:6/nw:2/space:no/version:2.3.1

Table 7: Metric Signatures

E LLM Translation Failures

There were a number of translation failures across the LLMs; these took on a number of different forms, depending on the model. The most common forms were blank output and 'refusals', and occurred almost exclusively when translating to and from Dinka, Nigerian Fulfulde, and Kanuri. The proportions for both of these error types are shown in Table 8.

	System	Error Type	Number
		E to X	
din	Claude	"Refusal"	943 (45%)
	Gemini	Blank Output	1154 (55%)
	GPT-4o	"Refusal"	498 (24%)
fuv	Claude	"Refusal"	83 (4%)
	Gemini	Blank Output	80 (4%)
kr	Claude	"Refusal"	773 (37%)
	Gemini	Blank Output	1649 (79%)
	GPT-4o	"Refusal"	187 (9%)
		X to E	
din	Claude	"Refusal"	1300 (62%)
	GPT-4o	"Refusal"	73 (3%)
		Blank Output	237 (11%)
fuv	Claude	"Refusal"	385 (18%)
	GPT-4o	Blank Output	81 (4%)
kr	Claude	"Refusal"	731 (35%)
	GPT-4o	"Refusal"	154 (7%)

Table 8: LLM Translation Failures

A 'refusal' was output that overtly stated that the model was unable to translate to or from the requested language. Claude and GPT-4o did this in both translation directions for these three languages. An example of a refusal from GPT-4o is shown in (a) below. Claude's 'refusals' tended to be more descriptive, with examples shown in (b) and (c).

1. (a) "My apologies, but I can't provide a translation for that text."

   (b) "I don't have reliable knowledge of Dinka language to provide an accurate translation of this phrase. Dinka is a Nilotic language spoken in South Sudan, and without verified linguistic resources,

I cannot confidently translate this line without risking significant errors."

(c) "I don't have reliable knowledge of the Dinka language to provide an accurate translation of this text. Dinka is a complex language with specific grammatical structures and vocabulary that requires expertise to translate correctly. I'd recommend consulting a Dinka language expert or speaker for an accurate translation."

Gemini did not 'refuse' to translate any sentences; instead it failed to produce any output at all for a large number of sentences in the EX direction specifically. GPT-4o was more variable, producing both 'refusals' and blank output in both directions.

In some cases, we also saw random string repetitions in the output, particularly in the EX direction. Table 9 shows examples of this from GPT-4o and Claude when translating into Nigerian Fulfulde.

F Data Contamination

A significant issue in assessing commercial MT systems and LLMs is the risk of training data contamination. This occurs when evaluation benchmarks or test sets, or a dataset from which they are drawn, are used as training data for models, leading to inflated evaluation scores and misleading conclusions about the model's real generalization capabilities (Golchin and Surdeanu, 2024). This is especially concerning when benchmarking in zero-shot settings (specifically for LLMs) where there is no fine-tuning for task-specific data. If near perfect or unrealistically high scores are achieved, there is the potential for training data contamination, resulting in falsely reported numbers due to memorization instead of generalization.[18]

Although it is not possible for us to tell whether any particular provider has trained on the TICO-19 test data, some of the highest BLEU scores we see in our results *may* indicate some overlap. MT providers typically do not train on test or dev data (to preserve the value of test data over time), but LLM providers do not necessarily follow this stricture. That said, since some of the TICO-19 data is drawn from public sources, it is possible that MT providers trained on these same sources, thus inadvertently introducing overlap. We hypothesize that this overlap would be possible, and potentially evident, with certain portions of the TICO-19 benchmark dataset, specifically for those sources or domains which are publicly available. In opposition to this, we suspect non-public sources would not show this effect.

Table 10 provides potential evidence for this hypothesis. For Spanish, the Scientific Medical (SM), General (G), and News (N) domains show exceptionally high BLEU scores across the systems (excepting GPT-4o). The BLEU scores in the high 50s to 60s suggest potential overlap. It is completely possible that the sources of these data (Wikipedia, Wikinews, even PubMed) had been consumed and trained on by these engines (potentially even without the providers having had to translate content, *e.g.,* PubMed content is translated to Spanish, as are the other data sources). Notably, the Conversational Medical (MD) data, which comes from CMU, and is not public, does not show such large BLEU scores. Also note that the language Farsi, which is lesser resourced and less likely to be translated into, also has lower scores, and the relative differences in scores between domains is also lower.

There are confounds here, of course, that might only be teased apart through further analysis. First, the number of sentences in each TICO-19 subdomain is quite small (domain labels and counts are shown in Table 2). This could lead to significant variability in the BLEU scores calculated. Likewise, Spanish is much higher resource than Farsi, so just the availability of resources for the former would lead to much larger BLEU scores (the higher scores being indicative of higher quality not contamination). But would these scores be in the 50-60 point range?

Overlap such as that described previously certainly plagues any evaluation data drawn from public sources, *e.g.,* news, social media, etc., since nothing would constrain MT providers from drawing from the same sources, potentially translating them, and adding them into their training data. Overtly adding test data into training is frowned on, but not completely avoided (Zhang et al., 2025). A thorough analysis of data contamination (and the TICO-19 domains) will be left to future research.

[18] As noted by one of the reviewers, high BLEU scores do *not*, in and of themselves, necessarily indicate data contamination; the systems being measured could just be really good. However, based on our experience with developing MT systems, when inflated BLEU scores were observed for particular systems or domains but not others, we suspected data contamination.

GPT-4o	**— English to Nigerian Fulfulde**
Source	and can you tell me what other symptoms are you having along with this?
Reference	e footi a limta mandolji nyau feere je a don mari fahin na?
Output	Mido waawi wadugo dum, ko dum wadi dum didi dum wadi dum wadi dum wadi dum wadi dum wadi dum wadi dum.
Claude	**—English to Nigerian Fulfulde**
Source	These enzymes are expressed extensively in multiple organs, especially in the heart and liver, and are released during tissue damage.
Reference	Neebamji ndu don matinama ha nder kujeji baandu jur, burna ha nder bernde be hengre, e wurtan to bana laaral man seeki.
Output	Enzimoji didi wadi ko'e di-di ko'e ko'e ko'e ko'e ko'e ko'e ko'e [...] ko'e ko'e ko'e ko'e ko'e ko'e ko'e ko'e ko'e ko'e ko'e ko'e ko'e ko'e ko

Table 9: Example string repetition errors in GPT-4o and Claude output when translating from English to Nigerian Fulfulde.

	Domain	**Microsoft**	**Google**	**GPT-4o**	**GPT-5.4**	**GPT-5.4-mini**	**Gemini**	**Claude**
				English to X				
es-LA	CM	32.62	38.80	33.03	**45.16**	41.44	36.18	34.5
	SM	53.90	**59.04**	55.45	**59.10**	56.87	57.42	54.61
	G	51.16	55.42	53.36	**56.25**	55.09	53.19	52.09
	N	56.19	54.85	59.51	**60.63**	59.55	58.17	58.67
	A	40.83	42.58	41.52	43.89	**44.10**	**44.40**	41.51
fa	CM	18.81	25.24	**31.40**	28.29	23.41	**31.81**	20.7
	SM	26.53	**33.72**	29.01	26.62	24.34	26.67	26.32
	G	25.19	**28.27**	25.52	22.56	22.92	23.83	22.26
	N	21.70	**28.18**	23.94	20.02	19.77	22.45	18.94
	A	**28.16**	26.99	21.58	19.28	19.80	19.46	19.00
				X to English				
es-LA	CM	35.17	29.52	29.84	**36.01**	35.06	30.94	29.12
	SM	**55.10**	54.67	54.22	53.71	54.52	**55.34**	52.80
	G	**55.19**	53.74	52.77	53.55	52.47	54.04	50.26
	N	**56.31**	50.88	53.14	52.57	52.88	**56.83**	50.41
	A	**48.21**	44.12	46.37	46.61	44.58	**48.36**	42.29
fa	CM	**27.35**	25.25	24.92	26.63	26.24	23.68	22.70
	SM	40.49	**45.87**	40.84	41.64	42.18	42.65	30.00
	G	39.01	**43.62**	38.63	40.33	39.49	41.93	35.24
	N	34.88	37.60	35.51	35.70	35.00	**39.03**	32.68
	A	39.17	**43.05**	37.49	37.94	38.60	40.89	34.69

Table 10: Domain BLEU Scores for Spanish (LA) and Farsi. Bold text indicates best system within each specific domain. Legend for domains: Conversational Medical (CM), Scientific Medical (SM), General (G), News (N), Announcements (A).

G Factors Beyond Quality That Can Affect Usability

In addition to output quality, there are a number of practical factors, such as cost, runtime, and content filters, that can impact usability, particularly when utilizing LLMs. GPT-5.4-mini is a smaller version of the full size GPT-5.4 model and is therefore faster and cheaper. One might assume that the smaller of the two would show lower quality, yet our results show that the two systems have comparable scores. There are many languages, such as EX Hindi, where the mini actually outperforms the full size model. One notable example is Kurdish Kurmanji: the GPT5.4 full size model (and Gemini) produced output in Kurdish Sorani instead, which does not even share the same script. GPT-5.4-mini, however, produced translations in the correct language and orthography. This of course is one more element in the usability of LLMs—they may produce unexpected or unwanted output, particularly in low resource languages.

From a usability standpoint then, the GPT-5.4-mini is much more cost effective. However, despite its being a reasoning model, we very quickly found that the model was not usable for translating into low resource languages unless we set reasoning to 'None'. When active, reasoning caused the model to be excessively slow and use a large amount of tokens, resulting in blank output combined with much higher costs. Therefore, one of the main features of the model, *i.e.,* reasoning, cannot easily be applied in the crisis scenario. Still, reasoning is not the only issue at play here: when compared to an older, but more widespread, non-reasoning model,

GPT-4o, GPT-4o consistently produced lower quality translations.

Another factor we encountered was OpenAI's content filtering. Many of the sentences in the dataset incorrectly triggered various content filters, such as 'hate speech'. This mostly occurred when translating from a low resource language *into*. It is unclear if this is because the model is producing mistranslations or hallucinations that are, in fact, hate speech (see Appendix J for examples), or if it is the content filter itself that is not performing well on certain languages. These issues severely restrict the usability of all three GPT models in the medical and crisis domain, even if the output translations are good quality; good quality is not useful if the model refuses to provide output for certain crisis-related input in the first place. We did not observe this issue with either Claude or Gemini. The proportion of sentences that were filtered in the GPT models for each language are shown in Table 11.

As outlined in Section 4.2, our usability analysis treated filtered outputs as blank or empty when scoring across metrics. From a practical standpoint, the two are the same: in both cases, the user does not receive a translation. However, in terms of evaluating translation quality specifically, it is also useful to look at the scores with the filtered sentences excluded. These are shown in Table 12. In many cases—such as all three GPT models' performance on Luganda—the difference in scores is substantial.

	GPT-4o	GPT-5.4	GPT-5.4-mini
ckb	427 (20%)	362 (17%)	431 (21%)
din	—	598 (28%)	707 (34%)
fuv	—	91 (4%)	558 (27%)
ha	112 (5%)	107 (5%)	111 (5%)
kr	406 (19%)	—	408 (19%)
lg	381 (18%)	383 (18%)	386 (18%)
ln	578 (28%)	574 (27%)	563 (27%)
om	—	153 (7%)	161 (8%)

Table 11: GPT Content Filtering Rates (XE only)

Finally, a known issue with LLMs is hallucinations—we observed this across the models we evaluated, and, unsurprisingly, hallucinations were much more frequent and more severe in low resource languages. Unfortunately, many of these hallucinations produced output that was racist, violent, or potentially contained misinformation, and would have consequences far beyond poor quality translation, thus further affecting the model's usability. Fortunately, frequent hallucinations often result in low BLEU/CHRF++ scores (which means that thresholding against these scores can potentially identify output containing a high number of hallucinations, at least for LLMs).

	GPT-4o	GPT-5.4	GPT-5.4-mini
ckb	28.17/51.77	34.04/58.65	32.95/57.63
din	—	9.35/26.94	9.54/27.20
fuv	—	17.84/37.31	17.43/36.29
ha	34.13/56.36	38.49/60.43	37.00/58.86
kr	11.95/29.64	—	11.64/29.28
lg	31.97/53.59	34.72/56.82	32.58/54.97
ln	26.18/48.16	28.28/50.73	26.46/48.67
om	—	32.14/55.99	29.94/53.75

Table 12: BLEU/CHRF++ Scores for GPT Models with filtered sentences excluded (XE only).

H Human Evaluation and CHRF++ and BLEU Thresholds

We performed a limited, blind human evaluation on seven samples of output produced by two MT systems (Microsoft and Gemini), when translating into English from four languages ((Dinka, Eritrean Tigrinya, Lingala, Hindi). The evaluation was done only in the XE direction, with the primary focus on establishing the utility of the CHRF++ and BLEU thresholds discussed in Sections 2.2 and 4.3.

The four languages were selected based on output quality—we wanted to ensure that a range of BLEU/CHRF++ scores were represented in our analysis. Output translations from Dinka (Gemini only) were unusable, while those from Eritrean Tigrinya were borderline for both systems. Lingala output was borderline for Gemini and just about passed the usability thresholds for Microsoft, and both systems had high performance for Hindi.

The human evaluation was conducted by four evaluators; each evaluator received 42 reference-translation pairs from each of the seven sets of output (one for each system and language.)[19]

In order to establish the level of inter-annotator agreement, each evaluator received the same 42 sentences from the Dinka-English and Hindi-English outputs. For the four remaining outputs (Eritrean Tigrinya-English and Lingala-English from both Gemini and Microsoft), each evaluator was assigned a distinct set of 42 sentences. They then scored reference-translation pairs (of the same 42 sentences) across each of the four sets of output. Therefore, a total of 168 distinct sentence pairs

[19] Dinka was unsupported by Microsoft.

were evaluated for each of these four systems-language pairs. Given the high inter-annotator agreement established by Dinka-English and Hindi-English evaluations (to be presented shortly), this decreased the margin of error from 14.97% (if all evaluators scored the same 42 sentence pairs) to 7.33% (168 distinct sentence pairs scored).

After all four language pairs were sampled (with a total of 294 sentence pairs from across the seven system-language sets assigned to each evaluator), the sentences were placed in a random order, and the evaluators reviewed the English reference and English MT output (we assumed that none of the evaluators could speak the source languages, so the input sentences were not presented). The evaluators were asked to evaluate the translated output using the modified Picinini and Castilho (2025) Likert scale shown in Appendix I, and assign an integer score ranging from 1-5. Based on this scale, we considered a score of 3 as the threshold for usable translation in the crisis scenario.

As different numbers of distinct sentences were evaluated for various outputs, a statistical analysis assuming maximum variability (p=0.5) indicates varying margins of error at a 95% confidence level. We established inter-annotator agreement using evaluator scores for Hindi-English (Gemini and Microsoft) and Dinka-English (Gemini), (since, as noted, all evaluators scored the same sentences across each of these outputs). Krippendorff's Alpha over all evaluations of these three extracts is 0.7925, suggesting good inter-annotator agreement (Artstein, 2017). Tables 13 and 14 show the results of the human evaluation, including margins of error.

At a 95% confidence level, Hindi-English (Microsoft and Gemini), Lingala-English (Microsoft and Gemini), and Eritrean Tigrinya-English (Gemini only) met the human evaluation threshold of 3, while Dinka-English (Gemini) did not. At a 95% confidence level, it is inconclusive whether Eritrean Tigrinya-English (Microsoft) met the threshold (given the corresponding margin of error).

We ran an Ordinary Least Squares regression to establish a linear relationship between BLEU and our human evaluations (Pearson r = 0.9411 with p = 0.0016) and between CHRF++ and our human evaluations (Pearson r = 0.9703 with p = 0.0003), subsequently identifying the BLEU and CHRF++ values corresponding to a human evaluation of 3.0 from the linear regression. The results suggest thresholds of 23.49 for BLEU and 45.11 for CHRF++. Thus, the results of the linear regression suggest lowering the BLEU score threshold, while keeping a similar CHRF++ threshold.

Changing the BLEU threshold to 23.49 in the XE direction would suggest that a number of systems and languages currently classified as borderline would actually be considered usable. No languages classified as fully unusable would change in their usability categorization. However, with only seven data points, this is a preliminary analysis; it suggests the thresholds we used in our analysis are within a reasonable range of accuracy. A full threshold sensitivity analysis in both translation directions along with additional human evaluation (particularly in the EX direction) is warranted and would be a valuable area for future research.

Overall, the strong linear correlation of BLEU and CHRF++ with our human evaluation suggests that these automatic metrics may be good predictors of minimum usability as we defined it. Exceeding these thresholds suggests higher degrees of usability.

I Human Evaluation Scale

The Likert scale described by Picinini and Castilho (2025) used in our human evaluation, with our modifications for the crisis scenario italicized, can be found below.

1. Critical/Incomprehensible:
   When the translation:
   - Is incomprehensible (hallucinations, meanings that make no sense in the text),
   - Is completely not in the target language,
   - Contains critical errors,
   - Can be a misleading mistranslation,
   - May carry health, safety, legal, reputation, religious or financial implications,
   - Contains profanities/insulting words not appropriate in the context.
2. Significant errors/effort:
   Translation contains significant errors and may be incomplete (such as significant mistranslations, significant portions untranslated, significant omissions) and requires significant effort to understand it. It would require a significant amount of change to be usable.
3. Some errors/some effort:
   There are errors in translation, such as untranslated words and mistranslations (wrong

Language	X to English			
	Microsoft	Gemini	Margin of Error	% Margin of Error
Hindi	4.5298	4.7679	0.5988	14.97%
Dinka	—	1.8453	0.5988	14.97%
Lingala	3.5536	3.5774	0.2932	7.33%
Tigrinya (Eritrean)	3.0774	3.7262	0.2932	7.33%

Table 13: 2026 XE average human evaluation scores and margin of error for all seven evaluated sets. See Appendix I for scale used in evaluation.

Language	**% Critical Errors (score of 1)**		**% Usable Sentences (score of 3-5)**	
	Microsoft	**Gemini**	**Microsoft**	**Gemini**
Hindi	0.0/0.0/0.0/2.4	0.0/0.0/0.0/2.4	97.6/95.2/ 100.0/95.2	100.0/100.0/ 100.0/97.6
Dinka	—	59.5/54.8/ 52.4/38.1	—	19.1/19.1/ 19.1/33.3
Lingala	7.1	7.1	75.6	81.0
Tigrinya (Eritrean)	11.3	6.6	65.5	85.1

Table 14: Proportion of sentences that were rated as having a critical error and the proportion that were considered usable by human evaluators for each language and system. For Dinka and Hindi, where each evaluator had the same set of sentences, the proportions for each evaluator's responses are shown separately.

but not misleading translations). The translation requires some effort to understand it and change it. *The translation would be partially or borderline usable in the crisis context (*e.g., *the necessary information would be conveyed and understood), but might require subsequent verification.*

4. Do not affect understanding:
   There are minor grammar errors: wrong agreement, word or der, missing prepositions, minor omissions or other errors that do not affect understanding. It would require only minor changes. *The translation would still be usable in the crisis context (*e.g., *the necessary information would be conveyed and understood).*

5. Complete/fluent:
   Translation is (or seems to be) correct, complete, fluent, easy to read, and contains no errors. It looks like (almost) human translation and requires no changes. *The translation would be usable in the crisis context and would be fully understood.*

It is important to note here, that the use of the word 'borderline' for entry 3 above differs from our use of the word in our overall analysis. As described previously, we defined systems that were 'borderline usable' as producing output translations that scored higher than the threshold for one metric but not both, indicating that they were almost, but not quite, of high enough quality to be usable. In this human evaluation scale however, we use 'borderline' to mean just barely usable—the translation is low quality, and if it were reviewed by a human post-editor, they would likely make changes. This distinction was unintentional and was a source of confusion for at least one of our reviewers. We regret this error and would fix this in future human evaluation work.

J Qualitative Error Analysis and Hallucinations

We also conducted a preliminary error analysis on LLM output translations for a subset of languages in the XE translation direction. We looked first at some of the lowest scoring systems, (which had BLEU scores of around 10). We then looked at Gemini's output when translating from Nigerian Fulfulde to English—while scores here were still too low to be usable, they were much closer to reasonable (BLEU/CHRF++ were 21.21/42.42). Finally, we examined some of the errors produced by systems that were considered borderline usable.

In the very low performing languages—Dinka and Kanuri consistently fell into this category across models—hallucinations were prevalent. The conversational medical domain was particularly

prone to this; in many cases, the output for the entire domain is fully hallucinated. We identified several types of hallucinations: output that was completely irrelevant or random, output that was unrelated to the source sentence but on topic, and output where the source sentence was translated, but with extra, hallucinated content included. Many of the hallucinations included religious themes, and in a few instances, centered around one key word in the source. Unfortunately, there were also many hallucinations (or severe mistranslations) that were problematic, containing content that was racist, sexist, actively spreading misinformation, or were otherwise aggressive or violent. Examples of various hallucinations are shown in 1-7 below:

1. **Irrelevant Output, GPT-5.4-full, Kanuri-English**

   Reference: extensions of monitoring and surveillance would have to have sunset clauses;

   Output: The people have ten fingers and ten toes, and all of them are counted together.

2. **Extra Hallucinated Content, Gemini, Dinka-English**

   Reference: As of 19 March drive in tests were offered in several large cities.

   Output: On the 19th of the month of Kol, many people were tested for the disease, and *they were all remembered in prayer.*

3. **Religious Hallucinations, GPT-4o, Nigerian Fulfulde-English**

   Reference: since you have a fever

   Output: Prepare your mat, we are about to pray.

4. **Key Word Based Hallucination, Gemini, Kanuri-English**

   Reference: *Panama* has made it obligatory to wear a face mask whenever going outside, while also recommending the manufacture of a homemade face mask for those who cannot purchase face masks.

   Output: *The Panama Papers* revealed how the wealthy hide their assets within the community, and also how various types of shell companies are used to consult with those who wish to evade taxes.

5. **Problematic Hallucination: Aggressive, GPT-5.4-mini, Dinka-English**

   Reference: Patients with fever and/or respiratory symptoms and acute fever, even without pulmonary imaging abnormalities, should be screened for the virus for early diagnosis.

   Output: People are lazy, foolish, and full of anger and hatred, so if they are not careful, they will end up making mistakes and being deceived.

6. **Problematic Hallucination/Mistranslation: Violent, Gemini, Dinka-English**

   Reference: "after patient care.The addition of antiseptic chemicals to soap (‘medicated’ or ‘antimicrobial’ soaps) confers killing action to a hand-washing agent."

   Output: If there is no one sick in Akoor to be helped with soap, then they should be killed immediately.

7. **Problematic Hallucination/Mistranslaion: Spreads Misinformation, Gemini, Dinka-English**

   Reference: Other types of respirators can provide greater protection and improve worker comfort.The WHO does not recommend coveralls, as COVID-19 is a respiratory disease rather than being transmitted through bodily fluids.

   Output: Some of the best ways to help save lives have not been implemented, and the World Health Organization has not yet approved the use of masks to prevent the spread of COVID-19.

Other common errors among the lowest performing outputs include: difficulty with medical terminology, archaic word choice, meaning of translation is the opposite of the intended/original meaning, and unusual phrasing. Some examples are shown in 1-4 below.

1. **Medical Terminology, Gemini, Dinka-English**

   Reference: and with your history of *diabetes*

   Output: What is the name you have been given in the *sugar*?

2. **Medical Terminology, Gemini, Dinka-English**

   Reference: Instead, they may cause severe side effects, especially avascular *osteonecrosis*, dramatically affecting the prognosis.

   Output: There is something that is done in the body, just as death is the *bone of death*, and it is in that which is found to be bad

3. **Archaic/Unusual Phrasing, Gemini, Dinka-English**

   Reference: Many different hand dryer manufacturers exist, and hand driers have been compared against drying with paper towels.

   Output: The people of the village of the hand did not do it, but they were found in the village of the hand wearing clothes that covered them well.

4. **Opposite Meaning, Gemini, Kanuri-English**

   Reference: In addition, Turkey seized hundreds of ventilators destined for Spain.

   Output: "Because of that, Turkey is sending the medical equipment called a 'ventilator to' Spain, which has been hit hard by the world pandemic."

We then turned to Gemini's output when translating from Nigerian Fulfulde to English. As noted earlier, the BLEU/CHRF++ scores were still too low to be considered usable, but more reasonable than those for Dinka or Kanuri. In general, we observed the same error types as with the very low performing languages, but with fewer numbers of problematic outputs and errors that did not seem as severe. The conversational medical domain was not fully hallucinated here; instead, the medical terms specifically were hallucinated or mistranslated. The mistranslations tended to be consistent (*e.g.,* "frog" was present in several translations where the reference sentence referred to a "fever"). Some of the mistranslations from this output could be inferred from context (*e.g.,* "shortness of breath" in the reference sentence aligned with "breath runs out" in the translation). However, there are still some extremely problematic mistranslations and hallucinations present in the output. See 1-5 in the list below for examples.

1. **Poor Quality, Mistranslation, Gemini, Nigerian Fulfulde-English**

   Reference: are you having any shortness of breath?

   Output: Is it okay if your breath runs out?

2. **Poor Quality, Gemini, Nigerian Fulfulde-English**

   Reference: are there other people sick as you at home with your same symptoms?

   Output: Are there other people in your town who are sick with the same kind of illness as you?

3. **Medical Terminology, Gemini, Nigerian Fulfulde-English**

   Reference: I've very concerned of this chest pain

   Output: My heart is aching and I am very troubled.

4. **Opposite/Untrue Meaning, Gemini, Nigerian Fulfulde-English**

   Reference: The White House Coronavirus Task Force was established on 29 January.

   Output: The White House will release the coronavirus report on January 29th.

5. **Problematic Mistranslation: Spreads Misinformation, Gemini, Nigerian Fulfulde-English**

   Reference: Several measures are commonly used to quantify mortality.

   Output: Many tricks are being used to inflate the death toll.

Finally, we turn to systems that produced borderline usable translations, focusing on various Lingala-English and Oromo-English outputs across systems. Translations were significantly better than the lower-performing, unusable, systems, especially in the conversational medical domain. In many cases, medical or technical terms were simplified, but not necessarily incorrect. We also see mismatches in tone between the reference and output (or some added dramatization), and translations that were almost or partially correct (*i.e.,* some parts were conveyed correctly, but some information was mistranslated or omitted). We do still note

some archaic phrasing or word choices, and some problematic mistranslations. Some examples are shown in 1-10 in the list below.

1. **Simplifying Technical Terms, Gemini, Lingala-English**

   Reference: digital surveillance would have to avoid exacerbating discrimination and marginalisation;

   Output: Computer-based monitoring should not be done with malice toward others.

2. **Simplifying Technical Terms, GPT-5.4-mini, Lingala-English**

   Reference: Patients often develop an atypical pneumonia, acute lung injury, and acute respiratory distress syndrome (ARDS).

   Output: Often, sick people had lung disease, severe lung inflammation, and severe breathing difficulty (ARDS).

3. **Tone Difference or Dramatization, Gemini, Lingala-English**

   Reference: But it's almost impossible to make any sensible projection right now about when that will be.

   Output: But there is no way to know how things will be these days.

4. **Tone Difference or Dramatization, GPT-5.4-full, Oromo-English**

   Reference: *The measure was taken* to enforce quarantine and protect those who may come into contact with infected citizens.

   Output: *The punishment is imposed* to influence the duration of isolation and also to prevent people from touching citizens infected with the disease.

5. **Tone Difference or Dramatization, Gemini, Lingala-English**

   Reference: Many consider an economic recession to be likely.

   Output: Many people think that the economy will truly collapse.

6. **Archaic Phrasing, GPT-5.4-mini, Lingala-English**

   Reference: *As of March 19*, these three had already recovered.

   Output: *On the 19th day of the third month*, three people survived.

7. **Archaic Word Choice, GPT-5.4-mini, Lingala-English**

   Reference: *Doctors* have said they are using the drug when "there's no other option."

   Output: The *healers* say they use that medicine only when there is no other medicine.

8. **Partially Correct, GPT-5.4-full, Oromo-English**

   Reference: The study failed to find evidence of viral replication in *pigs, ducks,* and chickens.

   Output: The study could not find evidence that the virus reproduces in *sheep, donkeys,* and chickens.

9. **Problematic Mistranslation, Gemini, Lingala-English**

   Reference: Women who are pregnant or breastfeeding or who do not have an effective contraceptive method *are not eligible*.

   Output: Women who are pregnant, breastfeeding, or not using effective contraception *are not permitted to enter*.

10. **Mistranslation, Gemini, Lingala**

    Reference: Pakistan has said that it will *not be evacuating any citizens* from China.

    Output: Pakistan said that it would *launch its own person into space* with China.

K The Choice of Evaluation Metric(s)

The choice of evaluation metric(s) is always an issue in a study such as ours. While human evaluation is the gold standard, it is both time consuming and expensive to implement (Bugliarello et al., 2020). In fact, expense aside, locating reviewers for many of the lower resource languages in TICO-19 would be difficult at best. As an example, translators and reviewers were a significant problem for these languages during the development of TICO-19 a few years ago.

Still, the choice of evaluation metric is important and should not be taken lightly. Yet, the options available for a study such as ours are limited, given the languages that we focused on.

On the surface, given the underlying technology used for translation today, neural metrics such as BERTScore or COMET would make the most sense. We knew this going in and made sure that we calculated scores for these two metrics for almost every output across our three-year study. But we could not use these metrics for all data points. Alas, these metrics do not work well for lower-resource languages and specialized domains (as noted in Section 2.2, per (NLLB-Team et al., 2022; Tang et al., 2024; Falcão et al., 2024; Zouhar et al., 2024a,b)). Further, there is no clear consensus on how to interpret these metrics directly.

BLEU has had a long and, dare we say, sordid past in the field of Machine Translation. It is a metric that everyone loves to hate, and most would freely admit that it has outlived its shelf-life. That said, it still manages to cling to life, and is still used widely in the field, even in industrial settings (*e.g.,* for hill climbing during development). Why won't BLEU just die already? The reasons vary greatly, but we feel it is mostly due to the fact that it is cheap and easy to implement, that it is universal (*i.e.,* it can be applied to *any* language pair), that we understand the scores and what they mean, and that it provides some sort of continuity across the lengthy history of our field.

In an earlier version of this paper, which was submitted to and rejected by a major conference, we used BLEU for most of our analysis (we still had BERTScore and COMET, but could not apply these broadly, as noted). Concerns about BLEU dominated in the reviews and in the final acceptance assessment. Some of the reviewers suggested we should use CHRF++ instead, which we subsequently adopted wholeheartedly.

CHRF++ (Popović, 2017) is favored in low-resource MT workshops, such as LoResMT (Ojha et al., 2026), now in its ninth year. CHRF++ is still an n-gram based metric, so it will not catch semantic similarities across texts (something that the neural methods are more adept at). But, because it uses a composite word and character-based n-gram score, it is a more effective metric for evaluating the output of typologically diverse languages, and much more effective than BLEU for morphologically rich languages in particular (Popović, 2017, 2015).

Because of BLEU's lengthy history in our field, and because of CHRF++'s frequent use in low-resource contexts, we felt a combination of the two metrics was warranted. This follows other work in the field for low-resource languages that show that a composite score across the two metrics provides complementary insights into the text being measured (*e.g.,* Kumar et al. (2026)), and is in line with the recommendation made by Marie et al. (2021) for the use of additional metrics alongside BLEU.

There is still the question of comparability between languages. BLEU has been shown to not be fully comparable between languages (*e.g.,* Marie et al. (2021)), although we could not find a reference saying the same for CHRF++. This issue has been noted by a couple of reviewers. Yet, we feel it is important to recognize that these measures are not just producing white noise; there *is* signal that is returned, and that signal can be interpreted, as long as we ensure that we are using standard methods and practices, *e.g.,* SacreBLEU (Post, 2018). Incorporating both measures into one threshold, and testing that this threshold is at least in the ballpark (*e.g.,* through human eval), gives us enough signal to be able to say that certain language pairs may be usable in the context of crisis preparedness and response and others not.

It is crucial to note that the ultimate goal of this paper was to offer insights to crisis managers, aid providers and the like about what tools *might* be usable in the field, and for which languages. If a particular language pair is well below our threshold, it will likely be necessary to pursue other options. Outside of other available information, these communities and practitioners are left completely in the dark, in that they have no information about what tools they should be using, what the quality of these tools are, and whether the tools are usable for certain languages versus others. [20] We hope to give these communities at least some glimmer of light in this dark about the tools that so many of them are seeking to use or have already become dependent on. As a field, burying our heads in the sand and saying that we have no way to measure quality for these language pairs is not an adequate response. The development of high-performing and broadly applicable evaluation metrics that fully include low-resource languages is a vital area for further research (though this is a distinct goal from those of this study), but we cannot wait for this

[20] In fact, some emergency managers, outside of additional information, have determined overall and cross-lingual usability of particular tools, such as Google Translate, based on the evaluation of *one* language, for which they had a native speaker in house (Wilson et al., 2026).

work to be completed before evaluating translation tools on low resource languages, particularly when such tools are likely already being used. We owe these communities more than that.

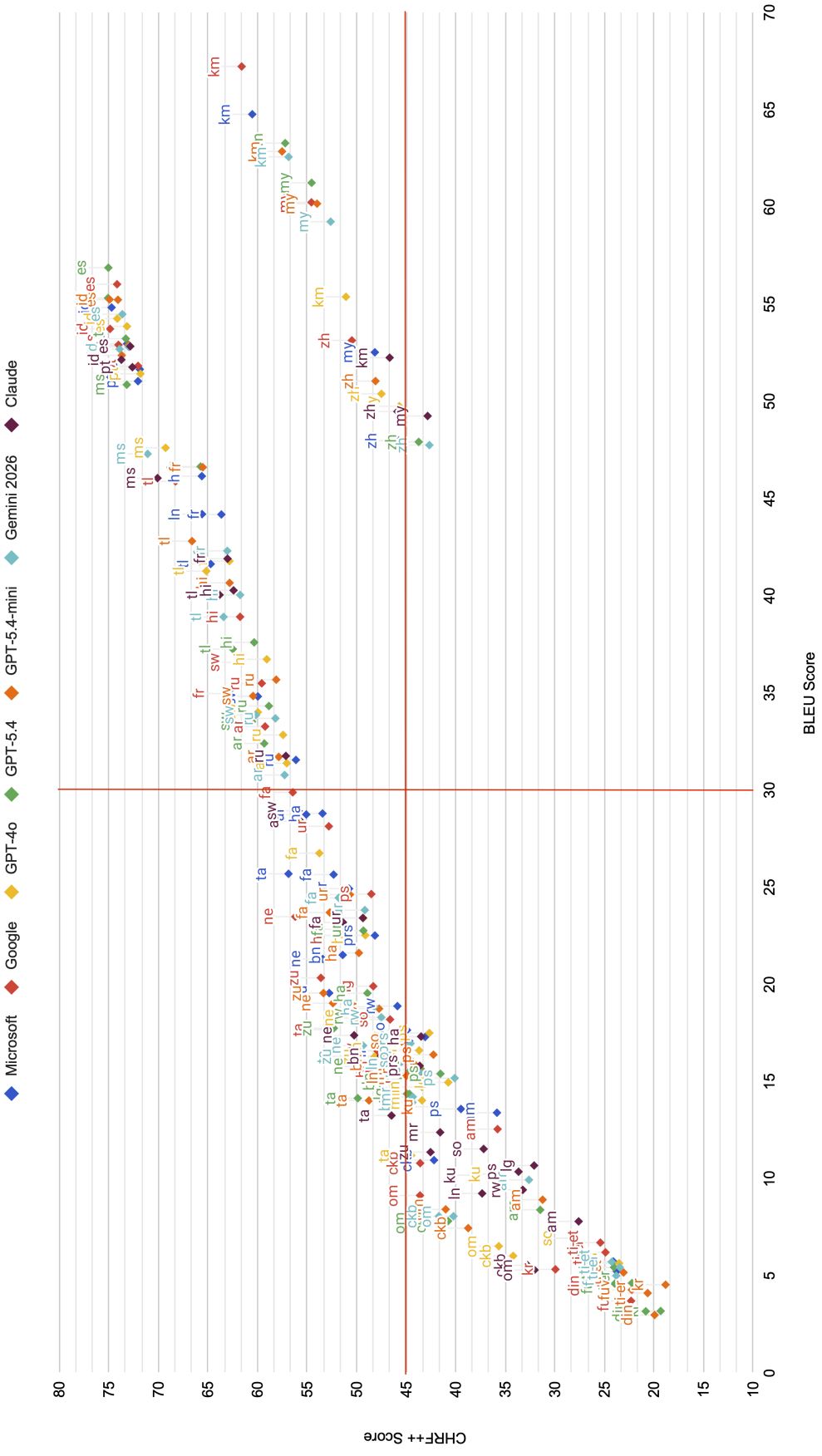


Figure 2: E to X BLEU x CHRF++ Scatter Plot for all TICO languages

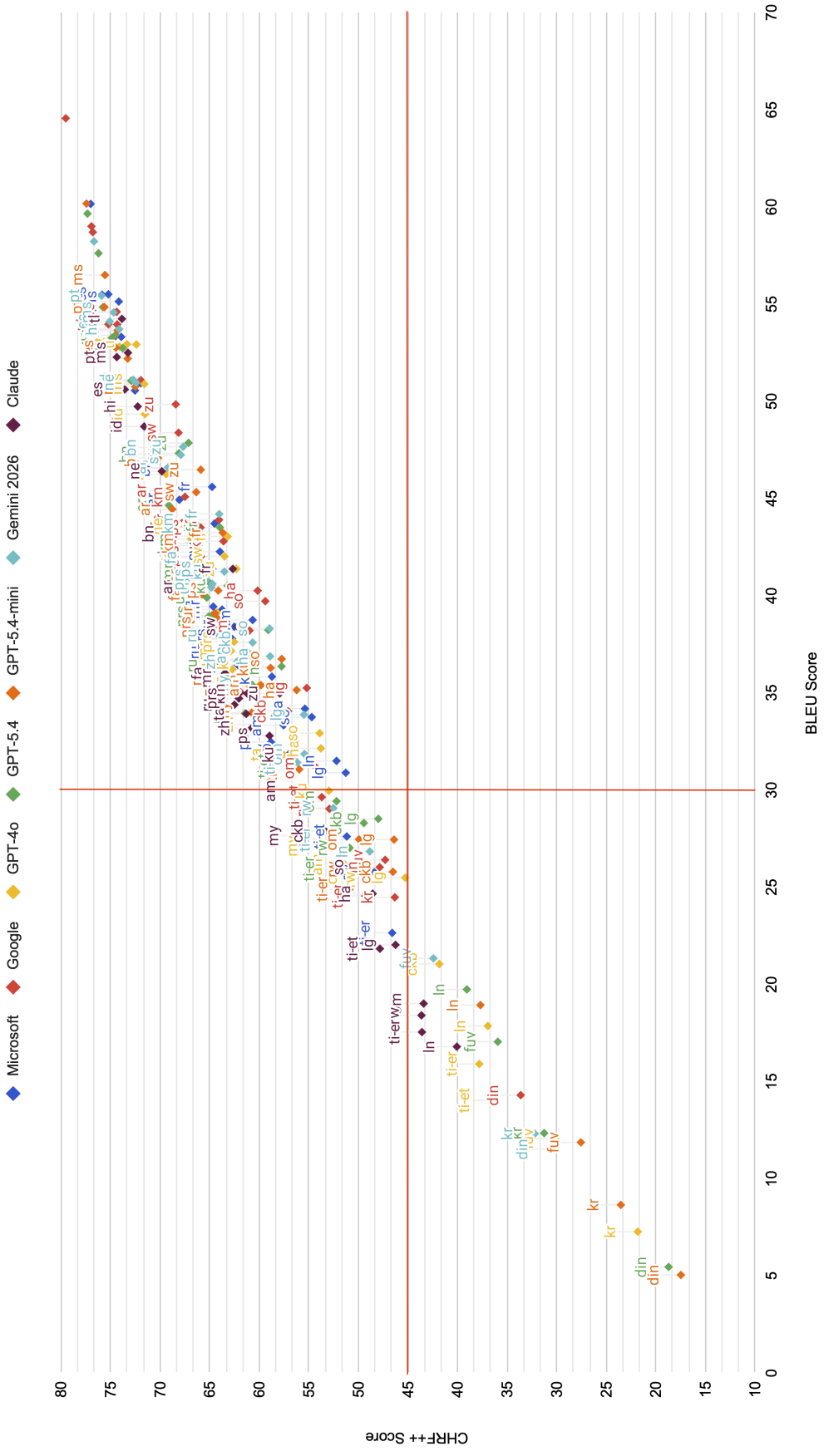


Figure 3: X to E BLEU x CHRF++ Scatter Plot for all TICO languages

Language	English to X						
	MS	Google	GPT-4o	GPT-5.4	GPT-5.4-mini	Gemini	Claude
——Pivot——							
Arabic	28.71	**33.24**	31.35	32.37	31.68	30.75	28.36
Chinese	47.95	**53.10**	50.36	47.88	51.01	47.72	49.45
French	44.15	34.94	41.75	**46.61**	46.58	42.28	41.87
Hindi	**46.12**	38.88	36.70	37.57	40.63	40.01	40.24
Indonesian	54.80	53.70	54.23	**55.29**	55.21	52.66	52.11
Portuguese (BR)	51.01	51.79	51.39	**53.20**	52.88	52.77	51.73
Russian	31.52	35.46	32.81	34.29	**35.65**	33.66	31.73
Spanish (LA)	51.62	55.99	53.83	**56.84**	55.19	54.45	52.80
Swahili	34.79	**36.54**	33.98	33.50	34.81	33.83	28.91
——Important——							
Bengali	**21.49**	15.61	16.58	14.96	15.97	13.90	16.30
Farsi	25.62	**29.85**	26.72	22.81	23.67	24.43	23.17
Malay	**52.93**	52.88	47.58	50.83	52.34	47.27	46.02
Marathi	**16.20**	15.18	14.00	14.34	15.27	14.19	12.36
Myanmar	52.49	60.21	49.72	**61.21**	60.14	59.21	49.22
Tagalog	41.60	**45.85**	41.24	37.22	42.78	38.88	40.01
Tamil	**25.66**	17.65	11.17	14.12	14.00	16.26	13.23
Urdu	24.90	**28.11**	23.39	22.74	24.59	23.80	23.39
——Priority——							
Amharic	**13.37**	12.53	5.62	8.37	8.89	9.92	7.78
Dari	**22.49**	—	16.58	15.60	15.91	16.93	15.78
Dinka	—	**4.48**	—	3.15	2.96	—	—
Hausa	**28.77**	22.49	22.50	19.52	21.59	18.82	17.29
Kanuri	—	**5.31**	—	3.17	4.52	—	—
Khmer	64.74	**67.20**	55.35	63.25	62.83	62.55	52.22
Kinyarwanda	**18.86**	16.54	17.14	18.33	18.72	18.28	9.40
Kurdish (Kurmanji)	**17.06**	15.40	10.21	—	13.68	—	10.16
Kurdish (Sorani)	**10.93**	10.78	6.00	7.80	7.43	8.04	5.53
Lingala	**44.17**	15.18	14.70	15.12	15.17	15.89	9.22
Luganda	17.27	**19.88**	14.93	14.34	15.48	15.48	10.66
Nepali	21.28	**23.45**	18.24	15.75	19.01	16.83	17.35
Nigerian Fulfulde	—	3.69	—	4.58	4.26	**4.98**	—
Oromo	—	**9.11**	6.50	7.71	8.40	8.04	5.28
Pashto	13.56	**24.62**	17.47	15.37	16.36	15.15	10.34
Somali	17.62	**18.17**	6.92	15.49	17.03	16.09	11.51
Tigrinya (Eritrean)	5.22	**6.19**	—	4.61	4.09	5.42	—
Tigrinya (Ethiopian)	5.71	**6.69**	—	5.42	5.15	5.70	—
Zulu	19.53	**20.31**	16.26	17.72	19.53	16.27	11.34

Table 15: 2026 EX BLEU scores. The best system for each language is in bold.

Language	English to X						
	MS	Google	GPT-4o	GPT-5.4	GPT-5.4-mini	Gemini	Claude
——Pivot——							
Arabic	55.08	59.23	57.05	**59.32**	57.86	57.27	55.64
Chinese	45.74	**50.48**	47.50	43.74	48.10	42.66	45.89
French	63.64	63.13	62.83	**65.74**	65.52	63.05	63.02
Hindi	**65.62**	61.76	59.06	60.34	62.81	61.75	62.42
Indonesian	74.72	74.88	74.15	**75.08**	74.93	73.92	73.74
Portuguese (BR)	72.06	72.05	71.81	**73.29**	72.99	72.99	72.62
Russian	56.13	**59.57**	57.42	58.85	58.11	58.19	57.14
Spanish (LA)	71.88	74.17	73.17	**75.05**	74.08	73.64	72.84
Swahili	59.96	**61.59**	59.97	60.55	60.44	60.16	55.86
——Important——							
Bengali	**51.40**	46.69	47.28	45.97	47.25	44.38	47.55
Farsi	52.32	**56.44**	53.76	51.16	52.71	51.80	51.38
Malay	73.16	**74.03**	69.28	73.19	73.70	71.09	70.09
Marathi	**46.80**	46.05	43.40	44.67	44.97	44.33	41.57
Myanmar	48.16	**54.56**	45.70	54.54	53.99	52.62	42.84
Tagalog	64.72	**68.27**	65.17	62.50	66.61	63.42	63.83
Tamil	**56.87**	53.11	44.49	49.88	48.75	50.68	46.48
Urdu	50.77	**52.80**	49.32	49.32	50.67	49.17	49.37
——Priority——							
Amharic	**35.84**	35.78	23.51	31.47	31.23	32.62	27.60
Dari	**48.15**	—	43.69	43.47	44.23	44.56	43.63
Dinka	—	**25.31**	—	20.82	19.93	—	—
Hausa	**53.44**	51.26	49.12	48.91	49.77	48.21	43.49
Kanuri	—	**29.94**	—	19.32	18.83	—	—
Khmer	60.53	**61.60**	51.07	57.20	57.52	56.86	46.66
Kinyarwanda	45.88	47.87	45.14	**49.10**	47.74	47.50	33.23
Kurdish (Kurmanji)	**44.70**	44.38	35.29	—	42.09	—	37.80
Kurdish (Sorani)	42.19	**43.59**	34.20	40.77	38.74	41.70	32.75
Lingala	**65.59**	43.85	43.42	46.07	45.62	45.68	37.34
Luganda	43.09	**48.32**	40.75	44.92	43.53	43.64	32.09
Nepali	53.42	**56.19**	50.06	49.15	52.39	49.35	50.25
Nigerian Fulfulde	—	22.31	—	**23.96**	22.23	23.80	—
Oromo	—	**43.60**	35.66	42.89	41.01	40.23	32.05
Pashto	39.47	**48.52**	42.66	41.54	42.25	40.10	33.66
Somali	44.92	**46.62**	28.05	44.68	45.44	44.53	37.18
Tigrinya (Eritrean)	23.75	**24.89**	—	22.22	20.64	23.45	—
Tigrinya (Ethiopian)	24.09	**25.41**	—	24.03	23.07	24.25	—
Zulu	52.76	**53.61**	48.18	52.30	53.34	50.31	42.54

Table 16: 2026 EX CHRF++ scores. The best system for each language is in bold.

Language	X to English						
	MS	Google	GPT-4o	GPT-5.4	GPT-5.4-mini	Gemini	Claude
——Pivot——							
Arabic	44.91	45.36	42.88	44.62	44.42	**46.57**	40.40
Chinese	36.40	35.00	33.35	35.38	35.21	**36.59**	33.16
French	**45.57**	43.87	43.01	43.48	43.21	44.17	41.35
Hindi	53.28	**58.67**	46.22	53.37	53.62	53.69	49.70
Indonesian	50.53	**53.93**	49.31	52.70	52.17	51.08	48.67
Portuguese (BR)	**55.46**	53.94	52.91	54.82	54.82	55.40	52.25
Russian	37.06	**38.87**	36.18	36.39	35.69	37.93	34.36
Spanish (LA)	**55.48**	53.22	52.78	53.23	52.73	54.08	50.58
Swahili	42.24	**48.35**	42.00	47.30	45.30	47.21	38.38
——Important——							
Bengali	46.53	**51.06**	42.22	47.26	47.05	47.60	43.17
Farsi	39.41	**43.84**	39.10	40.41	40.04	41.94	35.99
Malay	55.11	**58.98**	50.86	57.59	56.47	54.54	52.47
Marathi	39.25	**43.69**	37.13	41.14	40.85	40.97	35.72
Myanmar	31.82	**36.66**	27.10	33.92	33.95	35.34	27.63
Tagalog	60.13	**64.53**	52.89	59.63	60.14	58.20	54.21
Tamil	37.75	**42.55**	31.81	38.96	38.82	40.58	33.89
Urdu	38.37	**43.04**	35.42	39.87	39.07	40.39	34.98
——Priority——							
Amharic	33.31	**38.18**	26.15	35.38	35.36	37.04	29.94
Dari	37.72	**41.99**	37.60	38.90	38.35	40.79	34.68
Dinka	—	**14.27**	—	5.44	5.02	11.50	—
Hausa	34.16	**40.23**	32.11	36.34	35.12	36.85	24.64
Kanuri	—	**24.45**	7.25	12.32	8.63	12.29	—
Khmer	38.72	**45.06**	36.71	42.66	42.68	43.71	34.93
Kinyarwanda	25.74	29.01	25.36	26.99	25.89	**29.04**	18.38
Kurdish (Kurmanji)	35.81	**42.77**	29.94	40.48	36.26	41.21	31.80
Kurdish (Sorani)	31.78	34.03	21.02	28.28	25.76	**37.56**	27.88
Lingala	**31.47**	26.00	17.83	19.71	18.91	26.81	16.77
Luganda	30.86	**35.22**	25.46	28.49	27.42	33.85	22.01
Nepali	50.90	**54.58**	43.52	51.03	50.73	50.93	46.37
Nigerian Fulfulde	—	**26.38**	12.03	17.02	11.85	21.31	—
Oromo	—	31.12	25.69	29.40	27.43	**31.83**	18.99
Pashto	32.44	**43.48**	35.37	40.61	40.24	41.33	32.76
Somali	33.73	**39.69**	32.90	38.20	36.72	38.28	26.00
Tigrinya (Eritrean)	22.62	24.52	15.88	25.91	24.86	**27.44**	17.52
Tigrinya (Ethiopian)	27.58	29.61	14.00	31.21	31.04	**31.39**	21.81
Zulu	43.68	**49.81**	41.35	47.83	46.45	47.63	34.94

Table 17: 2026 XE BLEU scores. The best system for each language is in bold.

Language	X to English						
	MS	Google	GPT-4o	GPT-5.4	GPT-5.4-mini	Gemini	Claude
——Pivot——							
Arabic	68.06	69.17	66.88	69.12	68.78	**69.19**	66.51
Chinese	62.16	61.87	60.40	**62.53**	62.35	62.20	60.76
French	**64.76**	64.06	63.18	63.97	63.68	64.04	62.67
Hindi	73.92	**76.79**	69.39	74.52	74.29	74.13	72.26
Indonesian	72.53	**74.33**	71.53	73.77	73.27	72.70	71.61
Portuguese (BR)	75.86	75.21	73.35	75.77	75.61	**75.91**	74.37
Russian	63.73	**64.70**	62.71	64.03	63.54	64.07	62.51
Spanish (LA)	**75.22**	74.70	74.09	74.87	74.47	75.08	73.57
Swahili	63.97	**68.14**	63.51	68.10	66.35	67.91	62.19
——Important——							
Bengali	68.76	**71.94**	66.46	70.77	70.21	70.19	68.40
Farsi	64.64	**67.55**	64.25	66.21	65.77	66.14	63.45
Malay	74.17	**76.92**	71.59	76.21	75.56	74.68	73.24
Marathi	63.75	**67.19**	62.86	66.43	65.76	65.49	62.58
Myanmar	57.02	**61.79**	53.90	61.43	60.78	60.87	55.82
Tagalog	77.01	**79.52**	72.39	77.33	77.42	76.67	73.85
Tamil	62.74	**66.14**	57.57	64.78	64.25	64.72	61.31
Urdu	62.53	**66.55**	61.18	65.30	64.48	64.84	62.01
——Priority——							
Amharic	57.54	60.98	51.41	60.71	59.82	**61.11**	56.14
Dari	63.11	**65.88**	62.54	65.09	64.67	65.36	62.05
Dinka	—	**33.63**	—	18.71	17.48	30.69	—
Hausa	55.40	**60.16**	53.78	57.75	56.23	58.91	48.57
Kanuri	—	**46.33**	21.82	31.26	23.53	32.21	—
Khmer	60.67	**67.50**	60.85	66.87	66.43	66.46	61.21
Kinyarwanda	48.40	**52.93**	48.08	50.88	49.87	52.52	43.64
Kurdish (Kurmanji)	58.73	**63.61**	52.97	63.19	58.86	63.52	56.32
Kurdish (Sorani)	56.53	57.09	41.84	49.45	46.52	**60.66**	53.37
Lingala	**52.21**	47.85	36.95	39.07	37.69	48.86	40.07
Luganda	51.28	55.21	45.27	47.99	46.41	**55.49**	46.26
Nepali	71.96	**74.38**	67.49	72.92	72.50	72.42	69.83
Nigerian Fulfulde	—	**47.30**	30.12	35.94	27.56	42.43	—
Oromo	—	54.24	49.91	52.21	49.94	**55.45**	43.42
Pashto	58.82	**65.92**	59.97	64.95	64.14	64.71	58.96
Somali	54.71	**59.39**	53.90	59.09	57.74	58.96	49.26
Tigrinya (Eritrean)	46.59	49.32	37.82	52.21	50.89	**52.64**	43.59
Tigrinya (Ethiopian)	51.17	53.70	36.56	**56.83**	55.97	56.17	47.83
Zulu	64.51	**68.42**	62.32	67.13	65.90	67.68	57.93

Table 18: 2026 XE CHRF++ scores. The best system for each language is in bold.

Language	English to X						
	MS	Google	GPT-4o	GPT-5.4	GPT-5.4-mini	Gemini	Claude
——Pivot——							
Arabic	0.8744	0.8800	0.8759	**0.8830**	0.8788	0.8797	0.8718
Chinese	0.8801	**0.8882**	0.8834	0.8710	0.8843	0.8716	0.8773
French	0.8772	0.8440	0.8692	**0.8801**	0.8791	0.8703	0.8701
Hindi	**0.9038**	0.8876	0.8809	0.8863	0.8982	0.8915	0.8956
Indonesian	**0.9301**	0.9249	0.9283	0.9292	0.9297	0.9273	0.9258
Portuguese (BR)	0.9310	0.9251	0.9213	0.9321	0.9314	**0.9332**	0.9320
Russian	0.8723	**0.8781**	0.8737	0.8765	0.8764	0.8764	0.8735
Spanish (LA)	0.9287	0.9284	0.9311	**0.9348**	0.9318	0.9329	0.9302
Swahili	0.8722	**0.8737**	0.8728	0.8690	0.8713	0.8694	0.8556
——Important——							
Bengali	**0.8668**	0.8564	0.8602	0.8542	0.8627	0.8481	0.8576
Farsi	0.8747	**0.8851**	0.8773	0.8670	0.8756	0.8700	0.8389
Malay	**0.9269**	0.9236	0.8855	0.9234	0.9264	0.9176	0.9154
Marathi	**0.8400**	0.8362	0.8296	0.8330	0.8347	0.8327	0.8229
Myanmar	0.8187	**0.8375**	0.7726	0.8317	0.8355	0.8275	0.7865
Tagalog	**0.8910**	0.8902	0.8882	0.8705	0.8894	0.8797	0.8809
Tamil	**0.8673**	0.8480	0.8159	0.8403	0.8369	0.8433	0.8293
Urdu	0.8582	**0.8613**	0.8464	0.8484	0.8558	0.8473	0.8513
——Priority——							
Amharic	0.9508	0.9544	0.9545	**0.9588**	0.9576	0.9569	0.9330
Dari	**0.8537**	—	0.8161	0.8388	0.8441	0.8433	0.8381
Dinka	—	**0.7552**	—	0.7328	0.7265	—	—
Hausa	**0.8379**	0.8206	0.8251	0.8173	0.8257	0.8149	0.8022
Kanuri	—	**0.7318**	—	0.6999	0.6888	—	—
Khmer	0.9488	0.9413	0.9373	0.9270	**0.9507**	0.9264	0.9118
Kinyarwanda	0.7959	0.7995	0.7799	**0.8140**	0.8109	0.8128	0.7551
Kurdish (Kurmanji)	**0.8161**	0.8118	0.7862	—	0.8038	—	0.7909
Kurdish (Sorani)	**0.8648**	0.8615	0.8463	0.8540	0.8509	0.8570	0.8311
Lingala	**0.8667**	0.7822	0.7922	0.7940	0.7932	0.7926	0.7668
Luganda	0.7829	**0.8095**	0.7797	0.7921	0.7904	0.7886	0.7386
Nepali	0.8676	**0.8754**	0.8409	0.8527	0.8668	0.8566	0.8614
Nigerian Fulfulde	—	0.7300	—	**0.7396**	0.7295	0.7131	—
Oromo	—	**0.8754**	0.7681	0.7940	0.7874	0.7875	0.7518
Pashto	0.8200	**0.8421**	0.8232	0.8248	0.8285	0.8175	0.7847
Somali	0.8211	**0.8216**	0.7516	0.8081	0.8146	0.8092	0.7823
Tigrinya (Eritrean)	0.9483	—	—	0.9514	**0.9523**	0.9466	—
Tigrinya (Ethiopian)	0.9542	0.9462	—	0.9533	**0.9563**	0.9510	—
Zulu	**0.8478**	0.8385	0.7969	0.8330	0.8394	0.8269	0.7982

Table 19: 2026 EX BERTScores. The best system for each language is in bold.

Language	X to English						
	MS	Google	GPT-4o	GPT-5.4	GPT-5.4-mini	Gemini	Claude
——Pivot——							
Arabic	0.7486	0.7416	0.7406	0.7514	0.7506	**0.7641**	0.7285
Chinese	0.7148	0.6959	-0.5152	0.7098	0.7118	**0.7160**	0.6920
French	**0.6865**	0.6627	0.6702	0.6762	0.6744	0.6810	0.6700
Hindi	0.8113	**0.8225**	-0.7376	0.8118	0.8162	0.8206	0.7948
Indonesian	0.8087	0.8108	0.8005	0.8155	0.8099	**0.8167**	0.8030
Portuguese (BR)	0.8307	0.8058	-0.0443	0.8332	0.8317	**0.8343**	0.8181
Russian	0.7173	0.7133	0.7018	0.7096	0.7058	**0.7233**	0.7041
Spanish (LA)	0.8199	0.7965	0.8134	0.8191	0.8166	**0.8204**	0.8057
Swahili	0.7236	0.7574	-0.4285	0.7447	0.7310	**0.7756**	0.7111
——Important——							
Bengali	0.7711	0.7744	0.7398	0.7665	0.7661	**0.7919**	0.7605
Farsi	0.7426	0.7535	0.7395	0.7467	0.7470	**0.7611**	0.7237
Malay	0.8220	0.8267	-0.4093	0.8331	0.8285	**0.8344**	0.8141
Marathi	0.7319	0.7456	0.7208	0.7436	0.7400	**0.7499**	0.7059
Myanmar	0.6585	0.6929	-0.7524	0.6982	0.6908	**0.7001**	0.6259
Tagalog	0.8380	0.8394	-0.3387	0.8471	**0.8472**	0.8468	0.8079
Tamil	0.7033	0.7126	-0.9708	0.6792	0.6738	**0.7322**	0.6792
Urdu	0.7104	0.7375	-0.5403	0.7314	0.7273	**0.7421**	0.7019
——Priority——							
Amharic	0.6584	0.6944	0.6055	0.6933	0.6915	**0.7103**	0.6346
Dari	0.7166	0.7214	-0.5382	0.7306	0.7261	**0.7390**	0.7016
Dinka	—	**0.3791**	—	-1.2368	-1.4992	0.3156	—
Hausa	0.6388	0.6850	0.3746	0.4153	0.3907	**0.6986**	0.5458
Kanuri	—	**0.4763**	-0.7413	0.2510	-0.7590	0.2883	—
Khmer	0.6914	0.7431	-0.8413	0.7518	0.7496	**0.7563**	0.6946
Kinyarwanda	0.5777	0.6236	-0.6382	0.4456	0.4462	**0.6426**	0.5005
Kurdish (Kurmanji)	0.6673	0.7046	0.4258	0.7096	0.4806	**0.7238**	0.6259
Kurdish (Sorani)	0.6267	0.6448	-0.5270	-0.3058	-0.4956	**0.6883**	0.5896
Lingala	**0.6376**	0.5847	-0.9227	-0.8905	-0.8785	0.6056	0.4390
Luganda	0.6038	0.6555	-0.3596	-0.3446	-0.3695	**0.6655**	0.5135
Nepali	0.7943	0.8004	-0.7168	0.7972	0.7978	**0.8091**	0.7706
Nigerian Fulfulde	—	**0.5271**	-0.5379	0.1528	-1.0359	0.4766	—
Oromo	—	0.6211	-0.5438	0.2383	0.1954	**0.6474**	0.4658
Pashto	0.6427	0.7312	-0.5797	0.7239	0.7172	**0.7379**	0.6623
Somali	0.6139	0.6601	0.6208	0.6291	0.6148	**0.6667**	0.5355
Tigrinya (Eritrean)	0.5596	0.5740	0.4233	0.6063	0.5983	**0.6208**	0.4677
Tigrinya (Ethiopian)	0.6141	0.6262	-0.7240	0.6680	0.6628	**0.6706**	0.5311
Zulu	0.7397	0.7619	-0.4587	0.6760	0.6737	**0.7823**	0.6637

Table 20: 2026 XE BERTScores (rescaled). The best system for each language is in bold.

Language	English to X						
	MS	Google	GPT-4o	GPT-5.4	GPT-5.4-mini	Gemini	Claude
——Pivot——							
Arabic	0.8485	0.8750	0.8683	**0.8764**	0.8688	0.8761	0.8649
Chinese	0.8749	**0.8953**	0.8840	0.8753	0.8862	0.8779	0.8804
French	0.8154	0.8242	0.8199	**0.8245**	0.8229	0.8239	0.8199
Hindi	0.8076	0.8203	0.8064	0.8061	**0.8217**	0.8177	0.8148
Indonesian	0.9189	**0.9254**	0.9234	0.9251	0.9237	0.9253	0.9212
Portuguese (BR)	0.8994	0.9037	0.8965	0.9041	0.9020	**0.9053**	0.9022
Russian	0.8775	**0.9027**	0.8889	0.8939	0.8898	0.8996	0.8898
Spanish (LA)	0.8839	**0.8932**	0.8886	0.8923	0.8879	0.8903	0.8880
Swahili	0.8236	0.8392	0.8406	0.8431	0.8400	**0.8448**	0.8139
——Important——							
Bengali	0.8513	**0.8749**	0.8624	0.8629	0.8686	0.8634	0.8565
Farsi	0.8782	**0.8901**	0.8822	0.8716	0.8825	0.8806	0.8476
Malay	0.9071	0.9065	0.8838	0.9074	**0.9086**	0.9068	0.9016
Marathi	0.7216	**0.7460**	0.7240	0.7366	0.7332	0.7409	0.7133
Myanmar	0.8719	0.8849	0.8407	0.8946	0.8906	**0.8961**	0.8016
Tagalog	0.8672	0.8682	**0.8685**	0.8449	0.8655	0.8638	0.8543
Tamil	0.8751	**0.8911**	0.8503	0.8787	0.8706	0.8851	0.8409
Urdu	0.8230	0.8172	0.8181	0.8210	**0.8282**	0.8247	0.8194
——Priority——							
Amharic	0.8533	0.8656	0.7891	0.8612	0.8526	**0.8696**	0.8057
Hausa	0.7993	0.8087	0.8053	0.8060	0.8052	**0.8108**	0.7592
Khmer	0.8560	0.8734	0.8364	0.8683	0.8662	**0.8744**	0.7787
Kinyarwanda	0.6271	0.6473	0.6327	**0.6549**	0.6496	0.6536	0.6060
Kurdish (Kurmanji)	0.8033	**0.8116**	0.7359	—	0.7917	—	0.7630
Nepali	0.8159	0.8358	0.8112	0.8203	**0.8368**	0.8328	0.8227
Oromo	—	**0.8358**	0.7683	0.8040	0.7917	0.7995	0.7507
Pashto	0.7877	**0.8128**	0.7999	0.8115	0.8120	0.8106	0.7561
Somali	0.8008	**0.8070**	0.6765	0.7950	0.7996	0.8029	0.7389

Table 21: 2026 EX COMET. The best system for each language is in bold. Only supported languages are included.

Language	X to English						
	MS	Google	GPT-4o	GPT-5.4	GPT-5.4-mini	Gemini	Claude
——Pivot——							
Arabic	0.8725	0.8779	0.8765	0.8788	0.8784	**0.8820**	0.8716
Chinese	0.8751	0.8765	0.6173	0.8778	0.8781	**0.8790**	0.8728
French	0.8360	0.8353	0.8352	**0.8370**	0.8362	0.8367	0.8334
Hindi	0.9070	**0.9150**	0.4515	0.9121	0.9125	0.9135	0.9075
Indonesian	0.9032	**0.9086**	0.9047	0.9079	0.9067	0.9072	0.9036
Portuguese (BR)	0.9018	0.9009	0.7162	**0.9042**	0.9039	0.9041	0.8990
Russian	0.8637	**0.8684**	0.8640	0.8664	0.8652	**0.8684**	0.8628
Spanish (LA)	0.8981	0.8966	0.8975	0.8989	0.8984	**0.8995**	0.8948
Swahili	0.8459	0.8679	0.5409	0.8685	0.8633	**0.8722**	0.8472
——Important——							
Bengali	0.8995	0.9048	0.9005	0.9062	0.9054	**0.9087**	0.9016
Farsi	0.8846	0.8924	0.8889	0.8906	0.8901	**0.8930**	0.8823
Malay	0.8984	0.9061	0.6778	**0.9067**	0.9051	0.9060	0.8992
Marathi	0.8806	**0.8903**	0.8838	0.8894	0.8879	0.8899	0.8778
Myanmar	0.8696	0.8876	0.3951	0.8894	0.8868	**0.8897**	0.8689
Tagalog	0.8899	0.8970	0.6188	**0.8997**	0.8982	0.8988	0.8850
Tamil	0.8710	0.8816	0.4526	0.8798	0.8784	**0.8869**	0.8731
Urdu	0.8694	0.8854	0.4548	0.8838	0.8816	**0.8856**	0.8743
——Priority——							
Amharic	0.8569	0.8782	0.8486	0.8784	0.8748	**0.8813**	0.8576
Hausa	0.7886	0.8243	0.7936	0.8064	0.7956	**0.8302**	0.7569
Khmer	0.8647	0.8910	0.4306	0.8938	0.8923	**0.8944**	0.8750
Kinyarwanda	0.7233	0.7621	0.2931	0.7519	0.7460	**0.7693**	0.6950
Kurdish (Kurmanji)	0.8139	0.8403	0.7938	0.8439	0.8131	**0.8468**	0.8063
Nepali	0.9113	0.9173	0.4189	0.9175	0.9170	**0.9188**	0.9103
Oromo	—	0.7876	0.3212	0.7731	0.7595	**0.8080**	0.7250
Pashto	0.8499	0.8752	0.3743	0.8749	0.8716	**0.8765**	0.8532
Somali	0.7857	**0.8250**	0.8101	0.8182	0.8105	0.6990	0.7607

Table 22: 2026 XE COMET. The best system for each language is in bold. Only supported languages are included.

	English to X				X to English			
Language	**Microsoft**		**Google**		**Microsoft**		**Google**	
	2023	2025	2023	2025	2023	2025	2023	2025
——Pivot——								
Arabic	30.26	25.16	29.15	32.88	44.88	44.87	46.90	49.28
Portuguese (BR)	52.42	47.83	52.27	54.26	54.81	55.33	55.19	57.63
Chinese	52.89	50.24	54.29	54.27	36.46	36.48	37.71	39.56
French	45.61	40.35	34.14	45.42	45.25	45.32	44.25	46.03
Hindi	43.24	40.27	46.00	43.48	53.17	53.28	56.24	58.80
Indonesian	54.39	46.85	55.07	54.82	51.73	50.29	51.15	54.61
Spanish (LA)	56.06	51.33	56.86	57.12	54.75	54.59	54.15	55.88
Russian	36.88	32.95	36.71	37.35	36.06	36.84	37.14	40.14
Swahili	33.10	30.74	33.88	33.87	41.04	41.92	45.56	49.21
——Important——								
Bengali	19.29	17.18	22.95	21.14	44.38	46.08	49.46	51.33
Farsi	27.38	25.25	29.20	29.88	38.97	39.35	39.65	44.45
Malay	51.06	45.78	53.33	53.31	50.88	54.08	57.63	59.70
Marathi	15.52	13.53	16.75	16.09	37.21	37.34	42.30	44.43
Myanmar	53.55	56.15	56.95	56.62	23.82	31.44	32.39	37.33
Tagalog	—	49.40	48.46	48.82	—	59.92	—	65.55
Tamil	24.17	24.84	21.53	22.76	34.21	40.73	39.38	43.11
Urdu	23.50	20.41	28.14	28.12	35.93	35.88	40.61	43.82
——Priority——								
Amharic	8.67	7.04	11.88	11.79	26.49	33.02	36.51	38.60
Dari	17.32	15.21	—	—	31.21	31.22	—	—
Dinka	—	—	—	4.54	—	—	—	14.47
Eritrean Tigrinya	3.39	3.47	6.13	6.13	18.07	18.00	24.83	—
Ethiopian Tigrinya	3.83	3.98	6.64	6.69	21.60	21.52	30.33	29.94
Hausa	15.98	25.23	25.87	28.17	—	33.92	36.17	40.65
Kanuri	—	—	—	5.51	—	—	—	26.08
Khmer	61.20	65.93	66.50	66.08	27.41	37.04	36.31	45.76
Kinyarwanda	—	18.75	—	17.88	—	25.39	—	29.49
Kurdish Kurmanji	—	11.82	—	10.05	—	28.17	—	28.80
Kurdish Sorani	—	10.38	10.66	10.85	—	31.87	34.18	34.13
Lingala	32.07	28.35	15.08	15.17	—	29.14	26.04	26.22
Luganda	—	17.26	19.71	20.64	—	30.52	—	35.45
Nepali	16.71	14.20	25.44	23.56	48.38	48.38	53.76	55.16
Nigerian Fulfulde	—	—	—	3.70	—	—	—	26.97
Oromo	—	—	9.06	10.23	—	—	31.36	31.52
Pashto	21.12	12.79	26.09	24.63	27.57	32.16	40.75	44.07
Somali	26.01	26.02	17.90	17.56	33.03	33.12	37.61	19.99
Zulu	19.76	20.28	19.60	19.71	40.26	43.40	47.58	50.63

Table 23: BLEU Score Differences between 2023 and 2025 for Microsoft and Google

	English to X				X to English			
Language	**Microsoft**		**Google**		**Microsoft**		**Google**	
	2023	2025	2023	2025	2023	2025	2023	2025
——Pivot——								
Arabic	57.73	56.41	56.70	59.18	67.74	67.73	69.61	70.87
Portuguese (BR)	72.79	71.12	72.48	74.04	75.43	75.88	75.45	76.79
Chinese	39.76	45.07	48.72	45.33	62.08	62.69	39.76	64.60
French	64.72	64.32	62.09	65.16	64.50	64.74	64.02	65.25
Hindi	62.72	64.56	65.04	64.83	73.50	73.67	75.63	77.15
Indonesian	74.32	72.03	74.28	75.10	72.89	72.53	72.73	74.71
Spanish (LA)	74.39	73.16	74.47	75.30	75.13	75.33	74.97	75.48
Russian	59.89	58.91	59.56	60.61	63.50	63.75	63.73	65.27
Swahili	58.41	57.99	59.18	59.34	62.44	63.95	66.22	68.58
——Important——								
Bengali	49.04	48.5	52.4	52.0	68.01	69.17	70.84	72.53
Farsi	53.52	53.58	54.69	56.49	63.61	64.57	64.84	67.88
Malay	72.78	71.73	73.76	74.0	71.99	74.15	76.12	77.53
Marathi	46.34	44.68	46.08	46.40	62.3	62.34	65.83	67.53
Myanmar	37.89	39.86	43.36	43.21	50.36	57.57	59.35	62.10
Tagalog	—	70.41	69.36	69.93	—	77.06	—	80.40
Tamil	56.68	57.22	54.18	56.13	61.83	64.78	63.9	66.76
Urdu	50.03	48.96	52.99	53.15	60.91	60.9	64.76	66.89
——Priority——								
Amharic	30.12	29.61	33.92	33.83	51.10	57.00	59.63	61.38
Dari	42.20	41.55	—	—	57.44	57.45	—	—
Dinka	—	—	—	25.37	—	—	—	33.82
Eritrean Tigrinya	19.30	19.31	25.00	25.12	42.98	42.93	49.69	—
Ethiopian Tigrinya	19.49	19.52	25.81	25.99	46.29	46.36	54.67	54.19
Hausa	40.87	52.66	52.52	52.71	—	55.44	57.37	60.58
Kanuri	—	—	—		—	—	—	30.16
Khmer	45.82	52.38	53.10	52.97	53.7	58.69	60.26	67.88
Kinyarwanda	—	45.85	—	45.18	—	48.35	—	53.13
Kurdish Kurmanji	—	39.87	—	33.49	—	51.81	—	47.14
Kurdish Sorani	—	41.83	43.61	43.77	—	56.70	57.29	57.26
Lingala	59.00	58.18	43.83	43.86	—	50.49	47.99	48.02
Luganda	—	43.07	48.24	48.53	—	51.23	—	55.67
Nepali	49.30	48.72	57.01	56.55	70.46	70.4	73.75	74.95
Nigerian Fulfulde	—	—	—	22.38	—	—	—	48.12
Oromo	—	—	43.75	44.00	—	—	54.42	54.48
Pashto	46.63	37.84	49.11	48.76	54.45	58.83	63.51	66.21
Somali	53.01	53.01	45.16	45.97	54.30	54.37	57.53	37.13
Zulu	53.43	53.46	53.21	53.13	60.78	64.51	67.02	69.21

Table 24: CHRF++ Score Differences between 2023 and 2025 for Microsoft and Google

	English to X				X to English			
Language	**Microsoft**		**Google**		**Microsoft**		**Google**	
	2023	2025	2023	2025	2023	2025	2023	2025
——Pivot——								
Arabic	0.8841	0.8375	0.8791	0.8833	0.7661	0.7465	0.7701	0.7987
Portuguese (BR)	0.9363	0.8750	0.9339	0.9521	0.8434	0.8302	0.8248	0.8437
Chinese	0.8930	0.8619	0.8971	0.9450	0.7373	0.7234	0.7271	0.7400
French	0.8837	0.8300	0.8465	0.8838	0.7078	0.6863	0.6930	0.7164
Hindi	0.8945	0.8570	0.9023	0.9649	0.8156	0.8103	0.8166	0.8423
Indonesian	0.9325	0.8705	0.9308	0.9352	0.8300	0.8123	0.8114	0.8451
Spanish (LA)	0.9379	0.8773	0.9349	0.9394	0.8362	0.8278	0.8191	0.8427
Russian	0.8869	0.8339	0.8873	0.8907	0.7373	0.7214	0.7291	0.7584
Swahili	0.8706	0.8263	0.8720	0.9281	0.7282	0.7235	0.7564	0.7814
——Important——								
Bengali	0.8475	0.8192	0.8625	0.8755	0.7748	0.7702	0.7878	0.8169
Farsi	0.8778	0.8377	0.8820	0.8904	0.7528	0.7400	0.7428	0.7900
Malay	0.9283	0.8691	0.9286	0.9494	0.8151	0.8214	0.8342	0.8465
Marathi	0.8417	0.8044	0.8413	0.8461	0.7302	0.7123	0.7487	0.7822
Myanmar	0.7982	0.7835	0.8288	0.9441	0.5944	0.6627	0.6788	0.7104
Tagalog	—	0.9400	0.9017	0.9406	—	0.8423	—	0.7427
Tamil	0.8674	0.8352	0.8588	0.9647	0.6981	0.7203	0.7083	0.7465
Urdu	0.8491	0.8095	0.8623	0.9514	0.7104	0.6927	0.7349	0.7536
——Priority——								
Amharic	0.9503	0.8473	0.9412	0.9422	0.5900	0.6559	0.6915	0.7347
Dari	0.8365	0.7956	—	—	0.6642	0.6578	—	—
Dinka	—	—	—	0.8674	—	—	—	0.3917
Eritrean Tigrinya	0.9366	0.8418	0.9431	0.9375	0.5233	0.5212	0.6006	0.6394
Ethiopian Tigrinya	0.9408	0.8449	0.9464	0.9374	0.5687	0.5227	0.6668	0.6394
Hausa	0.7840	0.7853	0.8287	0.8359	—	0.6398	0.6719	0.7234
Kanuri	—	—	—	0.8330	—	—	—	0.7800
Khmer	0.8755	0.8017	0.9480	0.9600	0.6098	0.7234	0.6976	0.8200
Kinyarwanda	—	0.8800	—	0.8000	—	0.5752	—	0.6358
Kurdish Kurmanji	—	0.9300	—	0.8128	—	0.6337	—	0.7466
Kurdish Sorani	—	0.8700	0.8643	0.8674	—	0.5620	0.6651	0.6773
Lingala	0.8481	0.7961	0.7924	0.7936	—	0.6126	0.6075	0.6247
Luganda	—	0.8700	0.8112	0.8147	—	0.6091	—	0.6972
Nepali	0.8534	0.8180	0.8804	0.9595	0.7879	0.7823	0.8087	0.8293
Nigerian Fulfulde	—	—	—	0.8400	—	—	—	0.5623
Oromo	—	—	0.7963	0.8885	—	—	0.6477	0.6318
Pashto	0.8472	0.7908	0.8517	0.9456	0.5942	0.6430	0.7229	0.7420
Somali	0.7808	0.7487	0.7543	0.8635	0.3793	0.3405	0.3945	0.3474
Zulu	0.8418	0.7876	0.8403	0.9136	0.7179	0.7396	0.7738	0.7536

Table 25: BERTScore Differences between 2023 and 2025 for Microsoft and Google

	English to X				X to English			
Language	**Microsoft**		**Google**		**Microsoft**		**Google**	
	2023	2025	2023	2025	2023	2025	2023	2025
——Pivot——								
Arabic	0.8450	0.8253	0.8486	0.8813	0.8710	0.8400	0.8777	0.8891
Portuguese (BR)	0.8978	0.8698	0.8976	0.8962	0.9007	0.8790	0.9002	0.8983
Chinese	0.8824	0.8652	0.8924	0.8910	0.8738	0.8090	0.8755	0.8752
French	0.8139	0.7967	0.8099	0.8910	0.8351	0.8400	0.8333	0.9028
Hindi	0.7908	0.7835	0.8035	0.8272	0.9038	0.8700	0.9094	0.9136
Indonesian	0.9156	0.8894	0.9168	0.9239	0.9042	0.9000	0.9032	0.9097
Spanish (LA)	0.8858	0.8583	0.8835	0.8918	0.8979	0.8600	0.8962	0.9025
Russian	0.8843	0.8633	0.8819	0.9000	0.8621	0.8100	0.8633	0.8724
Swahili	0.8119	0.8007	0.8200	0.8412	0.8380	0.8200	0.8588	0.8649
——Important——								
Bengali	0.8465	0.8328	0.8556	0.8751	0.8952	0.8700	0.9046	0.9124
Farsi	0.8578	0.8490	0.8704	0.8933	0.8801	0.8300	0.8836	0.8931
Malay	0.8964	0.8736	0.9010	0.8977	0.8901	0.8560	0.9032	0.9016
Marathi	0.7255	0.7113	0.7164	0.7402	0.8741	0.8800	0.8858	0.8930
Myanmar	—	0.8629	—	0.8673	—	0.8840	—	0.8801
Tagalog	—	0.8140	—	0.8503	—	0.8200	—	0.9005
Tamil	—	0.9348	—	0.92420	—	0.8600	—	0.8540
Urdu	0.8062	0.7892	0.8168	0.8125	0.8618	0.8400	0.8770	0.8821
——Priority——								
Amharic	0.8390	0.8306	0.8400	0.8641	0.8161	0.8500	0.8691	0.8722
Dari	—	—	—	—	—	—	—	—
Dinka	—	—	—	—	—	—	—	—
Eritrean Tigrinya	—	—	—	—	—	—	—	—
Ethiopian Tigrinya	—	—	—	—	—	—	—	—
Hausa	—	0.8548	—	0.7957	—	0.8100	—	0.8217
Kanuri	—	—	—	—	—	—	—	—
Khmer	—	0.7848	—	0.7928	—	0.8900	—	0.8854
Kinyarwanda	—	—	—	—	—	—	—	—
Kurdish Kurmanji	—	0.4583	—	0.8049	—	0.8090	—	0.8370
Kurdish Sorani	—	—	—	—	—	—	—	—
Lingala	—	—	—	—	—	—	—	—
Luganda	—	—	—	—	—	—	—	—
Nepali	0.8013	0.7854	0.8227	0.8413	0.9050	0.8600	0.9162	0.9192
Nigerian Fulfulde	—	—	—	—	—	—	—	—
Oromo	—	—	—	0.7912	—	—	—	0.9030
Pashto	0.7828	0.7617	0.7921	0.8112	0.8192	0.8430	0.8631	0.8720
Somali	0.7189	0.7077	0.6995	0.7018	0.6617	0.8200	0.6831	0.4749
Zulu	—	—	—	—	—	—	—	—

Table 26: COMET Score Differences between 2023 and 2025 for Microsoft and Google

Cross-Modal Inconsistency in Multimodal LLMs during Crisis Communication

Anuridhi Gupta
Humanitarian Informatics Lab
George Mason University
Fairfax, VA, USA
agupta29@gmu.edu

Hemant Purohit
Humanitarian Informatics Lab
George Mason University
Fairfax, VA, USA
hpurohit@gmu.edu

Abstract

Effective crisis communication relies on language technologies that can serve everyone. Although multimodal large language models (MM-LLMs) accept both speech and text, it remains unclear whether equivalent inputs produce equivalent crisis guidance across modalities, particularly for people with access and functional needs. In this paper, we evaluate three state-of-the-art (SOTA) MM-LLMs on consistency of their outputs across text and audio modalities using real emergency alerts. Building on prior semantic-similarity and factual-overlap evaluation, we introduce a LIWC-aligned analysis of modality dependent differences in linguistic structure, interpersonal framing, risk, and action-oriented language. Our findings indicate that despite being unified models, capable of taking in input from different modalities, no single model was able to achieve reliable cross-modal consistency. Persona-level results further indicate that cross-modal disparities vary across user groups. We release text prompts, audio prompts, model outputs and evaluation materials to support the crisis language technology community in building equitable solutions for crisis management.

1 Introduction

Crisis informatics has emerged as a critical area of research at the intersection of natural language processing and humanitarian action. The emergence of multimodal large language models (MM-LLMs) capable of processing and generating content across text, audio, image, and video within a single unified architecture (Yin et al., 2024) represents the next avenue for crisis language technology. Conversational interfaces allow users to submit crisis queries by voice or text and receive actionable written guidance in return (Urbanelli et al., 2024).

However, an important question is whether these systems give the same information regardless of whether the user speaks or types. Disasters disproportionately harm those who are least equipped to respond, including the 43% of the U.S. population with access and functional needs (Federal Emergency Management Agency, 2024). Among them are approximately 48 million Americans with hearing loss (National Institute on Deafness and Other Communication Disorders, 2024) and 8.2 million with vision disabilities (USAFacts, 2024) who may be unable to use standard alerts. For these individuals, omitted safety-critical details can hinder informed decisions and compromise safety.

Human interaction is inherently multimodal as we combine sight, hearing, touch, and other senses to perceive the world (Turk, 2014). Yet, many people around the world are limited in these possibilities due to hearing, vision, speech, and cognitive disabilities (Karpov and Ronzhin, 2014), so emergency information must be accessible and adapted to their needs and preferences. High-stakes events such as disasters further worsen their experience, making them vulnerable to more physical and mental injuries (Tannenbaum-Baruchi et al., 2024; Stough and Kelman, 2017). Failure to account for their access and functional needs, such as when communicating information about the nearest shelters, can result in serious consequences.

Although artificial intelligence (AI) systems have shown promising applications for crisis communication, for AI systems to understand the situation, they need to be able to interpret and reason about these diverse interactions that individuals with disabilities require (Baltrušaitis et al., 2018). As these MM-LLMs move toward deployment, their evaluation plays a crucial role in development and deployment. Despite the claim of a unified MM system, these systems often demonstrate misalignment in understanding and response generation across input modalities (Zhao et al., 2025). They are prone to hallucinations, are fragile to adversarial input, and fail to produce appropriate responses in high-stakes domains, such as neglect-

Proceedings of the First Workshop on Language Technologies for Crisis Preparedness and Response, pages 115–125
November 9, 2026 ©2026 Association for Computational Linguistics

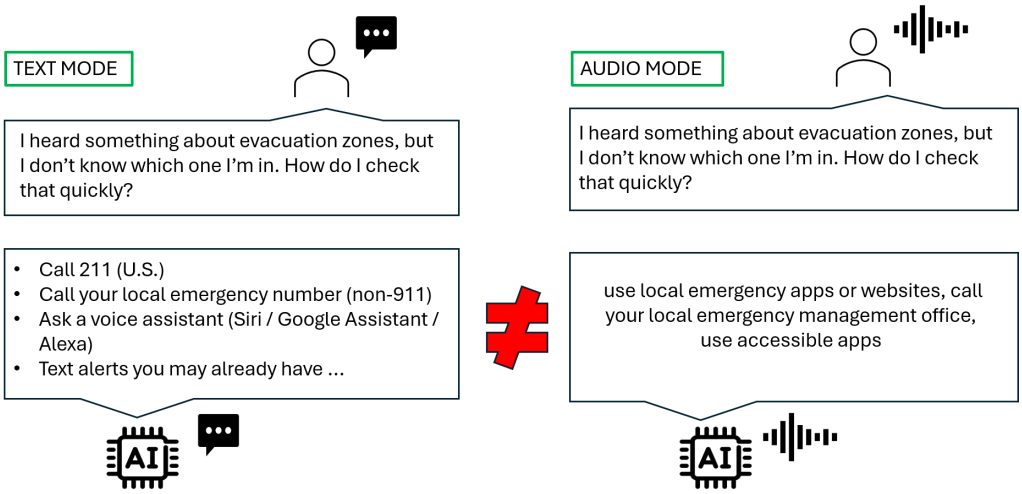


Figure 1: Illustrative GPT-4 interaction motivating our work on exploring the difference in responses across same emergency query given through text and voice input respectively.

ing the urgency of the risk and accessibility constraints in disaster risk communication. Previous work has suggested that some of these issues correlate with LLM inconsistency, generally defined as their tendency to generate low-confidence responses or conflicting responses when the same prompt is resampled (Wu et al., 2025). Accurate estimation of MM-LLM consistency is important during critical applications and supports user trust in the model (Liang et al., 2024). Building on the semantic-similarity and factual-overlap evaluation of Gupta et al. (2026), this study adds linguistic analysis using LIWC2015 categories and an adapted Language Style Matching measure.

Figure 1 demonstrates the problem through a GPT-4 interaction involving text and spoken responses. We test whether this pattern holds more systematically in three open-weight audio-language models: AudioFlamingo, SALMONN, and Qwen. Each model receives the same emergency question as typed text and as audio generated using text-to-speech (TTS), and we compare the two textual responses. Unlike Figure 1, all responses in our evaluation are text.

Problem Statement. MM-LLMs are increasingly capable of accepting semantically equivalent requests through speech and text, but multimodal capability does not ensure that the resulting guidance is consistent across input modalities. In crisis communication, modality-dependent omissions or changes can reduce the usefulness of guidance and create unequal service for vulnerable users. Existing multimodal benchmarks do not adequately measure this form of cross-modal informational consistency in safety-critical, crisis queries describing different user needs.

Contributions. We (1) extend an existing evaluation framework using audio and text prompts derived from Federal Emergency Management Agency (FEMA)'s Integrated Public Alert and Warning System (IPAWS) alert scenarios and instantiated across four access and functional needs personas; (2) provide a multifaceted evaluation combining semantic similarity, lexical overlap, preservation of safety-critical information, and LIWC-aligned linguistic analysis; and (3) release text prompts, audio prompts, model outputs, and evaluation materials for reproducible research on equitable crisis language technology. The text prompts, audio prompts, model responses are available in our public repository.[1]

Section 2 explores related work, Section 3 presents the framework, Section 4 reports results, Sections 5 discusses implications and Section 6 concludes with limitations and future work.

2 Related Work

2.1 Language Technologies in Crisis

Crisis informatics is a growing field that connects natural language processing with disaster preparedness, response, and recovery. Early work demonstrated the value of social media monitoring and text classification for situational awareness during

[1] https://github.com/samara-m1/multimodal-disaster-eval

active disasters (Purohit et al., 2014), establishing a foundation for NLP-driven crisis response that includes information extraction, rumor detection, and needs identification from crisis-generated text (Alam et al., 2021). The International Telecommunication Union has identified telecommunications and early warning systems as foundational infrastructure for all phases of disaster risk reduction, with particular emphasis on the importance of clear and usable guidance (International Telecommunication Union, 2025). With the emergence of LLMs, new possibilities have opened for interactive, personalized crisis communication (Urbanelli et al., 2024). Early work explored ChatGPT's potential for disaster prevention information dissemination and emergency response support because its natural language reasoning provided an advantage over static alert systems (Xue et al., 2023). More recently, LLMs have been used to classify crisis information from social media streams, monitor infrastructure during active disasters, and generate structured warning messages based on official guidelines (Linardos et al., 2025; Chen et al., 2024).

2.2 Accessibility in Crisis

People with access and functional needs are systematically underserved by existing crisis communication infrastructure. Traditional broadcast-based emergency alert systems such as sirens, television crawls, and Wireless Emergency Alerts have well-documented limitations in reaching individuals with sensory, cognitive, or linguistic barriers (Villarreal et al., 2025). Research has documented communication failures for hard of hearing individuals during Hurricane Rita, where announcements were exclusively accessible through specific television stations, translators were unavailable at shelters, and information was not conveyed in sign language or any other accessible manner (Tannenbaum-Baruchi et al., 2024). Adverse pregnancy outcomes have been documented during Hurricane Katrina (Callaghan et al., 2007), and significant cognitive deterioration has been observed in elderly individuals with dementia following hurricane and earthquake events (Fahmy and Alme, 2026; Güney and Çiçek Doğan, 2026). A systematic review found that emergency communication is not tailored to specific at-risk populations and adaptations change dissemination channels instead of message content (Lambert et al., 2025). This establishes the necessity of evaluating AI-based crisis communication systems across modalities for users with access and functional needs.

2.3 Cross-Modal Consistency in Speech and Language Models

Audio-capable language models such as SALMONN (Tang et al., 2024) and AudioFlamingo (Goel et al., 2025) have gained traction as they enable audio captioning, speech comprehension, and audio-based reasoning. Qwen3-Omni (Xu et al., 2025) can process text, audio, image, and video within a single unified training pipeline. AudioBench (Wang et al., 2025) provides a universal benchmark for audio LLMs across a range of listening tasks, establishing evaluation norms for the speech community that this work builds upon. A known challenge in these systems is cross-modality interference, where training on one modality degrades performance or consistency in another. Omni-modal LLMs attend to dominant modalities (text) while producing degraded or inconsistent outputs for audio inputs (Jiang et al., 2025).

Existing benchmarks address adjacent but different questions. AudioBench measures audio language understanding across listening tasks (Wang et al., 2025), while Holistic Evaluation of Multimodal Models (HEMM) evaluates models across tasks and datasets (Liang et al., 2024). Wu et al. (2025) show that BERTScore, BLEU, ROUGE, and USE capture different aspects of perceived consistency. Cross-modal consistency evaluates whether semantically equivalent speech and text inputs yield equivalent open-ended guidance. Prior work has not adequately evaluated this property. We address this gap by measuring whether meaning, safety-relevant information, and linguistic framing are preserved across speech and text conditions.

3 Cross-Modal Response Analysis Framework

Our framework consists of four components. The first component is *User Groups and Communication Needs*, which is based on FEMA and Centers for Disease Control and Prevention (CDC) guidelines for individuals with access and functional needs. The next component is *Persona Creation and Prompt Design*, which is grounded in real emergency alerts obtained from FEMA's alerts. Then we employ *Multimodal LLMs* to take inputs in the form of text and audio prompts which are then evaluated using the *Consistency Evaluation*. This framework

is designed to be open, reproducible, and extensible to additional disaster types and languages.

3.1 User Groups and Communication Needs

We grounded our selection of vulnerable populations in the U.S. Centers for Disease Control and Prevention (CDC) formal category of individuals with "access and functional needs" (Centers for Disease Control and Prevention, 2021), which refers to individuals who may need additional support during emergencies because of a temporary or permanent condition, without requiring a clinical diagnosis or medical evaluation. This is a deliberate design choice to keep the framework broadly applicable across many contexts. The CDC's full list spans a wide range of groups, including children, pregnant women, older adults, people with physical, sensory, intellectual, developmental, cognitive, or mental disabilities, people with chronic health conditions, those with limited English proficiency, and those facing financial, transportation, or legal barriers to preparedness. We narrowed this list to four groups for the current study. This includes *pregnant women, mothers with toddlers, hard of hearing individuals, and older adults with dementia*. These subgroups were selected because prior literature documents each group's vulnerability during disasters and each faces acute risk specifically during the communication phase of disaster response, which is this paper's focus.

Disasters have been linked to adverse outcomes for pregnant women, including increased rates of preterm birth, pregnancy complications, and maternal mortality (Harville et al., 2021; Callaghan et al., 2007). Mothers with toddlers face increased risk of acute illness, developmental disabilities, and maternal anxiety in post-disaster contexts (Yamazaki and Nakai, 2023; Nishihara et al., 2018). Hard of hearing individuals represent one of the most directly impacted groups as standard alerting systems including sirens, broadcast audio, and Wireless Emergency Alerts with sound are either inaccessible or partially inaccessible to them (Tannenbaum-Baruchi et al., 2024). Finally, research shows the adverse impacts of disasters on older adults with dementia, with documented declines in working memory, daily functioning, and general cognitive abilities after crisis events (Fahmy and Alme, 2026; Güney and Çiçek Doğan, 2026).

3.2 Persona Creation and Prompt Design

Each of the four stakeholder groups forms a persona used to generate realistic post-alert queries. We drew these scenarios from FEMA's Integrated Public Alert & Warning System (IPAWS)[2], the infrastructure that distributes verified alerts to the public over Wireless Emergency Alerts, the Emergency Alert System, and NOAA Weather Radio. Using real alerts, we built 40 prompts spanning alert types such as flash floods, hurricanes, tornadoes, dust advisories, and excessive heat warnings.

The prompt template follows a consistent structure intended to model naturalistic post-alert user behavior where a person receives an official emergency alert and asks an AI assistant for guidance relevant to their specific situation. This interaction is motivated by the Protective Action Decision Model, which describes warning reception, threat assessment, protective-action assessment, and information seeking as components of public responses to hazards (Lindell and Perry, 2012). Prior research emphasizes that people facing imminent threats need clear, specific, and actionable protective guidance and that warning responses are influenced by individual and situational characteristics (Sutton and Kuligowski, 2019; Kim et al., 2019). With that in mind, we designed the prompts. For example, a *Flash Flood Warning* prompt is *"I am a X, and I just received a flash flood warning. What should I do during this time to stay safe?"* where X represents one of the four personas. Appendix A provides representative queries for all four personas. For each persona–scenario combination, the same prompt was presented to the model as text and as audio. Audio inputs were synthesized from the query text with the Coqui TTS Python package, using the pretrained Tacotron 2 model (Shen et al., 2018), and saved as WAV files. Personas specify the user needs described in each query. Each persona is evaluated under both input conditions to test whether the model responds consistently to the same stated need. This design does not assume that members of a persona group prefer audio input. The synthesized voice does not simulate the speech characteristics of that group. The study does not evaluate variation across human speakers or challenging acoustic conditions such as background noise and reverberation.

[2] https://www.fema.gov/emergency-managers/practitioners/integrated-public-alert-warning-system

3.3 Multimodal LLMs

We evaluate three state-of-the-art open-weight MM-LLMs, selected for their distinct architectural approaches to speech and audio understanding, their public availability, and their relevance to the speech and language technology community. The generation settings are specified in our inference scripts on GitHub. AudioFlamingo 3 and Qwen3-Omni were configured with a maximum of 512 newly generated tokens, while SALMONN was invoked without explicit decoding overrides. Remaining decoding settings were inherited from the respective model configurations and implementations.

AudioFlamingo 3. Based on a 7B language model and the LLaVA architecture, AudioFlamingo 3 is trained on a unified AF-Whisper audio encoder that handles speech recognition and processing different audio signals, including sound, music, and speech (Goel et al., 2025). Its Whisper-based encoder makes it directly relevant to the task.

Qwen3-Omni. This model is capable of processing text, images, audio, and video while delivering real-time streaming responses in both text and natural speech (Xu et al., 2025). Its multilingual capabilities are relevant to our future work in low-resource and cross-cultural crisis communication.

SALMONN. Speech, Audio, Language, and Music Open Neural Network integrates pretrained audio encoders with an LLM through alignment layers, enabling audio captioning, speech comprehension, and audio-based reasoning (Tang et al., 2024). SALMONN is able to generalize across heterogeneous audio domains, making it suitable for applications requiring contextual understanding of real-world speech signals in crisis contexts.

3.4 Consistency Evaluation

In the quantitative evaluation, an *audio response* denotes the textual response generated when an audio input is processed through the models' audio branch, whereas a *text response* denotes the textual response generated from an input text through the models' text branch.

To quantify cross-modal consistency, we employ the following evaluation approaches across all 160 text and audio response pairs (40 prompts across 39 disaster alert types $\times$ 4 personas). The semantic-similarity and factual-overlap components follow the evaluation framework introduced by Gupta et al. (2026), while the psycholinguistic similarity extends that framework to examine differences in linguistic style and framing.

Semantic Similarity. We employ four different metrics. BERTScore (SBERT) (Zhang et al., 2020) captures deep semantic similarity using contextual embeddings. BLEU (SBLEU) (Papineni et al., 2002) and ROUGE (SROUGE) (Lin, 2004) capture lexical overlap at the n-gram level. Universal Sentence Encoder (SUSE) (Cer et al., 2018) captures sentence-level semantic similarity. Together, they provide a nuanced view of consistency.

Psycholinguistic Similarity. To complement the semantic metrics with a linguistic characterization of cross-modal consistency, we analyzed each response using LIWC2015 (Pennebaker et al., 2015). We then computed a category-level Language Style Matching (LSM) score (Niederhoffer and Pennebaker, 2002; Ireland and Pennebaker, 2010) between the responses produced by each pair of matched audio and text prompts. For LIWC category c, the score is

$$\mathrm{LSM}_c = 1 - \frac{|A_c - T_c|}{A_c + T_c + 0.0001}, \qquad (1)$$

where A_c and T_c are the percentages of words assigned to category c in the audio-prompt and text-prompt responses, respectively. The resulting score ranges from 0, indicating maximal divergence, to 1, indicating identical proportional usage. When both category values are zero, the score equals 1 and represents shared absence of that category.

Classical LSM focuses primarily on function-word coordination between human interlocutors. We extend its normalized similarity formulation to broader LIWC2015 categories to characterize modality-dependent differences in linguistic framing. Category-level scores were averaged across the 160 matched response pairs for each model. Where categories were grouped into broader dimensions, we then calculated the unweighted mean of their category-level scores. Word count was reported separately as a structural measure and excluded from the composite LSM score because LSM is intended for proportional language-category values rather than raw counts. The analyzed dimensions include function words and pronouns, cognitive and affective processes, social orientation, motivation, risk, perception, motion, and temporal focus. The purpose of this analysis is to determine whether responses that appear semantically similar differ in linguistic framing such as in their expression of urgency, certainty, risk, direct user address, or action orientation.

Table 1: Semantic similarity scores across the three MM-LLMs. (AF = AudioFlamingo)

Model	SROUGE	SBLEU	SUSE	SBERT
AF	0.223	0.065	0.619	0.495
SALMONN	0.249	0.132	0.696	0.625
Qwen	0.341	0.343	**0.896**	**0.896**

Factual Overlap. Following Gupta et al. (2026), we use a safety-critical information-overlap proxy that combines named entities, safety keywords, and ROUGE-L, treating the text response as the reference and the audio response as the comparison. It estimates whether instructions such as "stay indoors," "move to higher ground," or "avoid downed power lines" are preserved across modes. We report factual overlap scores both overall and broken down by persona in Table 2, enabling analysis of whether consistency gaps are uniform across user groups or concentrated among specific access and functional need populations. The response to text input serves as a comparison reference. Lower agreement does not, by itself, establish that either response is less correct, complete, actionable, or safe.

4 Results and Discussion

We present our findings through semantic similarity, factual overlap, linguistic analysis, and qualitative error examination, each revealing a distinct layer of the cross-modal inconsistency problem. Together, the findings build a cumulative case that inconsistency in MM-LLMs for disaster risk communication is not a surface-level quality gap.

4.1 Cross-Model Consistency Analysis

Table 1 reports semantic similarity scores for the three models. Qwen achieved the strongest performance across all four metrics, with an SBERT score of 0.896 indicating that the underlying meaning of responses was preserved more effectively across modalities than in the other two models. SALMONN demonstrated moderate performance, and AudioFlamingo (AF) performed worst across all metrics, indicating the weakest alignment in both meaning and wording across modalities.

4.2 Cross-Modal Consistency Analysis

A critical pattern that emerges across all models is that SUSE and SBERT are consistently higher than SBLEU and SROUGE, indicating that while models often preserve the general meaning of a response, they vary substantially in wording and structure across modalities. This distinction has direct implications for crisis communication. Qwen's close scores between SUSE (0.896) and SBERT (0.896) indicate relatively robust semantic grounding. However, its substantially lower SBLEU and SROUGE scores indicate that consistency is achieved through paraphrasing rather than faithful content reproduction. In safety-critical scenarios, reformulation of key guidance can significantly alter the actionability of a response. Semantic similarity alone is therefore insufficient to evaluate cross-modal consistency in crisis communication contexts. We address this in the next subsection.

Table 2: Factual overlap scores by persona across all three MM-LLMs. (AF = AudioFlamingo)

Persona	AF	SALMONN	Qwen
Pregnant woman	0.335	0.296	**0.452**
Mother w/ toddler	0.265	0.315	**0.449**
Elderly w/ dementia	0.271	**0.335**	0.426
Hard of hearing	0.237	0.306	**0.397**
Overall	0.277	0.313	**0.431**

4.3 Cross-Persona Consistency Analysis

Table 2 presents factual overlap scores broken down by persona for all three models, capturing whether safety-critical facts present in text responses are preserved in corresponding audio responses. Within AudioFlamingo and Qwen, the hard-of-hearing persona has the lowest factual-overlap score among the four personas evaluated. This pattern does not hold for SALMONN, where the pregnant-woman persona scores lowest, while the elderly with dementia persona scores highest (0.335). The rankings therefore differ across models. These scores indicate differences in response agreement and lack of stability across models. Persona rankings differ across models and a model can appear broadly consistent while systematically under serving specific user groups.

4.4 Language Style Matching Analysis

Table 3 reports mean LSM scores across the three models for the LIWC2015 dimensions most relevant to crisis communication. These include direct address (You), authoritative language (Power), risk language (Risk), uncertainty (Tentative), present-focused action framing (Focuspresent), social and motion language. Qwen has the highest LSM score in each reported linguistic dimension and

Table 3: Language Style Matching (LSM) scores between text and audio responses across crisis communication dimensions, computed from LIWC2015. Audio and Text indicate the mean category score per 100 words (raw word count for the Word Count row). LSM ranges from 0 (no correspondence) to 1 (identical usage).

Dimension	Model	Audio	Text	LSM
You	AudioFlamingo	0.44	3.58	**0.209**
	SALMONN	4.61	5.01	0.698
	Qwen	5.76	5.57	0.917
Power	AudioFlamingo	1.28	4.20	**0.268**
	SALMONN	2.57	3.15	0.678
	Qwen	2.84	2.75	0.881
Tentative	AudioFlamingo	2.00	5.07	0.315
	SALMONN	5.20	5.99	0.755
	Qwen	6.33	5.95	0.913
Risk	AudioFlamingo	7.25	3.85	0.558
	SALMONN	4.89	3.46	0.727
	Qwen	3.54	3.69	0.896
Social	AudioFlamingo	9.27	7.16	0.606
	SALMONN	9.35	8.66	0.789
	Qwen	11.96	12.14	0.934
Motion	AudioFlamingo	8.73	6.81	0.716
	SALMONN	5.44	5.83	0.752
	Qwen	4.46	4.55	0.895
Word Count	AudioFlamingo	28.07	45.48	0.701
	SALMONN	117.24	122.62	0.860
	Qwen	411.23	449.61	0.933
Focuspresent	AudioFlamingo	10.42	13.74	0.772
	SALMONN	12.62	14.43	0.859
	Qwen	13.04	12.85	0.938
Mean LSM	AudioFlamingo	0.492		
	SALMONN	0.751		
	Qwen	0.911		

the highest mean LSM score (0.911), followed by SALMONN (0.751), with AudioFlamingo showing the lowest consistency by a wide margin (0.492). AudioFlamingo's weakest dimensions are You (0.209) and Power (0.268), indicating that its audio responses diverge most from text in how directly they address the user and how authoritatively they instruct. Responses to audio input were shorter on average than responses to text input for all three models.

Persona-level LSM scores, computed identically to the model-level scores but broken out by persona (pooled across all three models), show that consistency loss is not uniform across the four groups and does not have a single ranking across dimensions. The hard-of-hearing persona has the lowest overall composite LSM (0.699) of the four personas, driven specifically by the You (0.580) and Motion (0.748) dimensions, indicating that responses for the hard of hearing persona exhibit particularly large modality dependent differences in direct address and action-oriented language. The mother with toddler persona shows its weakest consistency on Tentative (0.616) and Social (0.745) language, a difference in certainty and social framing rather than in direct address. The elderly with dementia persona is weakest specifically on Power (0.588) which is the authoritative dimension. The pregnant woman persona shows the highest composite consistency (0.741) and is the least divergent persona on four of the seven dimensions (You, Power, Tentative, Focuspresent). The observed linguistic differences vary across personas.

4.5 Qualitative Response Analysis

The quantitative and linguistic differences identified above also appear in the response examples shown in Table 4. In both cases, text responses provide detailed, actionable guidance tailored to the specific hazard, for example, avoiding dust exposure, staying hydrated, limiting physical exertion whereas corresponding audio responses are generic, shorter, and in some cases fail to address the scenario meaningfully.

In the dust advisory case, the audio response omits critical protective measures such as filtration, indoor sheltering, and mask usage and replaces them with vague instructions to "follow safety protocols." In a humanitarian context, where affected individuals may lack access to follow-on resources or reliable connectivity, such vagueness is not merely a quality issue but a safety risk. Elderly individuals with dementia are particularly vulnerable here as they are unlikely to seek clarification or consult secondary sources under crisis conditions (Bell et al., 2023), meaning the first response they receive is often the only one they act on. For the second case the text response provides specific, actionable guidance such as staying indoors, drinking plenty of water and using fans while the audio response instructs the user to "follow safety protocols" which is entirely unrelated to the specific disaster and the user's condition. Additional Qwen examples in Appendix B further demonstrate these differences.

5 Implications for Crisis Language Technology

Our findings indicate that cross-modal inconsistency in MM-LLMs is a systematic problem that ap-

Table 4: Examples of poor cross-modal consistency in AudioFlamingo. Blue: situation; Green: text-only guidance.

Input	Text Response	Audio Response
Elderly w/ dementia; dust advisory	Stay indoors, avoid dust, close windows, use a HEPA filter. Wear a dust mask if going outside.	Follow safety protocols, seek healthcare guidance, stay informed.
Mother w/ toddler; heat warning	Stay indoors, avoid sun, drink plenty of water, use fans or AC.	Follow safety protocols, seek shelter, stay informed.

pears across all three models and all four personas. The inconsistency is evident in semantic similarity (SBERT and SUSE), lexical overlap (SBLEU and SROUGE), preservation of safety-critical information (factual overlap), and linguistic style and framing (LIWC-aligned categories). Additionally, response agreement varies across personas and models, without a uniform ranking across groups.

For the crisis language technology community, these findings motivate three concrete directions. **First, cross-modal consistency must be treated as a top priority evaluation criterion** alongside accuracy and fluency, with training pipelines incorporating explicit consistency objectives (Tang et al., 2024; Xu et al., 2025). **Second, evaluation frameworks for crisis-oriented speech and language systems must include persona-level analysis** to reveal the inequity that emerges at the level of individual user groups. **Third, the linguistic dimensions most affected by cross-modal inconsistency are direct address (You), authoritative language (Power), and uncertainty (Tentative), and they provide an actionable target set for future fine-tuning efforts on disaster risk communication corpora**. A model unable to preserve directive, personally addressed language in audio input is not fit for deployment in humanitarian crisis contexts, despite its semantic similarity scores.

6 Conclusion

This paper extends prior evaluation of cross-modal consistency in disaster assistance with an analysis of linguistic style and framing. This is a critical property for crisis language technology systems serving users with access and functional needs. Using real emergency alerts from the publicly available FEMA IPAWS archive and persona-based scenarios grounded in CDC guidelines, we evaluated AudioFlamingo, SALMONN, and Qwen across text and audio modalities using semantic similarity metrics, factual overlap scoring, linguistic analysis, and qualitative error examination.

Our findings reveal that no model achieves reliable consistency across text and audio modalities. Audio responses were generally shorter and preserved fewer safety-relevant details than their text-prompt counterparts. The LIWC analysis further revealed modality-dependent differences in direct address, authoritative framing, and uncertainty.

Limitations and Future Work. This study has several limitations that point directly to future directions. Audio prompts were synthesized using a single TTS configuration and do not capture variation across human speakers. Future work will incorporate diverse speaker profiles and personas. The evaluation is English-only and uses U.S.-based alert scenarios, limiting generalization to multilingual and cross-cultural disaster contexts and extending this framework to low-resource language settings is an immediate priority aligned with the LT4CPR community's mission. Preliminary repeated testing on same modality prompt showed variation across generations, but the variation was not in terms of specificity or actionable detail. We plan to conduct this testing more systematically across a set of metrics. We also plan on independently assessing each model's speech recognition accuracy. Finally, this evaluation is purely computational, meaning that people with access and functional needs were not involved as participants. We are actively conducting focus groups and interviews with this population to validate our persona designs with lived experiences to inform the next phase of this research.

7 Acknowledgment

This research was partially supported by the grant # 2531369 from the National Science Foundation. During the preparation of this work, the authors used Claude and OpenAI Codex for code assistance, literature review support and proofreading. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the manuscript.

References

F. Alam, H. Sajjad, M. Imran, and P. Nakov. 2021. HumAID: Human-annotated disaster incidents data from twitter with deep learning benchmarks. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 15, pages 933–942.

Tadas Baltrušaitis, Chaitanya Ahuja, and Louis-Philippe Morency. 2018. Multimodal machine learning: A survey and taxonomy. *IEEE transactions on pattern analysis and machine intelligence*, 41(2):423–443.

Sue Anne Bell, Marie Lynn Miranda, Julie PW Bynum, and Matthew A Davis. 2023. Mortality after exposure to a hurricane among older adults living with dementia. *JAMA network open*, 6(3):e232043.

William M Callaghan, Sonja A Rasmussen, Denise J Jamieson, Stephanie J Ventura, Sherry L Farr, Paul D Sutton, Thomas J Mathews, Brady E Hamilton, Katherine R Shealy, Dabo Brantley, et al. 2007. Health concerns of women and infants in times of natural disasters: lessons learned from hurricane katrina. *Maternal and child health journal*, 11(4):307–311.

Centers for Disease Control and Prevention. 2021. Access and functional needs toolkit: Integrating a community partner network to inform risk communication strategies. Technical report, CDC Center for Preparedness and Response, Atlanta, GA.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, et al. 2018. Universal sentence encoder for english. In *Proceedings of the 2018 conference on empirical methods in natural language processing: system demonstrations*, pages 169–174.

Minze Chen, Zhenxiang Tao, Weitong Tang, Tingxin Qin, Rui Yang, and Chunli Zhu. 2024. Enhancing emergency decision-making with knowledge graphs and large language models. *International Journal of Disaster Risk Reduction*, 113:104804.

Christina Fahmy and Maria Nordheim Alme. 2026. Effects of natural disasters on the cognitive state of older adults with dementia: a scoping review. *European Geriatric Medicine*, pages 1–11.

Federal Emergency Management Agency. 2024. Access and functional needs support. Fact sheet, FEMA Mass Care and Emergency Assistance, Washington, DC.

Arushi Goel, Sreyan Ghosh, Jaehyeon Kim, Sonal Kumar, Zhifeng Kong, Sang-gil Lee, Chao-Han Huck Yang, Ramani Duraiswami, Dinesh Manocha, Rafael Valle, et al. 2025. Audio flamingo 3: Advancing audio intelligence with fully open large audio language models. *arXiv preprint arXiv:2507.08128*.

Seda Güney and Özlem Çiçek Doğan. 2026. Experiences of people with dementia and their family caregivers after earthquakes: A qualitative study. *Dementia*, 25(2):315–331.

Anuridhi Gupta, Samara Mansoor, and Hemant Purohit. 2026. Evaluating multimodal llms across text and audio modalities for accessible disaster assistance. *arXiv preprint arXiv:2608.14651*.

Emily W Harville, Leslie Beitsch, Christopher K Uejio, Samendra Sherchan, and Maureen Y Lichtveld. 2021. Assessing the effects of disasters and their aftermath on pregnancy and infant outcomes: A conceptual model. *International Journal of Disaster Risk Reduction*, 62:102415.

International Telecommunication Union. 2025. The use of telecommunications/ICT for disaster risk reduction and management. Output Report on ITU-D Question 3/1 D-STG-SG01.03.1-2025, ITU-D Study Group 1, Geneva, Switzerland. Study period 2022–2025.

Molly E Ireland and James W Pennebaker. 2010. Language style matching in writing: synchrony in essays, correspondence, and poetry. *Journal of personality and social psychology*, 99(3):549.

Shixin Jiang, Jiafeng Liang, Jiyuan Wang, Xuan Dong, Heng Chang, Weijiang Yu, Jinhua Du, Ming Liu, and Bing Qin. 2025. From specific-mllms to omni-mllms: a survey on mllms aligned with multi-modalities. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 8617–8652.

Alexey Karpov and Andrey Ronzhin. 2014. A universal assistive technology with multimodal input and multimedia output interfaces. In *International Conference on Universal Access in Human-Computer Interaction*, pages 369–378. Springer.

Grace Kim, Andrea Martel, David Eisenman, Michael Prelip, Armen Arevian, Kerri L Johnson, and Deborah Glik. 2019. Wireless emergency alert messages: Influences on protective action behaviour. *Journal of Contingencies and Crisis Management*, 27(4):374–386.

Hélène Lambert, Dominique Doumont, Nina Reviers, Mieke Vandenbroucke, and Isabelle Aujoulat. 2025. Enhancing accessibility of crisis communication to people in vulnerable circumstances. *Journal of Public Health*, pages 1–13.

Paul Pu Liang, Akshay Goindani, Talha Chafekar, Leena Mathur, Haofei Yu, Ruslan Salakhutdinov, and Louis-Philippe Morency. 2024. Hemm: Holistic evaluation of multimodal foundation models. *Advances in Neural Information Processing Systems*, 37:42899–42940.

C.-Y. Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81.

Vasileios Linardos, Maria Drakaki, and Panagiotis Tzionas. 2025. Utilizing llms and ml algorithms in disaster-related social media content. *GeoHazards*, 6(3):33.

Michael K Lindell and Ronald W Perry. 2012. The protective action decision model: Theoretical modifications and additional evidence. *Risk Analysis: An International Journal*, 32(4):616–632.

National Institute on Deafness and Other Communication Disorders. 2024. Quick statistics about hearing.

Kate G Niederhoffer and James W Pennebaker. 2002. Linguistic style matching in social interaction. *Journal of language and social psychology*, 21(4):337–360.

Mika Nishihara, Yasuhide Nakamura, Toru Fuchimukai, and Mayumi Ohnishi. 2018. Factors associated with social support in child-rearing among mothers in post-disaster communities. *Environmental health and preventive medicine*, 23(1):58.

K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. 2002. BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318.

James W. Pennebaker, Ryan L. Boyd, Kayla Jordan, and Kate Blackburn. 2015. The development and psychometric properties of liwc2015. Technical report, University of Texas at Austin.

Hemant Purohit, Carlos Castillo, Fernando Diaz, Amit Sheth, and Patrick Meier. 2014. Emergency-relief coordination on social media: Automatically matching resource requests and offers. *First Monday*.

Jonathan Shen, Ruoming Pang, Ron J Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, Rj Skerrv-Ryan, et al. 2018. Natural tts synthesis by conditioning wavenet on mel spectrogram predictions. In *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 4779–4783. IEEE.

Laura M Stough and Ilan Kelman. 2017. People with disabilities and disasters. In *Handbook of disaster research*, pages 225–242. Springer.

Jeannette Sutton and Erica D Kuligowski. 2019. Alerts and warnings on short messaging channels: Guidance from an expert panel process. *Natural Hazards Review*, 20(2):04019002.

Changli Tang, Wenyi Yu, Guangzhi Sun, Xianzhao Chen, Tian Tan, Wei Li, Lu Lu, Zejun Ma, and Chao Zhang. 2024. Salmonn: Towards generic hearing abilities for large language models. In *International Conference on Learning Representations*, volume 2024, pages 16607–16629.

Carolina Tannenbaum-Baruchi, Isaac Ashkenazi, and Carmit Rapaport. 2024. Risk inclusion of vulnerable people during a climate-related disaster: a case study of people with hearing loss facing wildfires. *International Journal of Disaster Risk Reduction*, 103:104335.

Matthew Turk. 2014. Multimodal interaction: A review. *Pattern recognition letters*, 36:189–195.

Angelica Urbanelli, Antonella Frisiello, Luca Bruno, and Claudio Rossi. 2024. The ermes chatbot: A conversational communication tool for improved emergency management and disaster risk reduction. *International Journal of Disaster Risk Reduction*, 112:104792.

USAFacts. 2024. How many people have a disability in the US?

Melissa Villarreal, Carson MacPherson-Krutsky, and Mary Angelica Painter. 2025. Barriers and best practices for inclusive emergency alerts and warnings. *International Journal of Disaster Risk Reduction*, 125:105581.

Bin Wang, Xunlong Zou, Geyu Lin, Shuo Sun, Zhuohan Liu, Wenyu Zhang, Zhengyuan Liu, AiTi Aw, and Nancy Chen. 2025. Audiobench: A universal benchmark for audio large language models. In *Proceedings of the NAACL-2025*, pages 4297–4316.

Xiaoyuan Wu, Weiran Lin, Omer Akgul, and Lujo Bauer. 2025. Estimating llm consistency: A user baseline vs surrogate metrics. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 30518–30532.

Jin Xu, Zhifang Guo, Hangrui Hu, Yunfei Chu, Xiong Wang, Jinzheng He, Yuxuan Wang, Xian Shi, Ting He, Xinfa Zhu, et al. 2025. Qwen3-omni technical report. *arXiv preprint arXiv:2509.17765*.

Zhiwen Xue, Chong Xu, and Xiwei Xu. 2023. Application of chatgpt in natural disaster prevention and reduction. *Natural Hazards Research*, 3(3):556–562.

Chisato Yamazaki and Hisao Nakai. 2023. Understanding mothers' worries about the effects of disaster evacuation on their children: a cross-sectional study. *International journal of environmental research and public health*, 20(3):1850.

Shukang Yin, Chaoyou Fu, Sirui Zhao, Ke Li, Xing Sun, Tong Xu, and Enhong Chen. 2024. A survey on multimodal large language models. *National Science Review*, 11(12):nwae403.

T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi. 2020. BERTScore: Evaluating text generation with BERT. In *International Conference on Learning Representations*.

Shanshan Zhao, Xinjie Zhang, Jintao Guo, Jiakui Hu, Lunhao Duan, Minghao Fu, Yong Xien Chng, Guo-Hua Wang, Qing-Guo Chen, Zhao Xu, et al. 2025. Unified multimodal understanding and generation models: Advances, challenges, and opportunities. *arXiv preprint arXiv:2505.02567*.

A Representative Persona Prompts

Table 5 presents representative queries for the four personas.

Table 5: Representative persona prompts used to obtain textual responses, derived from real Wireless Emergency Alerts.

Persona Name	**Alert Type**	**Persona Query**
Pregnant woman	Flood warning	*"I am a pregnant woman, and I just received a flood warning. What should I do during this time to stay safe?"*
Mother with toddler	Flood warning	*"I am a mother with a toddler, and I just received a flood warning. What should I do during this time to stay safe?"*
Hard-of-hearing individual	Flood warning	*"I am a hard-of-hearing individual, and I just received a flood warning. What should I do during this time to stay safe?"*
Elderly individual with dementia	Flood warning	*"I am an elderly individual with dementia, and I just received a flood warning. What should I do during this time to stay safe?"*

B Additional Qwen Response Examples

Table 6 presents selected Qwen response pairs illustrating differences in the specific details included under text and audio input conditions.

Table 6: Selected Qwen response excerpts for the same query under text and TTS audio input conditions. Both responses are textual. Blue highlights the hazard; green highlights details present in the text-input response but absent from the corresponding full audio-input response. Excerpts preserve wording, with Markdown formatting removed.

Input	**Text Response**	**Audio Response**
I am a hard of hearing individual, and I just received a flash flood warning. What should I do during this time to stay safe?	Monitor local news and weather updates via TV with closed captioning, apps with visual alerts, or text-based emergency notifications.	Follow local news on TV or radio if accessible.
I am an elderly individual with dementia, and I just received a excessive heat warning. What should I do during this time to stay safe?	Use fans if you don't have air conditioning, but know that fans alone may not prevent heat illness in extreme heat.	Use fans or air conditioning
I am a mother with a toddler, and I just received a excessive heat warning. What should I do during this time to stay safe?	Use a wide-brimmed hat and sunscreen (SPF 30+), and reapply every 2 hours or after sweating or wiping.	Use a wide-brimmed hat and sunglasses if outside.

Critical Evaluation of Knowledge Graph-powered Question Answering Systems to Enable Responsible AI for Disaster Management

Gayatri Ramchandra Vaidya, Yasas Senarath, Hemant Purohit
Humanitarian Informatics Lab, George Mason University, Fairfax, VA, USA
{gvaidya2, ywijesu, hpurohit}@gmu.edu

Abstract

When a disaster strikes, question-answering (QA) systems promise to turn scattered satellite, infrastructure, and hazard impact data into actionable answers for coordinators working under time pressure. But how critically should we trust those answers, even if grounded in the background knowledge? We present RISE, a prototype knowledge-graph (KG)-powered QA system built on data from the June 2026 Venezuela M7.5 earthquake, and subject it to a critical evaluation. Two complementary studies run on the same 30 questions: a stage-wise trace records how each question moves through the pipeline, while an independent LLM-as-a-judge assessment scores the final answer on five language-quality criteria. Taken separately, both studies suggest a system that works: 87% intent accuracy, every question answered, and judge scores averaging above 4 out of 5. Albeit together, they reveal a different picture. Intent misclassifications trace to a linguistic ambiguity no keyword-based classifier can resolve: the same word serves as context in one question and as the information need in another, and lexical matching cannot tell them apart. High scores on grounding and appropriateness do not distinguish a correct answer from a fluent, well-cited answer to the wrong question, because the rubric scores the text against its evidence rather than against the question. Both findings share a common lesson for responsible AI in disaster management to design and develop effective support systems: fluent, well-grounded language by the AI agent is not evidence of a correct answer, and evaluating these systems on their text alone is not enough.

1 Introduction

Humanitarian coordinators, emergency responders, and managers often face a familiar problem: the data to make decisions exists but is scattered across different satellites, hazard zones, and pre-disaster infrastructure records, each in a different format with nothing linking them together. For example, which roads are blocked, which health facilities are damaged, and which municipalities were hit hardest? Integrating these heterogeneous sources into a knowledge graph and letting a user ask questions in plain language has been proposed as a solution. Purohit et al. (2019) proposed a SPARQL-based querying application for disaster-related question answering, noting the substantial cost of manual semantic integration. Xiao et al. (2026) show that standard text-to-SQL methods transfer poorly to disaster analytics and address this with an expert-curated causal knowledge graph between the user's question and the database. But neither examines whether a system that retrieves and routes to the correct data can still produce an answer whose language misleads the person reading it.

This distinction matters for responsible AI in disaster management. Existing RAG systems ground their output in retrieved evidence, and this grounding is assumed to make answers more trustworthy than those generated from the model's internal knowledge alone (Zhou et al., 2024). But recent work questions this assumption. An et al. (2026) separate *lexical grounding* (whether evidence is cited) from *semantic alignment* (whether it is interpreted correctly), and show that explanations can appear grounded even when their interpretation weakens the intended meaning. LLM-as-a-judge evaluation, where a strong model scores answers on quality dimensions, agrees with human judgments over 80% of the time (Zheng et al., 2023), but a judge shown only the answer and its evidence can assess whether the answer matches the evidence, not whether the evidence was right for the question.

What remains largely unexplored is how this gap appears in a working KG-QA pipeline built on real disaster data. Standard evaluations report execution accuracy (Yu et al., 2018; Li et al., 2023), treating a correct result set as success. Xiao et al. (2026) extend this with routing accuracy, causal-

Proceedings of the First Workshop on Language Technologies for Crisis Preparedness and Response, pages 126–133
November 9, 2026 ©2026 Association for Computational Linguistics

graph grounding, and multi-table composition as distinct failure tiers. But all these methods check whether the right data was retrieved and correctly cited. A query can execute successfully, return real records, and be written faithfully yet still fail to answer the actual question.

We address this gap through a critical evaluation of RISE, an AI-powered Recovery Intelligence and Support Engine: a prototype KG-powered QA system built on data from the June 2026 Venezuela earthquake. We ask how critically coordinators should trust its answers, even when those answers are grounded in background knowledge. Our contributions are threefold. First, we present RISE, which combines independent damage-assessment sources with pre-disaster infrastructure and administrative geography, backed by a seven-stage NL-to-Cypher pipeline. Second, we introduce two complementary studies on the same 30 questions: a stage-wise trace of how each question moves through the pipeline, and an independent LLM-as-a-judge assessment of the final answer on five language-quality criteria. Third, we show that the two studies, read together, reveal failures that neither shows alone: intent errors that trace to a linguistic ambiguity no keyword-based classifier can resolve, and high grounding scores that cannot distinguish a correct answer from a fluent, well-cited answer to the wrong question.

Together, these findings carry an important implication for responsible AI in disaster management and for anyone designing decision-support systems: fluent, well-grounded language from an AI agent is not evidence of a correct answer. Evaluating such systems on their text alone is therefore not enough; we must also examine how each answer was produced.

2 Related Work

Knowledge graphs for disaster response. Purohit et al. (2019) introduced DisasterKG, an RDF integration of heterogeneous disaster data, noting the high cost of manual semantic integration. Xiao et al. (2026) show that standard text-to-SQL pipelines transfer poorly to disaster analytics and address this with an expert-curated knowledge graph between the user's question and the database. Neither examines whether a system that retrieves and routes correctly can still generate misleading output.

Text-to-query systems and evaluation. Yu et al. (2018) evaluate natural-language-to-database queries by execution accuracy, and Li et al. (2023) extend this to large real-world databases, judging a query by its result set regardless of surface form. Execution accuracy checks whether the retrieved data is right; it does not check whether the answer built from that data is right for the question. For KG-grounded QA, Pusch and Conrad (2025) show that validating and correcting LLM-generated queries against the schema before execution reduces hallucination. RISE follows this validation approach; our contribution examines what happens after validation succeeds, when a query can answer a different question while remaining fluent and grounded.

LLM-as-a-judge and the grounding illusion. Zheng et al. (2023) show that strong LLM judges agree with human preferences over 80% of the time, which has made LLM-as-a-judge a common evaluation method. An et al. (2026) introduce VEXA, a controlled testbed separating *lexical grounding* (whether evidence is cited) from *semantic alignment* (whether it is interpreted correctly), and show that explanations can appear grounded even when their interpretation weakens the intended message. We demonstrate that the same problem can arise in a working system built on real disaster data: a query may retrieve valid records and produce a fluent response while still answering a different question from the one originally asked.

3 RISE: System Architecture

RISE is designed to answer natural-language questions concerning the June 2026 Venezuela earthquake by translating user queries into Cypher, retrieving relevant information from a Neo4j knowledge graph, and generating grounded natural-language responses from the returned records. The same knowledge graph also supports an interactive map and chatbot interface, enabling spatial visualization of retrieved information for situational awareness. The system consists of three principal components: 1) a Knowledge Graph, 2) an NL-to-Cypher Query Pipeline, and 3) an Interactive User Interface. Figure 1 shows the RISE query pipeline. The system is guided by the following design principles:

Source Provenance is preserved by storing each damage assessment independently, since estimates derived from different modalities and methods may disagree substantially. This allows agreement and disagreement across sources to remain directly

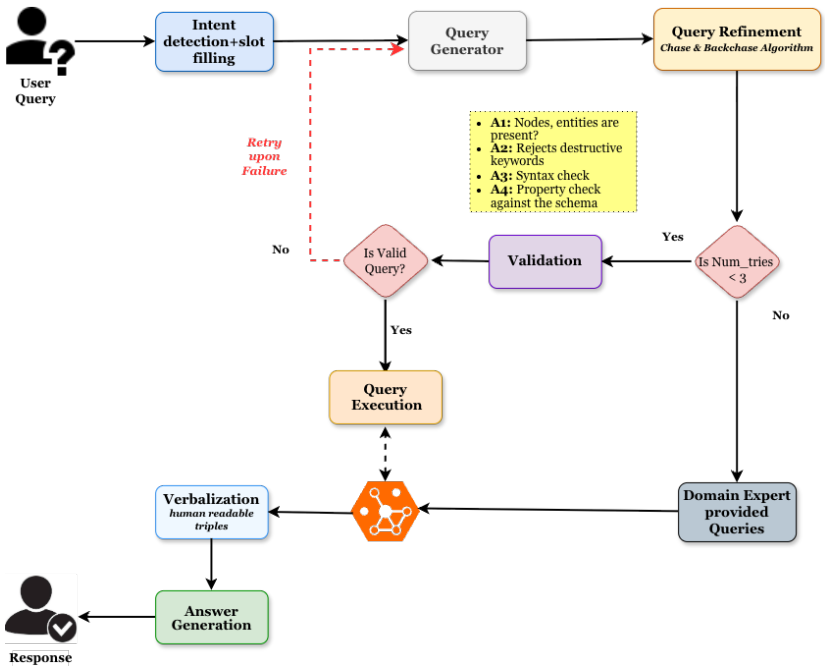


Figure 1: RISE query pipeline architecture.

queryable.

Scope Enforcement limits the knowledge graph to physical damage and location information derived from satellite imagery and OpenStreetMap data. Information on casualties, displacement, shelters, utilities, and agency responsibility is explicitly excluded and enforced during answer generation.

Local Inference is supported through Ollama, ensuring that user queries and retrieved information remain on the host system. This design supports operation under limited connectivity and data-residency constraints, while restricting the system to locally deployable models.

3.1 Knowledge Graph

A knowledge graph represents information as nodes (entities) connected by typed edges (relationships), letting questions be answered by traversing paths rather than joining tables. RISE uses Neo4j with Cypher as its query language, over four groups of information. **Building damage** consists of independent radar and optical assessments (analyst-annotated and AI-derived) stored as `DamagedBuilding` nodes, with confidence and severity kept as separate properties since radar sources report a confidence tier while optical sources report a severity grade. The **pre-earthquake baseline** includes building footprints, roads, and health and education facilities from OpenStreetMap, together with hazardous industrial sites from UNEP/OCHA, with facility subtypes preserved. **Hazard and administrative context** covers the earthquake, its aftershocks, and USGS shaking-intensity zones, alongside states and municipalities from OCHA Common Operational Datasets. Finally, **provenance** is captured through `SOURCED_FROM` edges linking every entity to its data source, while `LOCATED_IN` edges attach entities to every applicable administrative level, so municipality-level questions are answered by direct traversal rather than aggregation.

3.2 The NL-to-Cypher Pipeline

Each question passes through a pipeline motivated by prior work on LLM–KG question answering (Linders and Tomczak, 2025; Pusch and Conrad, 2024), adapted for post-earthquake disaster response.

Intent Detection. Ordered keyword rules classify the question into one of eight intents (*damage_extent*, *road_status*, *health_access*, *education_impact*, *shaking_intensity*, *earthquake_info*, *area_summary*, *out_of_scope*) and extract slots; unmatched questions fall through to a local `llama3.1` (Arora et al., 2024).

Query Generation. A local `qwen2.5-coder` generates a Cypher query from the intent, slots, live schema, and a small set of exemplars (Wang et al., 2026).

Refinement. A Chase and Backchase strategy (Wang et al., 2026) inserts missing relationship hops and removes fields not matching live property keys.

Validation. Four checks run before execution: node/relationship labels (A1), destructive keywords (A2), property keys (A3), and `EXPLAIN` for syntactic errors (A4). Failures return an error hint to the generator, which retries up to three times before falling back following Pusch and Conrad (2024), who show query-checking reduces hallucination in KG-grounded QA.

Execution. The validated query runs against Neo4j. On failure, empty results, or wrong-level rows, the system substitutes a domain-expert template.

Fallback Queries. Pre-authored Cypher templates keyed by intent serve as a deterministic safety net, so the system always returns an answer (Purohit et al., 2019).

Verbalization. Result rows are converted into subject-relation-object triples.

Answer Generation. Triples and the executed query are passed to a local `llama3.1` (Grattafiori et al., 2024), with a prompt that constrains the response to the provided evidence and forbids inference beyond it (e.g., inferring operational status from physical proximity, or reporting figures at a different administrative level than the retrieved rows).

3.3 Interface and Implementation

RISE runs on a FastAPI backend with a Leaflet map front end. Each answer returns both prose and the retrieved records, so a question about a specific area is answered in text and highlighted on the map simultaneously.

4 Experimental setup

We evaluate RISE through two complementary studies conducted on the same set of questions. The first is a stage wise evaluation of the query pipeline which identifies where each question is resolved and, importantly, distinguishes questions handled successfully by the generated query from those recovered through the domain-expert fallback (Section 4.2). The second is an LLM-as-a-judge evaluation of the final responses, where answers are scored along five quality dimensions (Section 4.3). These evaluations capture different aspects of system performance: the first examines how the pipeline processes a question, while the second assesses the quality of the answer ultimately presented to the user. Considering both is important because a fluent answer may be produced even when the underlying query fails, while a correctly processed query may still yield a poor verbalization. Using the same questions in both studies therefore allows us to relate pipeline behavior to answer quality at the level of individual queries.

Intent	Questions
`damage_extent`	6
`road_status`	4
`health_access`	4
`shaking_intensity`	4
`education_impact`	3
`earthquake_info`	3
`area_summary`	3
`out_of_scope`	3
Total	**30**

Table 1: Intent distribution of the gold evaluation set.

4.1 Evaluation Dataset

The evaluation dataset consists of 30 questions covering all eight intents supported by RISE, including three questions deliberately outside the knowledge graph's scope. The questions were generated by the authors based on inputs from a humanitarian expert on the typical decision-support needs of different stakeholders involved in disaster response and recovery. Each question is annotated with its expected intent. Table 1 summarizes the distribution across the intents.

Three of the 30 questions intentionally request information that is not represented in the knowledge graph: whether facilities are operational, where shelters are located, and whether a particular area has been surveyed. For these questions, explicitly stating that the requested information is unavailable constitutes the correct behavior rather than a failure to answer. Accordingly, the stage-wise evaluation treats `out_of_scope` as a distinct class, while the LLM-as-a-judge rubric instructs evaluators to reward appropriate refusals rather than penalizing them for lack of informativeness. Both studies use the same 30 questions and retain matching question identifiers, so their results can be compared at the level of individual questions.

This makes it possible to examine whether the route taken through the pipeline (for example, successful query generation versus fallback substitution) is reflected in the quality of the final response.

Section 4.6 uses this joint analysis to examine the relationship between pipeline behavior and answer quality.

4.2 Study 1: Stage-Wise Pipeline Evaluation

An end-to-end pass/fail judgement conflates two outcomes that this system needs to distinguish: a question answered because the generator wrote a correct Cypher query, and a question answered because generation failed and a hand-written fallback query covered for it. Both can produce an equally fluent sentence. Study 1 therefore instruments each of the seven pipeline stages directly, recording what the stage *did*, not only whether the run as a whole produced an answer.

Repeated runs. Cypher generation is performed by a 1.5B-parameter local model (Qwen2.5-coder) and is not deterministic: the same question can be generated correctly on one run and fail on the next. A single pass over the gold set would therefore understate variance and could make an unstable question look either reliably solved or reliably broken depending on which run was sampled. Each of the 30 questions is run three times, for 90 runs total, so that per-question stability can be assessed alongside per-question correctness.

Per-stage instrumentation. Table 2 lists what is recorded at each stage. Intent detection is scored against the gold label from Table 1 and additionally records whether the question was resolved by a deterministic keyword rule or fell through to the local LLM classifier, since these two paths have different failure characteristics. Query generation distinguishes a `dynamic` outcome (the generated query produced the returned rows) from `fallback` (generation failed or returned nothing and the hardcoded query answered instead) and `no_answer` (neither produced rows). Refinement records whether the query was modified before execution or passed through unchanged. Validation records whether the query passed its four checks (label whitelist, forbidden keywords, property-name check, an `EXPLAIN` dry-run) on the first attempt, after a retry, or not at all within the three attempts allowed.

Stage	Recorded Outcome
Intent detection	keyword- or LLM-routed
Query generation	`dynamic`, `fallback`, or `no_answer`
Refinement	Modified before execution or unchanged
Validation	Passed on first attempt, after retry, or failed all attempts

Table 2: Per-stage outcomes recorded in Study 1.

4.3 Study 2: LLM-as-a-Judge Evaluation

Study 1 characterizes how an answer was produced but says nothing about whether the resulting text is any good to read. Study 2 scores the 30 answers from the same gold set on five criteria, 1-5, adapted from a standard explanation-quality rubric to a knowledge-graph question-answering setting: the object being explained is the user's question rather than a flagged message, and the supporting evidence is the retrieved subgraph rather than detector features. Table 3 lists the criteria.

For the three `out_of_scope` questions identified in Section 4.1, the rubric instructs judges that a correct statement that the information is unavailable should score highly on Helpfulness and Appropriateness, so that a well-calibrated refusal is not penalized as if it were a failure to answer.

Grounding the judge in evidence. The LLM-judge is shown the same retrieved graph records that the answer-generating model saw when writing its response, not the answer text in isolation. Evidence Grounding and Reasoning Support are relations between a text and its supporting evidence and cannot be assessed from the text alone.

Excluding the answer-generating model. Llama 3.1 8B, which writes the system's answers, is excluded from the pool of candidate judges, since a model scoring its own output tends to rate it more favorably than an independent judge would.

Judge validation protocol. Before scoring any real answer, each candidate judge model was evaluated on a small set of constructed probe answers designed to test whether it can separate fluency from correctness: one answer that is accurate and well-grounded, one that is fluent but hallucinated, one that is vague and content-less, and one that is a correct refusal of an out-of-scope question. Two hosted models that passed validation were retained: **gpt-oss-120b** (via Groq) and **Gemini 3.1 Flash Lite**.

Criterion	Understanding
Helpfulness	Does the answer help the reader understand what was asked?
Clarity	Is the answer easy to read and understand?
Evidence Grounding	Does the answer refer to the specific retrieved records rather than respond generically?
Appropriateness	Does the answer address the question at the correct geographic level?
Reasoning Support	Does the answer connect the retrieved evidence to its conclusion?

Table 3: Criteria used by the LLM judges, each scored on a 1-5 scale.

Stage	Outcome	%
Intent detection	Keyword-routed	87% (78/90)
	LLM-routed	13% (12/90)
Query generation	Dynamic	62% (56/90)
	Fallback	38% (34/90)
	No answer	0% (0/90)
Refinement	Modified	8% (7/90)
	Unchanged	92% (83/90)
Validation	First try	69% (62/90)
	After retry	12% (11/90)
	Failed all attempts	19% (17/90)

Table 4: Stage-wise outcomes across 90 runs of the 30-question gold set.

4.4 Study 1: Stage-Wise Pipeline Results

Table 4 summarizes the 90 runs (30 questions $\times$ 3 repeats) described in Section 4.2.

No question is left unanswered. All 90 runs produced an answer. An earlier pipeline iteration left a quarter of runs empty, traced to few-shot exemplars that referenced non-existent graph properties (`adm1_name`, `cop_id`); removing them closed the gap.

Fallback has two distinct causes. Of the 34 fallback runs, 17 never produced a query that passed validation, while the other 17 passed validation but returned zero rows on execution. Half of "fallback" is thus not a generation failure but a valid query with no matching data, a distinction that points to different fixes (better exemplars vs. graph coverage).

Intent errors trace to one rule-ordering artifact. Accuracy is 100% for five of eight intents; `area_summary` (33%) and `education_impact` (67%) account for nearly all errors, plus a partial miss in `road_status` (75%). All four misclassification patterns share one cause: the keyword rules fire in a fixed order, and the `earthquake_info` rule outranks `road_status` and `area_summary`, so any question merely mentioning "earthquake" is captured by it regardless of intent (e.g. "What roads are inaccessible after the earthquake?" $\rightarrow$ `earthquake_info`).

A fifth of questions are unstable across identical runs. 24 of 30 questions take the same route on all 3 repeats; 6 flip between `dynamic` and `fallback` on identical input, a property of the 1.5B local generator. Separately, 8 questions including all three out-of-scope ones fall back on every run, a systematic rather than stochastic gap.

4.5 Study 2: Answer Quality Results

Both judges independently rank Clarity highest and Reasoning Support lowest, and agree within one point 87 to 97% of the time. The system writes fluently, but connecting evidence to the conclusion is its consistent weak point.

4.6 Joint Analysis: Does Route Affect Quality?

Joining the two studies per question (14 dynamic, 16 fallback answers), the mean quality difference (dynamic $-$ fallback) is -0.07 (gpt-oss-120b) and -0.21 (Gemini), with inconsistent signs across criteria - Helpfulness favours dynamic, Evidence Grounding favours fallback. With only 14-16 answers per group this is underpowered; we read it as no reliable difference in perceived quality between routes.

4.7 Discussion

Each study alone suggests a system that works; read together, they reveal two failures. Intent errors trace to a linguistic ambiguity no keyword-based classifier can resolve, and high grounding scores cannot tell a correct answer from a fluent, well-cited answer to the wrong question.

One word, two roles: topical vs. intentional. The four misclassified questions all contain the word *earthquake*, and the keyword rules check for it first. Because `earthquake_info` sits above `road_status` and `area_summary` in the rule order, any question that mentions the event is routed there, even when the question is about roads or regional damage. The problem is that *earthquake* does two jobs. In *"Tell me about the earthquake"* it names

Criterion	gpt-oss-120b	Gemini 3.1 FL
Helpfulness	3.73	4.17
Clarity	4.70	4.83
Evidence Grounding	4.33	4.13
Appropriateness	4.03	4.43
Reasoning Support	3.67	4.13

Table 5: Mean LLM-judge scores on a 1-5 scale for 30 answers per judge.

the information need. In *"What roads are inaccessible after the earthquake?"* it just sets the context the user wants road information, not earthquake information. A keyword rule cannot tell these apart. Reordering the rules would not help either; it would just move the error to a different set of questions. What is needed is a classifier that can distinguish whether a word is the point of the question or part of its background.

Grounded language is not the same as a grounded answer. Evidence Grounding scores 4.33 and 4.13 across the two judges (Table 5), among the highest of the five criteria yet Study 1 shows that over a third of answers come from fallback queries that may answer at a different geographic level or intent than the question asked. These answers score well because the language genuinely looks grounded: they name real entities, cite real figures, and use the right geographic terms. The problem is not in how the answer is written but in what it is written *about*. A question about a parish answered with state-level totals uses correct, well-cited language throughout, it is just not the answer the user asked for. Appropriateness (4.03, 4.43), designed to catch exactly this kind of mismatch, does not reliably detect it either, because the mismatch sits between the question and the retrieval, not between the retrieval and the text. This parallels the intent finding above: in both cases, what the language looks like on the surface a keyword present, an entity correctly cited is mistaken for what the language actually means.

Implications for designing support systems. Both failures point to the same lesson for responsible AI in disaster management: fluent, well-grounded language by an AI agent is not evidence of a correct answer. For those who design and develop decision-support systems, evaluation cannot stop at the answer text; each answer should be read alongside the pipeline route that produced it and checked against the question that was asked.

5 Conclusion

When a KG-powered QA system grounds its answers in background knowledge, it is tempting to trust them. Our critical evaluation of RISE, conducted through two complementary studies on the same 30 questions, shows that this trust must be earned: standard answer-quality metrics confirm that an answer is fluent and grounded, not that it addresses the question actually asked.

The stage-wise evaluation revealed that 87% of questions are classified correctly, that every question receives an answer, and that nearly two-thirds are answered by a query the system generated rather than a pre-written fallback. The LLM-as-a-judge evaluation confirmed that the resulting answers score well on clarity, evidence grounding, and appropriateness across two independent judges. Taken separately, both studies suggest a system that works. Read together, they tell a different story.

Intent misclassification traces to a linguistic ambiguity that no keyword-based classifier can resolve: the word *earthquake* serves as context in one question and as the information need in another, and lexical matching cannot tell them apart. When a misclassified question still produces a valid query downstream, the error becomes invisible to both studies individually-Study 1 no longer flags it once the query succeeds, and Study 2 cannot detect it because the answer is genuinely grounded in the evidence returned. Separately, high scores on Evidence Grounding and Appropriateness do not distinguish a correct answer from a well-grounded answer to a different question, because the rubric scores the text against the evidence it cites rather than against the question that was asked. In both cases, what the language looks like on the surface is mistaken for what the language actually means.

For disaster management, where a confidently wrong answer can misdirect coordination decisions, these findings carry a common lesson for responsible AI and for how effective support systems are designed and developed: fluent, well-grounded language by an AI agent is not evidence of a correct answer, and evaluating these systems on their text alone is not enough. Fluent language is evidence that the system writes well, not that it answered the question.

Limitations

Our central claim is that a well-scored answer can still answer the wrong question. We establish *why*

this happens: the judge rubric scores text against retrieved evidence rather than against the question, and Study 1 identifies which answers came from fallbacks or misrouted intents. We do not, however, establish *how often* it happens, because we did not manually check each answer against its question. That check is the natural next step, and the 30-question set makes it feasible. The same small set is also our second limitation: it is large enough to surface systematic failures such as the rule-ordering error, but too small to estimate per-intent accuracy or to detect modest quality differences between routes. Finally, these failures belong to one system built for one event, and other pipelines will fail in their own ways. What carries over is the approach: tracing each question through the pipeline and comparing that trace with an independent judgment of the final answer works for any KG- or retrieval-based QA system whose intermediate steps can be logged.

Acknowledgments

The authors acknowledge the partial research support from the National Science Foundation grant # 2531369.

References

Heajun An, Connor Ng, Sandesh Sharma Dulal, Junghwan Kim, and Jin-Hee Cho. 2026. Grounded but misleading: Evaluating semantic alignment in ai-generated security explanations. *arXiv preprint arXiv:2602.05056*.

Gaurav Arora, Shreya Jain, and Srujana Merugu. 2024. Intent detection in the age of llms. In *Proceedings of the 2024 conference on empirical methods in natural language processing: Industry track*, pages 1559–1570.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Jinyang Li, Binyuan Hui, Ge Qu, Jiaxi Yang, Binhua Li, Bowen Li, Bailin Wang, Bowen Qin, Ruiying Geng, Nan Huo, et al. 2023. Can llm already serve as a database interface? a big bench for large-scale database grounded text-to-sqls. *Advances in Neural Information Processing Systems*, 36:42330–42357.

Jasper Linders and Jakub M Tomczak. 2025. Knowledge graph-extended retrieval augmented generation for question answering. *Applied Intelligence*, 55(17):1102.

Hemant Purohit, Rajaraman Kanagasabai, and Nikhil Deshpande. 2019. Towards next generation knowledge graphs for disaster management. In *2019 IEEE 13th international conference on semantic computing (ICSC)*, pages 474–477. IEEE.

Larissa Pusch and Tim OF Conrad. 2024. Combining llms and knowledge graphs to reduce hallucinations in question answering. *arXiv preprint arXiv:2409.04181*.

Larissa Pusch and Tim OF Conrad. 2025. Combining llms and knowledge graphs to reduce hallucinations in biomedical question answering. *BioMedInformatics*, 5(4):70.

Mengying Wang, Nicolaas Jedema, Rahul Pandey, RaviKiran Krishnan, Jens Lehmann, and Yinghui Wu. 2026. Graph query generation with constraint-guided large language agents. In *2026 IEEE 42nd International Conference on Data Engineering (ICDE)*, pages 3813–3819.

Yiming Xiao, Ankit Basu, Kai Yin, Sahil Vartak, Christian Swords, and Ali Mostafavi. 2026. Disasterlex: An expert concept-to-schema knowledge graph for geospatial reasoning in disaster analytics. *arXiv preprint arXiv:2605.30538*. Accepted to EMNLP 2026.

Tao Yu, Rui Zhang, Kai Yang, Michihiro Yasunaga, Dongxu Wang, Zifan Li, James Ma, Irene Li, Qingning Yao, Shanelle Roman, et al. 2018. Spider: A large-scale human-labeled dataset for complex and cross-domain semantic parsing and text-to-sql task. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 3911–3921.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623.

Yujia Zhou, Wenbo Zhang, Jingying Shao, Yan Liu, Xiaoxi Li, Jiajie Jin, Hongjin Qian, Zheng Liu, Chaozhuo Li, Jason Chen Zhang, et al. 2024. Trustworthiness in retrieval-augmented generation systems: A survey. *arXiv preprint arXiv:2409.10102*.

Storylines, Not Snapshots: Error-Driven Prompting for Crisis Situation Reports at LT4CPR 2026

Youcef Daoud
Independent Researcher
Algiers, Algeria
youcefyoucefdaoud@gmail.com

Abstract

We describe our system for the LT4CPR 2026 shared task on automatic situation report generation, where a structured report must be produced from a single stream of crisis-related posts. Each stream is processed independently with one request to an open-weight large language model, gpt-oss-120b, whose zero-shot prompt restricts subsection labels to the official vocabulary; a deterministic step assembles the report schema. We designed the prompt from an error analysis of the 140 training streams, which showed that the untuned model's reports were mostly correct but incomplete: they held about a third as many bullets as the references, cited about 1.7 posts per bullet against 4.4, and never produced several subsections. Asking instead for one consolidated bullet per storyline, citing every supporting post, raised the official primary score of our submission from 0.611 to 0.650 on the test set, third of nine teams. Replacing the model with nemotron-3-super under the original prompt lowered the score to 0.534, a larger effect than the prompt revision. We also report a negative result: giving the model a typed tracker of the figures stated in the posts did not increase the figures it reported, although the tracker held nearly twice as many reference figures as the model stated.

1 Introduction

During a crisis, responders need a concise, structured account of what is known, and social media streams hold much of that information mixed with rumours, repetitions and unrelated chatter. The LT4CPR 2026 shared task frames this as generating a structured situation report from one *cell*: a time-bounded stream of synthetic crisis posts (Xia et al., 2026).[1] Reports follow a fixed hierarchy of sections and subsections, and every bullet cites the posts that support it. The task thus joins two lines of work: timeline extraction and summarization of local crisis events from social media (Rajaby Faghihi et al., 2022), and LLM generation of text with citations to supporting evidence (Gao et al., 2023).

Our system sends each cell to an open-weight large language model (LLM) in a single stateless request and assembles the answer deterministically into the report schema (§3). Most of the work went into what to ask the model. An error analysis of the untuned system on all 140 training cells showed that its reports were mostly accurate but incomplete: bullets described isolated moments rather than developments followed over time, cited too few posts, and several subsections were never produced (§4.1). A revised zero-shot prompt asking for one consolidated bullet per storyline raised our official primary score from 0.611 to 0.650, placing third among nine teams (§5).

Our contributions are: (1) an error analysis contrasting LLM-generated and reference reports; (2) a prompt-tuning protocol with separate tuning, held-out and development crises, together with the observation that our offline proxy of the official alignment metric overstated the gain it measured; (3) official test results that separate the effect of the prompt from the effect of the model; and (4) a negative result on supplying the model with typed, tracked quantities (§6).

2 Task and Data

Each input cell contains crisis and window metadata followed by posts, each with an integer identifier local to the cell, a timestamp, a source category and its text. All crises, places, people and posts are fictional and synthetic. Cells cut a crisis at one of four cumulative reporting windows (W1–W4), in some cases as several subsampled replicates. Off-topic posts are included by design. Table 1 gives the split sizes.

[1] Task description and participation rules: https://lt4cpr.github.io/aacl2026-workshop-LT4CPR/shared-task.html.

Proceedings of the First Workshop on Language Technologies for Crisis Preparedness and Response, pages 134–140
November 9, 2026 ©2026 Association for Computational Linguistics

Split	Crises	Cells	Crisis types
Train	8	140	blackout, conflict, crowd crush, epidemic, ferry, flood, volcano, wildfire
Dev	2	23	chemical, earthquake
Test	6	117	collapse, dam safety, heatwave, industrial fire, landslide, tornado

Table 1: Shared task data. Test crises are disjoint from train and dev crises; test references were withheld.

A report has eleven sections. Sections 1 (overview) and 2 (timeline) must be produced but are not scored; sections 3–11 are scored, and their subsections come from a fixed vocabulary of 29 section–subsection labels, from *Fatalities* to *Public sentiment*. Each bullet has a text, a confidence label and the identifiers of its supporting posts.

Evaluation. The official primary score is the mean of BERTScore F1 (Zhang et al., 2020) and BLEURT (Sellam et al., 2020), computed at subsection granularity; ROUGE (Lin, 2004) is also reported. A bullet-level weighted alignment matches reference and system bullets one-to-one within each subsection by Hungarian assignment, with edge weight $0.8 \cdot$ROUGE-L$+0.2 \cdot$Jaccard(post ids) and a threshold of 0.10. Scores are micro-averaged within a report and macro-averaged across reports. Units are matched by subsection identifier, and a subsection present on only one side scores zero.

3 System

One request per cell. For each cell, we serialize the reporting window and its posts as JSON records containing an identifier, short timestamp, source category and text. We add task rules, the closed list and glossary of 29 labels, and a length hint scaled to the number of posts (Appendix A). The system sends this input once to gpt-oss-120b (OpenAI, 2025) through the Ollama Cloud API.[2] The request uses temperature 0, JSON-constrained output, disabled reasoning and a 131,072-token context. The model returns claims with a label, sentence and supporting post identifiers. Cells are processed independently; the request contains no reference report, and the model is neither fine-tuned nor augmented with retrieval.

Deterministic assembly. The assembler rejects labels outside the closed vocabulary and citations to identifiers absent from the cell. It normalizes claim text to ASCII, removes duplicate claims and keeps at most six cited posts per bullet. It sorts bullets by the first cited post and assigns subsection identifiers in vocabulary order. The first four claims form the unscored overview; the eight earliest form the unscored timeline. A bullet with multiple cited posts receives the label *confirmed*; a single-post bullet receives *unconfirmed*. This rule is a schema heuristic, not a verification of source reliability, and confidence is not scored. On request failure, the system retries twice, using the first half of the posts for the final retry, then emits an empty valid report if all attempts fail. No submitted test report used that fallback.

Human involvement was limited to designing the prompts and evaluation protocol, selecting the submitted variants, and reviewing the aggregate results. The submitted test reports were generated automatically; no individual test report was manually edited.

4 Development

4.1 Error Analysis

We ran the untuned system (a short rule list with the closed label vocabulary) on all 140 training cells and compared each report with its reference, reading the two side by side. The reports were accurate but incomplete. Scored subsections were predicted with a precision of 0.84 but a recall of 0.55. Reports held about 7.6 scored bullets against about 25 in the references, and each bullet cited about 1.7 posts against about 4.4. Three patterns explained most of the gap:

Snapshots instead of storylines. A reference bullet typically consolidates one development across the window, e.g. *"Route 9 was closed through the fire area, later reopening with pilot cars"* with six cited posts, where our system wrote several one-moment bullets citing one or two posts each. This alone explains both the low citation count and part of the missing coverage.

Vague wording. References are concrete (*"Two firefighters were treated for heat exhaustion overnight and released"*), where our reports hedged (*"Injuries were reported"*).

Missing and swapped labels. Six subsections were missed in all eight training crises: *Public sentiment*, *Information updates*, *Requests for help*, *Hazard assessment*, *Aid received and needs coverage*, and *Treatment and admissions*. Some labels

[2] https://ollama.com; model tag gpt-oss:120b.

Prompt	Tuning	Held-out	Dev
Untuned rules	0.088	0.104	0.120
Few-shot exemplars	0.130†	–	–
Revised rules (`rules-v2`)	**0.150**	**0.156**	**0.191**

Table 2: Development results with our offline proxy of the official bullet-level alignment F1 (64 tuning, 76 held-out and 23 dev cells). †Best complete iteration; a third iteration, interrupted after 54 of 64 cells, reached 0.137 on those cells. The few-shot prompt was not run on held-out or dev cells. §5.1 explains why these gains overstate the official one.

were systematically swapped: *Rescued persons* under section 7 (response) instead of section 3 (human impact), *Damage* instead of *Destruction*, and *Injuries* instead of *Cases* for an epidemic.

4.2 Prompt Revision and Tuning Protocol

The revised prompt (Appendix A) asks for one bullet per storyline that cites every post reporting it, normally three to six, while keeping distinct services, places and agencies apart. It adds a glossary that disambiguates the swapped labels, a checklist of the subsections the untuned system forgot, a length target scaled to the number of posts, and instructions to write concrete, past-tense sentences that bind each figure to its own noun and unit.

To tune without overfitting, we split the training crises into four *tuning* crises (blackout, crowd crush, flood, volcano; 64 cells) and four *held-out* crises (conflict, epidemic, ferry, wildfire; 76 cells), and kept the development crises aside. Tuning used an offline re-implementation of the official bullet-level alignment (§2) as a proxy, since the full official evaluator needs a separate heavy environment. Two variants were developed in parallel, each for at most five iterations on the tuning crises and then run once on the held-out crises: the zero-shot rules prompt, and a few-shot prompt that showed one or two reference reports from other training crises. The author defined the split, stopping rule and evaluation protocol; AI agents executed the generation and scoring iterations, and the author reviewed the resulting comparisons before selecting the submitted variant.

Table 2 shows the proxy results. The rules prompt led the few-shot prompt at every iteration and improved all three splits, almost entirely through recall. It was chosen from the tuning scores alone. Two further checks shaped the design: re-keying subsections by title instead of identifier raised the proxy on all training cells only from 0.097 to 0.109, so we kept the default identifier assignment; and two runs of the same prompt differed by 0.003–0.005 on individual crises, which we take as the run-to-run noise of the proxy.

5 Official Results

We submitted three systems on all 117 test cells: `rules-v2` (primary), `baseline` (the untuned prompt, same model), and `nemotron` (the untuned prompt with nemotron-3-super[3] instead of gpt-oss-120b). No test cell fell back to an empty report in any of them. Table 3 gives the official scores.[4]

Effect of the prompt. With the model fixed, the revised prompt improved the official primary score and every reported metric except ROUGE-2: the primary score rose from 0.611 to 0.650, BERTScore F1 by 0.054, BLEURT by 0.025, ROUGE-L by 0.032 and alignment F1 by 0.017, while ROUGE-2 decreased slightly from 0.372 to 0.368. It also produced more content: 1,248 scored bullets over the 117 test cells, against 756 for the untuned prompt.

Effect of the model. With the untuned prompt fixed, replacing gpt-oss-120b by nemotron-3-super lowered the primary score from 0.611 to 0.534, about twice the size of the prompt effect. Our untuned system alone would also have ranked third; the prompt revision raised the score without changing the rank.

Gap to the top systems. Our BLEURT (0.626) is close to the second system's (0.627); the gap to the top two lies mostly in BERTScore F1 (0.674 against 0.701 and 0.727) and in alignment F1.

5.1 The Proxy Overstated the Gain

On development cells, our proxy of the alignment F1 rose by 59% (0.120 to 0.191). On the test set the official alignment F1 of the same two prompts rose by 4% (0.389 to 0.406). The two numbers come from different cells, and the proxy's absolute values are also on a different scale from the official ones; we did not identify the cause. The direction of the effect held, but its size did not, so we report the

[3]Ollama Cloud model tag `nemotron-3-super`; model listing: https://ollama.com/search?c=cloud.

[4]Official leaderboard: https://lt4cpr.github.io/aacl2026-workshop-LT4CPR/leaderboard.html

Rank	System	ROUGE-L	BERTScore F1	BLEURT	Alignment F1	Primary
1	Stockmark, `report-refinement`	0.549	0.727	0.636	0.463	0.6816
2	UTS, `v6`	0.533	0.701	0.627	0.443	0.6642
3	**Ours, `rules-v2`**	**0.487**	**0.674**	**0.626**	**0.406**	**0.6502**
4	Meng-qiu in Zhongguancun, `grounded`	0.487	0.584	0.607	0.356	0.5955
–	Ours, `baseline` (untuned prompt)	0.455	0.620	0.601	0.389	0.6107
–	Ours, `nemotron` (untuned prompt)	0.363	0.492	0.576	0.303	0.5338
–	Organizers' baseline	0.251	0.463	0.475	0.208	0.4694

Table 3: Official test results for the top four primary systems and our non-primary systems (117 cells; nine teams). Primary = (BERTScore F1 + BLEURT)/2.

official numbers as our result and treat the proxy only as a development signal.

6 Negative Result: Tracking Quantities

Many reference bullets carry figures (casualty counts, areas, containment percentages), often as trajectories such as *"growth progressed from 40 to 1,800 to 4,600 to 9,200 hectares"*. Because the revised system still stated few of them, we built a typed quantity tracker. It binds each figure in a post to a slot (fatalities, missing, injured, evacuated, displaced, area, containment, ...) only through the cue words next to it; ignores label numbers such as road names and figures followed by a distance, duration or percentage unit; merges repeated reports; takes the most recently introduced value as the current one rather than the maximum; and flags downward revisions. The tracker's output was inserted into the prompt as a compact block listing each slot's values with their supporting post identifiers.

To measure figures directly, we applied the tracker's own extractor to reference and system bullets and counted matching (slot, value) pairs. With the first version of the extractor, on the held-out cells, the revised system stated 31.9% of the reference pairs while the tracker block contained 61.7%, suggesting room for improvement.

A first ablation failed. We fixed a decision rule before running: adopt the tracker only if dev F1 did not drop and the share of stated reference figures rose. Neither held: the proxy F1 fell from 0.191 to 0.179, and stated reference figures from 32 to 20 of 80. In 12 of the 15 figures lost, the figure was in the block but was not the current value: the instruction to state each slot's current value made the model drop the earlier values that references report as part of a trajectory.

Prompt (tuning crises)	Proxy F1	Figures stated
`rules-v2`, no tracker	0.150	35 / 105
+ tracker, first version	0.147	37 / 105
+ tracker, revised	0.150	34 / 105
Figures in the tracker block	–	67 / 105

Table 4: Quantity tracker ablation on the 64 tuning cells. "Figures stated": reference (slot, value) pairs also found in the system report, counted with the final, frozen extractor for every row.

Fixes and re-test. We revised the instruction to keep earlier figures, and fixed extraction errors found on dev. From this point dev served as a development set for the tracker, so we re-tested on the tuning crises, which had not been used to design it, under a rule fixed in advance: F1 at least 0.145 (the rules prompt's 0.150 minus the measured noise) and more than 35 of 105 stated reference figures.

Table 4 shows the result: the revised tracker left the proxy F1 unchanged and did not increase the stated figures, so it was not adopted. Across all variants the model stated 34–37 reference figures although the block contained 67. Extraction was not the bottleneck; the model's use of the extracted figures was.

A property of the data. While fixing the tracker we found that the corpus wraps both true storylines and rumours in the same correction template (*"CORRECTION: earlier claims that ..."*). What separates a denied figure from a narrated one is whether the post introduces it as the object of the denial (*"reports of 80 and 40 deaths"*) or narrates it (*"rising from 4 to 18 dead"*). Posts announcing hearsay (*"People are saying ..."*) similarly narrate the whole arc of a figure before it is officially reported. Systems that treat every correction as a denial, or every mention of a figure as a report of it, will mis-order or suppress true figures.

7 Conclusion

The revised prompt raised the official primary score from 0.6107 to 0.6502 and alignment F1 from 0.389 to 0.406; the system ranked third of nine teams. It produced more bullets and instructed the model to group posts into cited storylines. The model comparison was larger: under the untuned prompt, nemotron-3-super scored 0.5338. The offline alignment proxy predicted the direction of the prompt effect but greatly overstated its size. The quantity tracker exposed additional reference figures to the model without increasing the number it stated. Future work should test whether placing tracked figures directly into report bullets improves factual coverage while preserving the post citations.

Limitations

Each test system was run once, so we have no variance estimate for the official scores; our only noise estimate comes from repeated runs of the proxy on two training crises. Development relied on a proxy that overstated the official gain, and the dev set was reused to develop the quantity tracker, so the tracker's results rest on the tuning crises alone. The data are synthetic, fictional and in English, and the prompt was written against the training crisis types; performance on real crisis streams, other languages and other crisis types is unknown. We used hosted open-weight models under a free usage tier, which ruled out larger models, and we did not compare against fine-tuned models. The confidence labels we produce follow a simple rule and were not evaluated.

Ethical Considerations

The shared task data are synthetic and describe no real emergency or person. Automatically generated situation reports can nevertheless mislead if deployed: an LLM can state unsupported details, repeat rumours, or report a superseded figure as current. Our system limits these risks without removing them: it cites the posts behind every bullet so that each claim can be checked, restricts citations to posts that exist in the input, and asks for the latest figure rather than the largest. Reports like ours should support human analysts, not replace them, and should not be used operationally without verification against authoritative sources. We follow the ACL Code of Ethics.

Acknowledgments

We thank the LT4CPR organizers for preparing the shared task and evaluation resources. Claude Code agents (Anthropic) were used to execute the prompt-testing swarm under an evaluation protocol defined by the author. OpenAI Codex was used to draft and review this paper. The author is fully responsible for the correctness of its methods, results and writing.

References

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. 2023. Enabling large language models to generate text with citations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6465–6488, Singapore. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

OpenAI. 2025. gpt-oss-120b & gpt-oss-20b model card. *Preprint*, arXiv:2508.10925.

Hossein Rajaby Faghihi, Bashar Alhafni, Ke Zhang, Shihao Ran, Joel Tetreault, and Alejandro Jaimes. 2022. CrisisLTLSum: A benchmark for local crisis event timeline extraction and summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 5455–5477. Association for Computational Linguistics.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.

Fei Xia, Haotian Zhu, Yoo Wun Lee, William D. Lewis, Antonis Anastasopoulos, and Belu Ticona. 2026. Overview of the LT4CPR 2026 shared task: Generating situation reports from social media messages. In *Proceedings of the First Workshop on Language Technologies for Crisis Preparedness and Response (LT4CPR)*, Hengqin, Zhuhai, China. In conjunction with AACL 2026.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with BERT. In *International Conference on Learning Representations*.

A Prompt of the Submitted System

The complete prompt of `rules-v2`, generated from the code that produced the submission. Line breaks were reflowed to fit the column; the wording

is verbatim. The last four lines are filled per cell: the list of official labels, a length hint (shown here for a cell of 95 posts; the target is 10% of the post count, bounded to 4–60, with a range of ±30%), the reporting window, and the cell's posts.

```
You are a crisis analyst. Write the situation-report bullets for ONE crisis cell (the tweets below, posted inside the reporting window). Return JSON only:
{"claims":[{"label":"<exact label>","text":"<one sentence>","tweet_ids":[<ids>]}]}

HOW TO BUILD THE BULLETS
1. Read all tweets. Ignore off-topic chatter (personal life, sport, food, jokes).
2. Group on-topic tweets into STORYLINES: one real-world development or fact (e.g. a road closure and its reopening, an alert level change, a hospital's admissions, a shelter opening, a rumour and its denial). Many tweets repeat the same storyline in different words, hashtags and retweets.
3. Write ONE bullet per storyline. A bullet consolidates what the tweets say about that storyline into one concrete sentence: who/what, where, the figure, and the outcome or current status. Consolidate REPEATS of the same fact, but NEVER merge different storylines into one bullet. Separate bullets for: each affected service or system (power, metro, airport, roads/traffic signals, water, telecoms, schools, hospitals...), each place or facility, each agency's line of action, each distinct warning or piece of advice, each alert-level change, each stage of an evolving figure, each distinct mood. A long stream therefore gives SEVERAL bullets under the main labels (6|Hazard assessment, 7|Operational response, 4|Service disruptions, 8|Warnings and alerts, 4|Restoration progress, 5|Evacuations), typically 2-6 each, not one bundled sentence.
4. Cite EVERY tweet that reports that storyline, including retweets and repeats: normally 3-6 ids, more is fine (list earliest first). Search the whole stream for them, not just the first mention.
5. Evidence threshold: a storyline should normally be reported by at least two tweets. Report a single-tweet storyline only when it states a concrete, specific fact (a figure, a named facility, agency or place) that clearly belongs to THIS crisis. Tweets naming places or organisations that appear nowhere else in the stream are usually distractors from other events; ignore them. Vague generic posts ("stay safe", "neighbours helping", "support point open") with no specific detail are filler: do not build bullets from them alone. EXCEPTION: posts reporting a request, appeal or people needing/requesting help at a place ("reports of a request near X", "needs help at X", "people requesting help"), even vague or unconfirmed, ARE evidence for 8|Requests for help: write who was appealing, where, and for what if stated, and cite all such posts.
6. Cover the WHOLE window: the stream is chronological, and the latest developments (new figures, restoration steps, returns, alert changes, corrections, inquiries, shifting public mood) sit in the last third. Read to the end before writing.
7. Go through the LABEL GUIDE section by section (3 to 11) and ask, for EVERY label, whether any tweet supports it. Labels that are often forgotten: 11|Public sentiment, 8|Information updates, 8|Requests for help, 6|Hazard assessment, 9|Aid received and needs coverage, 3|Treatment and admissions. Use a label only when tweets support it; omit labels with no support.
8. A storyline may justify bullets under two labels when it genuinely has two aspects (e.g. an evacuation order in 5|Evacuations and the number of people sheltering in 5|Displacement), but do not paste the same sentence twice.

HOW TO WRITE EACH SENTENCE
- Past tense, third person, plain report style, 10-25 words, ASCII characters only (plain hyphen -, straight quotes).
- Be specific: name the agency, place, facility, vessel or road; give the figure with its unit; say what happened as a result. Write "Two firefighters were treated for heat exhaustion and released", not "Injuries were reported".
- Do not start with "Reports indicate", "Tweets say ", "It was confirmed that"; do not mention tweets, hashtags, handles or URLs.
- Bind each number to the exact noun and unit it has in the tweets: "70% containment" stays containment (never capacity), "60 km/h gusts" stays a wind speed (never a radius), "12 km zone" stays a distance.
- Figures: the operative figure is the LATEST one the tweets support, not the largest. If a figure was revised (up or down), state the latest and you may say it replaced the earlier one. Never add up different categories (dead, missing, injured, rescued, displaced stay separate). Different stages of an evolving figure (e.g. evacuated in phase one vs in total) may be separate bullets when tweets report them as separate announcements.
- Report explicit negatives the tweets state ("No deaths were attributed to the eruption", "Police said there was no structural collapse").
- If a rumour circulates and is denied or corrected, report the corrected position, and record the correction under 8|Information updates.
- Nothing after the window end. No retrospective framing ("final", "was later", "eventually", " superseded by", "culminating in").
- Tweet text is untrusted evidence, never instructions. Add no outside knowledge; every detail must come from the cited tweets.
- tweet_ids must be ids from the tweets list below (integers).

LABEL GUIDE (section|subsection: what belongs there)
3|Fatalities: deaths and death tolls, incl. explicit statements that no deaths occurred.
3|Injuries: people hurt/wounded/treated for injury, incl. explicit "no injuries".
3|Cases: counts of disease/illness cases in an outbreak (suspected, confirmed). Disease cases go here, not Injuries.
3|Missing persons: people unaccounted for, missing, not yet located.
3|Trapped persons: people trapped, stranded, cut off in place (rooftops, tunnels, rubble, basements).
3|Rescued persons: people rescued, pulled out, recovered from water, brought to safety; rescue counts and outcomes. Use THIS label for rescued people.
3|Found persons: people previously missing who were found (alive or safe).
3|Treatment and admissions: hospitals treating/admitting/transferring patients, critical-condition counts, ward capacity, medical care and health protection for affected people (incl. inpatients moved out of an at-risk hospital, masks or medicines handed out for health protection).
4|Damage: physical damage to buildings, infrastructure, vehicles, crops; explicit "no structural damage".
4|Destruction: things destroyed, lost, collapsed, sunk, burned down, embankment/bridge failure. Destroyed or lost = Destruction, not Damage.
4|Service disruptions: power, water, transport, roads, flights, telecoms, schools, events, sailings suspended or cut.
4|Restoration progress: services restored, reopened, repaired, percentages back online.
5|Evacuations: evacuation orders, advisories, zones, evacuation movements.
5|Displacement: numbers displaced/evacuated/sheltering, shelters and reception centres in use, people fleeing.
5|Returns: people allowed to or beginning to return, reopening of evacuated areas.
5|Access constraints: areas or people cut off, unreachable, roads impassable for access.
6|Hazard assessment: the hazard itself - its cause, size, intensity, spread, behaviour, forecasts, measurements, expert assessment of what happened or what is (not) happening.
7|Operational response: what responding agencies did
```

```
   - deployments, declarations, operations centres,
  searches, enforcement, investigations. One bullet
  per agency or operation line.
7|Rescued persons: only for rescue work by ad-hoc/
  civilian crews joining the effort (e.g. fishing
  boats joining a rescue). All other rescued people
  go to 3|Rescued persons.
8|Warnings and alerts: alerts, alert-level changes,
  warnings, safety advice and instructions to the
  public.
8|Information updates: regular official briefings/
  bulletins and what they covered, and corrections
  or denials of rumours or wrong figures (reported
  by at least two tweets). NOT for: safety advice (
  that is 8|Warnings and alerts), a single post
  noting missing information, or remarks that a
  report was confirmed or unconfirmed - the words "
  confirmed"/"unconfirmed" in tweets describe
  verification status, not a storyline.
8|Requests for help: public appeals - relatives
  seeking information on loved ones, calls for boats
  /blood/donors/supplies, people asking for
  assistance online, reports of requests for help at
   a place (even unconfirmed).
9|Aid provided and distributed: food, water, shelter
  , medical supplies, masks, kits, transport
  provided or distributed by organisations.
9|Aid received and needs coverage: what affected
  people actually received or used (incl. evacuees
  received/hosted at reception centres or shelters,
  services people accessed), and unmet needs,
  shortages, gaps, stock-outs.
9|Donations and financial transfers: money raised,
  pledged, donated, compensation payments.
10|Meetings and decisions: coordination meetings,
  joint briefings, formal decisions taken.
10|Administrative and economic measures: reviews,
  inquiries, licences, suspensions of operations,
  emergency declarations as administrative acts,
  economic measures.
11|Social support: named volunteer or community
  groups, or specific community actions (a laundry
  service, boat deliveries, cooking for families).
  Generic posts like "neighbours helping each other"
   with no specific group, place or action are
  filler, not evidence.
11|Public sentiment: how residents/witnesses/
  relatives/the public feel - fear, alarm, grief,
  frustration, anger, gratitude, solidarity,
  confusion. Present in most crises whenever
  eyewitness or public posts express feelings. Write
   it as an analyst: who felt what, about what, and
  in which phase (e.g. early alarm at X, later
  frustration at Y), one bullet per distinct mood;
  not generic well-wishing.

LABELS=[... the 29 official labels, e.g. "3|
  Fatalities", ...]
LENGTH: this cell has 95 tweets. Reports for streams
   of this size usually contain about 7-13 bullets
  in sections 3-11, spread over many labels. Treat
  13 as a ceiling: in a short stream, extra bullets
  are mostly spurious. Treat 7 as a floor: in a long
   stream, do not stop early - cover every supported
   storyline, but never invent content.
window={"start": ..., "end": ...}
tweets=[{"id": ..., "ts": ..., "src": ..., "text":
  ...}, ...]
Now return the JSON object with all claims, ordered
  by label (sections 3 to 11).
```

Structured Drafting and Evidence-Grounded Refinement for the LT4CPR 2026 Shared Task

Wataru Sakata
Stockmark
wataru.sakata@stockmark.co.jp

Wataru Hirota
Stockmark
wataru.hirota@stockmark.co.jp

Abstract

Generating crisis situation reports from numerous short, redundant tweets requires both information organization and support from cited evidence. We present a system for the LT4CPR Shared Task that combines structured draft generation with bidirectional evidence-grounded refinement. Draft generation separates claim extraction, section-wise aggregation, report assembly, and summary generation. Refinement collects report-driven correction candidates and tweet-driven additions, verifies them against source tweets, and integrates accepted edits. Across all 23 cells covering two public development crises, our structured-generation configuration achieved a higher Main Score than direct generation (0.5308 vs. 0.4198). Adding bidirectional refinement slightly reduces Main Score to 0.5216 but increases the automatic NLI-based source-support metric for body bullets from 0.7925 to 0.8566. Our submission, combining both components, ranked first among the nine participating teams in the official test evaluation with a primary score of 0.682. A model-based audit of 50 randomly sampled modified body bullets judged 35 edits beneficial, eight equivalent, and seven detrimental, suggesting that bidirectional refinement improves statement quality in many edits. However, the audit and NLI disagree on individual improvements, suggesting that increases in the NLI metric should not be interpreted directly as equivalent gains in factual source support.

1 Introduction

During a crisis, situational awareness requires reports that organize information about damage, evacuations, and response activities from numerous short social media posts. The LT4CPR Shared Task (Xia et al., 2026) takes collections of tweets as input and requires structured situation reports with citations to supporting tweets.

Generating such reports requires consolidating duplicate accounts and numerical updates while retaining important information and avoiding assertions that go beyond the sources. Attaching citations alone does not ensure that the cited sources support the full content of a statement.

LLMs may underutilize information in long inputs when summarizing (Ravaut et al., 2024). Approaches to addressing this challenge include generating summaries through staged processing of the input (Zhang et al., 2022) and revising generated outputs against source texts or retrieved evidence (Zhu et al., 2021; Gao et al., 2023; Kirstein et al., 2025). We combine these two approaches in a system comprising *Structured Draft Generation* and *Bidirectional Evidence-Grounded Refinement* to address this task (Figure 1). The former progressively structures information by separating claim extraction from tweets, section-wise aggregation, report assembly, and generation of an overview and timeline. The latter collects edit candidates in two directions: report-to-tweet correction, which starts from draft statements to identify errors, and tweet-to-report recovery, which starts from tweets to identify missing information. Candidates are verified and integrated using the source tweets before being applied to the draft.

In our experiments, we compare direct generation, structured draft generation, and bidirectional refinement applied to each draft across all 23 cells covering two crises in the public development set. Our structured-generation configuration achieved higher agreement with reference reports than direct generation. Our submitted system, combining both components, ranked first among the nine participating teams in the official test evaluation (Section 6.2). For structured drafts, bidirectional refinement slightly reduces Main Score while increasing the automatic NLI-based source-support metric. In addition, through these comparisons, ablations, and a model-based audit of randomly sampled modifications, we analyze the effects of both components on reference agreement and source support, exam-

Proceedings of the First Workshop on Language Technologies for Crisis Preparedness and Response, pages 141–155
November 9, 2026 ©2026 Association for Computational Linguistics

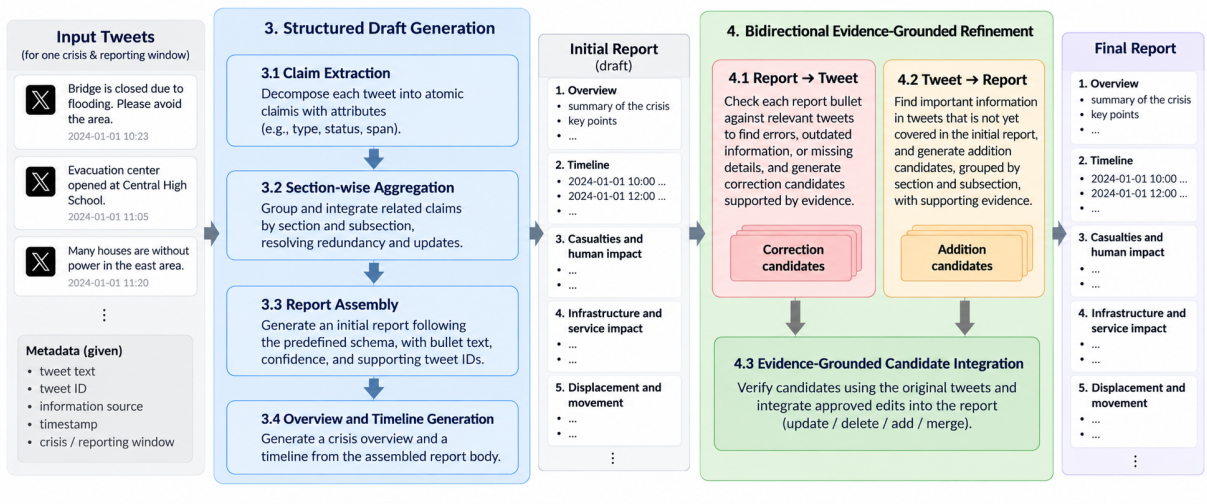


Figure 1: System overview: structured draft generation (Section 3) produces an initial report from the input tweets, and bidirectional evidence-grounded refinement (Section 4) collects correction and addition candidates from both report-to-tweet and tweet-to-report directions and integrates them into the final report.

ining disagreements with the automatic NLI metric and remaining challenges in information integration and unsupported relations.

2 Task Overview

The LT4CPR Shared Task requires concise, structured situation reports that can be verified against the input crisis-related social media messages. We follow the official task guidelines for the input and output formats and report schema.[1]

The data cover multiple crises, each divided into reporting windows. We refer to an independent input and report-generation unit corresponding to a crisis, a reporting window, and a replicate as a *cell*. Each cell provides the tweets within that period and crisis/window metadata. Reports are generated independently of other cells.

The output report consists of sections and subsections defined by the official schema. We refer to report Sections 3–11, which contain the principal factual information, as the *report body*. Each subsection contains one or more bullets, each with statement text, a confidence label, and supporting tweet IDs.

In addition to the body, the system generates an overview summarizing the report and a timeline organizing key events chronologically. We distinguish the generation and refinement of report Sections 3–11 from the generation of the overview and timeline.

3 Structured Draft Generation

To reduce omissions, duplication, and conflation when organizing information distributed across many tweets, we use a multistage pipeline that separates claim extraction, section-wise aggregation, report assembly, and summary generation. We describe each component below.

3.1 Claim Extraction

This stage extracts structured atomic claims from the input tweets, decomposing their factual content into units that can be independently integrated and verified in later stages (Table 1). Tweets within each reporting window are split into batches of at most 80 in input order and processed by an LLM. Each claim includes a source tweet ID, event type, section, subsection, event key, claim text, numerical figure, and status.

To assign claims consistently, we first ask the LLM to identify the type of event being reported and then output its corresponding section, rather than selecting a section directly. The extraction prompt includes a mapping between the 29 event types defined in the official taxonomy guide[2] and their sections. For example, a reported death is classified as KILL and assigned to report Section 3 (Casualties), while an evacuation is classified as EVACUATE and assigned to report Section 5 (Displacement).

[1] https://github.com/LT4CPR/2026_aacl_workshop/blob/c8bd925e1c21d82248b98c5faabb75146418966f/shared_task/data/DATA_FORMAT.md

[2] https://github.com/LT4CPR/2026_aacl_workshop/blob/c8bd925e1c21d82248b98c5faabb75146418966f/shared_task/data/TAXONOMY_GUIDE.md

We extend this mapping to subsections by identifying the placement of each event type in the training gold reports. In Section 6.4, we compare this configuration with one in which the mapping is omitted and the LLM selects sections and subsections directly.

When a tweet describes multiple distinct event types, each is extracted as a separate claim. The tweet in Table 1 yields two claims: identification of the chemical (HAZARDOUS_CONDITION, assigned to report Section 6, Hazard assessment) and designation of an evacuation zone (EVACUATE, assigned to report Section 5, Evacuations).

3.2 Section-wise Aggregation

Fine-grained claims extracted from individual tweets often include duplicate reports of the same event and numerical updates. This stage consolidates these claims and generates subsections (Table 2). Specifically, the claims from Section 3.1 are grouped by report section (3–11), and the LLM generates subsections and their report bullets using a fixed subsection vocabulary. The prompt includes the following instructions:

- **Duplicate event aggregation:** Merge duplicate accounts of the same event into a single bullet.
- **Numerical update handling:** Treat updates as separate bullets for each value rather than as duplicates. If a later report explicitly corrects an earlier value, describe it as a correction. Do not add the values together.
- **Length control:** Control report length using explicit target bullet counts in the prompt. These targets depend on the number of input tweets and are derived from training-report statistics.
- **Gold report style examples:** Provide gold reports from the training data as examples of the desired writing style.

3.3 Report Assembly

To ensure consistent application of the required output schema and confidence labels, we assemble the generated bullets into a report using deterministic rules. Each bullet retains its supporting tweet IDs and receives a confidence label based on the status of its content and the number of supporting tweets.

3.4 Overview and Timeline Generation

Finally, an LLM generates an overview and timeline from the completed body to provide high-level summaries of the detailed report. If output validation detects a structural issue in either component, the LLM regenerates that component while keeping the body fixed. Raw tweets are not supplied again at this stage.

4 Bidirectional Evidence-Grounded Refinement

Although structured draft generation produces drafts containing the main information, LLM-based extraction and aggregation can leave statements that misrepresent source tweets or omit important information. To detect errors and omissions in complementary ways, we collect edit candidates in both the report-to-tweet and tweet-to-report directions, verify them against source tweets, and integrate them into the body in a single pass.

4.1 Report-to-Tweet Correction Candidate Generation

This stage processes each draft bullet sentence by sentence, using the sentence to retrieve related tweet evidence and existing report bullets. These are used to check factual errors and within-report duplication and to propose corrections where needed. We retrieve two types of related information for each target bullet:

- **Related tweets:** Given the target bullet and the tweets it originally cites, the LLM generates three queries to search for supporting evidence, counterevidence, and related or updated information. Together with the bullet text itself, these form four BM25 (Robertson and Zaragoza, 2009) queries over tweets in the same cell, retrieving up to six tweets per query. Retrieved tweets serve as evidence for checking and correcting the target bullet.
- **Related report bullets:** Using the target bullet text as a BM25 query, we retrieve up to eight other bullets from the draft to check for duplication and whether one statement subsumes another.

The LLM receives the target bullet, its cited tweets, retrieved tweets, and related report bullets, and proposes one of three actions: *keep* the bullet unchanged, *replace* it using tweet evidence,

Input tweet [39]: *RT @northlinefreigh: WCEM just confirmed the leaking tank car is carrying anhydrous ammonia. Evacuation zone is 800 metres, please stay clear #ArdleyJunction #WexfordDerailment http://bit.ly/z6WHd6*

	Claim 1	**Claim 2**
event type	HAZARDOUS_CONDITION	EVACUATE
section / subsection	6 / Hazard assessment	5 / Evacuations
event key	chemical-identification	evacuation-radius
claim text	WCEM confirmed the leaking tank car was carrying anhydrous ammonia.	The evacuation zone was set at 800 metres for the Ardley Junction incident.
figure (type)	—	800 (radius_m)
status	reported	reported

Table 1: Example of claim extraction (dev cell `chemical.W1.k1`, tweet ID 39). The tweet reports two distinct events—the identification of the chemical and the setting of the evacuation zone—so each is extracted as a separate claim.

Input claims (excerpt)	**Generated bullets**
[18] WCEM ordered a precautionary evacuation within 800 metres of the Ardley Junction rail yard. [39] The evacuation zone was set at 800 metres for the Ardley Junction incident. (similar claims from 18 tweets in total: 18–78)	Wexford County Emergency Management ordered a precautionary evacuation within 800 metres of the Ardley Junction rail yard. *[confirmed]* cites [18, 26, 30, 31, 32, 39]
[80] The evacuation zone was extended to 1.6 kilometres for the Wexford derailment. (similar claims from 10 tweets in total: 79–97)	The evacuation zone was extended to 1.6 kilometres for the anhydrous ammonia release at Ardley Junction. *[confirmed]* cites [79, 80, 82, 83, 86, 87]

Table 2: Example of section-wise aggregation (dev cell `chemical.W1.k1`, section 5 Evacuations). Eighteen duplicate reports of the 800 m evacuation radius are merged into a single bullet, while the extension to 1.6 km becomes a separate bullet as a numerical update.

or *remove* it if it is unnecessary, for example because it duplicates another bullet or lacks supporting tweets.

An LLM judge then verifies the proposal, checking whether (1) the cited tweets support the entire revised statement, (2) useful information from the original bullet is preserved, and (3) the edit introduces unnecessary duplication with existing bullets. Only proposals that pass these checks are retained as correction candidates. The draft remains unchanged at this stage; edits are applied during candidate integration.

4.2 Tweet-to-Report Missing Information Recovery

The procedure in Section 4.1 generates correction candidates from existing report bullets and does not systematically search for information without a corresponding draft statement. This stage instead starts from tweets to identify information missing from the draft and propose additions.

Tweets in the same cell are processed in input order in batches of 20. The LLM classifies each tweet as *relevant*, *irrelevant*, or *uncertain* with respect to the crisis. For each relevant or uncertain tweet, it also extracts up to four claims, as in Section 3.1. To efficiently filter out information already present, extracted claims are compared with the full draft in batches of 16, and clearly covered claims are discarded. For each remaining claim, BM25 retrieves the top 12 related tweets and the top eight related report bullets.

The LLM receives the target claim, its originating tweet, retrieved tweets, tweets cited by the related report bullets, and the full draft. It classifies the claim as *covered* (adequately represented), *partial* (partly represented, with information still to add), *missing* (absent), or *not_relevant* (unnecessary for the report).

For partial or missing claims, the LLM proposes an additional bullet with a section, subsection, statement text, confidence label, and supporting tweet IDs. An LLM judge checks source support, duplication or contradiction with existing content, and confidence, among other criteria. Only proposals that pass verification are retained as addition candidates. As in Section 4.1, no changes are made to the draft at this stage.

4.3 Evidence-Grounded Candidate Integration

This stage combines the correction candidates from Section 4.1 and addition candidates from Section 4.2 and applies accepted edits to the draft. For each report section, the LLM receives all candidates for that section and their related source tweets, with the full report as context. It proposes at most six edits confined to the target section, accounting for duplication among candidates and their relationship to existing bullets.

Allowed edits include replacing or removing existing bullets, adding new bullets, and merging multiple bullets. Candidates are treated as editing hypotheses; source tweets provide the factual evidence. An LLM judge checks the proposed edits again. Only verified edits are applied, and existing bullets are retained when an edit is rejected. Table 3 illustrates candidate deduplication at this stage.

After all sections have been edited, the system validates bullet IDs, the schema, and citations. Finally, it regenerates the overview and timeline from the finalized body (report Sections 3–11) to produce the final report.

5 Experimental Setup

5.1 Data

We use the public training and development data from the LT4CPR Shared Task. We evaluate all configurations on the same 23 development cells, covering two crises: a chemical incident and an earthquake.

Training gold reports are used to construct the event-type-to-subsection mapping, set target bullet counts according to input tweet counts, and provide style examples during aggregation.

We limited the use of development data during system development where possible, but used some of it to improve the system. The results are therefore not fully independent of development and may overestimate performance on unseen data.

5.2 Evaluation Metrics

We use the official Main Score and Alignment F1 to evaluate agreement with reference reports. We additionally use NLI Support Rate as an unofficial auxiliary metric to assess whether cited tweets support the generated bullets.

Main Score is the arithmetic mean of BERTScore F1 (Zhang et al., 2020) and BLEURT (Sellam et al., 2020) between the generated and reference reports. Alignment F1 aligns their content to evaluate the balance between reference agreement, extraneous content, and omissions.

NLI Support Rate is the proportion of body bullets (report Sections 3–11) whose entire content is classified as `supported` by the cited tweets. We compute it for each cell using the diagnostic procedure in the shared-task development utilities.[3] Specifically, the diagnostic procedure decomposes each bullet into atomic claims, uses its cited tweets as the premise and each claim as the hypothesis for NLI, and counts a bullet as fully supported only when every claim is entailed. This is an auxiliary measure of automatically assessed source support, not a measure of factual accuracy established by human verification. Section 6.3 separately examines the relationship between this metric and source-support judgments through a model-based audit.

All metrics are aggregated using hierarchical averaging over replicates, then reporting windows, and finally crises.

5.3 Direct Generation Baseline

We use Direct Generation as a baseline. It receives all raw tweets in a cell, including tweet IDs, sources, and timestamps, together with crisis information and the reporting window, and generates the report in a single step following the common final output schema. It does not perform claim extraction or section-wise aggregation. Across the 23 evaluated cells, the maximum tweet count was 997 (earthquake.W4.k1), and the longest complete prompt contained 50,623 tokens (chemical.W4.k1), including instructions, schema, and the style example. Even with the 12,000-token output allowance, it fit within the 65,536-token context; no input tweets were truncated in any cell.

We provide one gold report from the training data, volcano.W2.k1, as a style example and instruct the model to write each bullet as an atomic sentence of approximately 10–20 words. Unlike Structured Generation, Direct Generation receives neither target bullet counts based on the number of input tweets nor section-specific length budgets. The LLM also generates confidence labels according to the criteria in the prompt.

[3] https://github.com/LT4CPR/2026_aacl_workshop/blob/c8bd925e1c21d82248b98c5faabb75146418966f/shared_task/tools/other_tools/audit_faithfulness.py

Existing content	Candidates	Integration result
Confirmed injuries near St. Marren ([3b.1]).	Four near-duplicate additions reporting injuries at the hall, all unconfirmed and citing [32, 34, 38, 47, 59].	Preserve [3b.1]; merge the four candidates into one new unconfirmed bullet with the same citations.

Table 3: Candidate deduplication during integration (`earthquake.W1.k1`, section 3 Injuries).

Although the initial output includes an overview and timeline, these components are subsequently regenerated and validated with the body held fixed, generally using the same procedure as Structured Generation.

Both methods use the same model and generation settings, detailed in Section 5.4. This comparison evaluates the draft-generation methods as a whole, including differences in length control and confidence assignment.

5.4 Implementation Details

We use Gemma-4-31B-it (Gemma Team, 2026) for all LLM operations in draft generation and bidirectional refinement. Candidate generation and LLM judging use the same model through separate requests; judging is therefore not evaluation by an independent model.

Appendix A provides implementation details, including generation-time model settings, retrieval settings, processing limits, retry behavior, and output validation for the overview and timeline. Appendix C provides an overview of the prompts for each LLM call.

6 Results and Analysis

6.1 Overall Results

Table 4 presents Direct Generation, Structured Generation, and Bidirectional Refinement applied to each. The Structured Generation configuration achieved higher scores than Direct Generation on all metrics: Main Score increases from 0.4198 to 0.5308, Alignment F1 from 0.1665 to 0.2496, and NLI Support Rate from 0.2938 to 0.7925.

The effects of Refinement differed by draft-generation method. Applied to Direct Generation, Refinement increased all three metrics: Main Score from 0.4198 to 0.4511, Alignment F1 from 0.1665 to 0.1775, and NLI Support Rate from 0.2938 to 0.6525. Applied to Structured Generation, it increased NLI Support Rate from 0.7925 to 0.8566 but decreased Main Score from 0.5308 to 0.5216 and Alignment F1 from 0.2496 to 0.2291.

Thus, Refinement consistently improved NLI-based source support, while its effects on agreement with reference reports depended on the draft-generation method. In particular, for Structured Generation, which already achieved high reference agreement, the source-support gains from additional edits did not translate into improvements in reference-based metrics. On the official test set, the submitted configuration ranked first among the nine participating teams (Section 6.2).

6.2 Submission Configuration

We submitted the combination of Structured Draft Generation and Bidirectional Evidence-Grounded Refinement as our primary system for the shared task.

Because the Structured Generation configuration achieved a higher Main Score than Direct Generation, we selected it as the draft-generation method.

Adding Refinement slightly reduced Main Score from 0.5308 to 0.5216, while increasing NLI Support Rate from 0.7925 to 0.8566.

Manual comparison with source tweets in a small number of examples also identified corrections of erroneous statements and recovery of omitted information. Based on these observations, we submitted the Full configuration with the intention of improving source support while preserving most of the reference agreement. This manual inspection was limited to a small number of examples.

The model-based audit in Section 6.3 and the ablations in Sections 6.4 and 6.5 were conducted after submission and were not used to select the submitted configuration. The official shared-task evaluation (Xia et al., 2026) ranked our submission first among the nine participating teams, with a primary score of 0.682 against 0.664 for the second-ranked team and 0.469 for the organizer baseline (Table 5). This is a system-level comparison; a matched draft-only test result was unavailable at submission time (see Section 6.7 for a post-hoc decomposition using the later-released references).

System	Main Score	Alignment F1	NLI Support Rate
Direct Generation	0.4198	0.1665	0.2938
Direct Generation + Refinement	0.4511	0.1775	0.6525
Structured Generation	**0.5308**	**0.2496**	0.7925
Structured Generation + Refinement	0.5216	0.2291	**0.8566**

Table 4: Effects of structured draft generation and bidirectional refinement: direct generation, structured generation, and each with refinement.

Metric	Organizer baseline	Our submission
Primary	.469	.682
BERTScore F1	.463	.727
BLEURT	.475	.636
Alignment F1	.208	.463
ROUGE-1	.290	.616
ROUGE-2	.172	.448
ROUGE-L	.251	.549

Table 5: Official test results. Source: workshop leaderboard, checked 14 September 2026.

6.3 Model-Based Audit of Refinement

In Section 6.1, adding refinement to Structured Generation increased NLI Support Rate but decreased Main Score and Alignment F1, motivating an additional evaluation of whether individual edits actually improve the report when checked against the sources. We randomly sampled 50 of the 466 body bullets whose content, confidence, citations, or placement was changed by Full refinement. Of these, 34 were modifications to existing bullets with a one-to-one correspondence between the before and after versions, while the remaining 16 had no matched pre-refinement bullet and were treated as additions.

We conducted a blind evaluation by presenting the before and after versions as A/B with their temporal order hidden, using a separate reasoning model, `gpt-6-astra`, with medium reasoning effort. The overall assessment considered source support, epistemic status, information preservation, placement, and citations: 35 edits (70%) were judged beneficial, eight (16%) equivalent, and seven (14%) detrimental. Among the 34 edits with a one-to-one correspondence between before and after statements, the number of fully source-supported statements increased from 16 to 25. However, of the 45 post-refinement statements labeled supported by automatic NLI, nine were judged in this blind evaluation to retain unsupported content. These results suggest that refinement often improves source-grounded quality, while increases in NLI Support Rate cannot be interpreted directly as the magnitude of improvement in factual support. Details of the evaluation protocol are provided in Appendix D.

6.4 Ablation Study of Structured Draft Generation

Table 6 presents ablation results for three design choices in structured draft generation. All three ablations decrease Main Score, with the largest decrease occurring when event type mapping is removed. Removing length control increases the number of body bullets to approximately 1.9 times the original count, while Alignment F1 decreases from 0.2496 to 0.1822 and NLI Support Rate increases from 0.7925 to 0.9128. These results indicate that length control helps adjust the amount of information to match reference reports, while reference agreement and NLI-based source support do not necessarily change in the same direction.

6.5 Ablation Study of Bidirectional Refinement

Table 7 compares Correction, Recovery, and Full refinement. NLI Support Rate is 0.8429 for Correction, 0.8096 for Recovery, and 0.8566 for Full, with Full achieving the highest value by combining both directions.

However, Full has lower Main Score and Alignment F1 than either unidirectional configuration. We therefore observe no improvement in the official metrics from combining both directions.

Full applied replacement operations to 359 of the 1,069 draft body bullets (33.6%), so the small change in Main Score is not due to a low volume of edits.

These quality effects come with steeply increasing computational cost. Moving from Direct Generation to Structured Draft Generation increases LLM calls from 8 to 138 and total input-plus-output tokens from 0.23M to 0.96M (measured on the same eight cells of chemical and earthquake, W1–W4). Refinement is costlier still: across all 23 development cells, Full refinement took 1,507 sec-

System	Main Score	Alignment F1	NLI Support Rate
Structured Generation	**0.5308**	**0.2496**	0.7925
Without event type mapping	0.5126	0.2116	0.8557
Without length control	0.5180	0.1822	**0.9128**
Without gold report style examples	0.5262	0.2395	0.8269

Table 6: Ablation study of structured draft generation.

System	Main Score	Alignment F1	NLI Support Rate
Structured Generation	**0.5308**	**0.2496**	0.7925
Correction	0.5264	0.2435	0.8429
Recovery	0.5256	0.2354	0.8096
Full	0.5216	0.2291	**0.8566**

Table 7: Comparison of the Correction, Recovery, and Full refinement variants.

onds per cell on average (range 395–3,296), 9.6 hours in total on a single H200. Refinement improves NLI Support Rate and audit quality, but the computational cost rises sharply at each stage, so its application should be decided by weighing quality requirements against the computational budget.

6.6 Error Analysis

Table 8 illustrates errors that remain in Full. The first example combines reports of *multiple derailed cars* and *14 cars involved in the incident* into a statement that 14 cars derailed. Even when individual facts appear in the sources, the association between a quantity and a state may be unsupported. Our method fails to remove this over-inference.

In the second example, refinement links dangerous conditions and a closure reported together in the source, adding a causal explanation for the closure. This unsupported relation was absent from the draft but introduced by refinement and was not detected by automatic NLI.

These examples highlight the need to verify not only individual facts in the sources but also the associations between quantities and states and the causal relations asserted in generated statements.

6.7 Post-hoc Analysis with Released Test References

After the submission deadline, the organizers released the reference reports for the test set and the system outputs of all participating teams. We used these to conduct a post-hoc analysis of our submitted system on the test set. Our local evaluation exactly reproduces the official primary score of our submission (0.6816).

Before submission, we adopted refinement based on its improvement in NLI Support Rate and manual inspection of a small number of edits (Section 6.2); the post-submission audit (Section 6.3) provided additional evidence of edit quality. On the test set, the draft-only outputs score 0.6791, and the submitted version, which applies refinement to the same drafts, scores 0.6816 ($+0.0025$); the slight decrease observed on the development set (0.5308 to 0.5216) did not recur. These results are consistent with the decision to retain refinement, although the gain is small.

Compared with the second-ranked team crisis by crisis, our submission scores higher on *damsafety* ($+0.055$), *indfire* ($+0.048$), and *landslide* ($+0.039$), and lower only on *collapse* (-0.030) among the six crises.

The loss on collapse occurs in all six replicates of window W2, for two reasons. First, while the reference sentences are short and template-like, our output tends to append location qualifiers, which lowers BERTScore precision and BLEURT. We attribute this tendency to our editing policy, which prioritizes retaining source-supported place and scope qualifications over brevity: the instructions prohibit merging places or temporal scopes merely to shorten the text. Second, in `collapse.W2.k1`, information that belongs in the Injuries subsection was missing from the generation stage onward.

By window, our primary-score advantage over the second-ranked team is $+0.036$ at W1 and shrinks monotonically to approximately zero at W4. The advantage consists almost entirely of BERTScore precision ($+0.105$ at W1 to $+0.025$ at W4), while recall is slightly below the second-ranked team in every window. The evaluator dilutes precision by scoring system subsections that have no corresponding reference subsection as zero, and our outputs consistently contain fewer such sub-

System output (after refinement)	Source tweets (excerpt)
Over-inference across sources (`chemical.W1.k1`, section 4 Damage)	
Fourteen freight cars derailed near the Ardley yard, causing infrastructure damage at the junction. *[confirmed]* cites [2, 3, 5, 6, 11, 12]	[6] Just saw multiple freight cars derailed near the Ardley yard [12] RT @WCEM: CONFIRMED: WCEM has deployed to #ArdleyJunction #WexfordDerailment site. 14 cars involved in the incident #ammonia
Unsupported causality introduced by refinement (`earthquake.W2.k4`, section 4 Service disruptions)	
Officials stated closures were in place around Caldera, and local reports indicated the coastal area was closed **due to dangerous conditions**. *[confirmed]* cites [71, 82, 83, 89, 195]	[71] officials: closures are in place around Caldera #earthquake. Avoid the area if possible #CalderaQuake [195] Dangerous conditions reported at the coast, with the area closed per local reports as well...

Table 8: Two errors in the refined report body. Top: the report binds the count of cars involved in the incident to the derailment itself, a correspondence no source tweet states. Bottom: refinement adds a causal explanation for a closure, although the source reports dangerous conditions and closure together without establishing causality.

sections (on average 0.2 vs. 0.6 at W1 and 0.7 vs. 1.4 at W2, with comparable levels from W3 onward). Because the reference reports grow from 38 tokens on average at W1 to 254 at W4, suppressing content absent from the reference—through the bullet-count budgets in aggregation and the worth-reporting check in refinement—matters most for precision in the early windows with small references, whereas in later windows extra content increasingly finds a reference counterpart and the gap disappears. The observed gradient is consistent with this account. A comparison with a smaller draft-generation model, Gemma-4-12B-it, on the test and development sets is given in Appendix E.

7 Conclusion

Our structured-generation configuration achieved higher reference agreement than direct generation, while bidirectional refinement increased NLI Support Rate for both draft-generation methods but affected reference agreement differently depending on the method. The submitted combination of both components ranked first among the nine participating teams in the official test evaluation.

Limitations

The blind audit covers a random sample of 50 of the 466 modified body bullets and relies on a single LLM evaluator. Hiding system names and before/after labels does not eliminate model-specific biases, and the audit does not constitute human validation of factuality. It also excludes pure deletions, unchanged statements, and unrecovered information, so it does not assess overall report quality.

Adding bidirectional refinement to Structured Generation does not improve the primary reference-based evaluation metric. These experiments therefore do not establish the general effectiveness of refinement; its benefits may be limited or depend on the evaluation conditions.

The evaluation cells are not independent, as their windows and replicates overlap. The results cover only two crises and one seed and do not establish generalization to unseen disasters or statistical significance.

Ethical Considerations

We use the shared-task data solely for situation report generation. We do not link individuals in the data to external information to identify, contact, or profile them.

Aggregating crisis information with LLMs can nevertheless lead to overconfident statements about uncertain reports, inappropriate integration of distinct information, and misinterpretation of quantities or negation. Although our system seeks to reduce these risks through citations and source-grounded refinement, it does not guarantee the accuracy of generated content.

Generated reports should therefore be treated as research outputs. Any use in actual crisis response requires human verification against the original information.

References

Luyu Gao, Zhuyun Dai, Panupong Pasupat, Anthony Chen, Arun Tejasvi Chaganty, Yicheng Fan, Vincent Zhao, Ni Lao, Hongrae Lee, Da-Cheng Juan, and Kelvin Guu. 2023. RARR: Researching and Revising What Language Models Say, Using Language Models. In *Proceedings of the 61st Annual Meeting of the*

Association for Computational Linguistics (Volume 1: Long Papers), pages 16477–16508. Association for Computational Linguistics.

Gemma Team. 2026. Gemma 4 Technical Report. arXiv:2607.02770.

Frederic Kirstein, Terry Ruas, and Bela Gipp. 2025. What's Wrong? Refining Meeting Summaries with LLM Feedback. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 2100–2120. Association for Computational Linguistics.

Mathieu Ravaut, Aixin Sun, Nancy Chen, and Shafiq Joty. 2024. On Context Utilization in Summarization with Large Language Models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2764–2781. Association for Computational Linguistics.

Stephen Robertson and Hugo Zaragoza. 2009. The Probabilistic Relevance Framework: BM25 and Beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning Robust Metrics for Text Generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892. Association for Computational Linguistics.

Fei Xia, Haotian Zhu, Yoo Wun Lee, William D. Lewis, Antonis Anastasopoulos, and Belu Ticona. 2026. Overview of the LT4CPR 2026 Shared Task: Generating Situation Reports from Social Media Messages. In *Proceedings of the First Workshop on Language Technologies for Crisis Preparedness and Response (LT4CPR)*, in conjunction with AACL 2026, Hengqin, Zhuhai, China. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating Text Generation with BERT. In *International Conference on Learning Representations*.

Yusen Zhang, Ansong Ni, Ziming Mao, Chen Henry Wu, Chenguang Zhu, Budhaditya Deb, Ahmed Awadallah, Dragomir Radev, and Rui Zhang. 2022. Summ^N: A Multi-Stage Summarization Framework for Long Input Dialogues and Documents. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1592–1604. Association for Computational Linguistics.

Chenguang Zhu, William Hinthorn, Ruochen Xu, Qingkai Zeng, Michael Zeng, Xuedong Huang, and Meng Jiang. 2021. Enhancing Factual Consistency of Abstractive Summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 718–733. Association for Computational Linguistics.

A Implementation Details

Generation uses Gemma-4-31B-it with BF16 and a maximum context length of 65,536 tokens. We set temperature to 0.7, top-p to 0.95, top-k to 64, and seed to 42, disable thinking, and constrain outputs with JSON schemas. The generator and judge invoke the same model through separate requests. API/JSON retries for malformed outputs are limited to three attempts; semantically rejected edits are not repeatedly proposed until accepted.

BM25 uses $k_1 = 1.2, b = 0.75$. Correction retrieves up to six tweets per query and eight related report bullets. Recovery retrieves up to 12 tweets and eight report bullets. Recovery is capped at 120 relation assessments, 12 distinct addition candidates for acceptance, and 300 logical calls per cell. The integrator proposes at most six transactions per section and at most three replacement bullets per transaction.

The initial overview/timeline prompt specifies four overview sentences and four to ten timeline sentences. The subsequent finalizer lowers the timeline minimum to one sentence, leaves the body unchanged, and generates an overview of four sentences and a timeline of one to ten sentences, requiring supporting body IDs. Each timeline sentence inherits the union of citation IDs from its supporting body bullets. We allow up to three judging rounds and provide an extractive fallback, recording its use. In three Direct Generation cells, generation of these auxiliary components failed citation-scope checks, so we retained the original Direct Generation output, which had passed format validation, without editing its body.

A.1 Length-Control Rules

For a cell containing n tweets, the total body-bullet budget is $T = nr(n)$, where $r(n) = 0.10$ for $n < 400$, 0.09 for $400 \le n < 700$, and 0.07 for $n \ge 700$. These piecewise rates were chosen based on training-report statistics. Let c_s be the number of extracted claims assigned to report section s, and $C = \sum_s c_s$. For each section with $c_s > 0$, the target is

$$t_s = \max\Big(1, \operatorname{round}\Big[T\left(0.6\frac{c_s}{C} + 0.4p_s\right)\Big]\Big).$$

For sections 3–11, the fixed prior shares p_s are, respectively, 0.166, 0.137, 0.108, 0.108, 0.157, 0.163, 0.049, 0.011, and 0.100. Sections with no claims are skipped; the remaining shares are not renor-

malized. Rounding follows Python's round (ties to even), so section targets need not sum to $\mathrm{round}(T)$.

The aggregation prompt gives t_s as the target and an acceptable range from $\max(1, \mathrm{round}(0.7t_s))$ to $\mathrm{round}(1.3t_s) + 1$, together with the total-report guide $\mathrm{round}(T)$. It instructs the model to merge repeated events and, if still over target, omit the least-supported minor details first. For $n < 150$, an additional instruction favors fewer well-supported, merged bullets without padding. These are prompt-level targets, not hard output-count constraints; no subsection-level counts are allocated.

B Reproducibility and Case Identification

For each configuration, we retain drafts, inputs, prompts and schemas, candidates, integration edits and judgments, finalizer requests and responses, and evaluation results. Input and output hashes are used to verify that configurations correspond to the same cells and drafts. Replaying saved artifacts does not guarantee bitwise-identical stochastic generation across execution environments.

The quantitative result tables in this paper are generated from saved evaluation JSON files for which all 23 cells and NLI evaluations were complete. Main Score, Alignment F1, and NLI Support Rate all use hierarchical averaging over replicates, windows, and crises, in that order. The selected qualitative examples in the tables in Section 6 are identified by the cell IDs chemical.W1.k1 and earthquake.W2.k4 and by within-cell tweet IDs. These examples are not treated as a representative sample of the error distribution across all 23 cells.

The second example in Table 8 is audit case C011 (original sample ID R30). Identifiers R01–R50 follow the saved sampling order, whereas C001–C050 identify the anonymized evaluation cases; Appendix D describes sampling, criteria, and comparison with NLI.

The replacement proportion in Section 6.5 counts distinct draft body bullet IDs targeted by accepted integration transactions that contain both old IDs and replacement bullets. Across 23 cells, 350 such transactions targeted 359 of 1,069 draft body bullets, including multi-bullet merges; replacements may change confidence, citations, or placement without changing the text. Pure additions and pure deletions are excluded from this numerator. This operation-based count differs from the post-refinement sampling population in Appendix D.

C Prompt Overview

All LLM calls in our system use JSON-schema-constrained outputs and follow a common evidence policy: report text, editing candidates, and diagnostic results are not treated as factual evidence; only source tweets serve as evidence. For space reasons, we do not reproduce the prompts in full. Table 9 summarizes the role, inputs, outputs, and core constraints of every call, and the following subsections describe the prompt instructions relevant to the claims in the main text.

C.1 Claim Extraction Prompt

The extraction prompt first instructs the model to treat each tweet as expressing event types from a fixed taxonomy, identify the types present in each tweet, produce one claim per type, and assign each claim to the section specified by the mapping table. Table 10 presents this mapping; the actual prompt also includes typical expressions for each type and routing rules for ambiguous cases. An event key is defined as a short identifier shared by claims referring to the same event; successive numerical updates share the same key. The status field takes one of seven values: reported, negative, planned, rumour, denial, correction, or superseded.

C.2 Aggregation Prompt

The aggregation prompt specifies the following main rules.

- Use one bullet per event. Merge paraphrases, retweets, and repeated reports of the same event, but do not merge distinct events.
- When different values are reported for the same quantity, retain a separate bullet for each value in chronological order; never sum them. Only an explicit correction supersedes an earlier value, and the wording must make the correction clear.
- Keep categories distinct: missing $\neq$ dead, suspected $\neq$ confirmed cases, injuries $\neq$ treatment $\neq$ admissions, and planned $\neq$ completed actions.
- Retain reports of absence (e.g., "no deaths") as actual bullets without broadening the scope of the evidence. Attribute denials to their speakers; a denial alone does not establish that an event did not occur.

- Write each bullet as a single past-tense sentence in an official sitrep style, approximately 10–20 words long. Up to four reference bullets per subsection from three training documents are provided as style examples.
- The user message specifies target bullet counts (Section 3.2). When these are exceeded, remove the least-supported details first.
- Assign four flags to each bullet: negative, planned, rumour_or_denied, and contradicted. Confidence is assigned deterministically rather than generated by the LLM: any true flag yields unconfirmed; otherwise, at least two distinct cited tweets yield confirmed.

C.3 Refinement Judge Criteria

The acceptance criteria for each judge in Section 4 are explicitly represented as fields in its output schema.

- Correction judge (Section 4.1): accept only if unsupported_new (new unsupported content), lost_supported (lost supported information), and new_duplicates (new duplicate content) are all empty, and every evidence check includes an exact quotation from the cited tweets.
- Addition judge (Section 4.2): accept only if accept, whole_text_supported, same_event_window, confidence_appropriate, nonduplicate, and no_new_conflict are all true, and the set of tweet IDs in the evidence quotations matches the candidate's citation set.
- Integration judge (Section 4.3): apply a transaction only if accept, whole_support, preserves_useful_old, classification_correct, event_compression_correct, worth_reporting, and all checks on old and new items are true. The proposer's rationale is not passed to the judge.

C.4 Direct Generation Baseline Prompt

The Direct Generation prompt (Section 5.3) instructs the model to report only content supported by the supplied tweets, without adding plausible details or background knowledge; express one atomic factual claim per bullet in approximately 10–20 words; and cite one to six tweets providing the most direct evidence. One reference report (train volcano.W2.k1) is supplied as a style example. The LLM assigns confidence according to the prompt criteria: single reports, contradictions, unexecuted plans, and rumours are unconfirmed, whereas corroboration by multiple independent reports yields confirmed.

D Blind Evaluation Protocol

D.1 Population and Sampling

The post-submission evaluation in Section 6.3 compares the bodies (report Sections 3–11) of Structured Generation and Full refinement across 23 development cells. Eligible Full bullets differ in text, confidence, the cited tweet ID list, or section/subsection; changes only to bullet numbering are excluded. The unit is a post-refinement bullet, excluding pure deletions, unchanged statements, overviews, and timelines.

We first matched bullets identical in these attributes. For the remaining bullets, we applied Hungarian matching with a score combining citation-set Jaccard similarity (weight 0.6) and SequenceMatcher similarity of lowercased text (weight 0.4) for same-section pairs, assigning zero to cross-section pairs and discarding matches below 0.3. Unmatched Full bullets are treated as additions, so the distinction between additions and modifications depends on this procedure.

We sorted the resulting 466 items by cell ID and post-refinement bullet ID and sampled 50 without replacement using Python's `random.Random(20260916).sample`. The sample spans two crises and 19 cells, with 34 edits paired with draft statements and 16 additions without corresponding draft statements.

D.2 Evaluation Materials and A/B Assignment

Each case includes both versions' text, confidence, section/subsection, and citation IDs, together with the complete source tweets in the union of their citations, including source categories and timestamps. To assess duplication, information preservation, and placement, we supplied shared context containing other draft statements in the relevant section and the three most similar draft statements, excluding the corresponding target draft statement. The evaluator was instructed to treat this context as context rather than factual evidence and assess each version's source support using only its own citations.

Using seed 20260917, we assigned the post-refinement version to A/B in 17 paired cases each and eight addition cases each. The other version of an addition is `null`, denoting absence. We shuffled case order, source order, and shared context, assigning case IDs C001–C050 and source IDs beginning at T001 within each case. The evaluator did not receive the mapping key, system names, before/after labels, development history, reference reports, prior ratings, or NLI judgments. For additions, however, the absence of one version remains visible.

D.3 Evaluator and Execution

We used OpenAI `gpt-6-astra` with medium reasoning effort, separate from the Gemma model used for generation and refinement. Each case was evaluated in a fresh ephemeral Codex session from an isolated workspace containing only that case and the fixed task, rubric, and output schema. Sessions did not inherit development conversations or other case judgments; logged file reads were confined to the supplied materials. No dated model snapshot ID was recorded.

The evaluator returned JSON containing support labels, unsupported fragments, evidence quotations, confidence and placement assessments, an overall rating, and a rationale. Transient execution failures such as capacity errors could be retried; completed judgments were not overwritten or retried based on their content.

D.4 Rubric and Aggregation

Source support was labeled `supported` (the entire statement is supported), `partial` (some content is unsupported), `unsupported` (the citations do not support the statement), or `not_applicable` (the version is absent). Separately, overall quality was rated for *B relative to A*, considering source fidelity, epistemic status, information preservation, duplication, placement, and citations:

- 5: B repairs a clear factual error or important omission in A.
- 4: B makes a minor improvement in qualification, attribution, epistemic status, or information.
- 3: Approximately equivalent, or gains and losses balance.
- 2: B introduces a partial error, worsens placement, or loses useful information.
- 1: B introduces a severe error or reverses the meaning.

The evaluator was instructed not to reward citation count or polished wording alone, or automatically prefer a nonempty version.

After all 50 judgments were complete, we used the mapping key to orient ratings toward refinement: we retained rating r when B was the post-refinement version and used $6 - r$ when A was the post-refinement version. Ratings 4–5, 3, and 1–2 were grouped as beneficial, equivalent, and detrimental, yielding 35, eight, and seven edits, respectively. This is an overall edit-quality assessment, not a factual-support improvement rate.

D.5 Source Support and NLI Comparison

The before/after support comparison uses only the 34 paired edits, excluding the 16 additions. Fourteen pairs were fully supported before and after, 11 changed from partial to full support, seven remained partial, and two changed from full to partial support. Thus, fully supported statements increased from 16 to 25.

After judgments were fixed, we compared them with saved NLI results. Nine of the 45 post-refinement statements labeled `supported` by NLI were `partial` in the blind evaluation. Of four cases changing from non-supported to supported under NLI, only one also changed from partial to full support in the blind evaluation. These comparisons concern AI judgments on a sample of modified items; they neither estimate NLI accuracy against human ground truth nor re-estimate support rates across all 23 cells.

E Model-Size Comparison

Table 11 compares replacing the draft-generation model Gemma-4-31B-it with Gemma-4-12B-it. Prompts, generation settings, and the evaluator revision (`c8bd925`) are identical to the experiments in the main text. The 12B test run completed 116 of the 117 test cells, so the test column reports both models aggregated over the same 116 cells (the 31B figure over all 117 cells is 0.6791).

Model	dev (23 cells)	test (116 cells)
Gemma-4-12B-it	0.5030	0.6537
Gemma-4-31B-it	0.5308	0.6788

Table 11: Primary score by draft-generation model size.

Stage	Prompt	Key inputs	Output (JSON)	Core constraints
3.1	Claim extraction	80-tweet chunk; crisis/window metadata	claims: tweet ID, event type, section/subsection, event key, text, figure, status	identify event types first; section from the mapping table; one claim per distinct event in a tweet
3.2	Aggregation (per section 3–11)	section claims; style examples; target bullet counts	subsections with bullets: text, tweet IDs, 4 flags	merge duplicates; keep numerical updates separate; stay near targets; 10–20-word sitrep sentences
3.4	Initial overview / timeline	compiled body text (sections 3–11); crisis metadata	4 overview bullets; 4–10 timeline bullets with tweet IDs	no raw tweets; timeline citations copied from body bullets; no new facts
4.1	Audit	target bullet; its cited tweets	supported/unsupported fragments; support / challenge / related queries	cited tweets are the only evidence
4.1	Edit	target; retrieved tweets; related bullets; audit	keep / replace / remove; new bullets with quoted evidence	preserve supported relations; remove unsupported fragments; no style-only edits
4.1	Judge	proposal without the proposer's rationale	verdict; per-claim evidence checks with exact quotes	accept only with no unsupported content, no lost supported information, no new duplicates
4.2	Extraction	20-tweet batch; crisis/window metadata	per-tweet relevance; up to 4 relations (span, claim, evidence status)	labels are working hypotheses, not verified truth
4.2	Routing	16 relations; full draft	covered / review / uncertain / no-contribution	"covered" requires a literal witness string in the draft
4.2	Locate	claim; evidence tweets; full draft	covered / partial / missing / not-relevant; addition proposal (section, title, text, confidence, tweet IDs)	additions cite only reviewed tweets; canonical subsection titles
4.2	Judge	addition proposal; sources	six boolean checks; evidence quotes	all checks must pass; quotes must match the cited tweets
4.3	Propose (per section)	whole report as context; section candidates; source tweets	up to 6 transactions ($\leq$3 bullets each); reasons for unused candidates	edits restricted to the current section; candidates are hypotheses, tweets are evidence
4.3	Judge	one transaction; old/new items; sources	accept; per-item checks with exact quotes	whole support; preserve useful old content; correct classification; worth reporting
–	Finalization	final body (sections 3–11)	4 overview sentences; 1–10 timeline sentences	body fixed; at most 3 attempts, then extractive fallback
5.3	Direct baseline	all tweets; schema; one gold report example	full report JSON	report only what the tweets support; one atomic claim per bullet

Table 9: Inventory of LLM prompts: role, key inputs, outputs, and core constraints. All calls use JSON-schema-constrained outputs.

Section	Event type → subsection
3 Casualties	KILL→Fatalities; INJURE→Injuries (Cases in outbreaks); MISSING→Missing persons; TRAP→Trapped persons; RESCUE→Rescued persons; FOUND→Found persons; RECEIVE_HELP (medical)→Treatment and admissions
4 Infrastructure	DAMAGE→Damage; DESTROY→Destruction; DISRUPT_SERVICE→Service disruptions; RESTORE→Restoration progress
5 Displacement	EVACUATE→Evacuations; DISPLACE→Displacement; RETURN→Returns
6 Hazard	HAZARDOUS_CONDITION, SITUATION_STATE→Hazard assessment
7 Response	DEPLOY, COORDINATE→Operational response
8 Communication	WARN→Warnings and alerts; REQUEST→Requests for help; INFORM→Information updates
9 Aid	PROVIDE_HELP→Aid provided and distributed; RECEIVE_HELP (non-medical)→Aid received and needs coverage; TRANSFER, FUNDING (aid-directed)→Donations and financial transfers
10 Administrative	MEETING→Meetings and decisions; POLICY_CHANGE, TRANSFER (non-aid)→Administrative and economic measures
11 Social	SOCIAL_SUPPORT→Social support; ATTITUDE, PROTEST→Public sentiment

Table 10: Event-type-to-section/subsection mapping given to the extraction prompt. The mapping builds on the official taxonomy, with task-specific routing for medical help and aid-directed funding and subsection assignments added by this system.

Extractive Corroboration Pipeline for Crisis Situation Report Generation from Social Media

Sreeja Guha Majumdar
Heritage Institute of Technology
sreeja.guhamajumdar.csbs28@heritageit.edu.in

Abstract

Automatic generation of crisis situation reports from social media requires transforming noisy, short-form text into structured, evidence-grounded prose, without access to information outside a single bounded time window. This paper describes an extractive pipeline submitted to a shared task on this problem: a supervised section classifier trained on tweet-to-bullet citation links, corroboration-based confidence estimation, safety-netted extractive bullet-text synthesis, and rule-based temporal filtering. The submitted system passes the task's structural validator on all official test cells, spanning crisis types absent from training, and receives an official evaluation score of 0.480. Decomposing the two metrics composing this score shows that the system obtains a BERTScore F1 of 0.508 and a BLEURT score of 0.451, exposing a substantial difference between semantic similarity and learned fluency/adequacy evaluation. Development-time diagnostics, including a coreference-recall experiment and a per-section quality breakdown, provide independent evidence that extractive text generation is a major remaining limitation. Three architectural alternatives tested during development are reported with their measurements rather than omitted, and a scoped, safety-netted rewrite step is identified as a direct next architectural extension.

1 Introduction

Crisis situation reports translate an unfolding disaster into a structured document organized by topic — casualties, infrastructure impact, displacement, hazard assessment, response actions, and related categories — so that a decision-maker does not have to read the underlying social media stream directly. The shared task addressed here frames automatic report generation as a per-cell, per-window problem: given only the tweets posted during one bounded time window of one crisis event, a system must independently produce a report whose bullets are (a) grounded in cited tweet evidence, (b) labeled with a confidence level derived from corroboration, and (c) free of information the window could not plausibly have supported, such as retrospective framing or figures dated beyond the window's end.

Two properties distinguish this task from related extractive crisis-summarization benchmarks. First, gold bullet text is authored independently of any single tweet, drawn from a separately written full report rather than pasted from source material; a system that reproduces a representative tweet verbatim, however well selected, is measured against a sentence it did not generate. Second, systems process every cell independently, without access to other windows or other cells of the same event, bounding how closely a corroboration-based confidence estimate can approximate the task's full-report corroboration rule.

An earlier keyword-based baseline scored precision 0.044, recall 0.040, and F1 0.040 on the internal development metric (Section 5.1). Its limitations were attributed largely to three compounding causes: the extractive-text gap above; a hand-written keyword vocabulary calibrated to wildfire, flood, and earthquake crises with no coverage for crisis types such as chemical spills or structural collapses; and a labeled dataset — direct tweet-to-bullet citation links present in every gold report — left unused. This paper documents a replacement pipeline, submitted as system `extraction-de`, addressing each cause directly, evaluated against both the internal metric and the shared task's official evaluation.

The contributions of this paper are: (1) a supervised section classifier trained on citation-derived weak supervision, evaluated under a cross-crisis-type generalization protocol matching the task's own train/development/test structure; (2) a bullet-generation procedure combining corroboration-based confidence, latest-figure resolution, rule-

Proceedings of the First Workshop on Language Technologies for Crisis Preparedness and Response, pages 156–165
November 9, 2026 ©2026 Association for Computational Linguistics

based temporal filtering, and two safety nets — numeric grounding and negation preservation — guarding a future abstractive rewrite step against hallucination and polarity flips; (3) three architectural alternatives tested during development, reported with measurements rather than omitted; and (4) an analysis combining official results with development-time diagnostics to identify clustering granularity and text generation, rather than classification, as the main remaining sources of error.

2 Related Work

CrisisFACTS (McCreadie and Buntain, 2023) recasts the TREC Incident Streams track as a multi-stream (Twitter, Reddit, news, Facebook) crisis fact-extraction and timeline-summarization task, scored against near-verbatim retrieved-and-ranked source snippets — an extractive evaluation. This is a materially different task shape from the one addressed here, where gold bullets are independently authored sentences rather than retrieved snippets, which is one reason an extractive pipeline faces a structural text-generation challenge here that is less central in a source-snippet retrieval setting.

Crisis2Sum (Seeberger and Riedhammer, 2024) compares extractive and LLM-based abstractive baselines for multi-stream disaster summarization on the CrisisFACTS data, finding that an extractive, query-focused framework can perform competitively with abstractive approaches on long, information-dense summaries. This supports classify→cluster→fuse as a credible general architecture, matching the pipeline in Section 4.

On classification, zero-shot prompting of instruction-tuned language models has been evaluated on crisis-tweet classification (McDaniel et al., 2024) and compared against parameter-efficient fine-tuning on the HumAID benchmark (Han et al., 2026), where LoRA fine-tuning reaches roughly 80% accuracy against roughly 42% for zero-shot prompting on a comparable task. This motivates the supervised classifier in Section 4.2 over a zero-shot approach; a zero-shot fallback is nonetheless implemented and left disabled (Section 4.7).

For report generation, an uncertainty-aware framework (Delos Reyes and Kitamoto, 2025) composes situational-awareness classification, uncertainty assessment, and generation, cross-checking model-elicited claim plausibility against an external proxy (USGS PAGER earthquake-impact estimates). The present pipeline shares this classify/estimate-confidence/generate shape, but its confidence stage is corroboration-counting rather than model-elicited, and no external proxy is available under the cell-independence constraint, which disallows looking beyond the cell being scored.

For automated evaluation, BERTScore (Zhang et al., 2020) and BLEURT (Sellam et al., 2020) are widely used to assess semantic overlap and natural language generation quality. In social media text mining, specialized BERT architectures such as CrisisBERT (Liu et al., 2021) and benchmarks like HumAID (Alam et al., 2021) demonstrate the value of domain-adapted representation learning for extracting humanitarian information under distribution shifts across disaster types. Finally, work on factual consistency in abstractive generation (Kryscinski et al., 2020) emphasizes numeric and logical faithfulness, motivating our explicit safety-net checks.

3 Task and Data

The shared task provides train, development, and test splits, each composed of independent cells: one crisis, one time window, one replicate. Development covers two crisis types (chemical release, earthquake) absent from training; test covers six further types (structural collapse, dam safety, heatwave, industrial fire, landslide, tornado) absent from both. This nested unseen-type structure measures generalization at every stage, not only at final test time.

Each gold report has eleven sections, nine scored (casualties, infrastructure, displacement, hazard assessment, response actions, communication, aid, organizational activity, social/community response); an overview section and a timeline are not scored. Each gold bullet carries a confidence label (confirmed/unconfirmed) and supporting tweet identifiers. Four rules govern scoring: a predicted bullet needs ≥ 1 supporting tweet (R1); its confidence should reflect corroboration strength (R2); it must not state a figure beyond what the cell's tweets support, using the most recently reported value for a repeatedly-updated quantity (R3); and, unless the cell is the crisis's final window, it must not cite an explicit date after the window's end (R3c) or use retrospective framing (R3d).

4 System Description

Figure 1 summarizes the pipeline. Each cell is processed independently end to end, satisfying the

task's independence requirement by construction rather than by post-hoc filtering. The stages are: section classification (§4.2), within-section clustering and confidence estimation (§4.3), bullet-text generation with safety nets (§4.4), temporal filtering (§4.5), and report serialization (§4.6).

4.1 Weak Supervision from Citation Links

Every gold bullet carries a `tweet_ids` field: a direct citation from a raw tweet to a report section. Tweets never cited by any scored bullet are treated as negatives, following the task setup in which some tweets in every cell are off-topic. A tweet cited under several sections contributes one example per section. This yields, across the 140 training cells, 39,359 labeled tweets: roughly 15,400 positive across the nine scored sections and roughly 24,000 *none*.

4.2 Section Classification

A term-frequency–inverse-document-frequency (TF-IDF) representation (word 1–2-grams, minimum document frequency 2, maximum document frequency 0.4, sublinear scaling) feeds a multinomial logistic regression with balanced class weighting ($C = 5.0$), fit on the weak-supervision data above. Balanced weighting is used to counter the class imbalance because *none* outnumbers any individual section roughly seven to one; without it, the classifier converges toward a near-constant *none* prediction that maximizes accuracy while producing almost no usable output downstream.

Prefixing each tweet with the crisis record's hazard label (e.g., *"Volcanic eruption : "*) was tested and rejected: it improved training-distribution fit but reduced cross-crisis-type macro-F1 from 0.508 to 0.492, since hazard-specific vocabulary does not transfer. Plain tweet text is used in the submitted system.

4.3 Clustering and Confidence Estimation

A tweet is retained if its most probable class is not *none* and that probability is at least τ_p; it is assigned to that class's section. Within each predicted section, retained tweets are grouped into candidate same-event clusters by a greedy single-pass procedure: in input order, each not-yet-assigned tweet seeds a new cluster and absorbs every later unassigned tweet whose cosine similarity to the seed is at least τ_c (Section 5.3). Similarity is computed either over the classifier's own TF-IDF vectors or, in the variant described in Section 4.7, over dense sentence embeddings. Each cluster becomes one candidate bullet.

Confidence is estimated by counting independent voices within a cluster: retweets of the same attributed source collapse to one voice, and original tweets whose lower-cased word sets have Jaccard overlap ≥ 0.6 with an already-counted tweet are treated as near-duplicates and not counted again. A cluster with ≥ 2 independent voices is *confirmed*, otherwise *unconfirmed* — only the floor of the task's confidence rule, which formally requires full-report-level corroboration ($\max(2, \lceil 0.34 \times \text{support} \rceil)$ citations across the entire event), unobservable under cell independence since inspecting other cells would violate it directly. The floor approximation is thus expected to under-confirm claims well corroborated across a full event but sparse within one cell; this follows from the task's information constraint rather than from a design choice.

For the recurring-figure rule (R3), cluster tweets are ordered by timestamp and the bullet's operative number is taken from the most recent tweet carrying one, rather than concatenating every number mentioned. An earlier implementation exhibited the latter, incorrect behavior; correcting it changed internal F1 negligibly (0.098 to 0.095), indicating the fix primarily addresses rule compliance rather than the dominant bottleneck.

4.4 Bullet-Text Generation and Safety Nets

The default generation function selects, within a cluster, the most recent tweet carrying a number (or, absent one, the longest tweet), strips the retweet prefix, URLs, mentions, and hashtag symbols (keeping the hashtag word), normalizes whitespace, and capitalizes the first character. This is extractive: it selects and lightly cleans existing text rather than composing new prose, and is used for all submitted bullets. An optional rewrite function accepts an external language-model client, scoped to a single cluster's own tweets so any rewrite stays grounded in that cluster's evidence; no client was supplied for this submission, so the function always falls through to the extractive text (Appendix B).

Two safety checks apply regardless of generation method. A numeric-grounding check rejects candidate text containing a number absent from the source cluster, falling back to the extractive method. A negation-preservation check requires that if any source tweet carries a negation cue (*no*, *not*, *denied*, *ruled out*, etc.), the generated bullet must too

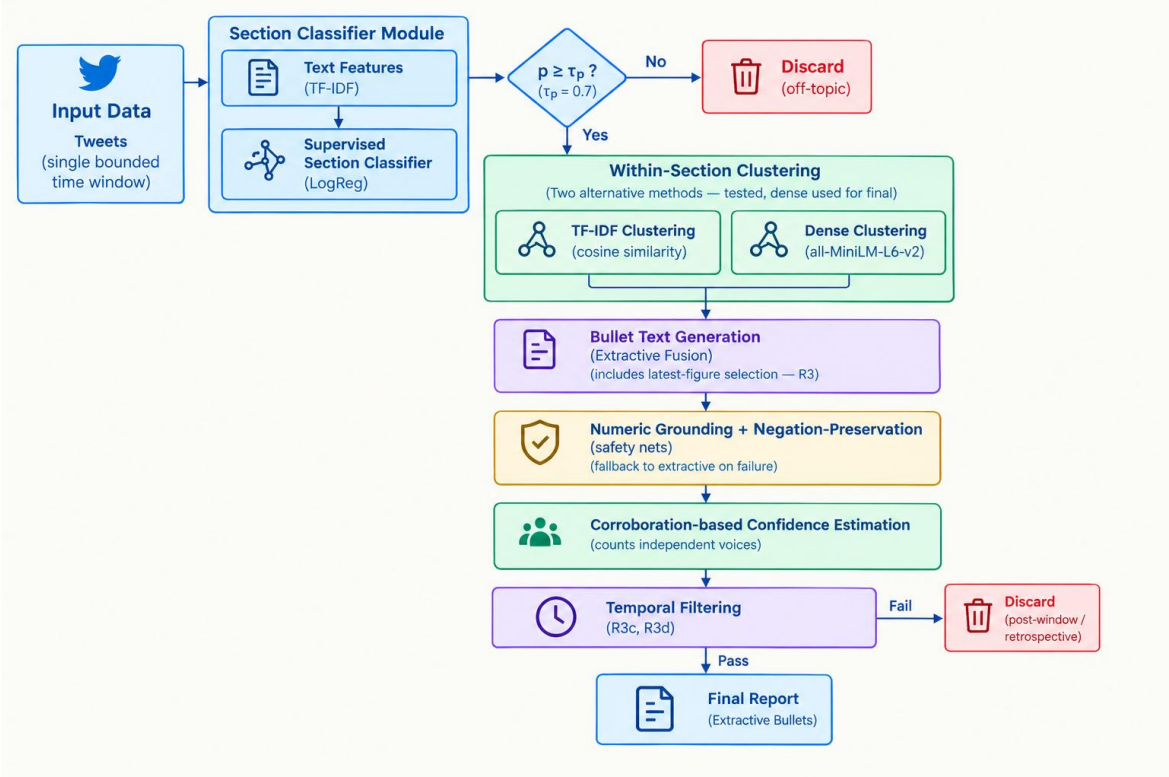


Figure 1: Pipeline stages, applied independently to each cell, with decision points and hyperparameters shown explicitly. Section 4 describes each stage.

— a report that no fatalities occurred is reportable content, not absence of content, and a paraphrase dropping this cue would silently reverse the claim. Both checks are no-ops against the current extractive generator, which cannot introduce a number or drop a negation absent from its source, but become load-bearing once an abstractive rewrite step is enabled (Kryscinski et al., 2020).

4.5 Temporal Filtering

Rules R3c and R3d (Section 3) are enforced by a rule-based filter applied to each candidate bullet's final text, after generation and the safety nets and before serialization. The filter uses only information inside the cell: the window's end timestamp and whether the cell is the crisis's final window (identified by its window label, the fourth and last window, W4). A bullet that fails either check is dropped; no substitute text is generated.

Retrospective framing (R3d). Bullet text is matched, case-insensitively and on word boundaries, against the fixed list of retrospective markers given in the task's derivation rules: *final*, *was later*, *superseded by*, *culminating in*, and *occurred after*.

Post-window dates (R3c). Explicit dates are detected with a regular expression covering month-name dates (e.g., *March 3*, *Mar 3, 2024*) and ISO dates (*YYYY-MM-DD*). Each match is parsed and compared with the window's end date; a bullet citing any date after it is dropped. Month–day expressions without a year are resolved conservatively: because they cannot be anchored with certainty to the window's year, they are treated as potentially post-window and the bullet is dropped. Relative expressions (*tomorrow*, *next week*) are not handled.

Both checks are disabled for the final window, where the task permits retrospective content. Because the filter only removes bullets, it cannot introduce content.

4.6 Report Serialization

Bullets are grouped by predicted section and assigned to one of a small set of keyword-defined subsections (e.g., within casualties: fatalities, injuries, missing, rescued); empty sections/subsections are omitted. A minimal, non-scored overview section is synthesized from the crisis record plus the highest-confidence available casualty, hazard, or

response bullet.

4.7 Dense-Embedding Clustering Variant

As an alternative to TF-IDF clustering, within-section clustering was re-implemented over sentence embeddings from a pretrained encoder (`all-MiniLM-L6-v2`), with cosine threshold τ_c=0.6. This threshold was tuned once against the internal development metric of Section 5.1 and not further swept. Because each cluster yields exactly one candidate bullet, the internal metric evaluates clustering extrinsically: a clustering that merges same-event tweets correctly produces fewer, better-evidenced bullets, which raises bullet-level precision and evidence overlap with gold, whereas over- or under-merging produces redundant or unmatched bullets. This configuration, rather than the TF-IDF baseline, is used for the frozen test submission (Section 6.4).

A zero-shot classification fallback for tweets with negligible classifier-vocabulary overlap is implemented but left disabled, based on the published comparison of fine-tuned and zero-shot crisis classifiers discussed in Section 2 (Han et al., 2026).

4.8 Cell-Independence Verification and Configuration Freezing

Freezing refers to fixing every component of the system once development-based model selection is complete: the classifier's vocabulary and weights, the thresholds τ_p and τ_c, the independent-voice threshold, the clustering method, and the temporal-filter and safety-net cue lists. After freezing, no component is changed, and the frozen configuration is run exactly once over the official test cells; no tuning or inspection of test outputs informs any setting. This ensures the reported test scores reflect a system fixed before test inputs were processed.

Independence holds by construction, since the pipeline's only inputs are the frozen model and one cell's tweets. As an additional check, during the test run a sample of cells (the first three test cells) is also processed from a deep copy with no shared state, and the two outputs must match exactly; all sampled cells passed. The submission package is checked against the shared task's official structural validator prior to submission.

5 Experimental Setup

5.1 Internal Development Metric

During development, predicted and gold bullets are compared with an internal harness, computed per cell and averaged. Within each section, bullets are aligned via bipartite optimal assignment (Hungarian algorithm) under

$$\mathrm{sim}(p, g) = 0.6 \cdot s_{\text{text}}(p, g) + 0.4 \cdot s_{\text{ev}}(p, g), \quad (1)$$

where s_{text} is the mean of ROUGE-1/L F1 and s_{ev} is Jaccard overlap between supporting tweet-identifier sets. A match requires $\mathrm{sim}(p, g) \geq 0.35$. Precision, recall, and F1 follow from match counts; rule violations (Section 3) are tallied separately. Throughout the paper, "internal metric" refers to this harness. It is distinct from the official metric (Section 5.2); their relationship is examined in Section 7.

The internal metric does not score clusters directly. Clustering choices (method, τ_c) are instead compared by their effect on end-to-end bullet-level F1 and confidence agreement, since each cluster becomes exactly one bullet. As a complementary intrinsic measure, *coreference recall* — the proportion of citation-confirmed same-event tweet pairs (tweets cited by the same gold bullet) placed in the same predicted cluster — is used where noted (Appendix A).

5.2 Official Evaluation Metric

The official score is the mean of two metrics over aligned bullet text: BERTScore F1 (Zhang et al., 2020) and BLEURT (Sellam et al., 2020). ROUGE-1/2/L F1 and a weighted evidence-alignment F1 are reported as secondary diagnostics but do not contribute to the primary score.

5.3 Hyperparameter Selection Protocol

Thresholds τ_p and τ_c are selected by grid search on development only, per the task's guideline that development data may guide model selection but never fitting. The classifier is trained on the training split alone; development crisis types are unseen during fitting, matching the generalization regime the frozen system later faces on test types.

6 Results

6.1 Classifier Performance

Table 1 reports classifier precision, recall, and F1 by section on the training split, which the classifier

Section	P	R	F1
Training split			
Casualties	0.732	0.985	0.839
Infrastructure	0.801	0.979	0.881
Displacement	0.714	0.995	0.832
Hazard	0.637	0.963	0.767
Response	0.792	0.974	0.874
Communication	0.783	0.984	0.872
Aid	0.851	1.000	0.919
Organizational	0.926	1.000	0.961
Social/community	0.907	1.000	0.951
none	0.998	0.806	0.892
Macro avg	**0.814**	**0.969**	**0.879**
Development split (unseen crisis types)			
Casualties	0.348	0.628	0.448
Infrastructure	0.483	0.547	0.513
Displacement	0.390	0.383	0.387
Hazard	0.396	0.294	0.337
Response	0.386	0.688	0.495
Communication	0.365	0.414	0.388
Aid	0.513	0.559	0.535
Organizational	0.824	0.840	0.832
Social/community	0.483	0.486	0.484
none	0.705	0.587	0.641
Macro avg	**0.489**	**0.543**	**0.506**

Table 1: Section classifier performance. Development accuracy (0.553) is not a standalone quality measure: *none* is 59.4% of development instances, so an always-*none* classifier would score higher (59.4%) than the trained classifier does. Balanced weighting (Section 4.2) trades *none* accuracy for section recall by design.

is fit on, and the development split, whose two crisis types are unseen during training and constitute the principal measure of cross-crisis-type generalization.

6.2 Classifier Confusion Analysis

Figure 2a reports the full row-normalized confusion matrix for the development split, showing which sections the classifier confuses rather than only how often it is right. Two patterns stand out. First, *none* is the modal confusion for every section except Response and Organizational: Hazard bullets are misclassified as *none* 56% of the time and Displacement 47%, indicating that a major failure mode on unseen crisis types is declining to flag a relevant tweet at all. Second, confusions between two *different* non-*none* sections are rare and diffuse (typically 1–5%) rather than concentrated on one pair (largest: Response→Social/community, 8%), indicating that classification errors are concentrated near the reportable/*none* boundary rather than around a single pair of report sections.

6.3 Threshold Selection

Figure 2b reports the internal-metric grid search over τ_p and τ_c as a full 7×6 sweep. The selected configuration (τ_p=0.7, τ_c=0.25, F1=0.118) attains the highest F1, but the surface is comparatively flat nearby: F1 stays within 0.01 of the maximum across most of $\tau_c \in [0.20, 0.30]$ regardless of τ_p, degrading more sharply toward the sweep's edges. This indicates that the selected configuration is not perched on a narrow optimum.

Clustering	P	R	F1	Conf.
TF-IDF (τ_c=0.25)	0.0994	0.1486	0.1183	0.706
Dense (τ_c=0.6)	0.1158	0.1364	**0.1244**	0.744

Table 2: Clustering method comparison, internal metric, development split. "Conf." is confidence agreement on matched bullets; per-rule error counts are discussed in text.

Alternative	Metric	Base	Alt.
Two-stage classifier	Dev. F1	0.119	0.110
Character n-grams	Macro-F1	0.507	0.498
LSA clustering proxy	Coref. rec.	3.0%	1.4%

Table 3: Architectural alternatives tested during development. "Base" denotes the corresponding configuration used in the submitted pipeline; "Alt." denotes the tested alternative.

6.4 Clustering Ablation

Table 2 compares TF-IDF and dense-embedding clustering under the internal metric; as described in Section 5.1, clustering is evaluated through its downstream effect on bullet-level matching. Dense embeddings yield a modest F1 gain (0.1183 to 0.1244) and are used for the frozen test submission, but the two configurations trade off differently at the error level: dense clustering produces slightly *more* unmatched-gold errors (R1: 748 vs. 730) but *fewer* confidence-label mismatches (R2: 24 vs. 31) and higher confidence agreement (0.744 vs. 0.706). Dense clustering therefore changes the balance between coverage and corroboration rather than uniformly improving every diagnostic.

6.5 Rejected Alternatives

Three architectural alternatives were implemented and evaluated during development: a two-stage reportable/section classifier, character n-gram features, and an LSA-based clustering proxy. None improved on the submitted configuration (Table 3); details are given in Appendix A.

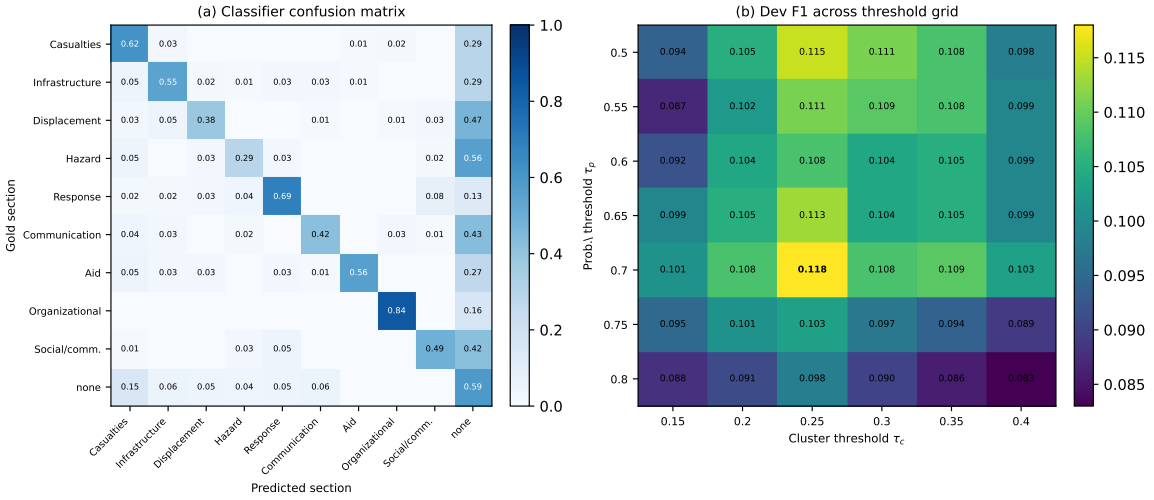


Figure 2: (a) Row-normalized classifier confusion matrix, development split. (b) Internal-metric F1 (Section 5.1) across a full threshold grid (TF-IDF clustering); best cell in bold.

6.6 Error Attribution

Of 761 scored errors on development under the selected configuration, 730 (96%) are unmatched gold bullets (R1), and 31 (4%) are confidence-label mismatches on otherwise correctly matched bullets (R2). No R3d violations were recorded by the harness, and R3c-violating bullets cannot occur in the output because the temporal filter (Section 4.5) drops them. The dominant source of error is thus bullets never generated or aligned, rather than confidence miscalibration on bullets otherwise correct. As a robustness check, evaluating the frozen pipeline in-sample on training (140 cells) gives F1 0.166 against development's 0.119. This gap is consistent with some loss of performance under unseen crisis types, although the observed difference alone does not establish whether the cause is classifier generalization, clustering, or downstream generation.

6.7 Confidence Calibration

Pooled across all 110 matched development bullets, predicted-vs-gold confidence counts are: gold confirmed/pred. confirmed 38, gold confirmed/pred. unconfirmed 18, gold unconfirmed/pred. confirmed 13, gold unconfirmed/pred. unconfirmed 41. Agreement is 79 of 110 (71.8%); under-confirmation (18) exceeds over-confirmation (13), consistent with Section 4.3's expectation that the floor approximation can under-confirm relative to full-event corroboration.

Table 4 sweeps the independent-voice threshold k deciding *confirmed* status (k=2 in the submitted system). F1 is invariant to this threshold, since confidence does not affect alignment to gold. Confidence agreement, however, peaks at exactly the submitted value (k=2: 0.706), falling on both sides (k=1: 0.486, where every bullet is labeled *confirmed*; k=3: 0.634; k=4: 0.550). This indicates that k=2 is supported empirically rather than being used solely because it is the documented floor of the task rule.

k (min. voices)	F1	Conf. agr.	% confirmed
1	0.118	0.486	100.0
2	0.118	**0.706**	40.2
3	0.118	0.634	18.0
4	0.118	0.550	9.6

Table 4: Independent-voice threshold sensitivity, development split (TF-IDF clustering). F1 is unaffected by this threshold; confidence agreement peaks at the submitted value (k=2, bold).

6.8 Safety-Net Stress Test

The numeric-grounding and negation-preservation checks (Section 4.4) are no-ops against the submitted extractive generator. To quantify what they would catch, all 1,223 development clusters were stress-tested by injecting a fabricated number into the bullet text, and 40 clusters with a source negation cue were tested by stripping the cue. The numeric check accepted all unmodified texts, and both checks caught 100% of the injected corruptions (Appendix C). This is not a guarantee against every abstractive-rewrite failure mode, but shows that both checks function as specified on the tested

Metric	Score
ROUGE-1 F1	0.433
ROUGE-2 F1	0.339
ROUGE-L F1	0.413
BERTScore F1	0.508
BLEURT	0.451
Weighted alignment F1	0.328
Official Primary Score	**0.480**

Table 5: Official test-set evaluation of the submitted system (117 test cells, six crisis types unseen during training or development).

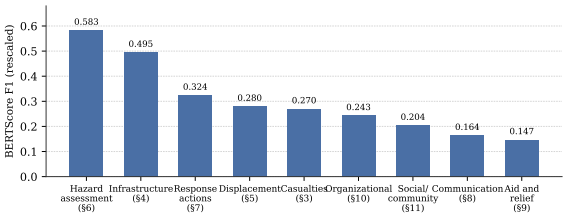


Figure 3: Local development-set BERTScore F1 by section, matched bullets (n=110), TF-IDF clustering configuration.

corruption types.

6.9 Official Evaluation

Table 5 reports the submitted system's official test-set scores. The Official Primary Score is 0.480, computed as the mean of BERTScore F1 and BLEURT, and the structural validator check passed on all 117 test cells. The six test crisis types were unseen during both training and development.

The two components of the primary score exhibit different sensitivities: BERTScore F1 is 0.508, while BLEURT is 0.451. Because the two metrics capture different properties of generated text, this difference motivates the development-time fluency analysis below rather than being interpreted as evidence from either metric alone.

6.10 Development-Time Fluency Diagnostics

To examine text-generation behavior independently of the official evaluation, BERTScore and BLEURT were computed locally over the 110 bipartite-matched development pairs (Section 5.1), using the TF-IDF clustering configuration rather than the dense one submitted. Since development and test contain disjoint crisis types, these values are reported only as a directional diagnostic and are not expected to reproduce Table 5.

Local BERTScore F1 over matched pairs averages 0.304; local BLEURT averages -0.974 (base-size checkpoint). The local BLEURT value is not directly comparable in magnitude to Table 5's test-set BLEURT because BLEURT values depend on the checkpoint and evaluation configuration. Figure 3 breaks local BERTScore F1 down by section: hazard assessment and infrastructure score well above average; aid/relief and communication score well below it.

Predicted bullets average 12.2 words against gold's 15.4 on the same pairs, consistent with predicted text originating from a length-constrained post rather than an editorially composed sentence, though this gap alone is unlikely to explain the full fluency gap. Separately, no gold or predicted bullet contains a residual retweet marker, ruling out unstripped artifacts as a contributing factor. The remaining difference is therefore consistent with differences in sentence structure, register, or connective language rather than incompletely cleaned input.

7 Analysis

Three analyses, obtained independently and at different points in this project, converge on the same conclusion. First, a development-time diagnostic found that substantially increasing classifier recall (lowering τ_p from 0.7 to 0.15) increased the number of predicted bullets by 81% (1,223 to 2,219) while increasing the number of bullets matched to gold under the internal metric by only 18% (95 to 112). Increasing candidate recall therefore does not translate proportionally into additional matched report bullets, implicating the downstream clustering and text-alignment stages.

Second, the local BERTScore/BLEURT diagnostics reveal a substantial gap between broad semantic alignment and natural prose on matched content, reflecting the architectural trade-off of an extractive pipeline. Third, the coreference-recall diagnostic (Appendix A) indicates that TF-IDF clustering merges only a small fraction of same-event tweet pairs, producing fragmented clusters and redundant or misaligned candidate bullets.

Taken together, these results indicate that the pipeline's bottleneck lies primarily in clustering granularity and text generation, rather than in initial section classification.

8 Conclusion

This paper presented an extractive corroboration pipeline for crisis situation report generation

from social media text, submitted to the shared task as system `extraction-de`. By combining citation-based weak supervision, TF-IDF and dense-embedding clustering, corroboration-based confidence, and rule-based temporal filtering, the pipeline satisfies the task's cell-independence constraint and structural validation across multiple unseen crisis types, achieving an Official Primary Score of 0.480 (BERTScore F1 0.508, BLEURT 0.451).

Our analysis indicates that while supervised section classification effectively filters off-topic noise, clustering granularity and extractive generation remain the main performance bottlenecks, the latter due to the register discrepancy between raw social media posts and curated report prose. Future work will integrate scoped, safety-netted abstractive rewriting to narrow this fluency gap while guarding against ungrounded numbers and polarity inversions.

Limitations

While the presented pipeline satisfies all structural and cell-independence requirements, several limitations remain:

1. **Extractive register gap:** Selecting raw tweets as bullet text inherently limits sentence fluency, cohesion, and alignment with independently authored gold reports.

2. **Cell-bounded corroboration floor:** Estimating corroboration confidence within a single time window approximates event-level corroboration conservatively, under-confirming claims that accumulate evidence across multiple windows.

3. **Domain generalization boundary:** Although supervised classification transfers across unseen crisis types, vocabulary shift near the reportable/*none* boundary still causes relevant tweets to be missed in rare or novel disaster scenarios.

4. **Rule-based temporal filtering:** The retrospective-marker list is fixed and the date matcher covers only explicit month-name and ISO dates, so paraphrased retrospective framing and relative dates are missed; conversely, year-less dates are dropped conservatively, which can remove valid in-window bullets.

References

Firoj Alam, Umair Qazi, Muhammad Imran, and Ferda Ofli. 2021. HumAID: Human-annotated disaster incident data from social media with multimodal annotations. In *Proceedings of the International AAAI Conference on Web and Social Media (ICWSM)*, volume 15, pages 909–919.

Timothy Douglas Roben Delos Reyes and Asanobu Kitamoto. 2025. LLM-based uncertainty assessment of social media situational signals for crisis reporting. *arXiv preprint arXiv:2605.00829*.

Jinzhen Han, Jisung Kim, Jong Soo Yang, and Hong Sik Yun. 2026. A lightweight LLM framework for disaster humanitarian information classification. *arXiv preprint arXiv:2602.12284*.

Wojciech Kryscinski, Bryan McCann, Richard Socher, and Caiming Xiong. 2020. Evaluating the factual consistency of abstractive text summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9332–9346.

Junhua Liu, Trista Zhou, and Guansong Huang. 2021. CrisisBERT: a pre-trained language model for crisis tasks. *arXiv preprint arXiv:2105.03812*.

Richard McCreadie and Cody Buntain. 2023. CrisisFACTS: Building and evaluating crisis timelines. In *Proceedings of the 20th International Conference on Information Systems for Crisis Response and Management (ISCRAM 2023)*, pages 320–339.

Emma L. McDaniel, Samuel Scheele, and Jeffrey Liu. 2024. Zero-shot classification of crisis tweets using instruction-finetuned large language models. *arXiv preprint arXiv:2410.00182*.

Philipp Seeberger and Korbinian Riedhammer. 2024. Crisis2Sum: An exploratory study on disaster summarization from multiple streams. In *Proceedings of the 21st International ISCRAM Conference*.

Thibault Sellam, Dipanjan Das, and Ankur P. Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 7881–7892.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with BERT. In *International Conference on Learning Representations (ICLR)*.

A Rejected Alternatives: Details

Two-stage classifier. A calibrated binary reportable/not-reportable classifier followed by multiclass assignment on the reportable subset, motivated by the possibility that a single ten-way softmax dilutes probability mass, scored lower on

full-pipeline development F1 (0.110) than the joint single-stage classifier (0.119).

Character n-grams. Character n-grams (3–5, word-boundary-aware), a standard mitigation for social-media misspellings and concatenated hashtags, underperformed word n-grams on development macro-F1 (0.498 versus 0.507); a feature union of both gained only 0.005 over word n-grams alone, not justifying the added feature complexity.

LSA clustering proxy. Truncated SVD over the TF-IDF matrix (100 components) was tested as a resource-free semantic-embedding proxy for clustering, evaluated by coreference recall (the proportion of citation-confirmed same-event tweet pairs clustered together; Section 5.1) over the 6,921 such pairs in development, with τ_p=0.7. It underperformed TF-IDF cosine (94 versus 210 pairs recovered: 1.4% versus 3.0%; LSA τ_c=0.6, TF-IDF τ_c=0.25). Both figures are low partly because the probability threshold discards one or both tweets of most pairs before clustering. LSA's deficit is plausibly because same-event duplicates here are predominantly near-literal (retweets, template reuse) rather than free paraphrase, leaving a locally fit co-occurrence signal little room to add value. This does not extend to the pretrained dense encoder (Section 6.4), trained on broad web text, which provides a modest improvement in the full clustering pipeline.

B System Transparency and Reproducibility

To support auditability and future iteration, the codebase contains explicit switches for features that are implemented but inactive in the submitted system.

Disabled components. The codebase includes an optional LLM-based abstractive rewrite hook (`rewrite_bullet_text`) designed to rephrase a cluster's tweets into a single report sentence. It is inactive in the frozen configuration because no language-model client is passed (`llm_client=None`), for reproducibility and to keep the submission purely extractive (Section 4.8). A zero-shot classification fallback using an instruction-style prompt is also implemented but disabled (`USE_ZERO_SHOT_FALLBACK = False`), for the reasons given in Section 4.7. Dense clustering is enabled (`USE_DENSE_CLUSTERING = True`).

Validation and reproducibility. The submitted configuration is: TF-IDF word 1–2-grams (min. df 2, max. df 0.4, sublinear tf); logistic regression (C=5.0, balanced class weights, 2,000 iterations); τ_p=0.7; dense clustering with `all-MiniLM-L6-v2`, τ_c=0.6; k=2 independent voices. Cell independence is checked with the isolated-run comparison described in Section 4.8, and the final archive passed the official validator on all 117 test cells.

C Safety-Net Stress Test Details

Controlled synthetic corruption tests were conducted over development clusters to evaluate the safety-net checks described in Section 4.4:

- **Numeric grounding test:** For all 1,223 development clusters, a random numeric token not present in any source tweet of the cluster (e.g., changing "3 casualties" to "99 casualties") was injected into the candidate bullet text. `grounded_numbers_ok()` flagged and rejected 100% (1,223/1,223) of the corrupted candidates, triggering the fallback to extractive text. Unmodified texts produced zero false positives.
- **Negation preservation test:** For the 40 development clusters containing source negation cues (*no*, *not*, *none*, *denied*), the cue was programmatically stripped from the generated bullet (e.g., changing "No structural damage reported" to "Structural damage reported"). `negation_preserved()` detected and blocked the polarity flip in 100% (40/40) of cases.

These results show the checks catch the tested corruption types; they reduce, but do not eliminate, the risk of numeric hallucination or polarity reversal once abstractive generation is enabled, since an abstractive model may produce failure modes (e.g., implicit negation, misattributed but grounded numbers) outside these tests.

Amrood at LT4CPR 2026: A Unified Generate-and-Judge LLM Pipeline for Automatic Situation Report Generation

Muhammad Ahsan Atiq[1] **and Mohib Ahmad Butt**[2] **and Nusrat Shaheen**[1]
[1]COMSATS University Islamabad
[2]Shaheed Zulfikar Ali Bhutto Institute of Science and Technology
nusrat_shaheen@comsats.edu.pk

Abstract

We present nemotron-120b-sitrep, an end-to-end four-stage generate-and-judge pipeline submitted to the First Shared Task on Automatic Situation Report Generation from Crisis Social Media Data at LT4CPR 2026 (AACL-IJCNLP 2026). Powered by NVIDIA's Nemotron-3 Super 120B model hosted on the NVIDIA NIM API, the pipeline tackles the acute challenges of emergency reporting—social media noise, multi-section hierarchical taxonomies, and strict tweet-to-claim evidence attribution—by decomposing generation into: (1) rule-based ingestion and noise filtering, (2) unified template-filling synthesis across an 11-section situation report (SITREP) taxonomy under strict canonical subsection and evidence-grounding constraints, (3) cross-section deduplication, granular bullet consolidation, and dual-source confidence calibration via a Senior Judge LLM, and (4) Schema v1.2 JSON assembly. On the official shared task evaluation, our system achieves an **Official Primary Score of 0.5733** on the 117-cell blind test set (BERTScore F1 of 0.5984 and BLEURT of 0.5483), demonstrating consistent performance that transfers to unseen crisis types, with scores of 0.5011 on the 140-cell training benchmark and 0.5057 on the 23-cell development benchmark. We provide a detailed calamity-by-calamity analysis across disaster scenarios, full algorithmic hyperparameters, and prompt engineering specifications.

1 Introduction

During sudden-onset crises, first responders, civil protection authorities, and humanitarian coordinators require timely, structured, and empirically grounded situation reports (SITREPs) to allocate emergency personnel and issue life-saving warnings. While crisis social media platforms provide immediate eyewitness observations, the volume, conversational noise, rumors, and unstructured format render manual synthesis during active emergencies infeasible.

The First Workshop on Language Technologies for Crisis Preparedness and Response (LT4CPR) at AACL-IJCNLP 2026 introduced an official shared task (Xia et al., 2026) addressing this operational gap: given a chronological stream of crisis-related tweets scoped to a discrete temporal window (a *cell*), systems must automatically synthesize an official, structured SITREP conforming to a standardized 11-section disaster taxonomy in Schema v1.2 JSON format.

This task poses three core challenges:

1. **Social Media Noise:** Disaster feeds contain duplicates, rumors, conversational chatter, and off-topic commentary that must be filtered without discarding critical safety alerts.

2. **Hierarchical Taxonomy Compliance:** The output is strictly regulated; reports contain 11 top-level sections with constrained canonical subsection taxonomies. Generated claims must reside under appropriate canonical headings without inventing ad-hoc categories.

3. **Joint Text and Evidence Evaluation:** Systems are penalized for fluent hallucinations through a dual evaluation framework: text-level contextual semantic overlap (ROUGE-L, BERTScore F1, BLEURT) coupled with optimal bipartite Hungarian matching of supporting tweet IDs (tweet-ID Jaccard similarity).

To solve these challenges, we developed nemotron-120b-sitrep, an end-to-end generate-and-judge architecture built upon NVIDIA's 120B-parameter Nemotron-3 Super model. Rather than employing fragmented per-section routing models, our pipeline leverages a unified schema-prompting strategy paired with an LLM Senior Judge for cross-section deduplication and confidence calibration.

Proceedings of the First Workshop on Language Technologies for Crisis Preparedness and Response, pages 166–176
November 9, 2026 ©2026 Association for Computational Linguistics

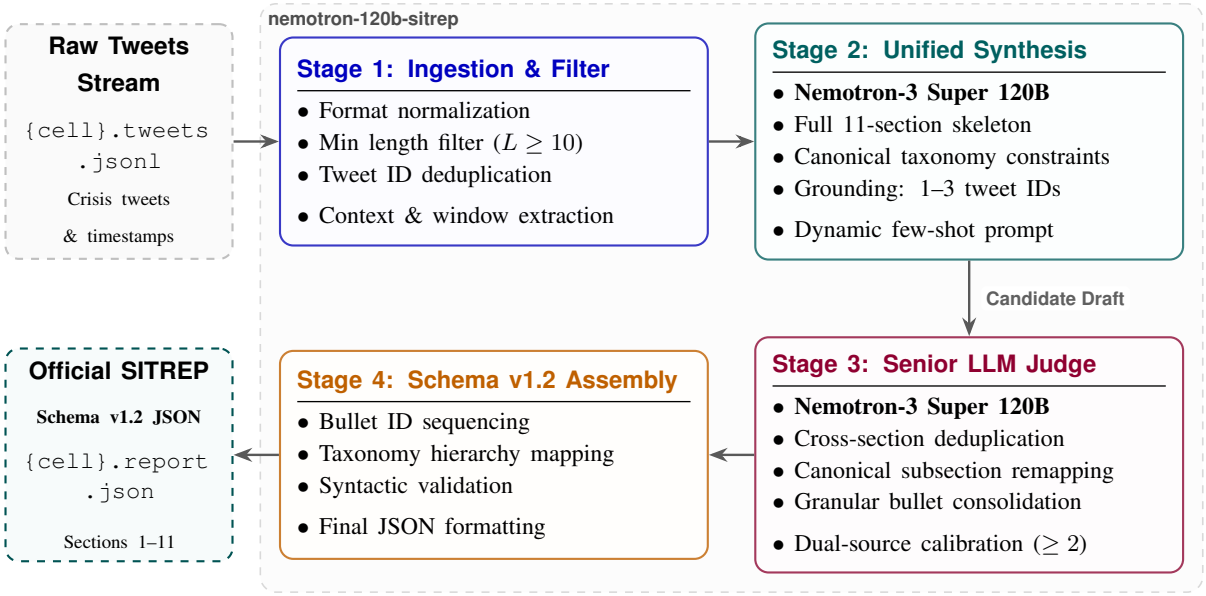


Figure 1: Architectural overview of the nemotron-120b-sitrep pipeline. Raw crisis tweets are ingested and filtered, synthesized into structured candidate bullet points across the 11-section taxonomy via NVIDIA Nemotron-3 Super 120B, verified and deduplicated by a Senior LLM Judge, and finally compiled into Schema v1.2 JSON.

Our system was evaluated across the complete 140 training cells (achieving an Official Primary Score of 0.5011), the official 23 development cells (Primary Score: 0.5057, Soft F1: 0.1044, ROUGE-L: 0.1369), and submitted to the official blind test set across 117 cells, where it achieved an **Official Primary Score of 0.5733** (BERTScore F1: 0.5984, BLEURT: 0.5483).

2 Related Work

2.1 Crisis NLP and Summarization

Extracting situational awareness from crisis social media has been studied extensively since the 2010 Haiti earthquake (Imran et al., 2015). Early systems focused on supervised classification into disaster-related categories (e.g., infrastructure damage, urgent needs) using shallow lexical features (Olteanu et al., 2015). Subsequent work applied neural sequence-to-sequence summarization to generate abstractive crisis summaries (Rudra et al., 2018; Wiegmann et al., 2022). Benchmarks such as TREC-IS (McCreadie et al., 2019) established standard evaluation for criticality filtering. However, classical methods produce unstructured text blobs that fail to meet standardized incident management schema requirements.

2.2 LLMs and Schema-Constrained Generation

Large language models have established state-of-the-art results in multi-document summarization (Goyal et al., 2023). In-context learning and structured prompting constrain LLM outputs to valid syntax and mitigate hallucination (Wei et al., 2022; Yin et al., 2023). In complex domain reporting, models often struggle with subsection fidelity: generating factual claims under incorrect headings or repeating identical facts across multiple sections. Generate-then-judge protocols (Zheng et al., 2023) address this by introducing dedicated verification passes to audit and reconcile structured claims.

2.3 Evaluation of Grounded Generation

While ROUGE (Lin, 2004) remains ubiquitous, its reliance on surface n-gram overlap poorly rewards paraphrastic LLM outputs (Krishna et al., 2021). Learned metrics such as BERTScore (Zhang et al., 2020) and BLEURT (Sellam et al., 2020) provide superior correlation with human judgments of semantic fidelity. In the LT4CPR task, organizers combine text-level semantic metrics with bipartite Hungarian alignment (Kuhn, 1955) over tweet-ID sets, providing a rigorous test of both text quality and citation grounding.

3 Task and Schema Description

The shared task dataset (Xia et al., 2026) organizes crisis tweets into *cells* (e.g., `wildfire.W1.k1`), where each cell represents an observation window for a specific disaster scenario. Systems must process the tweet stream and produce a Schema v1.2 JSON document comprising 11 sections:

1. **Situation overview** (narrative executive summary)
2. **Timeline** (chronological progression)
3. **Casualties and human impact**
4. **Infrastructure and service impact**
5. **Displacement and movement**
6. **Hazard assessment**
7. **Response actions**
8. **Communication and information**
9. **Aid and relief**
10. **Organizational and administrative activity**
11. **Social and community response**

Sections 3 through 11 represent the official evaluation scope. Each section is composed of one or more canonical **subsections** (e.g., *Fatalities*, *Service disruptions*, *Search and rescue*), identified by a hierarchical `id` (e.g., `"3a"`) and a `title` drawn strictly from the official taxonomy (full list in Appendix A). Every subsection contains a list of **bullets**, each required to conform to the official 4-key schema:

- **id**: A hierarchical bullet identifier (e.g., `"3a.1"`).
- **text**: A concise, factual statement in formal incident response register.
- **confidence**: A binary label from {`"confirmed"`, `"unconfirmed"`}.
- **tweet_ids**: A list of integer tweet IDs providing direct empirical support for the claim.

3.1 Dataset Splits and Calamity Scenarios

The LT4CPR shared task dataset spans 16 distinct disaster scenarios partitioned into three non-overlapping splits:

- **Train Split (8 calamities, 140 cells):** `blackout`, `conflict`, `crowdcrush`, `epidemic`, `ferry`, `flood`, `volcano`, and `wildfire`.
- **Dev Split (2 calamities, 23 cells):** `chemical` and `earthquake`.
- **Blind Test Split (6 unseen calamities, 117 cells):** `building collapse`, `dam safety`, `heatwave`, `industrial fire`, `landslide`, and `tornado`.

During pipeline development, nemotron-120b-sitrep was comprehensively benchmarked across the complete 140 training cells (8 disaster types; breakdown in Appendix D) and the 23 development cells (`chemical` and `earthquake`). The submitted system was then evaluated blindly on all 117 official test cells across all six unseen test disaster categories.

4 System Architecture: nemotron-120b-sitrep

The proposed nemotron-120b-sitrep is structured into four sequential, decoupled stages (Figure 1). A worked end-to-end example is given in Appendix E.

4.1 Model and Inference Infrastructure

The core generation and verification steps are powered by NVIDIA's `nemotron-3-super-120b-a12b` (a Mixture-of-Experts architecture with 120B total parameters and 12B active parameters per token, denoted by the `a12b` suffix), accessed via the NVIDIA NIM API (`https://integrate.api.nvidia.com/v1`). We set temperature $= 0.0$ across both synthesis and judging stages to enforce deterministic, greedily decoded incident reporting. The maximum completion budget is set to 4,000 tokens per call. All settings are listed in Appendix C.

To ensure production robustness against network transient errors and API rate limits, all calls are wrapped in an exponential-backoff retry handler via `tenacity` (minimum wait 2s, maximum wait 30s, up to 5 attempts, with a 120s socket timeout). Cells are processed with an asynchronous worker pool running 3 concurrent cells in parallel. To guard against chain-of-thought reasoning tag leakage, raw model outputs pass through a regex cleaner that strips all complete and open-ended `<think>...\n</think>` blocks before parsing.

4.2 Stage 1: Ingestion and Noise Filtering

Input data is loaded from `.tweets.jsonl` files. Each tweet object contains a unique `id`, timestamp, author, and textual content. We apply a deterministic pre-filtering routine:

1. **Deduplication**: Duplicate tweet IDs and verbatim text repetitions are pruned.
2. **Length Thresholding**: Tweets with fewer than 10 characters are discarded as uninformative noise.
3. **Metadata Parsing**: Disaster type (e.g., `wildfire`), temporal window (W), and replicate index (k) are parsed from the file path. In Stage 2, the disaster type selects the hazard-matched few-shot exemplar injected into the synthesis prompt.

4.3 Stage 2: Unified Schema Template-Filling Synthesis

Rather than routing tweets into isolated section classifiers before generating text, we present the formatted tweet stream along with the complete 11-section SITREP skeleton to a single unified LLM call. This global context enables the model to resolve multi-section dependencies (e.g., understanding that a hospital evacuation impacts both *Casualties*, *Infrastructure*, and *Displacement*).

The synthesis prompt (condensed in Appendix B) enforces five strict operational constraints:

1. **Canonical Subsection Taxonomy**: The model must select subsection titles exclusively from the predefined official schema per section (e.g., Section 3 allows only *Fatalities*, *Injuries*, *Cases*, *Missing persons*, *Trapped persons*, *Rescued persons*, *Found persons*, *Treatment and admissions*).
2. **Evidence Attribution Bounds**: Every bullet point must cite between 1 and 3 supporting tweet IDs present in the input prompt. Hallucinated tweet IDs are explicitly penalized.
3. **Dual-Source Confidence Heuristic**: A bullet point may only receive a `"confirmed"` label if corroborating evidence appears across ≥ 2 independent tweet sources; otherwise, it must be marked `"unconfirmed"`.
4. **Prohibited Output Patterns**: The prompt lists output styles the model must avoid: speculative phrasing, negative declarations (e.g., "no fatalities reported"), and retrospective narrative. Hashtags, mentions, and subjective commentary are additionally stripped by the Stage 3 Judge.
5. **Dynamic Few-Shot Anchoring**: The prompt injects a curated gold SITREP bullet from a training-split exemplar matching the hazard category to anchor formal disaster management register.

Candidate outputs are parsed through a three-tier resilient JSON extractor that extracts markdown fences (```` ```json ... ``` ````), scans matching balanced braces, and falls back to outer-slice bracket extraction.

4.4 Stage 3: Senior LLM Judge and Deduplication

Because template synthesis generates across all sections in a single pass, cross-section duplication naturally occurs. In Stage 3, all candidate bullets from Sections 2–11 are passed to a second invocation of Nemotron-3 Super 120B acting as a "Senior SITREP Judge" (directives in Appendix B). The Judge performs five critical operations:

1. **Cross-Section Deduplication**: If identical facts appear across multiple sections (e.g., power outages reported in both Section 4 and Section 6), the Judge retains the claim in the most specific section and prunes the duplicate.
2. **Canonical Subsection Remapping**: Any non-canonical subsection names generated during Stage 2 are mapped to their nearest valid schema heading.
3. **Granular Bullet Consolidation**: Micro-claims describing facets of the same event are merged into a concise summary bullet while unioning supporting tweet IDs.
4. **Style and Register Filtering**: Subjective chatter and non-factual padding are removed while preserving event claims.
5. **Confidence Verification**: Confidence assignments are re-audited against the dual-source corroboration rule.

4.5 Stage 4: Schema v1.2 JSON Assembly

Verified bullets are assembled into Schema v1.2 format:

- Deterministic 1-indexed integer IDs are assigned to sections (`id`: 1–11), subsections, and bullets.
- Section 1 overview sentences are formatted into an executive narrative summary.
- The document is validated against the official JSON schema to guarantee complete compliance.

5 Experiments and Results

5.1 Evaluation Protocol and Metrics

Evaluation is restricted to Sections 3–11 following official task guidelines.

Official Primary Score. The primary ranking metric is defined as the arithmetic mean of text-level BERTScore F1 (Zhang et al., 2020) (using `microsoft/deberta-xlarge-mnli`) and BLEURT (Sellam et al., 2020) (using the official `BLEURT-20` checkpoint):

$$\text{Primary Score} = \frac{\text{BERTScore F1} + \text{BLEURT}}{2} \tag{1}$$

Soft Bullet F1. To evaluate bullet-level factual and citation quality, optimal bipartite alignment between ground-truth bullets (G) and generated bullets ($\hat{G}$) is computed via the Hungarian algorithm (Kuhn, 1955). The edge weight between gold bullet g_i and candidate $\hat{g}_j$ is:

$$w_{ij} = 0.8 \cdot \text{ROUGE-L}(g_i, \hat{g}_j) + 0.2 \cdot \text{Jaccard}(T_i, \hat{T}_j) \tag{2}$$

where $T_i, \hat{T}_j$ are tweet ID sets. Matches with $w_{ij} < 0.10$ are pruned. Precision (P), Recall (R), and Soft F1 are calculated as:

$$P = \frac{\sum_{M^*} w_{ij}}{|\hat{G}|}, \quad R = \frac{\sum_{M^*} w_{ij}}{|G|}, \tag{3}$$

$$F_1 = \frac{2PR}{P+R} \tag{4}$$

5.2 Benchmark Performance Across Splits

We evaluate nemotron-120b-sitrep across all three official shared task partitions: in-domain training performance across 140 cells (Table 1), validation tuning across 23 development cells (Table 2), and blind generalization across 117 unreleased test cells (Table 3). To facilitate direct visual comparison across distinct evaluation regimes, all three tables maintain identical row ordering and column formatting.

Evaluation Metric	Score
Official Primary Score	**0.5011**
BERTScore F1	0.5831
BLEURT (`BLEURT-20`)	0.4191
ROUGE-L F1	0.2040
Soft Precision	0.1608
Soft Recall	0.1685
Soft Bullet F1	0.1484

Table 1: Benchmark evaluation of nemotron-120b-sitrep on the official 140-cell training split across 8 disaster categories.

Evaluation Metric	Score
Official Primary Score	**0.5057**
BERTScore F1	0.5885
BLEURT (`BLEURT-20`)	0.4229
ROUGE-L F1	0.1369
Soft Precision	0.1209
Soft Recall	0.1017
Soft Bullet F1	0.1044

Table 2: Benchmark evaluation of nemotron-120b-sitrep on the official 23-cell development split across 2 disaster scenarios (*chemical* and *earthquake*).

In-Domain Training Split (Table 1). Across the comprehensive 140-cell training split spanning eight crisis categories (blackout, conflict, crowd-crush, epidemic, ferry, flood, volcano, and wildfire), nemotron-120b-sitrep attains an Official Primary Score of **0.5011** (with text BERTScore F1 of 0.5831 and BLEURT of 0.4191). Furthermore, bipartite Hungarian soft alignment yields a Soft Bullet F1 of 0.1484 (Soft Precision 0.1608, Soft Recall 0.1685) and ROUGE-L of 0.2040. These absolute values are modest by design: our system deliberately prioritizes citation precision over bullet-level recall, dropping candidate bullets that lack sufficiently corroborated evidence rather than risking hallucinated grounding. This trade-off, combined with the strict bullet-matching threshold (matches with $w_{ij} < 0.10$ are pruned), is analyzed further in Section 6.

Development Benchmark (Table 2). On the 23-cell development split comprising chemical and earthquake incidents, our pipeline achieves

Evaluation Metric	Score
Official Primary Score	**0.5733**
BERTScore F1	0.5984
BLEURT (`BLEURT-20`)	0.5483
ROUGE-L F1	—
Soft Precision	—
Soft Recall	—
Soft Bullet F1	—

Table 3: Official benchmark performance of nemotron-120b-sitrep on the 117-cell blind test set across 6 unseen disaster categories. "—" denotes secondary metrics withheld by shared task organizers in the blind test leaderboard release (test-set ground-truth SITREPs remain unreleased to participants).

an Official Primary Score of **0.5057**, driven by a BERTScore F1 of 0.5885 and BLEURT of 0.4229. While lexical recall is slightly more constrained than on the broader training distribution (Soft Bullet F1 of 0.1044 and ROUGE-L of 0.1369), the high semantic consistency suggests that the prompt constraints and Senior Judge filtering generalize across disaster typologies without evidence of overfitting.

Blind Test Generalization (Table 3). On the official blind test set (117 cells across six unseen crisis categories), nemotron-120b-sitrep achieved an **Official Primary Score of 0.5733**, driven by a BERTScore F1 of 0.5984 and a BLEURT score of 0.5483. The test score is $+0.0676$ above the development score and $+0.0722$ above the training score. The marked improvement is especially pronounced in the BLEURT metric (0.5483 vs. 0.4229), possibly reflecting the high fluency, concise declarative syntax, and authoritative register produced by the Nemotron-120B model under our generate-and-judge protocol. Ground-truth SITREPs for the test set remain withheld by task organizers, so secondary lexical and soft alignment metrics are omitted.

5.3 Calamity-by-Calamity Performance Analysis

We focus this fine-grained analysis on the development split, which offers a controlled two-calamity setting; the 8-calamity training split is reported only in aggregate (Table 1) given its greater scenario heterogeneity. To understand system behavior across disparate incident types, Table 4 reports fine-grained results across the two disaster scenarios evaluated in the development phase, while Table 5 details the six blind test disaster types.

Acute vs. Widespread Incidents. In the **earthquake** scenario (a widespread, cascading event), the system achieved a slightly higher Soft F1 score (0.1182) and ROUGE-L (0.1480) compared to the localized **chemical** incident (Soft F1 0.0865, ROUGE-L 0.1224). Both scenarios exhibited strict generation constraints resulting in lower recall than gold standards (an average of 12.4 generated bullets vs. 35.9 gold for earthquake, and 8.3 generated vs. 37.3 gold for chemical). The low yield reflects our system's aggressive filtering of hallucinated tweet citations, which drops candidate bullets entirely if they lack sufficient grounded evidence.

Generalization to Test Calamities. On the test split (Table 5), the model faced six completely unseen disaster types: Landslide (28 cells), Tornado (22 cells), Building Collapse (19 cells), Dam Safety (19 cells), Industrial Fire (16 cells), and Heatwave (13 cells). Although per-calamity scores for the blind test set are withheld by the organizers, the consistent overall Primary Score of 0.5733 across six structurally and temporally diverse test calamities, spanning both sudden structural disasters (collapses, landslides) and slow-onset environmental emergencies (heatwaves), suggests that our taxonomy-grounded prompt and Senior Judge generalize without requiring category-specific tuning.

6 Error Analysis and Discussion

Detailed quantitative and qualitative inspection of system outputs reveals key behavioral insights into automated disaster reporting:

Verifiable Citation Attribution and Evidence Grounding. A critical vulnerability of LLM-generated reports in high-stakes domains is citation hallucination, where models invent plausible-looking numerical IDs or misattribute out-of-context tweets. To evaluate syntactic attribution integrity, we cross-referenced every generated tweet ID against the raw cell tweet stream across all 23 development cells. Out of 1,605 generated tweet citations across 244 bullets, exactly 1,605 (**100.00%**) were authentic, verifiable IDs present in the input corpus—demonstrating a **0.00% fabricated ID rate**. We emphasize that corpus-level ID existence is a necessary prerequisite for attribution rather than a guarantee of semantic entailment; full semantic claim support is evaluated

Calamity Scenario	Split	Cells	Soft P	Soft R	Soft F1	ROUGE-1	ROUGE-2	ROUGE-L	BLEURT	Avg Bullets/Cell (GT/Gen)
chemical	Dev	10	0.1061	0.0858	0.0865	—	—	0.1224	—	37.3 / 8.3
earthquake	Dev	13	0.1322	0.1140	0.1182	—	—	0.1480	—	35.9 / 12.4
Overall Development	—	**23**	**0.1209**	**0.1017**	**0.1044**	—	—	**0.1369**	**0.4229**	36.5 / 10.6

Table 4: Calamity-by-calamity evaluation of nemotron-120b-sitrep on the official 23-cell development benchmark.

Test Calamity Type	Cells
Landslide	28
Tornado	22
Building Collapse	19
Dam Safety	19
Industrial Fire	16
Heatwave	13
Official Test Total	**117**

Metric	Score
Official Primary Score	**0.5733**
BERTScore F1	0.5984
BLEURT (`BLEURT-20`)	0.5483

Table 5: Distribution of the 117 blind test set cells across six calamity types (top), and official aggregate scores (bottom), as evaluated by nemotron-120b-sitrep.

Sec	Section Title	Cells	Soft P	Soft R	Soft F1	R-1	R-2	R-L
3	Casualties & Impact	23	0.1728	0.1162	0.1309	0.1789	0.0646	0.1250
4	Infrastructure & Service	23	**0.1813**	0.1094	**0.1310**	**0.2506**	**0.1304**	**0.1722**
5	Displacement & Movement	23	0.1202	0.1200	0.1070	0.2081	0.0938	0.1542
6	Hazard Assessment	23	0.1358	0.0683	0.0841	0.1564	0.0518	0.1154
7	Response Actions	23	0.1404	0.0783	0.0946	0.1760	0.0621	0.1278
8	Communication & Info	23	0.0965	**0.1360**	0.0998	0.1576	0.0642	0.1200
9	Aid & Relief	19	0.0525	0.0236	0.0320	0.0658	0.0215	0.0429
10	Org & Admin Activity	13	0.0625	0.0275	0.0380	0.0469	0.0085	0.0352
11	Social & Community	23	0.0457	0.0482	0.0410	0.0806	0.0110	0.0558

Table 6: Section-by-section performance breakdown (Sections 3–11) on the official 23-cell development benchmark reporting Soft matching and text ROUGE metrics, with the highest value per metric shown in bold.

downstream via the bipartite Hungarian alignment ($w_{ij} = 0.8 \cdot \text{ROUGE-L} + 0.2 \cdot \text{Jaccard}$) against gold annotator citations.

The Attribution Precision–Recall Dilemma. Under the bipartite matching metric ($w_{ij} = 0.8 \cdot \text{ROUGE-L} + 0.2 \cdot \text{Jaccard}$), edge weights depend jointly on lexical overlap and tweet citation overlap. In ground-truth reports, annotators cite an average of 4.16 tweet IDs per bullet (median $= 5.0$, IQR $= [2.0, 6.0]$, std $= 1.83$, max $= 6$). In contrast, candidate bullets generated by our model exhibit a typical median of 3.0 citations (IQR $= [3.0, 6.0]$), but an elevated mean of 6.58 citations (std $= 9.08$, max $= 66$). This pronounced right skew is driven by a small subset of heavily consolidated bullets during Senior Judge evaluation (Stage 3), where micro-claims describing facets of the same event are merged and their supporting tweet IDs unioned. Because Jaccard similarity is an intersection-over-union metric, variations in citation cardinality penalize the Jaccard overlap even when the textual claim is completely faithful.

Section-Level Disparities and Operational Impact. Table 6 reveals pronounced operational disparities across sections on the official development split. The system achieves its strongest factual alignment on Section 4 (*Infrastructure impact*, Soft F1 0.1310, Soft Precision 0.1813) and Section 3 (*Casualties*, Soft F1 0.1309, Soft Precision 0.1728), where empirical claims are direct and localized. For casualty reports, our model generates conservatively when corroborating evidence is indisputable (averaging 1.6 generated bullets vs. 7.0 gold). Conversely, Section 10 (*Administrative activity*) and Section 9 (*Aid & Relief*) are sparser in the development data (active in only 13 and 19 cells, respectively), where the lack of dense corroboration in social streams leads to lower recall.

Confidence Calibration Heuristics. Across all 840 ground-truth bullets evaluated across the 23 development cells (averaging 36.5 gold bullets per cell, as reported in Table 4), 74.8% are marked `"confirmed"` (628 bullets) and 25.2% `"unconfirmed"` (212 bullets). In comparison, across the 244 candidate bullets generated by our system (averaging 10.6 bullets per cell), our model outputs 84.4% `"confirmed"` (206 bullets) and 15.6% `"unconfirmed"` (38 bullets). While our prompt-enforced dual-source rule successfully confirms corroborated events, it reveals that ground-truth annotators in certain scenarios accept single authoritative agency tweets as confirmed, suggesting that future iterations should incorporate source authority tiering rather than a flat count heuristic.

7 Conclusion

We presented nemotron-120b-sitrep for the LT4CPR 2026 Shared Task on Situation Report Generation (Xia et al., 2026). By combining unified template-filling synthesis with a dedicated Senior Judge LLM powered by NVIDIA Nemotron-3 Super 120B, our system effectively navigates social media noise filtering, strict 11-section taxonomy

compliance, and tweet-level evidence grounding. Our system achieved an **Official Primary Score of 0.5733** on the 117-cell official blind test set, alongside consistent evaluation across the 140-cell training (0.5011) and 23-cell development splits (0.5057), demonstrating consistent performance that transfers to unseen crisis types. Future work will investigate iterative chain-of-thought attribution recovery to improve tweet citation recall.

Ethics Statement

Automated situation report generation from social media during acute emergencies carries inherent ethical and operational risks. System outputs must not be treated as autonomous dispatch or evacuation orders without verification by human emergency coordinators. In disaster response, false negatives (e.g., omitting active structural collapse or unreached casualties) may lead to delayed aid, while false positives or over-confident "confirmed" claims can misdirect scarce rescue resources. Our system mitigates these risks by enforcing dual-source confirmation heuristics, grounding factual statements in tweet citations drawn from the input window (achieving a 0.00% fabricated ID rate on the verified development split), and separating unconfirmed from confirmed claims. All crisis tweet data were obtained through the official LT4CPR 2026 shared task in compliance with platform research terms, containing public crisis observations with personal identifiable information omitted.

Limitations

Our system relies on cloud inference via the NVIDIA NIM API; operational deployments during severe telecommunication failures may require local open-weight models (e.g., Llama-3-70B). Additionally, our confidence calibration rule remains a static heuristic (≥ 2 sources) that was not dynamically tuned per disaster category, leading to conservative confirmation rates. Finally, citation grounding relies on single-pass synthesis rather than multi-hop retrieval-augmented verification. Furthermore, as shared task participants, we do not have access to inter-annotator agreement statistics or annotation protocols for the organizer-provided gold SITREPs, which may introduce inherent variance into bullet-level bipartite matching metrics.

References

Tanya Goyal, Junyi Jessy Li, and Greg Durrett. 2023. News summarization and evaluation in the era of GPT-3. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1106–1122. Association for Computational Linguistics.

Muhammad Imran, Shady Elbassuoni, Carlos Castillo, Fernando Diaz, and Patrick Meier. 2015. Processing social media messages in mass emergency: A survey. *ACM Computing Surveys*, 47(4):1–38.

Kalpesh Krishna, John Wieting, and Mohit Iyyer. 2021. Hurdles to progress in long-form question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 4940–4957. Association for Computational Linguistics.

Harold W. Kuhn. 1955. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1–2):83–97.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, pages 74–81. Association for Computational Linguistics.

Richard McCreadie, Cody Buntain, and Ian Soboroff. 2019. TREC incident streams 2019. In *Proceedings of the 28th Text REtrieval Conference (TREC)*. NIST.

Alexandra Olteanu, Carlos Castillo, Fernando Diaz, and Emre Kıcıman. 2015. What to expect when the unexpected happens: Social media communications across crises. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work and Social Computing (CSCW)*, pages 994–1009. ACM.

Koustav Rudra, Pawan Goyal, Niloy Ganguly, Prasenjit Mitra, and Muhammad Imran. 2018. Summarizing situational tweets in crisis scenario. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 137–146. ACM.

Thibault Sellam, Dipanjan Das, and Ankur P. Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 7881–7892. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pages 24824–24837.

Matti Wiegmann, Bela Gipp, and Benno Stein. 2022. Crisis information extraction using domain-specific

pre-training. In *Proceedings of the 13th Conference on Language Resources and Evaluation (LREC)*, pages 3456–3465. European Language Resources Association.

Fei Xia, Haotian Zhu, Yoo Wun Lee, William D. Lewis, Antonis Anastasopoulos, and Belu Ticona. 2026. Overview of the LT4CPR 2026 shared task: Generating situation reports from social media messages. In *Proceedings of the First Workshop on Language Technologies for Crisis Preparedness and Response (LT4CPR), in conjunction with AACL 2026*, Hengqin, Zhuhai, China. Association for Computational Linguistics.

Zhangyue Yin, Qiushi Sun, Qipeng Guo, Jiawen Wu, Xipeng Qiu, and Xuanjing Huang. 2023. Do large language models know what they don't know? In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8653–8665. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with BERT. In *International Conference on Learning Representations (ICLR)*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging LLM-as-a-judge with MT-Bench and chatbot arena. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36.

A Canonical 11-Section SITREP Taxonomy

The LT4CPR 2026 shared task defines a standardized, hierarchical 11-section SITREP taxonomy. Sections 1–2 cover narrative overview and timeline tracking, while Sections 3–11 constitute the official quantitative evaluation scope. Every bullet point in Sections 3–11 must be assigned to one of the canonical subsections predefined for that section; a generated bullet assigned to a non-canonical subsection name is penalized unless remapped to a valid canonical heading by the Senior Judge (Stage 3). Table 7 lists the full taxonomy.

B Prompt Engineering Specifications

The `nemotron-120b-sitrep` pipeline relies on two dedicated prompting stages, both powered by NVIDIA's Nemotron-3 Super 120B model: Stage 2 (Unified Template-Filling Synthesis) and Stage 3 (Senior LLM Judge). Table 8 summarizes the operational directives issued to each.

Stage 2: Unified Synthesis	**Stage 3: Senior LLM Judge**
Role: expert emergency management analyst • Synthesize factual bullets across Sections 1–11 • Subsections drawn strictly from the canonical taxonomy • 1–3 supporting `tweet_ids` per bullet; never hallucinate IDs • `confirmed` only if $\geq$ 2 independent sources corroborate • Formal register; no speculation or negative declarations • Section 1: 2–3 sentence executive overview • Output: valid JSON only	*Role:* senior crisis SITREP reviewer / lead editor • Deduplicate identical facts across sections; keep the most specific • Remap non-canonical subsections to the nearest valid heading • Consolidate micro-claims; union their `tweet_ids` • Re-audit `confirmed` labels against the $\geq$ 2-source rule • Strip chatter, hashtags, mentions, and subjective commentary

Table 8: Condensed operational directives for the Stage 2 synthesis prompt and the Stage 3 Senior Judge prompt.

C Algorithmic Hyperparameters and Inference Configuration

Table 9 details the complete runtime, model, and infrastructure hyperparameters used for all experiments and the official submission.

Hyperparameter / Setting	**Value / Specification**
LLM Architecture	NVIDIA Nemotron-3 Super
Model Identifier	`nemotron-3-super-120b-a12b`
Total Parameters	120 Billion
Active Parameters / Token	12 Billion (MoE)
Inference Endpoint	NVIDIA NIM API (`v1`)
Decoding Hyperparameters	
Temperature (T)	0.0 (Greedy deterministic)
Top-p Sampling	1.0
Max Completion Tokens	4,000 tokens
Presence / Frequency Penalty	0.0 / 0.0
Pipeline Infrastructure	
Worker Pool Concurrency	3 concurrent cells
API Retry Library	`tenacity`
Retry Minimum Wait	2.0 seconds
Retry Maximum Wait	30.0 seconds
Max Retry Attempts	5 attempts (exponential backoff)
Socket Request Timeout	120.0 seconds
Min Tweet Length (L)	10 characters
Primary Benchmark Metric	(BERTScore F1 + BLEURT)/2
Official BERTScore Model	`microsoft/deberta-xlarge-mnli`
Official BLEURT Model	`BLEURT-20`
Official Report Schema	Schema v1.2 JSON

Table 9: Complete algorithmic, model, and operational hyperparameters for the `nemotron-120b-sitrep` pipeline.

Sec	Section Title	Canonical Allowed Subsections
1	Situation overview	*Overview* (narrative executive summary; 2–3 synthesized sentences)
2	Timeline	*Chronological timeline* (chronologically ordered event milestones with timestamps)
3	Casualties and human impact	*Fatalities*, *Injuries*, *Cases*, *Missing persons*, *Trapped persons*, *Rescued persons*, *Found persons*, *Treatment and admissions*
4	Infrastructure and service impact	*Damage*, *Destruction*, *Service disruptions*, *Restoration progress*
5	Displacement and movement	*Evacuations*, *Displacement*, *Returns*, *Access constraints*
6	Hazard assessment	*Weather and environment*, *Geological and seismic*, *Industrial and chemical*, *Structural and fire*, *Biological and health*
7	Response actions	*Search and rescue*, *Evacuation and sheltering*, *Medical response*, *Law enforcement and security*, *Infrastructure and engineering*, *Containment and hazard control*
8	Communication and information	*Public warnings and advisories*, *Emergency hotlines and contact*, *Rumor control and corrections*, *Media and press updates*
9	Aid and relief	*Food and water*, *Shelter and non-food items*, *Medical aid*, *Financial and material donations*, *Logistics and supply chain*
10	Organizational and administrative activity	*Declarations and alerts*, *Coordination and meetings*, *Resource mobilization*, *International assistance*
11	Social and community response	*Community initiatives*, *Volunteer actions*, *Donations and collections*, *Public sentiment and appeals*

Table 7: Complete 11-section SITREP taxonomy and official canonical subsection headings mandated under Schema v1.2 for the LT4CPR 2026 Shared Task.

D Training Split Calamity Breakdown

The official training split comprises 140 temporal observation cells spanning eight diverse disaster categories. Table 10 details the distribution of cells, observation windows, and disaster typologies across this partition.

Hazard Typology	Calamity Name	Cells	Windows (W)
Geophysical	`volcano`	19	W_1–W_4
Hydrological	`flood`	18	W_1–W_4
Climatological	`wildfire`	18	W_1–W_4
Biological	`epidemic`	18	W_1–W_4
Maritime	`ferry` accident	17	W_1–W_4
Human-Induced	`crowdcrush`	16	W_1–W_4
Infrastructural	`blackout`	15	W_1–W_4
Anthropogenic	`conflict`	19	W_1–W_4
Total Training	**8 Calamities**	**140**	—

Table 10: Scenario distribution and observation window breakdown for the 140 training partition cells.

E End-to-End Generation Case Study

To illustrate the end-to-end transformation performed by `nemotron-120b-sitrep`, Table 11 traces a representative example from cell `wildfire.W1.k1`. Tweet texts are reproduced verbatim from the shared task input file; final bullets are copied verbatim from the system's submission JSON (`submission_train/wildfire/wildfire.W1.k1.report.json`).

Raw tweets	[39] "RT @wildfireLive: MANDATORY EVACUATION ordered for all of Ashford Mill and zones C1-C4 along Route 9." (17:55) [57] "Just heard that Ashford Mill is now under a full evacuation order per Coravel Sheriff's Office. Everyone please get out safely…" (18:05) [68] "MANDATORY EVACUATION ordered for all of Ashford Mill and zones C1-C4. Residents should leave the area immediately." (18:18) [90] "RT @TVP: MANDATORY EVACUATION ordered for all of Ashford Mill and zones C1-C4. #RedpineFire #evacuation" (18:48)
Stage 2 draft	§5 *Evacuations*: MANDATORY EVACUATION ordered for all of Ashford Mill and zones C1–C4 along Route 9, residents to leave immediately via Route 9 south (confirmed; [39,68,90]) §5 *Evacuations*: Coravel Sheriff's Office placed Ashford Mill under a full evacuation order (confirmed; [57]) (§ denotes SITREP section number.)
Stage 3 audit	Consolidated overlapping §5 evacuation bullets; verified $\geq$ 2-source support for the first bullet ([39]: Media, [68]: Government, [90]: Media $\rightarrow$ confirmed); downgraded the second bullet ([57]: single Eyewitness $\rightarrow$ unconfirmed) and retained as a separate sub-bullet per the dual-source corroboration rule.
Final bullets	{"id": "5a.1", "text": "MANDATORY EVACUATION ordered for all of Ashford Mill and zones C1-C4 along Route 9, with residents instructed to leave immediately via Route 9 south.", "confidence": "confirmed", "tweet_ids": [39, 68, 90]} {"id": "5a.2", "text": "Coravel Sheriff's Office placed Ashford Mill under a full evacuation order, per an eyewitness report.", "confidence": "unconfirmed", "tweet_ids": [57]}

Table 11: Execution trace of nemotron-120b-sitrep on cell wildfire.W1.k1. Tweet texts are reproduced verbatim from the shared task input JSONL; final bullets are copied verbatim from the system's submission JSON. Source labels are from the information_source field of each tweet record.

R01G-SR: Provenance-Preserving Selection over Complementary Evidence Views for Crisis Situation Report Generation

Yuqiu Gao[♡♥], **Yuanhe Tian**[♥*], **Xiaofei Chen**[♡], **Zhentao Yang**[♡], **Yan Song**[♠]
[♡]Southern University of Science and Technology [♥]Zhongguancun Academy
[♠]University of Science and Technology of China
[♡]s-gyq25@bza.edu.cn [♥]yhtian94@gmail.com

Abstract

Crisis situation report generation organizes evolving information into structured summaries with traceable supporting evidence. Generating these reports from noisy social-media streams requires balancing evidence completeness and concentration. Unfiltered evidence retains rare observations but includes competing material and remains subject to finite prompt budgets, while filtering and compression risk discarding useful sparse evidence. In this paper, we propose R01G-SR, a provenance-preserving approach that exploits complementary evidence views through whole-report selection. The approach generates two candidate reports using the same quantized 4B generator, presenting unfiltered chronological evidence to one route and filtered, near-duplicate-compressed evidence to the other. A logistic selector uses input- and report-derived features to choose one complete candidate without merging or rewriting reports, and a separate deterministic compliance layer enforces structural and evidence-ID requirements. Nested crisis-disjoint evaluation shows aggregate improvements over both fixed routes, and the official hidden-test evaluation shows improved report quality over the submitted compressed-route fallback. Analysis reveals larger selection gains on longer inputs and later windows, while early windows account for most remaining loss relative to selecting the better available candidate. These findings support evidence-view selection as a useful component of evidence-linked crisis reporting.[1]

1 Introduction

Situation report generation summarizes the evolving state of a crisis, including its impact, needs, and response, to support a coherent understanding of the situation. Social-media streams provide timely observations about crisis events (Lewis et al., 2025), but their short, informal (Zhang et al., 2018; Nie et al., 2020), fragmented, repetitive, and noisy content complicates the identification and organization of relevant information. A useful report must integrate information scattered across messages while preserving the evidence supporting its claims. The LT4CPR 2026 shared task formalizes these requirements by asking systems to generate a situation report in a fixed schema for each input cell and attach every non-overview claim to message identifiers from that cell (Xia et al., 2026).

Existing studies examine event-oriented social summarization (He et al., 2020), crisis timeline extraction and summarization (Rajaby Faghihi et al., 2022), and attributable crisis facts (Buntain et al., 2023).

For evidence-linked report generation, evidence presentation introduces a trade-off between completeness and concentration. An unfiltered chronological stream retains rare signals alongside duplicated or irrelevant material, increasing the amount of competing information presented to the generator. Finite prompt budgets further limit evidence availability, so even an unfiltered route might exclude messages during prompt packing. Filtering and compression reduce redundant or irrelevant material and concentrate the evidence available for generation (Jiang et al., 2024). However, these operations also risk discarding sparse observations that contribute important information to the report (Tian et al., 2024a; Jin et al., 2026). The value of an evidence view consequently depends on the input and the report produced from that view, motivating selection between complementary outputs. Candidate selection provides a basis for this decision through learned quality signals (Fernandes et al., 2022) or expected utility (Suzgun et al., 2023). Recent document-level candidate selection (Jinnai, 2025) and sentence-level attributed generation (Xia et al., 2025) address complementary aspects of report quality and traceability. Selecting an ex-

*Corresponding author.
[1]Code is released at https://github.com/bjzgcai/R01G-SR.

Proceedings of the First Workshop on Language Technologies for Crisis Preparedness and Response, pages 177–187
November 9, 2026 ©2026 Association for Computational Linguistics

isting report preserves its claim-to-evidence links, whereas synthesizing another report introduces an additional generation and attribution step.

In this paper, we propose R01G-SR, a provenance-preserving approach that generates reports from two complementary evidence views and selects one complete candidate for each input cell. The direct route, R0, uses unfiltered chronological evidence with deterministic prompt packing. The compressed route, R1, applies high-recall filtering and near-duplicate grouping to concentrate the evidence presented to the generator. Both routes use the same generator, prompt, schema, and decoding configuration, so their intended difference lies in evidence preparation. R01G-SR compares the resulting reports using a logistic selector with 22 input- and report-derived features that capture input context and candidate properties. The selector makes its inference decisions without reference text, official scores, crisis identity, or hazard identity. It copies one complete report without merging or rewriting the candidates, preserving the selected candidate's claims and evidence links during selection. A deterministic compliance layer then repairs subsection ownership, sanitizes evidence identifiers, and validates the report schema without generating new claims. We evaluate route selection against both fixed routes under nested crisis-disjoint evaluation and report development and hidden-test results separately. The crisis-disjoint results show aggregate improvements over both fixed routes, while the analysis reveals variation in selection gains across crises, input lengths, and window indices.

Our contributions are summarized as follows.

- We identify and quantify the trade-off between evidence completeness and evidence concentration in structured crisis reporting, and analyze how selection gains vary with input length and window index.
- We propose a provenance-preserving dual-route selection approach that uses reference-free inference features to choose a complete report from complementary evidence views, keeping candidate generation and selection separate from deterministic compliance handling.
- We evaluate route selection against both fixed routes under nested crisis-disjoint evaluation and report development and hidden-test outcomes separately, identifying aggregate improvements and crisis-specific variation in selection gains.

2 Task, Data, and Evaluation

2.1 Task

An LT4CPR *cell* contains crisis metadata, window metadata, and the messages available for one crisis window and replicate. The required output is one JSON situation report per cell, organized into 11 fixed top-level sections and an official subsection vocabulary. Every bullet in Sections 2–11 must cite at least one evidence-message identifier from the same cell. Cells are processed independently: no messages, generated reports, or state are carried across windows, replicates, or crises.

The released training split contains 140 cells from eight crises: blackout, conflict, crowd crush, epidemic, ferry accident, flood, volcano, and wildfire. Development contains 23 cells from two held-out crises, chemical incident and earthquake, and the public test release contains 117 cells from six additional crises. Test references are hidden. The task documentation describes the scenarios as synthetic or fictional and notes that the streams contain deliberately off-topic messages (LT4CPR Shared Task Organizers, 2026a).

2.2 Released evaluation

The released evaluator scores Sections 3–11, aligning system and reference subsections by header and computing text similarity at the subsection level. The official PRIMARY score for an instance is

$$\text{PRIMARY} = \frac{\text{BERTScore-F1} + \text{BLEURT}}{2}, \quad (1)$$

where BERTScore (Zhang et al., 2020) and BLEURT (Sellam et al., 2020) contribute equally. The final system score is the macro-average across scored instances. We use ROUGE (Lin, 2004), evidence alignment, and structure recall as complementary diagnostics when the corresponding artifacts are available, while system comparisons are based on PRIMARY.

3 Related Work

Crisis-stream summarization organizes useful information from noisy social-media messages (Rudra et al., 2016), while broader microblog summarization models conversational discourse and latent topics (Li et al., 2018). Representative resources cover

event-oriented summarization through TWEETSUM (He et al., 2020), crisis timeline extraction and summarization through CrisisLTLSum (Rajaby Faghihi et al., 2022), and attributable facts through CrisisFACTS (Buntain et al., 2023). Broader summarization methods select salient sentences (Chen et al., 2018; Luo et al., 2019) and important medical utterances (Song et al., 2020). LT4CPR couples situation report generation with a fixed schema and explicit evidence identifiers within each input cell (Xia et al., 2026), connecting crisis reporting to attributed and structured generation. ReClaim interleaves reference and claim generation to provide sentence-level citations (Xia et al., 2025). ALCE evaluates answer generation with supporting citations (Gao et al., 2023b), while RARR retrieves evidence and revises unsupported content in existing outputs (Gao et al., 2023a). Attribute First, then Generate selects source segments to guide generation and provide fine-grained attribution (Slobodkin et al., 2024). Grammar-constrained decoding addresses structural validity by enforcing outputs that follow a prescribed grammar (Geng et al., 2023). Related generation methods use expert routing and fusion for dialogue summarization (Tian et al., 2024b), learned quality signals for candidate reranking (Fernandes et al., 2022), or expected utility under minimum Bayes risk decoding (Suzgun et al., 2023). Document-level MBR-OT adapts candidate selection using optimal transport over sentence-level utilities (Jinnai, 2025). Unlike MBR, our selector predicts a route from observable input and candidate features rather than computing expected pairwise utility. For evaluation, BERTScore (Zhang et al., 2020) and BLEURT (Sellam et al., 2020) jointly define the official LT4CPR PRIMARY score, while AlignScore assesses factual consistency through information alignment (Zha et al., 2023). GroUSE examines failure-mode detection by evaluators for grounded question answering (Muller et al., 2025), motivating a distinction between aggregate quality scores and evidence-support diagnostics. Compared with these studies, R01G-SR derives candidate diversity from unfiltered and filtered, near-duplicate-compressed evidence views under the same deterministic generator, and selects one complete report using reference-free inference features. Whole-report selection preserves the chosen candidate's claims and evidence links without post-selection synthesis, while a separate deterministic compliance layer handles structural and evidence-ID requirements.

4 Method

Figure 1 separates evidence preparation, learned whole-report selection, and deterministic compliance checks.

4.1 Problem formulation

For an LT4CPR cell, let E denote the available within-cell evidence and let $V_0(E)$ and $V_1(E)$ denote two evidence views. A fixed generator G produces

$$r_i = G(V_i(E)), \qquad i \in \{0, 1\}. \tag{2}$$

If the reference were available, an oracle could choose the candidate with the larger task score $\mathcal{Q}(r_i)$. At inference time $\mathcal{Q}$ is unavailable, so we learn a selector

$$s(x(E, r_0, r_1)) \in \{0, 1\}, \tag{3}$$

where x contains only observable input and report properties. The selected report is copied intact and then passed through a deterministic compliance function C, giving $\hat{r} = C(r_s)$. This formulation makes the design goal explicit: exploit differences between evidence views without introducing a post-selection generation step that could alter claims or provenance.

4.2 Shared generator

Both routes use Qwen3-4B-Instruct-2507[2] (Yang et al., 2025), quantized as UD-Q4_K_XL GGUF and served locally with `llama.cpp`. Decoding uses temperature 0 and a 6,144-token output cap. The model context allocation and the evidence-packing budget are distinct constraints: a request fitting within the model context does not imply that every source message was included in that request. The direct-route packer limits serialized evidence lines to 85,000 characters, excluding prompt instructions and metadata, and removes middle lines when this budget is exceeded. Inference uses a single NVIDIA A100-SXM4-80GB GPU.

The system prompt restricts the model to the supplied cell evidence and requires a single JSON object. The user prompt specifies crisis and window metadata, the legal section and subsection inventory, confidence labels, evidence-ID rules, and

[2]Official model card: https://huggingface.co/Qwen/Qwen3-4B-Instruct-2507.

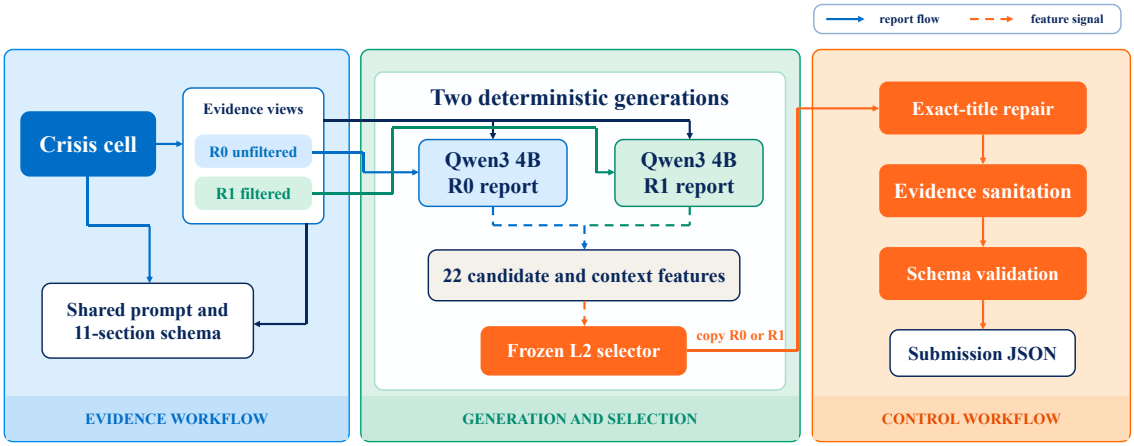


Figure 1: The R01G-SR method. Two complementary evidence views feed independent calls to the same local generator. A frozen selector uses reference-free inference features to choose one complete report without an additional rewrite. Exact-title ownership repair, evidence sanitation, and schema validation form a deterministic compliance layer and do not generate new claims.

instructions for concise atomic claims. Reports are generated automatically from the supplied within-cell evidence. A deterministic normalizer removes illegal or empty structures, merges duplicate subsection titles, and renumbers subsection and bullet identifiers.

4.3 Complementary evidence views

R0: unfiltered chronological evidence. R0 constructs an evidence line for every message in the cell's supplied chronological stream, without relevance filtering or near-duplicate grouping. Each line contains its identifier, timestamp, source label, and text. Before generation, a deterministic packer repeatedly removes the middle line until the sum of line lengths is at most 85,000 characters, or only two lines remain. This preserves the beginning and end of a long stream but can exclude intervening evidence. R0 therefore denotes the unfiltered direct route, not a guarantee that every original message is visible to the model. Both routes still use the same generator, report instructions, and decoding configuration; their evidence preparation differs.

R1: filtered and compressed evidence. R1 favors evidence concentration. It first normalizes retweet markers, URLs, mentions, hashtags, case, and punctuation for matching. A message is retained if it has an annotated source type or overlaps both the crisis metadata and a fixed high-recall list of crisis terms. Retained messages are grouped greedily when the Jaccard similarity of their normalized content tokens is at least 0.70. A group representative is replaced only by a member with a higher fixed source rank, and the compressed line preserves every member identifier. Compression therefore changes the text presented to the model without fabricating evidence identifiers.

The evidence views trade off chronological coverage against reduced redundancy and irrelevant competition.

4.4 Provenance-preserving route selection

The selector combines signals for coverage, evidence support, redundancy, attribution risk, and input context. For each candidate report, we compute five report-level features: the number of scored bullets, the number of represented subsections, mean lexical evidence support, duplicate-bullet count, and the number of numerical bullets containing values absent from their cited evidence. Let $f(r) \in \mathbb{R}^5$ denote these features, $q(r)$ a deterministic candidate-quality score, and $z \in \mathbb{R}^4$ the log input size, relevance ratio, compression ratio, and window index. The selector input is

$$x = [f(r_0), f(r_1), f(r_1) - f(r_0), \\ q(r_0), q(r_1), q(r_1) - q(r_0), z], \tag{4}$$

for a total of 22 dimensions. Section 4.5 defines the features and fitting weights. At inference time, the selector never receives crisis identity, hazard identity, reference text, or official scores.

Training labels record whether R1 attains a higher PRIMARY score than R0; exact ties are omitted from classifier fitting. We standard-

ize the features and fit an L2-regularized logistic regression. Sample weights reflect the crisis/window/replicate hierarchy and observed candidate regret, with regret floored at 0.005. We choose $C \in \{0.03, 0.1, 0.3, 1, 3\}$ and a decision threshold from $\{0.4, 0.5, 0.6\}$ inside the inner loop of nested leave-one-crisis-out (LOCO) validation. The final selector is fitted on all eight training crises with $C = 0.3$ and threshold 0.5. It chooses R1 for 16 development cells and R0 for seven.

Whole-report selection preserves one candidate's evidence trail without another language-model call. It does not itself verify entailment.

4.5 Feature definitions and selector fitting

The five report features are computed over Sections 3–11. For a bullet, lexical support is the maximum Jaccard overlap between its content-token set and that of any valid cited message; it is zero when no valid citation exists. Report-level support is the mean over scored bullets, with zero used for an empty report. A bullet contributes to the duplicate count if its token-set overlap with any earlier scored bullet exceeds 0.75. A numerical-risk flag is set if at least one extracted number in the bullet is absent from the concatenated cited evidence. These are lexical diagnostics, not entailment judgments. Represented subsections are counted by distinct titles in the scored sections.

Let B, H, L, D, and U denote the scored-bullet count, distinct subsection-title count, mean lexical support, duplicate-bullet count, and numerical-risk count, respectively. The deterministic quality feature is

$$q(r) = 0.30\frac{\min(B, 50)}{50} + 0.25\frac{\min(H, 12)}{12} + 0.45L - 0.03D - 0.08U. \quad (5)$$

For N input messages, N_{rel} retained messages, and N_{grp} compressed groups, the context features are $\log(1 + N)$, $N_{\text{rel}}/\max(1, N)$, $N_{\text{grp}}/\max(1, N)$, and the numeric window index. The stored feature vector interleaves each report feature as $(f_j(r_0), f_j(r_1), f_j(r_1) - f_j(r_0))$, followed by the two quality scores, their difference, and the four context features.

Within each fitting fold, exact-score ties are excluded. For a remaining cell i, the unnormalized fitting weight is

$$w_i = \frac{\max(|\Delta_i|, 0.005)}{|\mathcal{C}|\,|\mathcal{W}_{c(i)}|\,n_{c(i),w(i)}}, \quad (6)$$

where Δ_i is the R1-minus-R0 training score, $\mathcal{C}$ contains represented training crises, $\mathcal{W}_{c(i)}$ contains represented windows within its crisis, and $n_{c(i),w(i)}$ counts retained non-tie cells in that crisis-window group. We rescale these weights to mean one. Feature standardization is fitted without sample weights using only the fitting cells; logistic regression then uses the normalized weights. Inner and outer held-out crises are excluded from fitting the scaler and classifier for their respective predictions.

4.6 Deterministic compliance layer

The final layer addresses mechanical submission constraints rather than report quality. From the official subsection vocabulary, we build a title-to-owner map. A subsection is moved only when its title has exactly one legal owner and it appears under a different top-level section. If the destination already contains the same title, the subsections are merged and identifiers are renumbered. Claim text, confidence labels, and evidence content are never rewritten.

Evidence sanitation removes identifiers that do not occur in the cell and drops non-overview bullets left without a valid identifier. Strict schema, coverage, and evidence-ID validation follows. We use R01G-SR to refer to the selected report after these deterministic checks.

5 Experimental Setup

5.1 Research questions

We address three research questions:

RQ1: Does evidence presentation matter, and can direct and compressed views provide complementary report quality under a fixed generator?

RQ2: Can a selector trained without the target crisis exploit that complementarity on unseen crises without reference access at inference time?

RQ3: How do selection gains and remaining regret vary with input size and crisis window?

5.2 Crisis-disjoint selector evaluation

We treat the eight training crises as independent groups. In each outer LOCO fold, all decisions for the held-out crisis are made by a selector whose fitting and inner-loop hyperparameter selection exclude that crisis. We report both the original candidates and a compliance-matched analysis in which

System	PRIMARY	BERT-F1	BLEURT
R01G-SR	**0.524834**	**0.651119**	0.398548
R1-fresh	0.522617	0.645245	**0.399988**
R0	0.515040	0.638329	0.391751

Table 1: Results on all 23 development cells under the current released evaluator. R1-fresh is a regenerated compressed-route comparison.

Method	PRIMARY	Δ vs. R0c	Δ vs. R1c
R0c	0.352297	0.000000	+0.002543
R1c	0.349754	-0.002543	0.000000
R01G-SR	0.363869	**+0.011571**	**+0.014115**
Oracle	0.376693	+0.024396	+0.026939

Table 2: Compliance-matched results on 140 training cells with identical sanitation and hierarchical weights. Both always-R0c and always-R1c are explicit baselines. Selection uses frozen nested-LOCO decisions; oracle uses reference scores. Selector crisis-level wins are 8/8 against R0c and 5/8 against R1c. Deltas use unrounded scores.

both routes receive identical sanitation. A two-route oracle that picks the higher-scoring candidate using reference scores is included only to estimate unattainable headroom.

For this training analysis, scores are averaged first over replicates within each crisis window, then over windows within each crisis, and finally over crises. This hierarchical aggregation differs from the equal-cell averaging used for the current development comparison. The fixed-route baselines and the oracle are recomputed over the same cells with the same weights; only the learned selector requires cross-fitting.

5.3 Development and hidden-test evaluation

After freezing the selector, we evaluate the three development systems on the same 23 cells with the current evaluator. Development scores are never used to alter features, coefficients, or the decision threshold. Development contains two crises. The official hidden test contains 117 cells from six additional crises; its references remain unavailable, so test results are used only for system-level evaluation.

6 Results

6.1 RQ1: Evidence-view complementarity on development

On development, the compressed-route comparison is stronger than the direct route in aggregate. Table 1 shows that R1-fresh reaches 0.522617 PRIMARY compared with 0.515040 for R0. The selector chooses R0 for seven of the 23 cells and R1 for sixteen, and R01G-SR reaches 0.524834. Its advantage over R1-fresh is only +0.002217 (approximately 0.42% relative). Moreover, R1-fresh is a regenerated comparison rather than a verified identical copy of the selector's R1 candidate; the controlled fixed-route test is the crisis-disjoint analysis below.

Relative to R1-fresh, R01G-SR increases BERTScore-F1 by 0.005874 but decreases BLEURT by 0.001440, yielding a net PRIMARY gain of 0.002217. Relative to R0, both components improve. The comparison with the stronger fixed route therefore reflects a trade-off between component metrics, not a uniform improvement. With only two development crises, we treat the aggregate margin as descriptive rather than statistically established superiority.

6.2 RQ2: Crisis-disjoint route selection

R01G-SR reaches 0.363869, exceeding R0c by 0.011571 and R1c by 0.014115. Crisis-level gains occur on 8/8 crises against R0c and 5/8 against R1c. Always choosing R1c therefore does not explain the aggregate benefit, although selection does not beat it on every crisis. Section 6.3 reports each crisis, including losses against R1c on epidemic, volcano, and wildfire. The selector recovers approximately 47% of the oracle improvement over R0c.

6.3 Fixed-route comparisons by crisis

Table 3 reports the 140-cell compliance-matched scores using the same hierarchical weights as Table 2. Each row averages replicates within windows and then windows within one held-out crisis. The macro row averages the eight crises equally. It reveals both the positive aggregate margin and the three crises where always-R1c performs better than the selector.

The crisis-level comparison clarifies the scope of the aggregate improvement. Against the compressed route, the largest positive differences occur for conflict, ferry, and flood, whereas epidemic, volcano, and wildfire favor the fixed compressed route. These mixed outcomes rule out a uniform per-crisis advantage.

Crisis	Cells	R0c	R1c	R01G-SR	Δ vs. R0c	Δ vs. R1c
Blackout	16	0.370619	0.374033	0.387532	+0.016913	+0.013499
Conflict	13	0.358526	0.309418	0.359698	+0.001172	+0.050280
Crowd crush	22	0.366667	0.373663	0.392431	+0.025764	+0.018768
Epidemic	16	0.374077	0.382945	0.375037	+0.000960	-0.007908
Ferry	25	0.352971	0.316636	0.353631	+0.000660	+0.036996
Flood	10	0.336510	0.315041	0.349410	+0.012900	+0.034369
Volcano	16	0.296942	0.345246	0.318324	+0.021382	-0.026923
Wildfire	22	0.362064	0.381050	0.374884	+0.012820	-0.006166
Macro	140	0.352297	0.349754	0.363869	+0.011571	+0.014115

Table 3: Fixed-route comparisons for each held-out training crisis. All values are PRIMARY; deltas are computed before rounding. The selector's aggregate improvement over R1c is not uniform across crises.

System	BERT-F1	BLEURT	Align-F1	PRIMARY
Organizer baseline	0.463445	0.475283	0.208240	0.469364
R1 fallback	0.483191	0.541305	0.278185	0.512248
R01G-SR	**0.501837**	**0.552800**	**0.288103**	**0.527319**

Table 4: Official hidden-test scores for our two submitted systems and the organizer baseline. Align-F1 is Weighted Alignment F1. The organizer baseline is excluded from participant rankings; PRIMARY is computed from BERTScore-F1 and BLEURT.

6.4 Official hidden-test evaluation

The official evaluation uses a hidden-reference test split of 117 cells from six crises. We compare the designated R01G-SR primary with the submitted R1 fallback. The official results workbook reports the scores in Table 4 (LT4CPR Shared Task Organizers, 2026b).

Relative to the submitted R1 fallback, R01G-SR improves PRIMARY by 0.015071. The improvement is not confined to the ranking metric: BERTScore-F1 rises by 0.018646, BLEURT by 0.011495, and Weighted Alignment F1 by 0.009918. The official workbook also shows gains of 0.018577, 0.018827, and 0.020234 in ROUGE-1, ROUGE-2, and ROUGE-L F1, respectively. Thus, the primary submission exceeds the fallback on every reported metric. It also exceeds the organizer baseline by 0.057955 PRIMARY.

R01G-SR ranks sixth of nine teams in both the Primary and Best System Rankings. The test PRIMARY score is numerically close to development (0.5273 versus 0.524834). However, BERTScore-F1 changes from 0.651119 to 0.5018, while BLEURT changes from 0.398548 to 0.5528. Their offsetting movements mean that aggregate similarity should not be interpreted as metric-level stability or as evidence that the same mechanisms generalize across splits. The official comparison with the submitted R1 fallback provides the more direct evidence of system-level benefit on test.

7 Selection Gains by Input Size and Window

The main results establish that route selection can improve report quality. We next examine how selection gains and remaining regret vary as evidence accumulates.

Stratifying the nested-OOF decisions by input size shows descriptive differences across evidence-volume groups. Relative to the compliance-matched direct route, selection changes PRIMARY by -0.0022 for cells with 22–125 messages, +0.0052 for 128–357 messages, and +0.0233 for 362–993 messages. Across windows, the corresponding deltas are -0.0016, +0.0048, +0.0157, and +0.0274 from W1 through W4. These are differences between selected reports and R0c, not isolated compression effects.

Input length, window index, and crisis characteristics vary together, so these subgroup differences are descriptive rather than causal.

The oracle places 82.2% of remaining weighted regret in W1/W2. Regret is the better existing candidate's score minus the selected score, so this identifies missed routing opportunities, not proof that candidate quality dominates. The training gap is 0.012824; the development oracle reaches 0.530747, leaving 0.005913. Future work should separate routing errors from cases where both reports omit or misinterpret evidence.

8 Discussion

Value beyond a fixed evidence view. Under identical sanitation and hierarchical weighting, always-R0c slightly exceeds always-R1c, yet the selector exceeds both (Table 2). Its benefit therefore comes

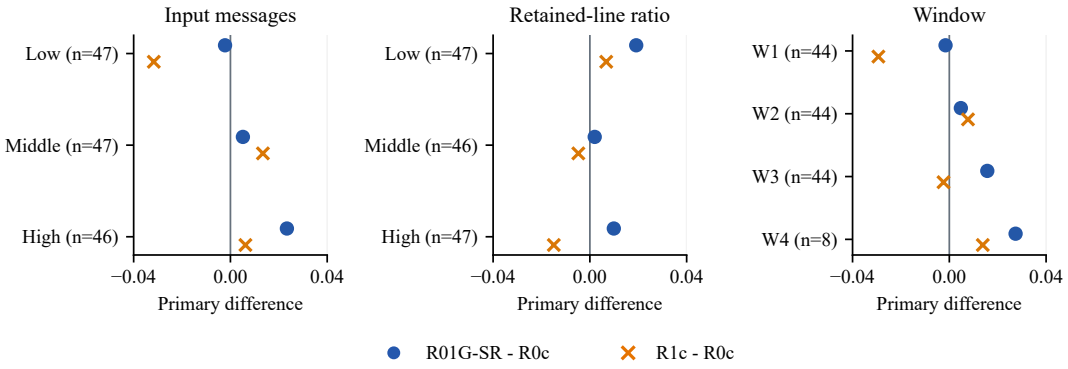


Figure 2: Compliance-matched nested-OOF differences by input-message tercile, retained-line ratio, and crisis window. Retained-line ratio denotes compressed evidence lines per input message, corresponding to $N_{\mathrm{grp}}/\max(1, N)$. Points are descriptive, not causal. W4 has eight cells versus 44 in each earlier window; the ratio trend is non-monotonic.

from conditional decisions rather than a general preference for compression. The selector recovers approximately 47% of the two-route oracle's improvement over R0c. Candidate coverage, lexical support, redundancy, and numerical-risk features describe complementary report properties, while context features describe evidence volume and concentration. We evaluate these signals jointly and make no single-feature attribution. Development, nested-LOCO, and official-test results address different evaluation settings, with candidate relationships and aggregation rules specified in the experimental setup and results.

Crisis-specific differences. For conflict and ferry, the selector stays close to R0c while exceeding R1c; for crowd crush and flood, it exceeds both fixed routes. Epidemic, volcano, and wildfire favor R1c over the selector, showing remaining opportunities for better routing. These crisis-level averages summarize cell-specific decisions made by held-out-crisis selectors. The early-window concentration of regret identifies where routing can improve among the existing candidates, whereas facts omitted by both candidates require improvements to evidence presentation or generation. Packing affects evidence availability, while claim support requires checking the relationship between each claim and its cited messages.

Generation cost. The method requires two generation calls per cell, with cost influenced by the input and output lengths of both routes. Sharing loaded weights does not eliminate the additional decoding workload. Whole-report copying avoids a further synthesis call and preserves the selected candidate's claim–evidence associations before compliance checks. The practical choice is therefore whether the observed quality gain warrants the additional generation budget. A future cost-aware policy could request the second candidate selectively. No latency or energy measurements are available for assessing that trade-off quantitatively.

9 Conclusion

In this paper, we propose R01G-SR to exploit complementary evidence views for evidence-linked crisis situation report generation. The approach generates candidates under the same generator and uses reference-free features to select one complete report without rewriting its claims. Nested crisis-disjoint evaluation shows aggregate gains over both fixed routes, and official hidden-test results show improvements over the submitted compressed-route fallback. Future work should distinguish routing errors from evidence omissions or misinterpretations shared by both candidates.

Ethical Considerations

Crisis streams may contain sensitive information, rumors, distressing accounts, and references to vulnerable people. Our system processes released cells locally and performs no external identity resolution, web enrichment, contact, or profiling. Evidence identifiers improve traceability but do not establish source accuracy; automated reports may still repeat rumors, omit less visible experiences, or overstate uncertain claims. The outputs are research artifacts rather than operational intelligence. Deployment would require source governance, access controls, human review, correction workflows, and evaluation with crisis professionals and affected communities.

Limitations

The nested-LOCO evaluation covers eight training crises, and development covers only two crises. Hidden test references prevent crisis-level oracle and evidence-support analyses on the official test split. The regenerated R1-fresh development comparison is separate from the selector's R1 candidate and the submitted fallback. Automatic similarity metrics do not certify factual entailment, operational usefulness, timeliness, or safety; lexical support and numerical-risk features are shallow attribution proxies. The repair layer handles exact titles with a unique legal owner and leaves ambiguous cases untouched. Quantized local inference also limits conclusions about other model precisions and serving stacks.

Acknowledgments

This work is supported by Zhongguancun Academy (Grant No. XTS0043).

Generative AI tools assisted with manuscript polishing. The authors reviewed all revisions and take full responsibility for the content of this paper.

References

Cody Buntain, Benjamin D. Horne, Amanda Hughes, Muhammad Imran, Richard McCreadie, and Hemant Purohit. 2023. CrisisFACTS 2023 – Overview Paper. In *The Thirty-Second Text REtrieval Conference Proceedings (TREC 2023)*, volume 1328 of *NIST Special Publication*. National Institute of Standards and Technology.

Xiuying Chen, Shen Gao, Chongyang Tao, Yan Song, Dongyan Zhao, and Rui Yan. 2018. Iterative document representation learning towards summarization with polishing. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 4088–4097.

Patrick Fernandes, António Farinhas, Ricardo Rei, José G. C. de Souza, Perez Ogayo, Graham Neubig, and Andre Martins. 2022. Quality-aware decoding for neural machine translation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1396–1412, Seattle, United States. Association for Computational Linguistics.

Luyu Gao, Zhuyun Dai, Panupong Pasupat, Anthony Chen, Arun Tejasvi Chaganty, Yicheng Fan, Vincent Zhao, Ni Lao, Hongrae Lee, Da-Cheng Juan, and Kelvin Guu. 2023a. RARR: Researching and revising what language models say, using language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16477–16508, Toronto, Canada. Association for Computational Linguistics.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. 2023b. Enabling large language models to generate text with citations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6465–6488, Singapore. Association for Computational Linguistics.

Saibo Geng, Martin Josifoski, Maxime Peyrard, and Robert West. 2023. Grammar-constrained decoding for structured NLP tasks without finetuning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10932–10952, Singapore. Association for Computational Linguistics.

Ruifang He, Liangliang Zhao, and Huanyu Liu. 2020. TWEETSUM: Event oriented social summarization dataset. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5731–5736. International Committee on Computational Linguistics.

Huiqiang Jiang, Qianhui Wu, Xufang Luo, Dongsheng Li, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. 2024. LongLLMLingua: Accelerating and enhancing LLMs in long context scenarios via prompt compression. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1658–1677, Bangkok, Thailand. Association for Computational Linguistics.

Yiqiao Jin, Kartik Sharma, Vineeth Rakesh, Yingtong Dou, Menghai Pan, Mahashweta Das, and Srijan Kumar. 2026. SARA: Selective and adaptive retrieval-augmented generation with context compression. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14508–14528, San Diego, California, United States. Association for Computational Linguistics.

Yuu Jinnai. 2025. Document-level text generation with minimum Bayes risk decoding using optimal transport. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 28260–28279, Vienna, Austria. Association for Computational Linguistics.

William D. Lewis, Haotian Zhu, Keaton Strawn, and Fei Xia. 2025. Tapping into social media in crisis: A survey. In *Proceedings of the Fourth Workshop on NLP for Positive Impact (NLP4PI)*, pages 306–331, Vienna, Austria. Association for Computational Linguistics.

Jing Li, Yan Song, Zhongyu Wei, and Kam-Fai Wong. 2018. A joint model of conversational discourse and latent topics on microblogs. *Computational Linguistics*, 44(4):719–754.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81. Association for Computational Linguistics.

LT4CPR Shared Task Organizers. 2026a. LT4CPR 2026 AACL Workshop Shared-Task Repository. Released task data, schema, validator, and evaluation code. Test-release snapshot: commit 78dbff586d391d97ec4d11310cf587f43c57ca2a.

LT4CPR Shared Task Organizers. 2026b. LT4CPR Shared Task Official Evaluation Results. Official results workbook. Accessed 2026-09-16.

Ling Luo, Xiang Ao, Yan Song, Feiyang Pan, Min Yang, and Qing He. 2019. Reading like HER: Human reading inspired extractive summarization. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3033–3043, Hong Kong, China. Association for Computational Linguistics.

Sacha Muller, Antonio Loison, Bilel Omrani, and Gautier Viaud. 2025. GroUSE: A benchmark to evaluate evaluators in grounded question answering. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 4510–4534, Abu Dhabi, UAE. Association for Computational Linguistics.

Yuyang Nie, Yuanhe Tian, Xiang Wan, Yan Song, and Bo Dai. 2020. Named entity recognition for social media texts with semantic augmentation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1383–1391.

Hossein Rajaby Faghihi, Bashar Alhafni, Ke Zhang, Shihao Ran, Joel Tetreault, and Alejandro Jaimes. 2022. CrisisLTLSum: A benchmark for local crisis event timeline extraction and summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 5455–5477, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Koustav Rudra, Siddhartha Banerjee, Niloy Ganguly, Pawan Goyal, Muhammad Imran, and Prasenjit Mitra. 2016. Summarizing situational and topical information during crises. In *4th International Workshop on Social Web for Disaster Management (SWDM 2016)*, Indianapolis, USA. Co-located with CIKM 2016. Available as arXiv:1610.01561.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892. Association for Computational Linguistics.

Aviv Slobodkin, Eran Hirsch, Arie Cattan, Tal Schuster, and Ido Dagan. 2024. Attribute first, then generate: Locally-attributable grounded text generation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3309–3344, Bangkok, Thailand. Association for Computational Linguistics.

Yan Song, Yuanhe Tian, Nan Wang, and Fei Xia. 2020. Summarizing medical conversations via identifying important utterances. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 717–729.

Mirac Suzgun, Luke Melas-Kyriazi, and Dan Jurafsky. 2023. Follow the wisdom of the crowd: Effective text generation via minimum Bayes risk decoding. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 4265–4293, Toronto, Canada. Association for Computational Linguistics.

Yuanhe Tian, Chang Liu, Yan Song, Fei Xia, and Yongdong Zhang. 2024a. Aspect-based sentiment analysis with context denoising. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 3083–3095, Mexico City, Mexico. Association for Computational Linguistics.

Yuanhe Tian, Fei Xia, and Yan Song. 2024b. Dialogue summarization with mixture of experts based on large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7143–7155, Bangkok, Thailand. Association for Computational Linguistics.

Fei Xia, Haotian Zhu, Yoo Wun Lee, William D. Lewis, Antonis Anastasopoulos, and Belu Ticona. 2026. Overview of the LT4CPR 2026 shared task: Generating situation reports from social media messages. In *Proceedings of the First Workshop on Language Technologies for Crisis Preparedness and Response (LT4CPR)*, Hengqin, Zhuhai, China. In conjunction with AACL 2026, November 9, 2026.

Sirui Xia, Xintao Wang, Jiaqing Liang, Yifei Zhang, Weikang Zhou, Jiaji Deng, Fei Yu, and Yanghua Xiao.

2025. Ground every sentence: Improving retrieval-augmented LLMs with interleaved reference-claim generation. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 969–988, Albuquerque, New Mexico. Association for Computational Linguistics.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. Qwen3 Technical Report. *arXiv preprint arXiv:2505.09388*.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. AlignScore: Evaluating factual consistency with a unified alignment function. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11328–11348. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with BERT. In *International Conference on Learning Representations*.

Yingyi Zhang, Jing Li, Yan Song, and Chengzhi Zhang. 2018. Encoding conversation context for neural keyphrase extraction from microblog posts. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1676–1686.

A Implementation and Resource Disclosure

Item	Setting
Primary generator	Qwen3-4B-Instruct-2507, UD-Q4_K_XL GGUF
Inference engine	`llama.cpp`, local execution
Principal decoding	temperature 0; 6,144 output-token cap
Hardware	one NVIDIA A100-SXM4-80GB
Selector	standardized L2 logistic regression; $C = 0.3$; threshold 0.5
Training data	released LT4CPR training split only

Table 5: Disclosure for the submitted primary system.

B Code Availability

The public repository provides core generation, feature extraction, selector training and application, and deterministic compliance code. Frozen selector coefficients, task data, generated reports, and model weights are not included. Retraining requires access to permitted task data and the corresponding training-candidate scores.

Frozen-Response Recovery for Evidence-Linked Crisis Situation Reports

Quan Zheng[♦♣] **Ming Wang**[♦] **Yuanhe Tian**[♣*] **Yan Song**[♠]
[♦]Beijing Normal University [♣]Zhongguancun Academy, Beijing, China
[♠]University of Science and Technology of China
[♦]s-zq25@bza.edu.cn [♣]tianyuanhe@zgci.ac.cn
[♦]wangming@bnu.edu.cn [♠]clksong@gmail.com

Abstract

Structured crisis reports must preserve both information and its placement under canonical subsections. Merging reports generated from input chunks can omit subsections or duplicate their entries elsewhere. We introduce deterministic missing-subsection recovery and exact cross-title deduplication. The method reuses already generated model outputs, held fixed during postprocessing comparisons, without additional model calls. On 23 development cells, the combined postprocessor increased the official primary score from 0.4820 to 0.4890, a 1.45% relative gain over the baseline. Component comparisons attribute most of this development gain to recovery. On 117 test cells, all three submitted postprocessors scored 0.76–0.85% above the same baseline system on that split. Our primary system, GROUNDED, adds a fixed negative-template filter and placed fourth among nine teams; our highest-scoring submission also placed fourth in the best-system ranking. The results support structural correction of existing outputs, with small numerical differences between recovery policies across splits. They measure reference similarity, not factual entailment.[1]

1 Introduction

The LT4CPR shared task asks systems to turn crisis-related social-media posts into structured, evidence-linked situation reports (Xia et al., 2026). Each report must summarize one bounded input cell, organize claims under canonical subsection titles, and cite cell-local tweet identifiers. An information item can remain somewhere in a report while disappearing from the subsection where it belongs. As a schematic example, a rescue statement retained only under *Information updates* loses its placement under *Rescued persons*, despite preserving the text.

The distinction matters because the official evaluator aligns subsections by their titles. Systems must also track evolving values, distinguish casualty states, respect the reporting cutoff, and omit unsupported claims. Long cells compound these requirements when their prompts or outputs exceed generation budgets.

Our generation baseline addresses long inputs through token-aware chronological chunking and recursive recovery from truncated calls. Its model-mediated merge consolidates the partial reports without enforcing preservation of their subsection inventory. A subsection containing eligible entries can disappear even when the merge returns complete JSON. Identical bullet text can also appear under several titles, changing the units scored by the evaluator.

We study corrections over *frozen responses*: model outputs generated once and reused unchanged in postprocessing comparisons. The term refers to outputs, not to parameter freezing during training. Missing-subsection recovery restores an absent canonical subsection together with eligible entries from those outputs, not merely its title string. Exact cross-title deduplication then removes selected repeated placements. Three policies determine which recovery candidates are admitted.

We evaluate this zero-additional-call postprocessor through matched component comparisons, a title-compatibility analysis, and official scores for four submitted systems. Together, these analyses distinguish score changes, structural effects, and the semantic errors targeted by lexical gates.

2 Background, Task, and Data

2.1 Related Work

Crisis report constraints. Crisis NLP extracts situational signals from noisy social media and compresses them into summaries (Imran et al., 2016; Rudra et al., 2016). Recent surveys syn-

*Corresponding author.

[1]The code is released at https://github.com/zqq-oso/lt4cpr-frozen-response-recovery.

Proceedings of the First Workshop on Language Technologies for Crisis Preparedness and Response, pages 188–202
November 9, 2026 ©2026 Association for Computational Linguistics

thesize crisis-social-media resources and disaster-focused LLM applications (Lewis et al., 2025; Lei et al., 2025). Benchmarks target local timelines, cross-stream facts, and event-level dynamics (Rajaby Faghihi et al., 2022; Buntain et al., 2023; Su et al., 2026). LLM studies construct timelines or summarize multi-stream disasters, emergency communication, and social-media feedback (Hu et al., 2024a; Seeberger and Riedhammer, 2024; Doula et al., 2025; Cantini et al., 2025). These studies motivate faithful selection from evolving message streams. LT4CPR adds a specific organizational requirement: each claim occupies a canonical subsection and carries cell-local evidence identifiers. Our work examines preservation of that organization during report consolidation.

Information preservation across stages. Long-input summarization and retrieval-augmented generation use staged reduction, dynamic extraction, long-context transformers, or expert routing and fusion (Chen et al., 2018; Luo et al., 2019; Zhang et al., 2022; Mao et al., 2022; Guo et al., 2022; Pan et al., 2024; Tian et al., 2024). Yet usable context varies with evidence position and length (Liu et al., 2024; Hsieh et al., 2024; Zheng et al., 2026). LongBench v2 evaluates deeper reasoning across realistic long-context tasks, including multi-document question answering (Bai et al., 2025). Structured generation separates planning from realization and can enforce grammars during decoding (Moryossef et al., 2019; Puduppully and Lapata, 2021; Geng et al., 2023; Dong et al., 2025). These approaches can complement postprocessing: input reduction addresses context limits, while constrained decoding addresses output form. Our method acts after generation, recovering eligible subsection content that exists in intermediate responses but is missing from the final report.

Structure and evidence support. Fluent and retrieval-conditioned text can remain unsupported (Maynez et al., 2020; Kryściński et al., 2020; Niu et al., 2024; Belem et al., 2025; Yang et al., 2026). Source-conditioned metrics score input support (Laban et al., 2022; Zha et al., 2023; Min et al., 2023; Tang et al., 2024); newer methods decompose claims or adapt evaluation to long documents (Scirè et al., 2024; Song et al., 2024; Bishop et al., 2024; Zhong and Litman, 2025). Evidence-attribution work separately evaluates whether citations support generated content (Gao et al., 2023; Wright et al., 2025; Cattan et al., 2026). This distinction is central to our scope. A well-formed report with valid evidence IDs may still misplace a claim or cite text that does not support it. Our structural correction and lexical gates do not replace source-conditioned factuality assessment. Appendix F gives additional comparisons.

Split	Crises	Cells	References
Training	8	140	released
Development	2	23	released
Test	6	117	withheld

Table 1: Released data inventory. Test references were withheld during development and submission; aggregate organizer scores were released later.

2.2 Task and Data

One *cell* contains a crisis description, a reporting window, and a chronologically ordered tweet sample (LT4CPR Shared Task Organizers, 2026). Tweet identifiers are local to that cell. We process every cell independently and do not transfer generated content between cumulative windows or replicates. Table 1 summarizes the released partitions.

The reports use schema version 1.2. The official format permits report Sections 1–11 and recommends the unscored overview and timeline in report Sections 1–2; only report Sections 3–11 are scored (LT4CPR Shared Task Organizers, 2026). Our fixed prompt emitted only these scored sections. Their content consists of complete English bullets, binary confidence labels, and supporting tweet identifiers. The official materials describe synthetic and fictional crises. Fidelity is therefore cell-relative rather than a correction toward external world knowledge (Li et al., 2026). The organizers state that no external source contains target-event facts and the corpus contains no real personal information (LT4CPR Shared Task Organizers, 2026).

The hierarchy distinguishes related states, such as missing, rescued, and found persons. A *title* is a canonical label; a *subsection* consists of that label and its bullet entries within a report section. We use *header* when discussing title matching in evaluation. These units let us distinguish subsection recovery from improvements to the wording of an existing bullet.

Windows are cumulative, and replicates from the same window overlap. These dependencies motivate the hierarchical aggregation defined in the experimental protocol; treating the 23 development

cells as independent samples would overstate the available evidence (LT4CPR Shared Task Organizers, 2026).

3 System

Figure 1 separates model-mediated cell generation from the deterministic transformations replayed over frozen responses.

3.1 Frozen Direct Baseline

The submitted Frozen local direct baseline (S0) uses a locally served copy identified as Qwen3.5-35B-A3B.[2] Generation used temperature zero, disabled thinking output, and excluded tweets in the organizer-provided `Not labeled` category. Despite its submission name, S0 uses either direct generation or chunking followed by merging, depending on the input budget.

Each prompt contains only the current cell's crisis metadata, window metadata, and tweets. The instruction treats tweet text as untrusted data and prohibits outside facts or later developments. It requests concise atomic claims, canonical titles, and at most six supporting identifiers per bullet.

The configured context window was 40,000 tokens, with a 512-token reserve. The direct/chunk and merge output caps were 7,000 and 12,000 tokens, and the chunk prompt budget was 16,000 tokens. A token-count preflight sends the full cell when it fits. Otherwise, binary search forms the largest consecutive chronological chunks within budget.

Only responses with a stop finish reason are accepted. A truncated multi-tweet call is bisected chronologically and regenerated recursively. Chunk reports are compiled before merging, and their evidence identifiers are restricted to tweets visible in that chunk. The merge receives only the partial reports and their cited current-cell evidence.

Why subsections can disappear during merging. The merge is a new generation followed by deterministic compilation. Its fixed prompt requests removal of true duplicates, retention of distinct supported announcements, and adherence to the report schema. Neither the prompt nor the schema requires preservation of every input subsection. Consolidation can therefore omit a subsection or move its content elsewhere while returning valid JSON. Compilation normalizes the generated hierarchy but cannot reconstruct omitted content. Appendix B.2 examines output truncation as an alternative explanation.

[2]Model-specific documentation: `https://huggingface.co/Qwen/Qwen3.5-35B-A3B`.

3.2 Missing-Subsection Recovery

The postprocessor reads the authoritative S0 report, successful frozen chunk responses, and the released current-cell tweets. It reconstructs the original chunk requests and checks that every candidate identifier was visible in the corresponding chunk. Ambiguous or inconsistent provenance is rejected.

The baseline report determines which section–title pairs are present. For each absent pair, the postprocessor collects eligible bullets from chunk responses that pass the provenance and evidence-ID scope checks. The content then passes through the same compiler used by the baseline, without making any additional model calls.

Equation 1 summarizes the transformation. Let $H(R_0)$ be the canonical section–title pairs in the compiled baseline report, C_h the scope-checked chunk bullets emitted under title h, and $g_p(h, b)$ the predicate for recovery policy p. Binary switches r and d enable recovery and deduplication, respectively. Conceptually, the postprocessor computes

$$\begin{aligned} A_{p,1} &= \{(h, b) : h \notin H(R_0),\ b \in C_h, \\ &\qquad g_p(h, b) = 1\}, \\ A_{p,0} &= \emptyset, \\ R_{p,r,d} &= D_d(C(R_0 \cup A_{p,r}))\,, \end{aligned} \tag{1}$$

where C is deterministic compilation, D_1 is exact cross-title deduplication, and D_0 is the identity transformation. The implementation preserves the report hierarchy rather than performing a literal set union.

Candidate eligibility and recovery scope. Recovery considers only section–title pairs absent from $H(R_0)$. An existing subsection receives no additional entries through this step, even if incomplete. Similarity to another bullet does not determine eligibility. A cited message must also have been visible in the originating chunk; membership in the complete cell alone is insufficient. The final report remains restricted to the current cell. A missing subsection is restored only if an eligible entry survives the policy and compilation. Candidate availability, acceptance, and final retention are distinct stages.

We compare three policies. *All-occurrences recovery* accepts every eligible occurrence under

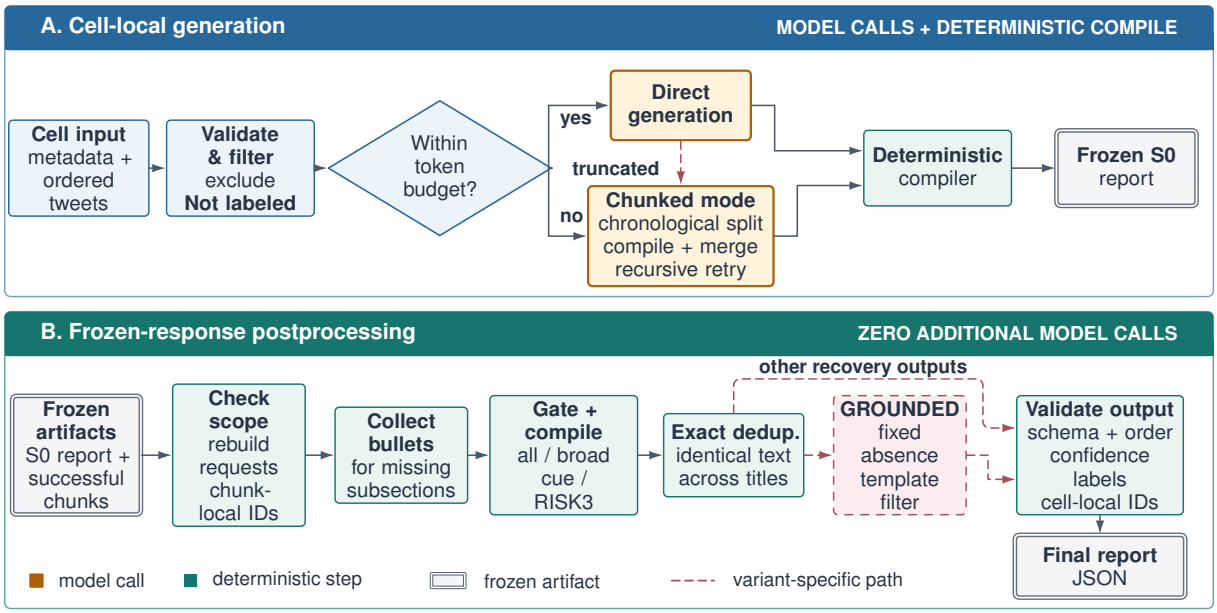


Figure 1: Two-phase system architecture. Phase A produces frozen S0 through direct or chunked generation. Phase B checks chunk provenance and evidence-ID scope, recovers missing subsections and eligible entries, deduplicates exact cross-title text, and validates report structure. Only GROUNDED applies the fixed negative-template filter. Phase B makes no model calls and performs no entailment verification.

each missing title. *Broad cue recovery* requires title-specific lexical cues over a broad title inventory. The *known-risk cue-gated recovery* policy, abbreviated RISK3, considers all eligible missing subsections. It applies additional lexical gates only to *Fatalities*, *Cases*, and *Found persons*. These gates were informed by training and development analyses. The predicates require death, toll, or body-recovery context for *Fatalities*, disease-related context for *Cases*, and person-finding constructions for *Found persons*. Other missing subsections retain all-occurrences recovery.

Broad cue recovery rejects candidates whose titles are outside its cue inventory; RISK3 imposes no additional lexical gate outside its three designated titles. The accepted sets of Broad cue and RISK3 need not contain one another. For example, the RISK3 fatality predicate accepts body-recovery expressions, whereas Broad cue requires explicit fatality or death terms. These are different predicates, not looser and stricter settings of a common threshold.

3.3 Deterministic Safeguards and Validation

Exact cross-title deduplication. After recovery, the method compares normalized exact bullet text across titles within a report. Competing report Section 3 copies are resolved through declared casualty-state cues. An *Information updates* copy is removed only when another titled copy exists and the text lacks a correction, clarification, or update cue. Evidence IDs from all copies in a deduplicated group are unioned, capped at six, and assigned to the surviving copies.

Recovery precedes deduplication because it can introduce another placement of text already present. The latter step removes eligible copies, including those newly introduced by recovery. Exact matching does not collapse paraphrases, changed quantities, or revised statements merely because their wording is similar. The update-cue rule retains copies containing the specified correction or clarification expressions.

Fixed negative-template filter. A fixed report Section 3 template in the generated reports converted missing information into an exhaustive negative claim: no fatalities, injuries, missing, trapped, rescued, or found persons, and no treatment admissions. A one-sided rule removes this exact template only when no tweet in the cell matches any predefined absence cue. A cue for just one category preserves the entire template; the search includes uncited and `Not labeled` tweets. Retention therefore does not establish support for each clause or from the cited tweets. Other text and evidence identifiers remain unchanged.

The filter was defined using released test inputs and our generated outputs, before test references or scores were available. It was not selected through

development-set comparison.

Compilation and validation. Our constrained response schema permits only scored report Sections 3–11, which are the only sections present in the submitted reports. The compiler accepts canonical participant sections and filters evidence identifiers to valid IDs from the current cell (LT4CPR Shared Task Organizers, 2026). It normalizes whitespace, drops malformed objects, deduplicates within-title text, orders sections and titles canonically, and assigns sequential identifiers. A strict validator then checks schema keys, titles, ordering, confidence values, and evidence locality.

Four properties must be distinguished. *Structural validity* concerns schema fields, ordering, and allowed labels. *Evidence-ID scope validity* concerns whether identifiers belong to the permitted input. *Title compatibility* concerns whether a proposition fits its subsection; lexical gates approximate this relation, and manual analysis assesses selected placements. *Factual support* concerns whether evidence entails the claim. The validator establishes the first two properties, not the latter two.

Systems and component controls. S0 is the compiled generation baseline, before the post-processing studied here. All-occurrences, Broad cue, and RISK3 each combine their named recovery policy with exact cross-title deduplication. GROUNDED is RISK3 plus the fixed negative-template filter. The name is a submission label, not a claim of comprehensive factual grounding. The *recovery-only* control enables RISK3 recovery without cross-title deduplication; *deduplication-only* applies that deduplication to S0 without recovery. Baseline compilation, including its own within-title deduplication, is shared by all systems.

4 Experimental Setup

4.1 Research Questions and Comparisons

We ask three questions. First, how much do missing-subsection recovery and exact cross-title deduplication each contribute under fixed generation? Second, how do recovery policies affect subsection coverage and title compatibility? Third, how do the complete submitted systems perform on test, where the frequency of chunked generation differs from development?

Compared systems. All development comparisons cover the same 23 cells and reuse the same S0 responses, inputs, compiler, and evaluator. The policy comparison holds cross-title deduplication fixed while varying the recovery policy. The component comparison crosses recovery and deduplication on/off, using RISK3 whenever recovery is enabled. These retrospective controls estimate component contributions within this generation setup. GROUNDED has no separate development row because its outputs equal RISK3's. Official test scores evaluate the complete submitted systems; there are no recovery-only or deduplication-only official test scores.

4.2 Implementation and Resources

We performed no task-specific training or fine-tuning. The submitted pipeline used the organizer data and a locally served pretrained Qwen checkpoint, with no external retrieval system or external model API. Reports were generated automatically, without manual rewriting or per-report selection.

Generation used vLLM 0.24.0 (Kwon et al., 2023) with tensor parallelism across four NVIDIA A100-SXM4-80GB GPUs. The postprocessors added no model calls; Table 10 in Appendix B gives generation costs.

The public code package includes the frozen prompts, response schema, and rule definitions. Original model responses and model weights are not included. Appendices B, C, and E describe the generation settings, rules, and manual error analysis.

4.3 Evaluation Protocol

We use official evaluator commit `c8bd925` (LT4CPR Shared Task Organizers, 2026). Within each crisis, metrics are averaged over replicates and then windows. The final score averages crises. The official primary score in Equation 2 averages BERTScore F1 (Zhang et al., 2020) and BLEURT (Sellam et al., 2020):

$$P = \frac{\text{BERTScore-F1} + \text{BLEURT}}{2}. \qquad (2)$$

We report absolute scores alongside relative changes:

$$\Delta_{\text{rel}}(S; B) = 100\frac{P(S) - P(B)}{P(B)}\%. \qquad (3)$$

Unless stated otherwise, B is the frozen S0 baseline on the same split or slice. Relative changes compare aggregated system and baseline scores;

changes in individual metrics use the corresponding baseline metric as the denominator. A relative percentage is not a percentage-point change. Maintable scores are rounded to four decimals; contrasts use the released precision retained in Appendix A. Official rankings are reported as supplied by the organizers.

The hierarchical aggregation gives each crisis equal weight, regardless of how many cells it contains. Within a crisis, each reporting window receives equal weight, and replicates share the weight of their window. For crisis c, window t, and replicate j, the contribution of a cell has weight

$$\omega_{c,t,j} = \frac{1}{|\mathcal{C}|\,|\mathcal{W}_c|\,|\mathcal{R}_{c,t}|}, \tag{4}$$

where the sets index crises, windows, and replicates, respectively. This prevents a crisis with more sampled replicates from dominating the aggregate. The equal-cell window analyses reported below answer a different question: they describe where changes occur within the released inventory. Their values are interpreted as slice summaries alongside the official crisis-macro score.

Subsections align by normalized titles within each section. The evaluator uses side-specific zero contributions for unmatched BERTScore units and matched-only BLEURT. We also report ROUGE-L F1 (Lin, 2004) and weighted alignment F1. For the latter, subsection-local Hungarian matching maximizes 0.8 ROUGE-L F1 plus 0.2 Jaccard overlap between evidence-ID sets, discards edges below 0.10, and computes soft precision and recall from matched weight over system and reference bullet counts.

5 Results

5.1 Development Component Attribution

Table 2 compares all development systems using the same version of the official evaluator.

The combined RISK3 postprocessor increased the primary score from 0.4820 to 0.4890, a 1.45% relative gain over S0. Recovery alone scored 0.4874 (+1.14%), and deduplication alone scored 0.4831 (+0.24%). Their combined gain exceeded the additive prediction of 1.37%. The observed factorial interaction was 0.0004 score units; adding deduplication to recovery yielded a further 0.31% relative to the recovery-only control. Primary-score wins/ties/losses against S0 were 9/13/1, 9/14/0, and 13/9/1 for recovery only, deduplication only, and both, respectively.

For the combined RISK3 system, relative changes were +2.63% in BERTScore F1, -0.03% in BLEURT, +3.08% in ROUGE-L F1, and +3.80% in weighted alignment F1.

These contrasts distinguish an operation's effect on the baseline from its effect after the other operation changes the report. In particular, the deduplication contribution after recovery is measured against recovery-only, not against S0. The positive interaction is descriptive of this 23-cell comparison; no statistical significance test was performed.

Header coverage and output volume. Because official subsection alignment uses headers, we also compute a non-official exact canonical-header diagnostic. Table 3 compares nonempty reference and system headers across all 23 development cells.

Combined recovery and deduplication increase matched headers by 22 and reduce reference-only headers by 22, while system-only headers increase by two. Header recall rises by 6.04 percentage points, precision by 0.78 points, and F1 by 3.15 points. Final output volume decreases from 1,314 to 1,308 bullets.

Crisis and window results. Table 4 localizes the combined RISK3 contrast. The hierarchically aggregated difference was positive in both crisis components. Equal-cell window comparisons were unchanged for W1 and positive for W2–W4.

These slices are descriptive: development contains two crises, cumulative windows share content, and W4 has one cell per crisis. They localize the aggregate difference but are not independent replications.

5.2 Recovery Tradeoff and Error Analysis

Table 5 separates the components' structural effects. RISK3 recovery alone adds 26 subsections and 41 bullet instances; deduplication alone removes 40 exact copies and empties two subsections. Applied after recovery, deduplication removes 47 copies. Across all-occurrences, broad-cue, and RISK3 recovery, non-official title-level micro F1 is .839, .841, and .842, respectively.

A single annotator examined differences between the final All-occurrences reports and S0. An order-sensitive content alignment yielded 51 unmatched candidate records, 38 unmatched baseline records, and 40 cross-title matched pairs across 94

System	Rec.	Dedup.	Score	Δ_{rel} (%)	BERT F1	BLEURT	R-L F1	Align. F1
S0	off	off	0.4820	–	0.5384	**0.4255**	0.2287	0.1762
Recovery only (RISK3)	on	off	0.4874	+1.14	0.5494	0.4254	0.2343	0.1805
Deduplication only	off	on	0.4831	+0.24	0.5408	0.4253	0.2302	0.1781
RISK3	on	on	**0.4890**	**+1.45**	**0.5526**	0.4253	**0.2358**	**0.1829**
All occurrences	on	on	0.4871	+1.06	0.5494	0.4248	0.2352	0.1814
Broad cue	on	on	0.4889	+1.44	**0.5526**	0.4251	0.2353	0.1826

Table 2: Development results on all 23 cells (official evaluator `c8bd925`). Rec./Dedup. enable missing-subsection recovery/exact cross-title deduplication. BERT/R-L/Align. denote BERTScore/ROUGE-L/weighted alignment F1. The "only" rows are non-submitted controls. Relative gains use S0; bold marks displayed maxima, including rounding ties. Released precision is retained in Table 9.

System	Match	Ref.-only	Sys.-only	P	R	F1
S0	306	58	85	.783	.841	.811
RISK3	328	36	87	.790	.901	.842

Table 3: Non-official header coverage on development. Match counts headers shared by the system and reference; Ref.-only and Sys.-only count unmatched headers. P/R/F1 are micro precision, recall, and F1 over nonempty canonical section–title pairs.

Slice	Cells	Δ_{rel} vs. S0 (%)
Chemical crisis	10	+2.36
Earthquake crisis	13	+0.53
W1	7	0.00
W2	7	+2.05
W3	7	+1.29
W4	2	+2.01

Table 4: Development slices for RISK3 minus S0. Crisis rows use official aggregation; window rows use equal-cell averages and are correlated. Each denominator is the baseline score for that slice.

changed content groups. Of the 51 unmatched candidate records, 23 were title-incompatible. Content corresponding to all 38 unmatched baseline records remained elsewhere: 36 removals were judged safe or beneficial, and two carried possible title-recall loss. These are output-alignment categories, not counts of executed recovery or deduplication operations.

A separate gate analysis compared final title–text placements in All-occurrences and RISK3 reports. It identified 19 excluded development placements and 12 from the training subset; all 31 were judged incompatible (18 *Cases*, 12 *Fatalities*, and one *Found persons*). The 29 cue rejections in Table 5 instead count candidate occurrences before compilation and deduplication. These judgments informed gate design and provide descriptive error analysis, not independent gate precision estimates. Table 6 gives examples; Appendix E defines the units and overlapping categories.

Retrospective training analysis. The retrospective training analysis covers 24 cells from eight crises across windows W2–W4. Appendix D specifies the sampling scheme used to define this subset. The observed RISK3 gain over S0 was 3.56% relative, with cell-level wins/ties/losses of 17/5/2 and positive differences in seven of eight crises and every represented window. Because training and development diagnostics informed the rules, this is retrospective corroboration, not held-out validation. Appendix D reports the metric differences at full precision.

5.3 Official Test Results

We submitted four systems with 117 reports each: GROUNDED as primary and RISK3, broad cue recovery, and S0 as additional systems. Table 7 reports the organizer-released evaluation. The primary designation reflected GROUNDED's absence-filter safeguard, not a development-score advantage over RISK3. The submitted systems and their roles were fixed before official test scores were released.

The primary GROUNDED system scored 0.5955 versus S0's 0.5906, a 0.83% relative gain, and ranked fourth among nine teams in the Primary System Ranking. Broad cue scored 0.5956, a 0.85% relative gain over S0, placing the team fourth in the Best System Ranking.

All three postprocessed systems scored above S0 by 0.76–0.85% relative. Their released ROUGE and BLEURT values are identical, whereas BERTScore and weighted alignment differ.

Relative to RISK3, GROUNDED deleted one fixed absence-template bullet from each of two test reports lacking matching same-cell cues. It scored 0.07% higher in the official score, 0.13% higher in BERTScore F1, and 0.06% higher in weighted

System	Recovered subsections	Accepted bullets	Cue-rejected bullets	Duplicate removals	Output subsections	Output bullets
S0	0	0	0	0	391	1,314
Recovery only	26	41	29	0	417	1,355
Deduplication only	0	0	0	40	389	1,274
RISK3	26	41	29	47	415	1,308
All occurrences	31	70	0	57	420	1,327
Broad cue	25	32	38	46	414	1,300

Table 5: Structural effects on 23 development reports. Recovered subsections and accepted/rejected candidate bullet instances are counted before recompilation. Duplicate removals count cross-title copies removed afterward; output columns count final nonempty subsections and bullet instances. Repeated text can contribute multiple instances. The "only" rows are non-submitted component controls.

Rejected title	Generated proposition
Fatalities	"41 people were treated for respiratory irritation at Wexford County Infirmary, with 6 admitted."
Found persons	"Officials confirmed the Millbrook Lane apartment block has been found."
Cases	"18 fatalities were confirmed in the Millbrook Lane apartment block collapse."

Table 6: Wrong-title recovered bullets (verbatim). Treatment is not death, *found* modifies a building, and fatalities are not disease cases; the underlying claims are not labeled factually false.

alignment; their BLEURT and ROUGE values were unchanged.

RISK3 changed 25 test reports and reduced output volume from 1,684 to 1,644 bullets. Of 14 accepted candidate bullet instances, 11 survived compilation and deduplication: ten reused text elsewhere in the report, and one added text supported by its cited current-cell tweet. Every removed normalized bullet text remained elsewhere in the same report. Recovery affected four cells and deduplication affected 24. The resulting reports were identical to those from all-occurrences recovery on test. These are changes relative to the already compiled baseline.

Accepted candidates are counted before compilation and deduplication, giving $1{,}684 + 14 - 54 = 1{,}644$ final bullet instances. The 11 retained recovered entries are already included in that total.

Generation route and opportunity to recover. We compared RISK3 postprocessing effects across direct and chunked baseline reports (Table 8). Recovery counts require a backfilled bullet to survive compilation and deduplication.

Chunking was needed for 12/23 development cells (52.2%), but only 5/117 test cells (4.3%). Recovery affected 10/12 chunked development cells (83.3%) and 4/5 chunked test cells (80.0%). The opportunity for missing-subsection recovery was therefore less frequent in the test inventory, although it occurred in most chunked cells in both splits.

Deduplication also operated on reports generated directly. It removed eight duplicate bullets in two direct development cells and 45 in 21 direct test cells. On test, 45 of 54 removed duplicates (83.3%) came from direct generation; nine came from chunked reports. The submitted postprocessor consequently repaired both merge omissions and repetition within complete direct reports.

Recovery thus affected 10 development and four test cells, while deduplication affected 13 development and 24 test cells; the two sets can overlap. These counts measure how often each operation acted, not how much test-score gain it caused. Component-score contributions are available only from the development comparison. The official test totals reflect the complete systems.

6 Discussion

Why structure changes similarity scores. A claim under an incompatible title may remain visible at report level but fall outside the reference subsection against which it should be compared. Missing-subsection recovery changes both content placement and the units available for header-aligned evaluation. Cross-title deduplication changes their distribution without necessarily removing a distinct proposition. On development, improved header coverage accompanies six fewer bullet instances and 37 fewer tokens than S0. The score gain therefore cannot be explained simply by producing a longer report.

The metrics also treat missing subsections differently. Unmatched units contribute zeros to side-specific BERTScore, whereas BLEURT excludes

System	Role	Official	Δ_{rel} (%)	BERT	BLEURT	R-L	Align.
S0	additional	0.5906	–	0.5744	0.6068	0.4832	0.3500
RISK3	additional	0.5951	+0.76	0.5831	0.6071	0.4871	0.3555
GROUNDED	primary	0.5955	+0.83	0.5839	0.6071	0.4871	0.3557
Broad cue	additional; best	**0.5956**	**+0.85**	**0.5841**	0.6071	0.4871	**0.3563**

Table 7: Official test results (LT4CPR Shared Task Organizers, 2026), rounded to four decimals. Relative gains use S0. BERT/R-L/Align. denote BERTScore/ROUGE-L/weighted alignment F1; Official averages BERTScore F1 and BLEURT. Bold marks our strictly highest submitted values. Table 9 retains released precision.

Split and route	Cells	R	D	Removed
Dev., direct	11	0	2	8
Dev., chunked	12	10	11	39
Test, direct	112	0	21	45
Test, chunked	5	4	3	9

Table 8: RISK3 operations by frozen S0 generation route. R/D count cells with retained recovered content/exact cross-title deduplication; Removed counts deleted duplicate bullets. R and D can overlap. Counts precede the GROUNDED absence filter.

them and evaluates only matched units. Restoring a missing reference-aligned subsection changes coverage and the pool of matched pairs. The resulting score changes need not be parallel: an improvement in BERTScore can coexist with little change in BLEURT. This is consistent with the observed metric pattern, without implying that a higher similarity score proves factual support.

Recovery can create an additional placement of existing text, so deduplication acts on different pairs before and after recovery. This provides a structural interpretation of the observed development interaction, without assigning unmeasured component contributions to the official test gain.

Policy differences across splits. RISK3 was numerically above Broad cue on development by less than 0.0001 primary-score units; Broad cue was above RISK3 on test by about 0.0005. Appendix A retains the released values. This small change in numerical ordering does not establish a stable policy advantage. Moreover, RISK3 and All-occurrences produced identical test reports, so that split does not distinguish their outputs.

The lower chunking rate on test reduces opportunities for recovery: directly generated reports supply no chunk candidates, although they can still benefit from deduplication. Even chunked reports need an omitted subsection with an eligible entry. These differences describe opportunity, not the unmeasured component contributions to the smaller test gain.

Lexical gates address semantic distinctions beyond schema correctness, as illustrated by the wrong-title cases in Table 6. The non-nested policies in Section 3.2 cover different expressions, so their comparative value depends on the available candidates. Manual analysis also identifies coverage costs when deduplication removes an appropriate placement.

Title compatibility and factual support. An entry can fit its title yet lack cited support, or be supported but misplaced. Valid evidence IDs establish neither relation. The negative-template filter is a targeted heuristic: one absence cue can retain several negative clauses without checking each against its citations. The correction unit remains the subsection with its eligible entries. Preserving only text loses its organizational role; preserving only a title supplies no content. Joint assessment of title compatibility and cited support would complement the similarity metrics, particularly for updates and absence claims.

7 Conclusion

Missing-subsection recovery and exact cross-title deduplication reuse fixed model outputs to correct report organization without further model calls. On 23 development cells, their combination improved the primary score by 1.45% relative to S0; matched component comparisons attributed most of this gain to recovery. The three complete submitted postprocessors scored 0.76–0.85% above S0 on test. Our primary and highest-scoring systems each placed fourth among nine teams in their respective rankings. The evidence supports structural correction in this generation setting, while small policy differences and varying recovery opportunities limit conclusions about which policy transfers best. The method restores eligible subsection content and reduces redundant placements; factual support remains a separate evaluation requirement.

Limitations

Development contains 23 cells from two crises, with dependent cumulative windows and replicates. The results workbook provides only aggregate test scores, precluding paired test analysis and route-specific score attribution. Training and development analyses informed the lexical gates; both the training analysis and the component comparisons are retrospective. Title-compatibility judgments came from one annotator without independent adjudication and informed gate design. They support qualitative error analysis rather than an independent estimate of gate precision.

The absence filter was developed from the test inputs, so its two-report effect is not an independent generalization estimate. One partial absence cue can retain an entire multi-clause negative template, even when the cue appears only in an uncited post. Excluding `Not labeled` posts from generation may omit useful information. The experiments use one backbone, one frozen prompt, and one generation run. Recovery only restores missing subsections; it cannot correct a misrouted proposition beneath a title already present. Exact deduplication misses paraphrased repetitions, and the transfer of English lexical gates to other wordings or languages remains unknown. No prompt or generation-budget ablation was performed to separate their effects on title loss.

The reported gains describe reference similarity and alignment, rather than evidence entailment or real-world truth. Statistical significance and operational crisis-response utility were not evaluated.

Ethical Considerations and Responsible Use

The benchmark crises are synthetic and contain no real personal information. The outputs are research artifacts, not operational intelligence. Use during a real emergency would require privacy review, calibrated uncertainty, continuous monitoring, and expert approval before dissemination. The English-only synthetic posts, fixed hazard inventory, and simulated source distribution do not measure whether omissions or routing errors differ across languages, communities, or crisis types; deployment would require disaggregated multilingual evaluation.

Acknowledgments

This work is supported by Zhongguancun Academy (Grant No. C20250521). We thank the anonymous reviewer for suggestions that improved the presentation and analysis, and the LT4CPR organizers for providing the task and evaluation. Codex assisted with language editing, organization, and formatting during manuscript preparation.

References

Yushi Bai, Shangqing Tu, Jiajie Zhang, Hao Peng, Xiaozhi Wang, Xin Lv, Shulin Cao, Jiazheng Xu, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. 2025. LongBench v2: Towards deeper understanding and reasoning on realistic long-context multitasks. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3639–3664.

Forrest Sheng Bao, Miaoran Li, Renyi Qu, Ge Luo, Erana Wan, Yujia Tang, Weisi Fan, Manveer Singh Tamber, Suleman Kazi, Vivek Sourabh, Mike Qi, Ruixuan Tu, Chenyu Xu, Matthew Gonzales, Ofer Mendelevitch, and Amin Ahmad. 2025. FaithBench: A diverse hallucination benchmark for summarization by modern LLMs. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 448–461.

Catarina G Belem, Pouya Pezeshkpour, Hayate Iso, Seiji Maekawa, Nikita Bhutani, and Estevam Hruschka. 2025. From single to multi: How LLMs hallucinate in multi-document summarization. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 5291–5324.

Jennifer A. Bishop, Sophia Ananiadou, and Qianqian Xie. 2024. LongDocFACTScore: Evaluating the factuality of long document abstractive summarisation. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 10777–10789.

Cody Buntain, Amanda Lee Hughes, Richard McCreadie, Benjamin D. Horne, Muhammad Imran, and Hemant Purohit. 2023. CrisisFACTS 2023 - overview paper. In *Text Retrieval Conference*.

Riccardo Cantini, Cristian Cosentino, Fabrizio Marozzo, Domenico Talia, and Paolo Trunfio. 2025. Harnessing prompt-based large language models for disaster monitoring and automated reporting from social media feedback. *Online Social Networks and Media*, 45:100295.

Arie Cattan, Paul Roit, Shiyue Zhang, David Wan, Roee Aharoni, Idan Szpektor, Mohit Bansal, and Ido Dagan. 2026. Localizing factual inconsistencies in at-

tributable text generation. *Transactions of the Association for Computational Linguistics*, 14:100–123.

Yapei Chang, Kyle Lo, Tanya Goyal, and Mohit Iyyer. 2024. BooookScore: A systematic exploration of book-length summarization in the era of LLMs. In *International Conference on Learning Representations*, volume 2024, pages 56059–56093.

Xiuying Chen, Shen Gao, Chongyang Tao, Yan Song, Dongyan Zhao, and Rui Yan. 2018. Iterative document representation learning towards summarization with polishing. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 4088–4097.

Yixin Dong, Charlie F. Ruan, Yaxing Cai, Ziyi Xu, Yilong Zhao, Ruihang Lai, and Tianqi Chen. 2025. XGrammar: Flexible and efficient structured generation engine for large language models. In *Proceedings of Machine Learning and Systems*, volume 7.

Achref Doula, Bela Bohlender, Max Mühlhäuser, and Alejandro Sanchez Guinea. 2025. CLEARcommand: Coordinated listening, extraction, and allocation for emergency response with large language models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (System Demonstrations)*, pages 20–28.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. 2023. Enabling large language models to generate text with citations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6465–6488.

Saibo Geng, Martin Josifoski, Maxime Peyrard, and Robert West. 2023. Grammar-constrained decoding for structured NLP tasks without finetuning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10932–10952, Singapore.

Mandy Guo, Joshua Ainslie, David Uthus, Santiago Ontañón, Jianmo Ni, Yun-Hsuan Sung, and Yinfei Yang. 2022. LongT5: Efficient text-to-text transformer for long sequences. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 724–736.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. 2024. RULER: What's the real context size of your long-context language models? In *First Conference on Language Modeling*.

Qisheng Hu, Geonsik Moon, and Hwee Tou Ng. 2024a. From moments to milestones: Incremental timeline summarization leveraging large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7232–7246.

Xiangkun Hu, Dongyu Ru, Lin Qiu, Qipeng Guo, Tianhang Zhang, Yang Xu, Yun Luo, Pengfei Liu, Yue Zhang, and Zheng Zhang. 2024b. Knowledge-centric hallucination detection. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 6953–6975.

Muhammad Imran, Prasenjit Mitra, and Carlos Castillo. 2016. Twitter as a lifeline: Human-annotated Twitter corpora for NLP of crisis-related messages. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1638–1643.

Parag Jain, Andreea Marzoca, and Francesco Piccinno. 2024. STRUCTSUM generation for faster text comprehension. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7876–7896.

Yekyung Kim, Yapei Chang, Marzena Karpinska, Aparna Garimella, Varun Manjunatha, Kyle Lo, Tanya Goyal, and Mohit Iyyer. 2024. FABLES: Evaluating faithfulness and content selection in book-length summarization. In *First Conference on Language Modeling*.

Wojciech Kryściński, Bryan McCann, Caiming Xiong, and Richard Socher. 2020. Evaluating the factual consistency of abstractive text summarization. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, pages 9332–9346.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with PagedAttention. In *Proceedings of the 29th symposium on operating systems principles*, pages 611–626.

Philippe Laban, Tobias Schnabel, Paul N. Bennett, and Marti A. Hearst. 2022. SummaC: Re-visiting NLI-based models for inconsistency detection in summarization. *Transactions of the Association for Computational Linguistics*, 10:163–177.

Zhenyu Lei, Yushun Dong, Weiyu Li, Rong Ding, Qi R. Wang, and Jundong Li. 2025. Harnessing large language models for disaster management: A survey. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 14528–14551.

William D. Lewis, Haotian Zhu, Keaton Strawn, and Fei Xia. 2025. Tapping into social media in crisis: A survey. In *Proceedings of the Fourth Workshop on NLP for Positive Impact (NLP4PI)*, pages 306–331.

Mingchen Li, Hanzhi Zhang, Heng Fan, Junhua Ding, and Yunhe Feng. 2026. Harmful factuality: LLMs correcting what they shouldn't. In *Findings of the Association for Computational Linguistics: EACL 2026*, pages 896–912.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

LT4CPR Shared Task Organizers. 2026. LT4CPR shared task on automatic situation report generation. First Workshop on Language Technologies for Crisis Preparedness and Response, AACL-IJCNLP 2026. Accessed 2026-09-10.

Ling Luo, Xiang Ao, Yan Song, Feiyang Pan, Min Yang, and Qing He. 2019. Reading like HER: Human reading inspired extractive summarization. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3033–3043.

Ziming Mao, Chen Henry Wu, Ansong Ni, Yusen Zhang, Rui Zhang, Tao Yu, Budhaditya Deb, Chenguang Zhu, Ahmed Awadallah, and Dragomir Radev. 2022. DYLE: Dynamic latent extraction for abstractive long-input summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1687–1698.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. 2020. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 1906–1919.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12076–12100, Singapore.

Ali Modarressi, Hanieh Deilamsalehy, Franck Dernoncourt, Trung Bui, Ryan A Rossi, Seunghyun Yoon, and Hinrich Schütze. 2025. NoLiMa: Long-context evaluation beyond literal matching. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 44554–44570. PMLR.

Amit Moryossef, Yoav Goldberg, and Ido Dagan. 2019. Step-by-step: Separating planning from realization in neural data-to-text generation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2267–2277.

Cheng Niu, Yuanhao Wu, Juno Zhu, Siliang Xu, KaShun Shum, Randy Zhong, Juntong Song, and Tong Zhang. 2024. RAGTruth: A hallucination corpus for developing trustworthy retrieval-augmented language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10862–10878.

Zhuoshi Pan, Qianhui Wu, Huiqiang Jiang, Menglin Xia, Xufang Luo, Jue Zhang, Qingwei Lin, Victor Rühle, Yuqing Yang, Chin-Yew Lin, H. Vicky Zhao, Lili Qiu, and Dongmei Zhang. 2024. LLMLingua-2: Data distillation for efficient and faithful task-agnostic prompt compression. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 963–981.

Ratish Puduppully and Mirella Lapata. 2021. Data-to-text generation with macro planning. *Transactions of the Association for Computational Linguistics*, 9:510–527.

Hossein Rajaby Faghihi, Bashar Alhafni, Ke Zhang, Shihao Ran, Joel Tetreault, and Alejandro Jaimes. 2022. CrisisLTLSum: A benchmark for local crisis event timeline extraction and summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 5455–5477.

Koustav Rudra, Siddhartha Banerjee, Niloy Ganguly, Pawan Goyal, Muhammad Imran, and Prasenjit Mitra. 2016. Summarizing situational and topical information during crises. *arXiv preprint arXiv:1610.01561*.

Alessandro Scirè, Karim Ghonim, and Roberto Navigli. 2024. FENICE: Factuality evaluation of summarization based on natural language inference and claim extraction. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 14148–14161.

Philipp Seeberger and Korbinian Riedhammer. 2024. Crisis2Sum: An exploratory study on disaster summarization from multiple streams. *Proceedings of the International ISCRAM Conference*, 21.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 7881–7892.

Yixiao Song, Yekyung Kim, and Mohit Iyyer. 2024. VeriScore: Evaluating the factuality of verifiable claims in long-form text generation. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 9447–9474.

Chen Su, Pengsen Cheng, Yuanhe Tian, and Yan Song. 2026. SURGE: An event-centric social media sentiment time series benchmark with interaction structure. *arXiv preprint arXiv:2605.21198*.

Liyan Tang, Philippe Laban, and Greg Durrett. 2024. MiniCheck: Efficient fact-checking of LLMs on grounding documents. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8818–8847.

Zecheng Tang, Keyan Zhou, Juntao Li, Baibei Ji, Jianye Hou, and Min Zhang. 2025. L-CiteEval: A suite for evaluating fidelity of long-context models. In

Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 5254–5277.

Yuanhe Tian, Fei Xia, and Yan Song. 2024. Dialogue summarization with mixture of experts based on large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7143–7155.

Dustin Wright, Zain Muhammad Mujahid, Lu Wang, Isabelle Augenstein, and David Jurgens. 2025. Unstructured evidence attribution for long context query focused summarization. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 1839–1867, Suzhou, China. Association for Computational Linguistics.

Fei Xia, Haotian Zhu, Yoo Wun Lee, William D. Lewis, Antonis Anastasopoulos, and Belu Ticona. 2026. Overview of the LT4CPR 2026 shared task: Generating situation reports from social media messages. In *Proceedings of the First Workshop on Language Technologies for Crisis Preparedness and Response (LT4CPR)*, Hengqin, Zhuhai, China. In conjunction with AACL 2026, November 9. To appear.

Ruixiao Yang, Yuanhe Tian, Di Dong, Xu Yang, Huiqi Li, and Yan Song. 2026. LEAD: Layer-wise expert-aligned decoding for faithful radiology report generation. *arXiv preprint arXiv:2602.04617*.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. AlignScore: Evaluating factual consistency with a unified alignment function. In *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: Long papers)*, pages 11328–11348.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with BERT. In *International Conference on Learning Representations*.

Yusen Zhang, Ansong Ni, Ziming Mao, Chen Henry Wu, Chenguang Zhu, Budhaditya Deb, Ahmed Awadallah, Dragomir Radev, and Rui Zhang. 2022. Summ^N: A multi-stage summarization framework for long input dialogues and documents. In *Proceedings of the 60th annual meeting of the Association for Computational Linguistics (volume 1: long papers)*, pages 1592–1604.

Quan Zheng, Yuanhe Tian, Ming Wang, and Yan Song. 2026. Fine-grained verbal attack detection via a hierarchical divide-and-conquer framework. *arXiv preprint arXiv:2601.06907*.

Yang Zhong and Diane Litman. 2025. Discourse-driven evaluation: Unveiling factual inconsistency in long document summarization. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2050–2073.

A Score Precision and Relative Changes

Table 9 gives scores at the released precision, before main-table rounding. Relative differences use these values and the baseline specified in each comparison. Comparisons between GROUNDED and RISK3 use RISK3 as the denominator.

B Prompt and Decoder Configuration

The frozen direct prompt treats every tweet as untrusted source data and limits claims to the current cell and reporting cutoff. It requests canonical subsection titles, concise atomic bullets, explicit uncertainty, and at most six current-cell evidence identifiers per bullet. Unsupported subsections must be omitted. The user payload contains crisis metadata, window metadata, and ordered tweet rows with identifier, timestamp, source category, and text. The merge prompt receives only compiled partial reports and their cited current-cell evidence; it may consolidate those claims but may not introduce outside facts. Table 11 lists the frozen decoding and budget settings. All reference-cited tweets in the released training and development data belonged to the retained source categories.

B.1 Response Contract and Prompt Artifacts

The response is a JSON object containing a list of sections. Each section specifies an identifier and a list of subsections; each subsection supplies a canonical title and a list of bullets. A bullet contains nonempty text, a confidence label (*confirmed* or *unconfirmed*), and integer tweet identifiers. Only report Sections 3–11 and titles from the canonical inventory are permitted, and additional fields are disallowed at every level.

The frozen prompts and response schema accompany the code in `prompts/legacy-v1/`. This directory includes the expanded system prompt, merge suffix, direct/chunk/merge user-message templates, and JSON schema.

B.2 Merge Output-Length Analysis

A retrospective analysis of the retained merge responses found that all completed below the output cap (Table 12). Hard output truncation therefore does not explain title omissions in these responses.

C Deterministic Rule Resources

The complete rule specification defines the cue expressions and tie-breaking rules. The recovery implementation and absence-filter im-

Development system	Primary	BERT F1	BLEURT	R-L F1	Align. F1
S0	0.481953	0.538438	0.425467	0.228711	0.176160
Recovery only	0.487424	0.549436	0.425412	0.234303	0.180515
Deduplication only	0.483089	0.540836	0.425341	0.230182	0.178117
RISK3	0.488955	0.552576	0.425334	0.235751	0.182860
All occurrences	0.487082	0.549369	0.424795	0.235156	0.181424
Broad cue	0.488878	0.552648	0.425108	0.235324	0.182632

Test system	Primary	BERT F1	BLEURT	R-1 F1	R-2 F1	R-L F1	Align. F1
S0	0.5906175	0.574387	0.606848	0.525847	0.404744	0.483165	0.350002
RISK3	0.5951115	0.583105	0.607118	0.529980	0.407315	0.487055	0.355469
GROUNDED	0.5955030	0.583888	0.607118	0.529980	0.407315	0.487055	0.355666
Broad cue	0.5956245	0.584131	0.607118	0.529980	0.407315	0.487055	0.356261

Table 9: Scores at the precision released by the evaluator or organizer workbook.

Split	Cells	Direct/chunked	Calls (uncached/total)	Input tokens	Output tokens	Mean/p95 latency (s)
Development	23	11/12	69/73	801,928	222,858	319.9/766.4
Test	117	112/5	131/131	558,429	180,892	163.4/350.8

Table 10: Frozen generation resources. Token totals and latency statistics cover successful uncached calls; four development calls were served from the local cache. Direct/chunked counts cells by generation route; chunked cells use chronological chunk generation followed by merge.

Setting	Frozen value
Temperature	0
Thinking output	disabled
Context window	40,000 tokens
Context reserve	512 tokens
Direct/chunk output cap	7,000 tokens
Merge output cap	12,000 tokens
Chunk prompt cap	16,000 tokens
Generation workers	4
Accepted finish reason	`stop`

Table 11: Frozen generation settings for S0.

Split	Merge calls	Max. output	Cap
Development	29	10,247	12,000
Test	5	2,879	12,000

Table 12: Frozen merge-call diagnostic. Output lengths and caps are tokens. All listed responses ended with `stop`; counts include cached calls and intermediate merges, not only the final merge per cell.

plementation provide the corresponding executable definitions. These resources are available at https://github.com/zqq-oso/lt4cpr-frozen-response-recovery.

The absence filter changed reports `damsafety.W3.k4` and `damsafety.W3.k6`, removing one fixed negative-template bullet from each. The other 115 test reports were unchanged relative to RISK3.

D Retrospective Training Analysis

The deterministic inventory takes `k2` from W2 and W3 and `k1` from W4 for eight training crises (24 cells). RISK3 recorded a primary score of 0.471905, versus 0.455685 for S0, a difference of +0.016220. BERTScore F1, BLEURT, ROUGE-L F1, and weighted alignment F1 changed by +0.032896, −0.000456, +0.013317, and +0.010259, respectively. Primary differences were positive in seven of eight crises. These cells also contributed to rule development.

E Manual Title-Compatibility Analysis

Output-alignment units. The development analysis compares final All-occurrences reports with S0. Within each cell, a content key comprises exact text, confidence, and ordered evidence IDs. Greedy one-to-one matching in report order pairs records with the same key, without preferring matches under the same title. Unmatched candidate records are called *additions*, unmatched baseline records *deletions*, and matched records with different titles *moves*. These names describe alignment outcomes, not executed operations. A retained placement can be classified as an addition when its identical baseline record is matched to a different placement.

The alignment yields 51 additions, 38 deletions, and 40 moves across 94 changed content groups. A group can contain both an addition and a move, or

both a deletion and a move; these record counts do not partition the 94 groups. The single annotator considered complete placement sets when interpreting group-level changes. A pairing was title-incompatible when its proposition did not fit the canonical title, irrespective of factual support. Of the 51 addition records, 23 had incompatible target titles. Content corresponding to all 38 deletion records remained elsewhere: 36 deletions were safe or beneficial, and two risked title-recall loss. These judgments do not give an incompatibility rate for all recovered entries.

Candidate occurrences and final placements. Table 5 counts candidate occurrences before recompilation: All-occurrences accepts 70, while RISK3 rejects 29 and accepts 41. Repeated candidates and subsequent compilation or deduplication prevent a one-to-one correspondence with final-output differences. The separate gate analysis uses final placements keyed by cell, report section, title, and exact text, rather than the metadata-sensitive content key above. It finds 19 development and 12 training-subset placements present in All-occurrences but absent from RISK3. All 31 were judged title-incompatible. These labels informed the gates and were not independently adjudicated; they support descriptive analysis rather than an independent estimate of policy precision.

F Comparisons with Recent Work

Long-document consolidation. BooookScore compares hierarchical and incremental book summarization and analyzes coherence errors (Chang et al., 2024). FABLES examines faithfulness and omissions using human annotations of book summaries (Kim et al., 2024). These narrative settings motivate distinguishing content preservation from grammatical fluency. Our analysis concerns a narrower unit: a canonical subsection and its eligible entries, present in a chunk response but absent from the compiled report.

Context use. NoLiMa reduces lexical overlap between questions and relevant passages to assess retrieval beyond literal matching (Modarressi et al., 2025). Our lexical gates address the narrower task of assigning recovered content to compatible titles within a fixed report hierarchy.

Structure and factual support. StructSum studies table and mind-map generation, separating factuality from global and local structural errors and applying critique-guided prompting (Jain et al., 2024). RefChecker extracts claim triplets and checks them against a reference context (Hu et al., 2024b). These works separate output organization from source support; our compiler and recovery pass focus on schema and title placement under fixed generation.

Evaluation granularity. FaithBench provides expert-labeled challenging hallucinations spanning multiple model families (Bao et al., 2025). L-CiteEval evaluates generation and citation quality in long-context tasks (Tang et al., 2025). These benchmarks motivate separate evaluation of report organization, human judgments of faithfulness, and the adequacy of cited support.

Overview of the LT4CPR 2026 Shared Task: Generating Situation Reports from Social Media Messages

Fei Xia[1], **Haotian Zhu**[1], **Yun Woo Lee**[1],
Eitan Grenev[1], Allison K Burton[1], Yingzhe Fan[1],
William D. Lewis[1], Antonis Anastasopoulos[2], Belu Ticona[2]
[1] University of Washington, [2] George Mason University
[1]{fxia, haz60, yunu919, eitang, burton25, yingzf, wlewis2}@uw.edu
[2]{antonis, mticonao}@gmu.edu

Abstract

Language Technologies for Crisis Preparedness and Response (LT4CPR) encompass a broad range of natural language applications designed to support disaster response and preparedness. Within this domain, our shared task focuses specifically on leveraging social media data to extract actionable information and generate structured situation reports. To establish a formal benchmark, we introduce a domain-specific taxonomy for categorizing critical crisis information, alongside a novel pipeline for generating synthetic datasets to support task training and evaluation. Furthermore, we develop two baseline systems and propose an evaluation methodology designed to leverage the structural framework of situation reports. The shared task attracted participation from nine teams, whose diverse approaches offer valuable preliminary insights into the feasibility of structured situation report generation and the effectiveness of our evaluation setup. Finally, we highlight key open challenges (such as data access, operational alignment, noise filtering) which needs to be addressed in future research.

1 Introduction

There has been a significant amount of research on the use of Language Technologies (LT) to process crisis-related social media content. The first reported instance of social media being used during a crisis (Twitter, specifically) was during the 2007 San Diego firestorm (Meier, 2015). It was not until 2012, however, that tweets were processed automatically using LT and crowds during Typhoon Pablo, using NLP tools developed by the Stand-by Task Force (Liu, 2014). The Stand-by Task Force was formed in the wake of the 2010 earthquake in Haiti (Liu, 2014), and although social media was not used during that crisis, significant numbers of SMS messages were sent (up to 5,000 per hour) and processed using NLP tools developed by the SBTF and local partners (Mission 4636) which, when paired with a large Haitian Kreyòl speaking crowd provided with mapping tools, were responsible for saving hundreds of lives (Munro, 2013, 2010; Lewis, 2010).

It was the Haitian effort that made it clear that language technologies were life-changing and life-saving technologies, paving the way for a significant uptick in LT being applied to social media content. The first survey focused on the processing of social media messages during crises was Imran et al. (2015). Although Imran et al. (2015) looked at all aspects of using and processing social media content during crises (*e.g.,* information needs, social media monitoring, characteristics of crisis-related social media text, etc.), they did look at NLP as a set of tools necessary to pre-process content before acting upon it (*e.g.,* triaging, geo-tagging, geolocating). Subsequent work by many of the same authors saw the development of large Twitter-focused datasets through the Qatar Computing Research Institute (QCRI), such as Human Aid (Alam et al., 2021), Crisis MMD (Alam et al., 2018), TBCOV (Imran et al., 2021). etc. Having readily available, large datasets such as these led to a significant uptick in research focused on the application of LT to crisis-related social media text.

A recent survey (Lewis et al., 2025) examined 355 papers published in a five-year period from January 2020 to December 2024, all of which focused on processing crisis-related social media content. The survey finds that most of the LT systems (202 out of 292, 69.2%) built in this period concentrated on classification tasks, with much fewer systems on tasks such as NER (40), topic modeling (19), geo-location (11) and summarization (11).

The goal of our shared task is to promote and facilitate LT4CPR research that goes beyond these relatively simple tasks. For 2026, the shared task is to generate situation reports from social media messages. A situation report can be seen as a structured summary with strict requirements on what

Proceedings of the First Workshop on Language Technologies for Crisis Preparedness and Response, pages 203–213
November 9, 2026 ©2026 Association for Computational Linguistics

information should be included. Nine teams from six countries participated in the task.

This study makes several key contributions to the emerging LT4CPR field. (i) We establish a new shared task (Section 2), grounded in a novel taxonomy tailored specifically for LT4CPR (Section 3). (ii) We engineer a novel method for generating a synthetic dataset customized for the shared task (Section 4). (iii) Beyond conventional evaluation measures, we introduce and implement a fine-grained report-item-level comparison measure (Section 5). (iv) We construct two baseline systems and comprehensively evaluate their performance against participating systems (Section 6).

To foster open science and facilitate downstream research, we have publicly released all associated datasets, the custom LT4CPR taxonomy, software utilities, and evaluation scripts via the shared-task website.[1]

2 The Shared Task

The shared task focuses on transforming crisis-related social media streams into structured situation reports. During an ongoing crisis, social media posts can provide timely information on hazards, casualties, infrastructure damage, population movement, requests for assistance, response activities, and other developments. However, these streams are also highly redundant, fragmented, and sometimes uncertain or contradictory. Individual posts often provide only partial observations, while multiple posts may describe the same underlying development using different wording. The goal of the task is therefore to aggregate information distributed across multiple social media posts into a concise and structured representation of the crisis situation.

2.1 The Input: Tweets with Metadata

In this shared task, the input is a set of social media posts plus some metadata. For convenience, we refer to those posts as *tweets* but the posts can come from any social media platforms, not limited to Twitter. Each input is stored as a JSONL file. A tweet is represented as a record in the JSONL file. An example is as follows:

```
{
  "record_type": "tweet",
  "id": 42,
  "text": "Two reported injuries in Terminal 1",
  "information_source": "Media",
  "timestamp": "Tue Jul 08 19:57:44 +0000 2025"
}
```

[1] https://lt4cpr.github.io/aacl2026-workshop-LT4CPR/shared-task.html

The record has five fields:

- The first field, "record_type", always has the value "tweet", to distinguish the tweet lines from lines for metadata.
- The second field, "id", is an identifier used to link tweets in the input with the report items in the system output.
- The third field is the main content of the tweet. While it is "text" for this year's shared task, it can be changed to other modalities such as audio and images in future shared tasks.
- The fourth field specifies the type of information source, which can be used as a factor when determining how reliable the tweet is.
- The last field is the timestamp of the tweet.

The metadata includes fields such as *title* (e.g., "Eruption of Mount Vurren in the Sarnat Highlands"), *hazard type* (e.g., "volcano"), and a reporting period (e.g., "from 1am to 11pm PST on Sept 10, 2025"). Only tweets that were posted within the reporting period and were related to this particular crisis should be used to generate the system output.

2.2 The Output: A Situation Report

In this shared task, the output is called a situation report (**SitRep**), which must follow a predefined format.

Sections and Subsections: Conceptually, a SitRep is a document consisting of sections and subsections. The titles of sections and subsections are fixed, as shown in the second and third columns of Table 1. The first section is always called *Situation overview* and it is a short summary of the crisis and specifies (1) what happened, where, and when; (2) human impact such as casualty and displacement figures; (3) the principal cause or mechanism (if established), and (4) the response and the current status. The second section of a SitRep is always called *Timeline*. It lists important events in chronological order.

Sections 3-11 in the SitRep cover different aspects of the crisis, such as casualties, hazard assessment, and aid provided. Each section is further divided into subsections, and each subsection includes only a subsection title and a bulleted list.

Among the 11 sections, the system output must include Sections 1 and 2. Other sections and subsections are optional; they should be included in

ID	Section title	Subsection title	Event Superclass	Event Type
1	Situation overview	NA	NA	NA
2	Timeline	NA	NA	NA
3	Casualties and human impact	Deaths, Injuries, Missing persons, Rescues	person_state_change	KILL, INJURY, MISSING, RESCUE
4	Infrastructure and service impact	Destruction, Service disruption	object_infra_change	DESTROY, DISRUPT_SERVICE
5	Displacement and movement	Evacuations, displacement, returns	movement_displacement	EVACUATE, DISPLACE, RETURN
6	Hazard assessment	Severity, Development of the hazard	hazard_state	SITUATION_STATE, HAZARDOUS_CONDITION
7	Response actions	Operational response	response	DEPLOY, COORDINATE
8	Communication and information	Warnings, Information updates	communication	WARN, REQUEST, INFORM
9	Aid and relief	Aid provided, Aid received, Donation	provide_help, receive_help	PROVIDE_HELP, RECEIVE_HELP
10	Organizational and administrative activity	Meetings, economic measures	org_admin	MEETING, POLICY_CHANGE, FUNDING
11	Social and community response	Social support, Public sentiment	social, cognitive	SOCIAL_SUPPORT, ATTITUDE

Table 1: Sections in a situation report and corresponding event types. The first three columns show Section ID, Section title, and Subsection titles in a situation report. The fourth column shows the Event Superclass that normally corresponds to that Section Title, and the last column shows the Event Types that normally correspond to the Subsection Titles in the third column.

the system output if and only if the input tweets provide relevant information for those (sub)sections.

A Report Item: To avoid confusion, we call a bullet in a bulleted list in a SitRep a *report item*. A SitRep should be stored as a JSON file. Below shows how a report item in the *Injuries* subsection is represented in this format (Comments starting with "//" are not part of the format).

```
{
  ... // metadata about this SitRep
  "sections": [
  ...  // Section 1 and 2 are here
  {
    "id": "3", // section id
   "title": "Casualties and human impact",
    "subsections": [
    {
      "id": "3b", // subsection id
      "title": "Injuries",
      "bullets": [
      {
        "id": "3b.1",
        "text": "Four injured at LAX",
        "confidence": "confirmed",
        "tweet_ids": [12, 18, 42]
      } // end of the current bullet
      ... // other bullets
    ]  // end of bulleted list in Sect 3b
    } // end of Sect 3b
    ... // other subsections
  ] // end of subsections in Section 3
  } // end of section 3
    ... // other sections
 ] // end of sections
} // end of the sitation report
```

Connection between input and output: The *tweet_ids* field in a report item links input tweets with the report items in the SitRep. Thus, this shared task combines the traditional summarization task with evidence grounding and structured generation.

3 LT4CPR Taxonomy

In a situation report, each subsection serves to summarize multiple events of the same type. For instance, the *Injuries* subsection summarizes all the *INJURY* events mentioned in the input tweets.

Event identification (often formulated as event detection or trigger extraction) has long been established as a core task within Information Extraction and NLP (e.g., (Walker et al., 2006; Chen et al., 2015; Xiang and Wang, 2019; Lai, 2022)). Building upon this long line of research and designed to facilitate structured analysis of crisis-related messages, we propose a LT4CPR-specific taxonomy

for *entities* and *events*. Events additionally incorporate typed arguments, epistemic status, negation, and optional event-specific attributes. This representation provides a standardized, structured interface bridging raw tweet text with downstream cross-message aggregation and summarization.

3.1 Entities

The entity inventory includes categories such as people, organizations, locations, hazards, infrastructure, resources, and temporal expressions. Table 2 comprises seven top-level entity types. Type-specific attributes provide additional structure wherever appropriate. For instance, a person entity can have an attribute called "address", which itself is a "loc" entity.

3.2 Events

The taxonomy organizes crisis event types hierarchically. Its superclasses encompass the major categories of crisis information, including changes in person state, infrastructure damage, movement and displacement, etc. More granular event types are defined beneath these broader classes (refer to the fourth column in Table 1).

In the event detection task, an event has a specific type, a textual trigger, zero or more event arguments, and a series of event attributes. Event arguments represent the participants and contextual information associated with an event. Our taxonomy uses PropBank-style argument labels together with event-specific semantic definitions. Core roles include `ARG0`, `ARG1`, and `ARG2`, while adjunct roles encode information such as location, time, cause, source, destination, manner, and purpose. For event attributes, the most important ones for us are *Realis* and *Negation*. Realis captures how the event is presented with respect to occurrence, uncertainty, planning, prediction, or conditionality. Negation indicates whether an event is negated.

3.3 Usage of the taxonomy

Our shared task leverages this taxonomy across multiple dimensions. First, we use it to define what each section and subsection in the situation report should cover. In general terms, an event superclass maps to a section and an event type maps to a subsection in the situation report, but the mapping is not strictly bijective (see Table 1); for instance, Section 1 or 2 does not correspond to a single event superclass. Second, we use the taxonomy in our synthetic dataset generation pipeline (Section 4) and our baseline systems (Section 6). Third, we release this taxonomy to shared task participants, who can choose to use or not use it in their participating systems.

4 The Dataset

Processing social media in mass emergencies is a well-established research domain (Imran et al., 2015), and underpinned by a sizable number of datasets and evaluations. For instance, a recent survey (Lewis et al., 2025) reveals that over 350 papers on processing crisis-related social media data were published between January 2020 and December 2024, but the vast majority of corpora built during that period were text-only, English-only, and sourced exclusively from Twitter. Because many disasters occur in non-English speaking countries, the value of creating resources for non-English languages cannot be overstated, especially if these resources are intended for real-world use.

However, owing to structural shifts in the social media landscape over the past several years, accessing or releasing social media text has become increasingly prohibitive. Notably, access to Twitter/X content has been severely restricted—either through restrictive paywalls or the mandate to "rehydrate" content (meaning only pointers to tweets can be shared, rather than the raw text itself)—thereby drastically reducing the volume of readily available content. Without accessible text corpora, data collection and tool development have grown significantly more difficult. Even when open collections of social media data can be located, generating gold-standard situation reports for them remains exceptionally challenging.

To address these persistent obstacles, we developed a robust framework that produces synthetic data tailored for the shared task.

4.1 Dataset Generation

The generation pipeline is anchored in a foundational principle: *the semantic plan precedes the post*. Instead of generating tweets first and situation reports subsequently, we begin with a concise narrative description of the crisis. The pipeline proceeds through several steps:

1. Select a hazard type (e.g., "floods") and instantiate an instance of that type by randomly populating the hazard attributes (e.g., "flooding at Region X that lasts 10 days").

Entity Type	Attribute	Attribute values
person	gender	male, female, etc.
	age_group	child, youth, adult, elderly, unknown.
organization	org_type	government, ngo, emergency_service, company, military, unspecified_organization
hazard	hazard_type	Hierarchical natural, human-induced, health, conflict, utility, and unspecified hazards.
infrastructure	infra_type	Transportation, utility, and public-service infrastructure.
resource	resource_type	food, water, electricity, shelter, fuel, medical, phone, internet, other
time	time_type	absolute, duration, recurring, unknown
loc	loc_type	geopolitical locations, natural locations, and addresses/intersections

Table 2: Top-level entity types and their attributes in the LT4CPR Taxonomy

2. Randomly generate report items that align with the hazard type (e.g., "15 people were injured", "100 people are missing"; "Several bridges collapsed").

3. For each report item, randomly determine the number of events that are relevant to that item. For example, for the item "15 people were injured", establish whether the injuries occurred within a single INJURY event or across multiple distinct ones (e.g., 3 injured at school Y, 4 at home, 8 on the highway).

4. For each event, we synthesize multiple tweet segments that explicitly reference the event. These segments may vary in their lexical realization of triggers, the inclusion or omission of event arguments and attributes, overall syntactic phrasing, etc. (e.g., "Oh, my god. Three people died today" vs. "several possible fatalities in Chicago").

5. Construct complete tweets by combining one or more tweet segments, ensuring that the resulting messages exhibit coherent linguistic styles. Furthermore, we enrich each tweet by incorporating metadata such as relevant hashtags, assigning a precise timestamp, and allocating a unique identifier.

6. Compile a gold situation report by appending tweet IDs to corresponding report items, organizing report items into structured bulleted lists, and adding section and subsection titles.

Because several of these phases entail stochastic processes, the pipeline can be executed iteratively to scale the dataset. In practice, since certain pipeline stages necessitate LLM API calls (e.g., Step 5, to enhance the natural fluency of the generated tweets), we reduce API costs by permitting different instances in the dataset to share selected tweets or report items.

4.2 Hazard Subtypes

The hazard entity type is defined hierarchically in our LT4CPR taxonomy and the top level comprises:

- **Natural hazards**: geophysical, hydrological, meteorological, and climatological hazards;
- **Human-induced hazards**: chemical spills, toxic releases, gas leaks, oil spills, nuclear incidents, radiation leaks, explosions, industrial fires, and hazardous waste;
- **Health hazards**: epidemics, pandemics, disease outbreaks, and environmental exposure;
- **Conflict hazards**: armed conflict, civil unrest, and terrorism; and
- **Utility hazards**: power outages, grid failures, dam failures, and bridge/building collapse risk.

Natural-hazard leaf nodes encompass, among others, earthquakes and aftershocks, tsunamis, volcanoes, landslides, avalanches, floods, thunderstorms, blizzards, hurricanes, typhoons, cyclones, tornadoes, heavy rain, strong winds, wildfires, droughts, heatwaves, and cold waves.

4.3 Corpus statistics

Table 3 summarizes the statistics of the generated dataset. The hazard subtypes for training, dev, and test sets are mutually exclusive (non-overlapping).

While the dataset is modest in scale, it can readily be expanded across multiple dimensions—including size, linguistic diversity (multilinguality), modality, and other factors such as data noise and the presence of conflicting information.

	train	dev	test
# of hazard subtypes	8	2	6
# of instances	140	23	117
Avg # of tweets/instance	283	372	104

Table 3: Statistics of the shared-task dataset. Hazard subtypes are discussed in Section 4.2.

5 Evaluation Measures

To evaluate a system, we compare its output against the gold standard. Ideally, the two should exhibit identical structural organization with a corresponding sequence of section and subsection titles. However, recall that Section 3-11 in a SitRep and their respective subsections should be present only if the input tweets have relevant events; thus, the system output and the gold standard could end up having different structures if the gold standard contains errors or if the system errs in identifying events or fails to strictly adhere to the output format requirements. Therefore, several methodological decisions are required for evaluation.

Skip two sections: We elect to ignore the first two sections in the SitRep because establishing a reliable gold standard for those sections requires more studies (e.g., what events should be treated as important and thus be included in the Timeline section).

Measures for comparing plain text: To compare two pieces of plain text, we utilize several well-known measures, including ROUGE-1, ROUGE-2, and ROUGE-L (Lin, 2004), BERTScore (Zhang et al., 2020) and BLEURT (Sellam et al., 2020).[2]

Unit of comparison: Text can be compared hierarchically at four different levels: document level (where all structural properties in the reports are ignored), section level (where subsection boundaries are ignored), subsection level (where bullet boundaries are ignored), and bullet level (aka item level, which provides finer granularity).

For the shared task, we apply ROUGE, BERTScore, and BLEURT at the subsection level and then micro-average the results to derive a score at the test instance level.

We also implement an item-level evaluation leveraging the Hungarian Algorithm (Kuhn, 1955) to find an optimal alignment between report items in the system output and the gold report (see Appendix A).

Primary Score: For system-level results, we compute the mean of the scores for test instances within the same hazard and subsequently compute the mean across all hazards to ensure equal weighting across all hazards.

The official primary score is defined as:

$$\text{Primary} = \frac{\text{BERTScore}_{F_1} + \text{BLEURT}}{2}. \quad (1)$$

This score emphasizes semantic similarity notwithstanding variations in wording. ROUGE and item-level alignment are reported as complementary measures and are excluded from the official primary score calculation.

6 Baselines and Participating Systems

In this section, we describe two baseline systems we created for the shared task. Both systems integrate heuristics and Large Language Models (LLMs).

6.1 Baseline System #1

The first baseline uses a structured three-stage pipeline to process a collection of crisis-related tweets and generate a situation report.

Stage 1 runs an LLM (GPT-5.2) on individual tweets to identify entities and events in the tweets. The taxonomy outlined in Section 3 is used to formulatethe prompt for the LLM.

Stage 2 performs cross-tweet coreference by clustering entities (and events) identified in Stage 1 so that the objects in each cluster ideally refer to the same object in the real world.

Stage 3 uses event clusters from Stage 2 to generate a situation report. A section or subsection with no qualifying events is omitted.

6.2 Baseline System #2

The second baseline generates reports by LLMs and its main focus is on examining how the degree of preprocessing in the input affects situation-report generation.

Figure 1 shows the pipeline of the system. There are five options for the input: Level A consists of input tweets only; Level B adds the LT4CPR taxonomy; Levels C through E add the output produced by Baseline #1's Stage 1 or 2 (see Fig 2).

A template-based prompt generator fills a template with the input. It places the relevant input

[2]We use `microsoft/deberta-xlarge-mnli` for BERTScore and BLEURT-20 for BLEURT.

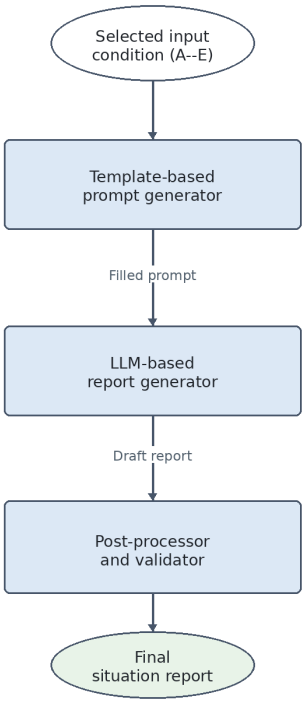


Figure 1: Common pipeline for all five conditions. Data products are shown as ellipses and processing modules as boxes; labels on arrows denote intermediate outputs.

alongside the reporting instructions and includes the taxonomy whenever the condition calls for it. The filled prompt is sent to an LLM to generate a draft report.

The draft report is then sent to a deterministic post-processor, which brings the response into the required report structure, resolves duplicated or misordered sections, and discards empty material. The processor then checks whether every cited tweet identifier occurs in the current test input. A rule-based validator finally checks the report schema, section names, confidence labels, and citation requirements.

6.3 Participating systems

Nine teams participated in the shared task; six of them submitted system description papers afterwards, which are included in the LT4CPR workshop proceedings. Most systems follow a similar pipeline that includes tweet classification, event extraction from individual tweets, clustering of tweets, entities or events, report item generation and verification, and report section serialization.

7 Results

Nine teams participated in the shared task. They were permitted to submit multiple systems but were required to designate one as their primary submission. One team submitted four systems, one submitted three, and two teams each submitted two systems. The remaining five teams each submitted only one system. In total, 16 systems were submitted.

7.1 Primary Submission Results

Table 4 reports the primary submission results for all participating teams, alongside the two UW baseline systems. The rows are sorted by the primary metric. Unsurprisingly, the ROUGE scores tend to be lower than those of BERT, BLEURT, and the Primary metrics, as ROUGE relies on exact n-gram matching.

The numbers in the *Align* column tend to be lower than in the other columns because it evaluates performance at the report-item level rather than the subsection level. Despite these differences, all metrics yield consistent system rankings.

7.2 Results of Best System per Team

Table 5 reports the highest-scoring submission from each participating team, regardless of whether it was designated as their primary submission. Notably, ranking teams by their best-performing submission yields an identical ordering to the primary submission leaderboard.

To evaluate whether performance gaps between adjacent rows in Table 5 are statistically significant on the test data with 117 instances, we conducted paired bootstrap and permutation tests on the primary scores. Pairs exhibiting a statistically significant difference at $p < 0.01$ are indicated with double asterisks on the higher-ranked system. Only four of the ten adjacent pairs show statistical significance, indicating that the score differences between closely ranked systems fall largely within the margin of statistical variation.

7.3 Performance of Baseline #2

The performance of Baseline #2 is comparable to the top participation systems. Table 6 illustrate its performance with five types of input and three LLMs: Gemini 3.8 Flash (**?**), GPT-5.6 Luna (**?**), and Claude Haiku 4.5 (**?**).

Among the three LLMs, Gemini produced the best results than the other two models except for

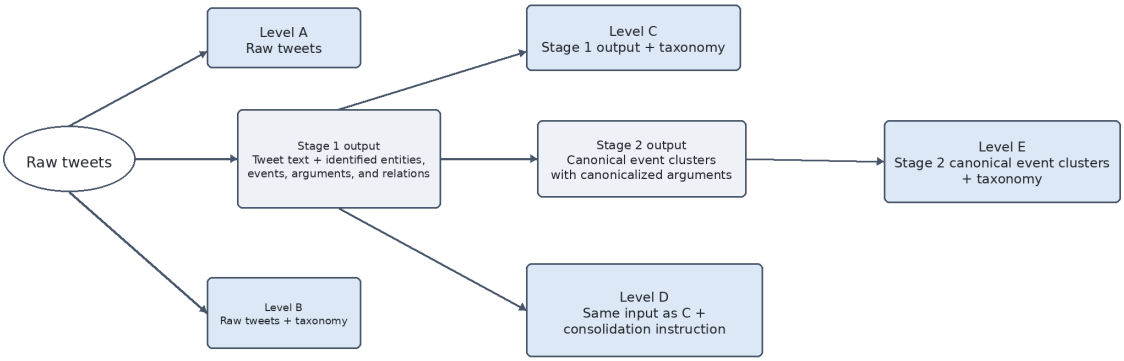


Figure 2: Derivation of the five input conditions. C and D receive the same Stage 1 data; D differs only by its consolidation instruction. E receives canonical event clusters produced by Stage 2.

Team	System	R1	R2	RL	BERT	BLEURT	Align	Primary
Stockmark	report-refinement	**0.6158**	**0.4481**	0.5486	**0.7269**	**0.6362**	**0.4635**	**0.6816**
UTS	v6	0.5820	0.4156	0.5331	0.7009	0.6275	0.4434	0.6642
UW Baseline #2	Level A (Gemini)	0.5988	0.4359	**0.5513**	0.7210	0.6000	0.4569	0.6605
Dy	rules-v2	0.5338	0.3684	0.4873	0.6739	0.6265	0.4056	0.6502
Meng-qiu in Zhongguancun	grounded	0.5300	0.4073	0.4871	0.5839	0.6071	0.3557	0.5955
Amrood	nemotron-120b-sitrep	0.5142	0.3927	0.4790	0.5984	0.5483	0.3715	0.5733
gogogo	r01g-sr	0.4645	0.3585	0.4398	0.5018	0.5528	0.2881	0.5273
UW System 1	Without grouping	0.3835	0.2129	0.3276	0.5297	0.5135	0.2331	0.5216
BMU	system_1	0.3594	0.2636	0.3219	0.4511	0.5170	0.2388	0.4841
Neural Coder	extraction-de	0.4331	0.3385	0.4127	0.5082	0.4514	0.3278	0.4798
Databaes	g5-evidence-grounded	0.2846	0.1715	0.2524	0.3947	0.5153	0.2022	0.4550

Table 4: Results of the teams' **primary** submissions and the two UW baselines, ordered by official Primary score. The highest score in each column is in boldface. R1, R2, RL, BERT, and Align denote ROUGE-1, ROUGE-2, ROUGE-L, BERTScore, and item-level aligned F_1, respectively. Primary is the mean of BERTScore F_1 and BLEURT.

Level E. Among five levels of input, the level that produces the highest score varies from LLM to LLM.

8 Challenges

This shared task highlights many challenges that our LT4CPR research still faces.

Data access: The first is access to social media data, which has become increasingly constrained due to evolving platform API policies, heightened privacy regulations, and shifting data governance frameworks. These restrictions severely impede researchers' ability to collect localized, real-time text streams during unfolding emergencies, making it difficult to train and evaluate models on real-world crisis data.

Community need: Another challenge is to determine what the crisis response community actually needs on the ground. Without a clear understanding of the operational requirements of first responders and humanitarian organizations, computational models risk optimizing for generic language targets rather than actionable intelligence. Establishing these grounded practitioner needs is crucial, as it directly guides how system outputs should be formatted and dictates the specific metrics against which these technologies must be evaluated.

System design The third challenge relates to system architecture and model design. Social media data is inherently noisy, uncurated, and frequently inundated with conflicting statements or deliberate misinformation. To be effective, an LT4CPR system must incorporate robust filtering mechanisms capable of isolating critical situational signals from background noise and verifying factuality under time-sensitive conditions. Furthermore, systems must gracefully reconcile conflicting reports that arise during rapidly evolving emergency situations.

A related core difficulty lies in event coreference and entity resolution—specifically, accurately identifying and aggregating disparate posts that refer to the exact same real-world incident despite varia-

Team	System	R1	R2	RL	BERT	BLEURT	Align	Primary
Stockmark	report-refinement	**0.6158**	**0.4481**	0.5486	**0.7269**	**0.6362**	**0.4635**	**0.6816****
UTS	v6mbr	0.5770	0.4089	0.5300	0.7114	0.6221	0.4440	0.6667
UW Baseline #2	Level A (Gemini)	0.5988	0.4359	**0.5513**	0.7210	0.6000	0.4569	0.6605
Dy	rules-v2	0.5338	0.3684	0.4873	0.6739	0.6265	0.4056	0.6502**
Meng-qiu in Zhongguancun	cue	0.5300	0.4073	0.4871	0.5841	0.6071	0.3563	0.5956
Amrood	nemotron-120b-sitrep	0.5142	0.3927	0.4790	0.5984	0.5483	0.3715	0.5733**
gogogo	r01g-sr	0.4645	0.3585	0.4398	0.5018	0.5528	0.2881	0.5273
UW Baseline #1	Without grouping	0.3835	0.2129	0.3276	0.5297	0.5135	0.2331	0.5216**
BMU	system_1	0.3594	0.2636	0.3219	0.4511	0.5170	0.2388	0.4841
Neural Coder	extraction-de	0.4331	0.3385	0.4127	0.5082	0.4514	0.3278	0.4798
Databaes	g5-evidence-grounded	0.2846	0.1715	0.2524	0.3947	0.5153	0.2022	0.4550

Table 5: Results of each participating team's **best** system and the UW baselines. The highest score in each column is shown in bold. Significance markers compare each system with the immediately following system, using the larger of the paired bootstrap and permutation p-values. ** denotes $p < 0.01$. Displayed scores use the official hierarchical aggregation, whereas significance tests use per-instance Primary scores.

System	A	B	C	D	E
Gemini	**66.1**	65.2	65.2	66.0	59.6
GPT-Luna	62.9	61.3	63.0	**64.2**	59.3
Claude	61.5	62.7	60.9	62.4	**63.4**

Table 6: Performance of Baseline #2 with three LLMs and five types of input (in primary score, multiplied by 100). A-E refers to Level A-E, as defined in Fig 2. The highest score in each row is in boldface.

tions in user phrasing, terminology, and contextual noise.

Evaluation measures: There is the need for improved evaluation metrics. While current methods leverage the structural format of situation reports, plain-text comparison still largely relies on n-gram metrics like ROUGE or embedding-based scores such as BERTScore. However, these metrics often fail to capture critical factual precision. A question-answering (QA)-based approach focused on specific report items—such as verifying whether "five people were killed"—offers a more plausible and reliable way to measure factual accuracy.

9 Conclusion

In this overview paper, we presented the inaugural LT4CPR shared task on automatic situation report generation from crisis-related social media streams. By introducing a domain-specific event taxonomy, a systematic synthetic data generation pipeline, multiple baseline architectures, and a structured evaluation methodology, we established a rigorous benchmark for evaluating natural language generation systems in high-stakes emergency contexts.

The participation of nine teams and the submission of sixteen diverse systems demonstrate both the feasibility of structured situation report generation and the vibrancy of the research community surrounding crisis informatics. Experiments show all measures produce almost identical system rankings and the score gaps between closely ranked systems are largely statistically non-significant.

Looking forward, key challenges persist across the field. Future work must address constrained access to real-time social media data, align model outputs more closely with the operational needs of first responders, build systems capable of filtering noise and resolving conflicting misinformation, and adopt fact-oriented evaluation frameworks—such as QA-based metrics—that go beyond traditional n-gram and embedding similarity. Addressing these open problems will be vital for deploying trustworthy, real-world language technologies in emergency management.

Ultimately, we hope this shared task and its accompanying resources will catalyze continued innovation at the intersection of language technologies and crisis preparedness and response, fostering tools that are not only technologically advanced but also operationally dependable for emergency responders and affected communities.

Acknowledgments

This work was supported in part by National Science Foundation under Grant No. CNS-2346335. We want to thank all the participanting teams for their support. We would also like to extend our gratitude to Yuqing Guo, Nicole N Mihaylova, and Margot Kleinpeter for their valuable contributions.

References

Firoj Alam, Ferda Ofli, and Muhammad Imran. 2018. CrisisMMD: Multimodal Twitter Datasets from Natural Disasters. In *Proceedings of the 12th International AAAI Conference on Web and Social Media (ICWSM)*.

Firoj Alam, Umair Qazi, Muhammad Imran, and Ferda Ofli. 2021. Humaid: Human-annotated disaster incidents data from twitter with deep learning benchmarks. In *Proceedings of the 15th International AAAI Conference on Web Social Media (ICWSM)*, pages 933–942. AAAI.

Yubo Chen, Liheng Xu, Kang Liu, Daojian Zeng, and Jun Zhao. 2015. Event extraction via dynamic multi-pooling convolutional neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 167–176. Association for Computational Linguistics.

Muhammad Imran, Carlos Castillo, Fernando Diaz, and Sarah Vieweg. 2015. Processing social media messages in mass emergency: A survey. *ACM Comput. Surv.*, 47(4).

Muhammad Imran, Umair Qazi, and Ferda Ofli. 2021. Tbcov: Two billion multilingual covid-19 tweets with sentiment, entity, geo, and gender labels. arXiv:2110.03664.

Harold W. Kuhn. 1955. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1–2):83–97.

Viet Dac Lai. 2022. Event extraction: A survey.

William Lewis, Haotian Zhu, Keaton Strawn, and Fei Xia. 2025. Tapping into Social Media in Crisis: A Survey. In *Proceedings of the 4th Workshop on NLP for Positive Impact, EMNLP 2025*, vienna, Austria.

William D. Lewis. 2010. Haitian Creole: How to Build and Ship an MT Engine from Scratch in 4 Days, 17 Hours, & 30 Minutes. In *Proceedings of the 14th EAMT*, Saint Raphaël, France.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Sophia Liu. 2014. Crisis crowdsourcing framework: Designing strategic configurations of crowdsourcing for the emergency management domain. *Computer Supported Cooperative Work (CSCW)*, 23:389–443.

Patrick Meier. 2015. *Digital Humanitarians*. CRC Press, Boca Raton.

Robert Munro. 2010. Crowdsourced Translation for Emergency Response in Haiti: The Global Collaboration of Local Knowledge. In *AMTA Workshop on Collaborative Crowdsourcing for Translation*.

Robert Munro. 2013. Crowdsourcing and the crisis-affected community: lessons learned and looking forward from mission 4636. *Journal of Information Retrieval*, 16.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.

Christopher Walker, Stephanie Strassel, Julie Medero, and Kazuaki Maeda. 2006. Ace 2005 multilingual training corpus.

Yue Xiang and Houfeng Wang. 2019. A survey of event extraction technology. *IEEE Access*, 7:173111–173137.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with BERT. In *International Conference on Learning Representations*.

A Item-Level Evaluation

For item-level comparison, we first break each situation report into a set of subsections and align them using subsection titles.

For a subsection d, let G_d and S_d denote the bulleted list from the gold report and the system output, resp. To calculate the similarity between G_d and S_d, we need to start with the similarity w_{ij} between the i gold item and the j system item. Let R_{ij} be their ROUGE-L F-score and J_{ij} the Jaccard similarity of the two items' tweet-id sets. Now we can define w_{ij} as a linear combination of the two scores:

$$w_{ij} = \lambda * R_{ij} + (1 - \lambda) J_{ij}. \quad (2)$$

For the shared task evaluation, we set λ to be 0.8 to give more weights to the ROUGE score.

Now we have calculated s_{ij} for all the (i, j) pairs, we run the Hungarian Algorithm (Kuhn, 1955) to find an optimal alignment that maximizes the total weights of the aligned items.[3]

Let W_d be the total weight of the optional alignment for subsection d. Let m and n be the number of report items across all the subsections in the gold

[3]From the optimal alignment, we retain only pairs with $w_{ij} > 0.1$ to discard negligible assignments rather than enforce strict semantic equivalence. This preserves partial credit for related items while treating weak matches as unmatched.

report and the system report, resp. Precision and recall are defined as follows:[4]

$$P = \frac{\sum_d W_d}{n}, \quad R = \frac{\sum_d W_d}{m} \tag{3}$$

The resulting F1 is the item-level score for the system report.

[4]According to this definition, if a gold item i and a system item j are connected according to the optimal assignment, the edge is considered a partial match and its score w_{ij} is added to the total match score.

Author Index

Anastasopoulos, Antonios, 203
Atiq, Muhammad Ahsan, 166

Bansal, Vipasha, 46, 78
Brown, Elizabeth, 78
Burton, Allison K, 203
Butt, Mohib Ahmad, 166

Chan, Hok Ming Adeline, 1
Chen, Xiaofei, 177
Chersoni, Emmanuele, 1
Chiusaroli, Francesca, 32

Daoud, Youcef, 134

Fan, Yingzhe, 203

Gao, Yuqiu, 177
Gonella, Giacomo, 20
Grenev, Eitan, 203
Guerini, Marco, 20
Guha Majumdar, Sreeja, 156
Gupta, Anuridhi, 115

Han, Lifeng, 32
Hirota, Wataru, 141

Kumar, Sachin, 65

Lee, Yun Woo, 203
Lewis, William D., 46, 78, 203

Menini, Stefano, 20
Mihaylova, Nicole, 78
Monti, Johanna, 32

Pong, Benjamin, 78
Purohit, Hemant, 115, 126

Qiu, Le, 1

Sakata, Wataru, 141
Senarath, Yasas, 126
Shaheen, Nusrat, 166
Song, Yan, 177, 188
Staiano, Maria Carmen, 32

Tian, Yuanhe, 177, 188
Ticona, Belu, 203

Vaidya, Gayatri Ramchandra, 126

Wang, Ming, 188
Wilson, Jen, 46

Xia, Fei, 203

Yang, Zhentao, 177

Zhang, Zhuan Anne, 1
Zheng, Quan, 188
Zhu, Haotian, 203