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

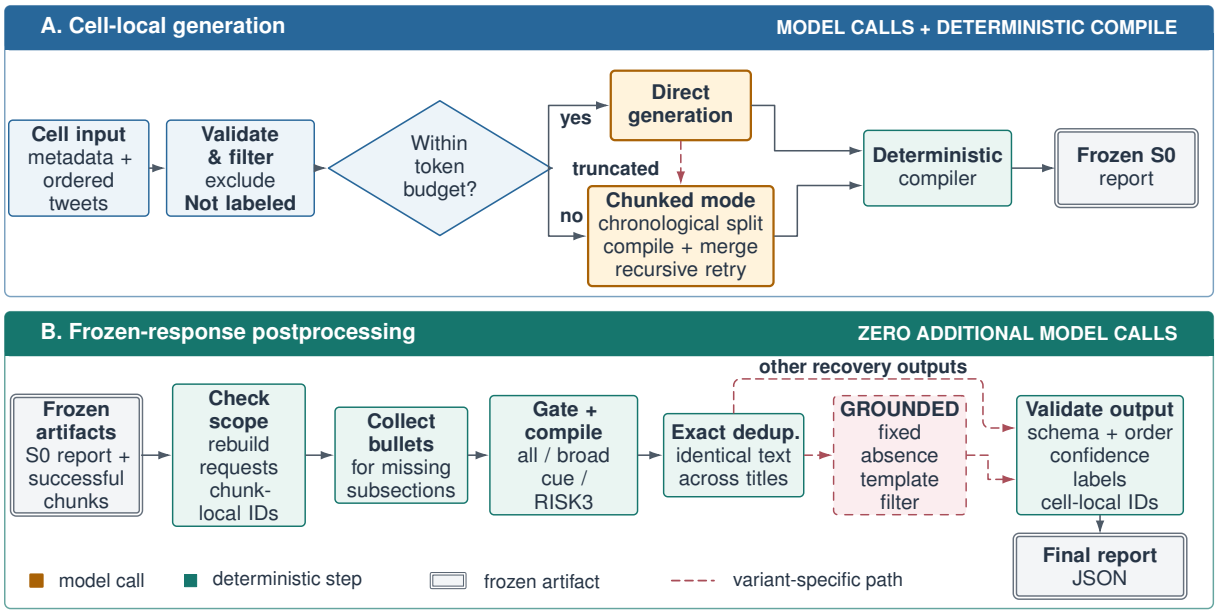


Figure 1: Two-phase system architecture. Phase A produces frozen S0 through direct or chunked generation. Phase B checks chunk provenance and evidence-ID scope, recovers missing subsections and eligible entries, deduplicates exact cross-title text, and validates report structure. Only GROUNDED applies 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.