LLMs as Span Annotators: A Comparative Study of LLMs and Humans

Zdeněk Kasner[1] **Vilém Zouhar**[2] **Patrícia Schmidtová**[1]
Ivan Kartáč[1] **Kristýna Onderková**[1] **Ondřej Plátek**[1]
Dimitra Gkatzia[3] **Saad Mahamood**[4] **Ondřej Dušek**[1] **Simone Balloccu**[5]
[1]Charles University [2]ETH Zurich [3]Edinburgh Napier University
[4]trivago N.V. [5]TU Darmstadt, Germany
Contact: kasner@ufal.mff.cuni.cz

Abstract

Span annotation – annotating specific text features at the span level – can be used to evaluate texts where single-score metrics fail to provide actionable feedback. Until recently, span annotation was done by human annotators or fine-tuned models. In this paper, we study whether large language models (LLMs) can serve as an alternative to human annotators. We compare the abilities of LLMs to skilled human annotators on three span annotation tasks: evaluating data-to-text generation, identifying translation errors, and detecting propaganda techniques. We show that overall, LLMs have only moderate inter-annotator agreement (IAA) with human annotators. However, we demonstrate that LLMs make errors at a similar rate as skilled crowdworkers. LLMs also produce annotations at a fraction of the cost per output annotation. We release the dataset of over 40k model and human span annotations for further research.[1]

1 Introduction

Fine-grained aspects of texts, such as semantic accuracy or coherence, depend on local lexical choices. To reflect these aspects in quality judgments of texts, techniques are needed that provide the appropriate amount of detail. However, most automatic evaluation metrics for Natural Language Generation (NLG) assign only singular scores for the whole text per each evaluated aspect (Gkatzia and Mahamood, 2015; Sai et al., 2023; Schmidtová et al., 2024). Although numerical values make it easy to rank systems, these metrics are too crude and susceptible to biases or miscalibration of the underlying models (Gehrmann et al., 2023; Liu et al., 2024; Wang et al., 2024; Gao et al., 2024).

The subject of our study, *span annotation* (Figure 1), offers an alternative approach. Instead of assigning a single score for each evaluated aspect, the goal of span annotation is to localize text spans of interest and classify them according to task-specific guidelines. Span annotations are aligned to specific parts of the evaluated text, which makes them more explainable and actionable than numerical ratings.

Despite its advantages, span annotation has not yet been widely applied in automatic NLG evaluation. The method traditionally required human annotators, making it costly and difficult to scale (Da San Martino et al., 2019; Thomson and Reiter, 2020; Popovic, 2020; Kocmi et al., 2024c). The *LLM-as-a-judge* paradigm recently emerged as a promising solution to this problem (Zheng et al., 2023; Gu et al., 2024), allowing task-specific applications (Kocmi and Federmann, 2023; Hasanain et al., 2024). However, to our knowledge, no study has systematically compared span annotation performance between LLMs and human annotators.

The central focus of our investigation is comparing human annotators and state-of-the-art LLMs on span annotation tasks. We select three span annotation tasks (cf. Section 3.1): evaluating data-to-text generation (Thomson and Reiter, 2020), identifying errors in machine translation (Kocmi et al., 2024a), and detecting propaganda techniques in human-written texts (Da San Martino et al., 2019).

Our contributions are as follows:

1. We establish that with structured outputs and detailed annotation guidelines, LLMs can serve as robust span annotators, yielding relevant spans for all three annotation tasks we work with (Sections 3.2 and 5.1).

2. We show that LLMs have moderate inter-annotator agreement with human annotators overall, but can reach the agreement level among verified crowdworkers who passed a qualification task (Section 5.2).

3. We discover the sources of model errors by

[1]Project website: https://llm-span-annotators.github.io

Proceedings of the First Workshop on Multilingual Multicultural Evaluation, pages 1–22

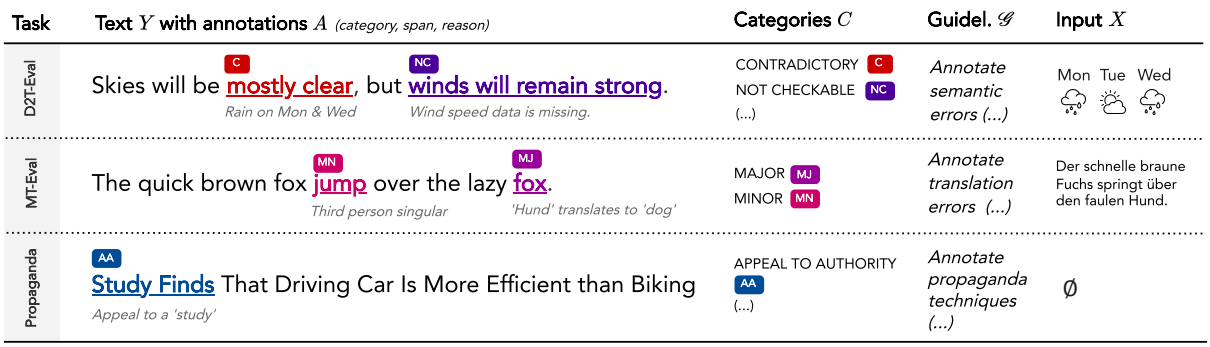


Figure 1: Examples of span annotation tasks that we automate with LLMs. We unify the setup for evaluation tasks (D2T-EVAL, MT-EVAL) and text analysis tasks (PROPAGANDA).

manually analyzing a subset of LLM annotation outputs (Section 5.3).

4. We release a dataset of more than 40k human and model annotations, including annotations collected from crowdworkers and reasoning traces from reasoning LLMs.

2 Related Work

LLMs for NLG Evaluation. Automatic NLG metrics traditionally assess text quality by measuring similarity to human-written reference texts (Sai et al., 2023; Schmidtová et al., 2024). As such, they are unable to quantify more fine-grained aspects (Gehrmann et al., 2023; Freitag et al., 2021) and do not correlate well with human judgments (Novikova et al., 2017; Reiter, 2018). With the emerging LLM-as-a-judge paradigm (Gu et al., 2024), LLMs have been applied as evaluators on various tasks, using simple numeric scoring (Bavaresco et al., 2025; Liu et al., 2023; Sottana et al., 2023; Leiter et al., 2023; Chiang and Lee, 2023), or free-form feedback (Li et al., 2024; Kim et al., 2024a,b; Kartáč et al., 2025). However, as these outputs are not firmly grounded in text, they tend to miss fine-grained aspects and are influenced by LLM biases (Stureborg et al., 2024; Koo et al., 2024; Wang et al., 2024).

Span Annotation Protocol. In machine translation (MT), span annotation is a long-standing component of protocols such as MQM or ESA (Lommel et al., 2014; Mariana, 2014; Popovic, 2020; Kocmi et al., 2024c), where human annotators mark erroneous spans in translations. In data-to-text (D2T) generation, span annotation was applied by Thomson and Reiter (2020), who introduced a span-based evaluation protocol for annotation of generated basketball match reports. Span annotation is also used to judge intrinsic text qualities, such as coherence or use of rhetorical devices, in tasks such as propaganda detection (Da San Martino et al., 2019) and text summarization (Subbiah et al., 2024). Unlike our work, these works focus on span annotation with human annotators.

Automatic Span Annotation. Early attempts at automating span annotation with ad-hoc guidelines were based on fine-tuned pre-trained encoder models. That includes evaluation of MT (Guerreiro et al., 2024), D2T generation (Kasner et al., 2021) or text summarization (Goyal et al., 2022), as well as propaganda detection (Martino et al., 2020; Goffredo et al., 2023; Piskorski et al., 2023). Automating span annotation with LLMs is more flexible and benefits from increasing LLM capabilities. We build on work that applies LLMs as a task-specific evaluation tool (Kocmi and Federmann, 2023; Fernandes et al., 2023; Hasanain et al., 2024; Kasner and Dušek, 2024; Chang et al., 2024; Zouhar et al., 2025; Kartáč et al., 2025; Ramponi et al., 2025). Furthermore, Semin et al. (2026) recently investigated various strategies for automatic span annotation with LLMs. Our work is the first that systematically compares the performance of LLMs to human annotators.

3 Automating Span Annotation with LLMs

We first formally introduce the span annotation process in Section 3.1. Next, we discuss how to automate the process with LLMs in Section 3.2 and how to evaluate the quality of span annotations in Section 3.3.

3.1 Span Annotation: Task Definition

The aim of span annotation is to annotate a **text sequence** $Y = \langle y_1, \ldots, y_n \rangle$ given:

- the set of **categories** $C = \{c_1, \ldots, c_k\}$,
- the annotation **guidelines** $\mathcal{G}$,
- the **source** X (such as the translation source; empty if we are annotating only intrinsic text aspects).

The output is a set of annotations $A = \{a_1, \ldots, a_m\}$, where each annotation a_i is a tuple $\langle s_i, e_i, c_i, r_i \rangle$:

- $s_i, e_i \in \{1, \ldots, n\}, s_i < e_i$ are the start and end indices of the annotated span,
- $c_i \in C$ is the assigned annotation category,
- r_i is a reason for the annotation (optional).

3.2 Span Annotation with LLMs

In our setup, annotations A for the given input $\langle Y, C, \mathcal{G}, X \rangle$ are collected from an LLM:

$$A = \text{LLM}(\text{prompt}(Y, C, \mathcal{G}, X)).$$

To obtain the annotations, we follow the setup of Kasner and Dušek (2024): we request the list of annotations in JSON format, using constrained decoding with a fixed JSON scheme to ensure that the output is syntactically valid. We require each annotation to contain the fields `reason` (the explanation r_i), `text` (the textual content of the span), and `type` (the integer index of the error category c_i).[2]

For reasoning models not supporting structured output, we retrieve the raw answer from the model, strip any parts within the `<think></think>` tags (if present), and consider the latest valid top-level JSON object as the model's response.

3.3 Evaluating Span Annotations

To compare annotations automatically, we need a notion of similarity between two sets of annotations $\mathcal{A} = \{A_1, A_2, \ldots, A_{|Y|}\}$ and $\hat{\mathcal{A}} = \{\hat{A}_1, \hat{A}_2, \ldots, \hat{A}_{|Y|}\}$ over a set of texts $\mathcal{Y} = \{Y_1, Y_2, \ldots, Y_{|Y|}\}$. Note that basic inter-annotator agreement metrics such as Cohen's κ (Cohen, 1960) are not applicable in our case, as they require a fixed set of annotation units, while the number and position of spans in the span annotation task may differ (Mathet et al., 2015). Therefore, we consider the following similarity metrics:

Pearson correlation ρ over counts. This metric compares how many spans were annotated for each example:

$$\text{Pearson}(\mathcal{A}, \hat{\mathcal{A}}) = \rho(\langle |A_Y|, |\hat{A}_Y| \rangle_{Y \in \mathcal{Y}}) \quad (1)$$

The correlation serves as a sanity check: a low value would suggest that an annotator either skips examples or over-annotates, indicating unclear annotation guidelines.

Precision, Recall, and $\mathbf{F_1}$. To quantify the degree of alignment between individual annotations, we compute precision, recall, and F_1 as defined by Da San Martino et al. (2019). These measures are on matching annotations, adjusted to give partial credit to imperfect matches:

$$\text{Precision}(A_Y, \hat{A}_Y) = \frac{1}{|A_Y|} \sum_{a \in A_Y} \frac{|a \cap \hat{a}|}{|a|}, \quad (2)$$

$$\text{Recall}(A_Y, \hat{A}_Y) = \frac{1}{|\hat{A}_Y|} \sum_{\hat{a} \in \hat{A}_Y} \frac{|a \cap \hat{a}|}{|\hat{a}|}, \quad (3)$$

where $a \cap \hat{a}$ is the character overlap between two annotation spans and $|a| = e - s + 1$ is the length of the annotation span in characters (see Section 3.1). Subsequently, we compute the F_1-score as the harmonic mean of precision and recall.

For each of the metrics, we consider *soft* and *hard* variants. The *hard* variant only considers overlaps where the span category matches, while the soft variant disregards the categories. We consider the hard variant to be the default. In addition, we report the difference $F_1\Delta = F_1(\text{soft}) - F_1(\text{hard})$.

Gamma γ. The F_1 score is sensitive to varying span granularities and does not consider near-matches with no overlap or agreement by chance. To this end, we follow Da San Martino et al. (2019) and Hasanain et al. (2024) in using the γ score (Mathet et al., 2015) as a complementary metric. The metric builds the best possible alignment between the sets of annotations A_Y and $\hat{A}_Y$ and computes the "disorder" of this alignment based on the *positional* and *categorical* dissimilarities of aligned annotations. The score ranges from $-\infty$ to 1, where 1 is achieved when the annotations

[2] Following Castillo (2024), we ensure that the `reason` field is generated first.

Task	# Cat.	# Texts	Avg. Len	Novel Data
D2T-EVAL	6	1,296	118/715	✔
MT-EVAL	2	2,854	26/185	✗
PROPAGANDA	18	100	914/4,659	✗

Table 1: Overview of span annotation tasks used in our experiments. *# Cat.* denotes the number of categories used in the task (see Appendix C for their listings), *# Texts* the number of texts annotated, *Avg. Len* the average number of words/characters in the output, and *Novel Data* indicates newly collected data.

are perfectly aligned. The γ score extends Krippendorff's α (Krippendorff, 1980), another popular metric, by computing the category-aware span alignments. We use the implementation of Titeux and Riad (2021).

$S_\emptyset$ score. For an output y, one or both annotation sets $A_Y, \hat{A}_Y$ may be empty. This is in fact desirable, e.g., if the goal is to annotate errors in an output that is entirely correct. However, these cases are not properly reflected by the other scores we are using: the F1 score only focuses on counting error spans and is not affected by true negatives, and the γ score is undefined if any of the two annotation sets is empty (these examples therefore need to be skipped during the γ computation). To compensate for this, we introduce a score $S_\emptyset$that is computed for examples where any of A_Y, $\hat{A}_Y$ is empty:

$$S_\emptyset = 1/(1 + |A|), \tag{4}$$

where:

$$|A| = \begin{cases} |A_Y| & \text{if } |\hat{A}_Y| = 0, \\ |\hat{A}_Y| & \text{otherwise.} \end{cases} \tag{5}$$

The score is equal to 1 for the cases where no annotator produced any annotation (i.e., a perfect match) and decreases proportionally to the number of annotations from the annotator that provided a non-zero number of annotations.

4 Experiments

4.1 Tasks

We cover three span annotation tasks of different qualitative aspects. We focus on tasks that do not have extensive training data resources and cannot be readily solved by encoder models (such as, e.g., named entity tagging): evaluating data-to-text generation (D2T-EVAL; Section 4.1.1), identifying errors in machine translation (MT-EVAL; Section 4.1.2), and detecting propaganda techniques (PROPAGANDA; Section 4.1.3). See Table 1 for an overview of our datasets.

4.1.1 D2T-EVAL: Evaluation of Data-to-text Generation

In D2T-EVAL, we use span annotation to evaluate semantic accuracy and stylistic aspects of data-to-text generation outputs (Sharma et al., 2022; Celikyilmaz et al., 2020). The inputs X are the structured data used to generate the output text Y.

We use D2T-EVAL as a control task to mitigate the effects of *data contamination*: the fact that the performance of the model might be inflated by previous exposure to publicly available benchmarks (Balloccu et al., 2024; Dong et al., 2024; Jiang et al., 2024). Instead of using an existing dataset, we use the QUINTD tool (Kasner and Dušek, 2024) to download structured inputs from multiple public APIs.[3] To obtain output texts for the structured data, we prompt LLMs in a zero-shot setting, asking them to generate a summary of the given data using approximately five sentences. Note that we do not need to deal with the factuality of outputs here, as the sole purpose of the texts is being the input to the annotation process (in fact, having some number of errors is desirable). See Appendix B.1 for more details.

To gather annotations for our dataset, we use crowdworkers from Prolific.com. We apply best practices for gathering human annotations, including an iterative process to refine annotation guidelines and preselecting the best-performing annotators using a qualification task (Tseng et al., 2020; Iskender et al., 2020; Huang et al., 2023; Zhang et al., 2023). Our process of collecting annotations proceeded in two stages, following the setup of Zhang et al. (2023): (1) a *qualification task* for preselecting skilled annotators, and (2) the *main task* for collecting the annotations. See Appendix B.2 for more details on collecting annotations.

For quality checks, we collect additional internal gold annotation (by the authors) for subsets of the data: $\mathcal{D}_{\texttt{dev}}$ for selecting the best prompt and $\mathcal{D}_{\texttt{iaa}}$ for validating the performance of human annotators (cf. Appendix B.2). Here is a complete overview of our data splits for D2T-EVAL:

- $\mathcal{D}_{\texttt{test}}$ (1200 outputs) – for LLM evaluation, annotated with crowdworkers,

[3] We selected two of the existing domains: `openweather` for generating weather forecasts and `gsmarena` for generating phone descriptions. We also add the `football` domain (using RapidAPI - API-Football) for generating soccer game reports.

- $\mathcal{D}_{\texttt{dev}}$ (84 outputs) – for the study of prompt variants, annotated internally,
- $\mathcal{D}_{\texttt{iaa}}$ (12 outputs) – control for human crowdworkers, annotated internally.

4.1.2 MT-EVAL: Identifying Errors in Machine Translation

For MT-EVAL, we use the dataset of system outputs from the WMT 2024 general shared task (Kocmi et al., 2024b). The system outputs were annotated with the Error Span Annotation (ESA) protocol (Kocmi et al., 2024c) by professional translators.

The inputs X for MT-EVAL are the texts in the source language used to produce the translation Y. We follow the WMT 2024 annotation guidelines, focusing on character-level span annotations of *Major* and *Minor* translation errors (see Table 6 for their definitions). Note that unlike for the other tasks, the annotations in MT-EVAL cannot overlap and need not be aligned with word boundaries.

We select the three textual domains present in the WMT 2024 shared task: `news`, `literary`, and `social`; using the data translated from English into other languages: Chinese, Czech, German, Hindi, Icelandic, Japanese, Russian, Spanish, and Ukrainian.

The original dataset has nearly 50k model outputs, making it too extensive for our evaluation campaign. Therefore, we used a balanced subsample: For each of the nine language pairs, we randomly sample ten input translation segments. We then take all available system outputs for these 90 input segments, making up 2,854 examples in total.

4.1.3 PROPAGANDA: Propaganda Technique Detection

For the PROPAGANDA task, we use the dataset of Da San Martino et al. (2019) containing news collected mostly from on-line propagandistic sources. The token-level annotations in the dataset created by expert annotators cover 18 categories of logical fallacies and persuasion techniques. We use the `test` split for our experiments. Inputs X are empty for this task, as all annotated categories are intrinsic to the evaluated text Y.

4.2 Collecting LLM annotations

Models For collecting span annotations with LLMs, we use a mixture of open and proprietary state-of-the-art models:

	Llama 3.3			DeepSeek-R1		
Prompt	F_1	γ	#a/o	F_1	γ	#a/o
$\mathcal{P}_{\texttt{base}}$	0.20	0.13	2.4	**0.25**	**0.20**	1.0
$\mathcal{P}_{\texttt{cot}}$	0.09	0.10	0.8	0.24	0.19	1.1
$\mathcal{P}_{\texttt{5shot}}$	**0.25**	**0.18**	2.5	0.21	0.16	1.4
$\mathcal{P}_{\texttt{noguide}}$	0.11	0.08	3.4	0.20	0.16	1.6
$\mathcal{P}_{\texttt{noreason}}$	0.22	0.13	2.2	0.24	0.18	1.1

Table 2: Comparison of prompting techniques on the $\mathcal{D}_{\texttt{dev}}$ (#a/o is the average number of annotations per output).

- **instruction-tuned models**: Llama 3.3 70B (Grattafiori et al., 2024), GPT-4o (Hurst et al., 2024), and Claude 3.7 Sonnet (Anthropic, 2025),
- **reasoning models**:[4] DeepSeek-R1 70B (DeepSeek-AI, 2025), o3-mini (OpenAI, 2025), and Gemini 2.0 Flash Thinking (Deepmind, 2025).

See Appendix A for details on our experimental setup.

Prompts We define several prompt variants for our experiments. $\mathcal{P}_{\texttt{base}}$ is the base prompt that includes the guidelines $\mathcal{G}$ as given to human annotators and asks the model to explain its annotation. By extending $\mathcal{P}_{\texttt{base}}$, we implement a few-shot prompt adding 5 examples ($\mathcal{P}_{\texttt{5shot}}$) and a chain-of-thought prompt simply asking the model to produce intermediate reasoning ($\mathcal{P}_{\texttt{cot}}$). We also ablate $\mathcal{P}_{\texttt{base}}$ by removing extended guidelines ($\mathcal{P}_{\texttt{noguide}}$) and not asking for explanations ($\mathcal{P}_{\texttt{noreason}}$). The full prompts can be found in Appendix D.

5 Results

We first investigate the effect of prompting techniques in Section 5.1. Next, we evaluate the LLM annotations using automatic metrics (Section 5.2) and manually analyze the errors in the model outputs (Section 5.3).

5.1 Prompting Techniques

We perform preliminary experiments on the D2T-EVAL task $\mathcal{D}_{\texttt{dev}}$ set using open models (Llama 3.3 and DeepSeek-R1) to study the differences between prompting techniques. The results are shown in Table 2.

[4] By *reasoning* models we understand the models that use extra inference time to generate a thinking trace before providing the answer (Marjanović et al., 2025).

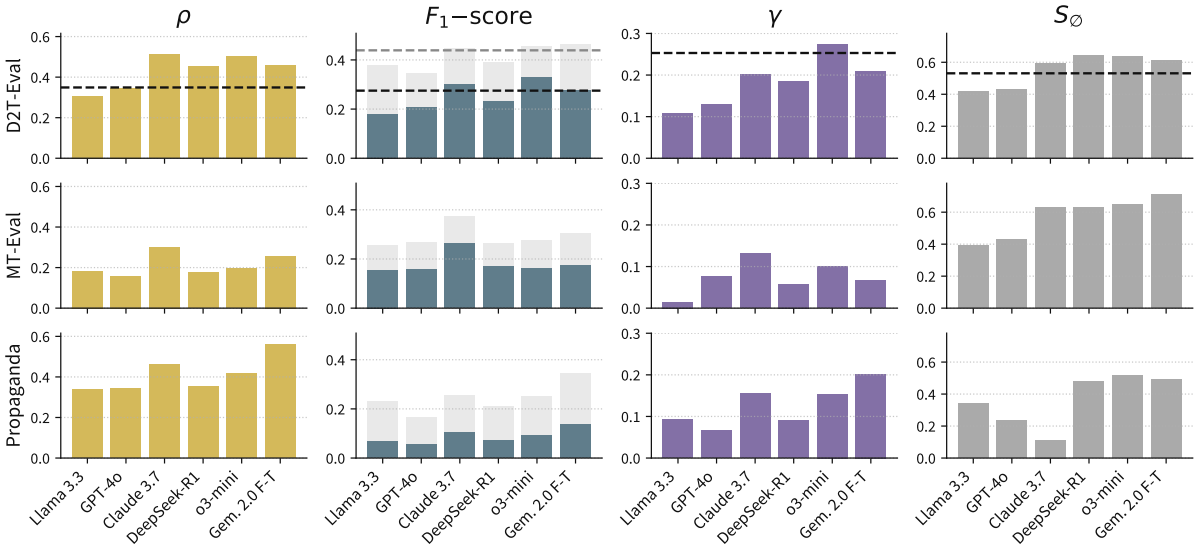


Figure 2: Comparison between LLMs using $\mathcal{P}_{\texttt{base}}$ and human annotators. Rows represent different tasks (see Section 4.1), columns show different annotation similarity metrics (see Section 3.3). For the F_1 score, the shadow bar denotes its *soft* variant. The dashed horizontal lines denote agreement between our human annotators for D2T-EVAL (the agreement is not available for the external datasets). More detailed results are included in Tables 11 to 14.

Including detailed guidelines seems beneficial: omitting the guidelines ($\mathcal{P}_{\texttt{noguide}}$) lowers the performance of both models. In contrast, not letting the model explain the annotation ($\mathcal{P}_{\texttt{noreason}}$) does not have a substantial effect. For Llama 3.3, the chain-of-thought (CoT) prompting ($\mathcal{P}_{\texttt{cot}}$) makes it produce fewer annotations per example than the base variant (0.8 vs. 2.4), leading to lower F1 and γ scores. Llama 3.3 with $\mathcal{P}_{\texttt{cot}}$ tends to "overthink" the annotations, deciding not to annotate cases of errors against which it can find some arguments.

Few-shot prompting ($\mathcal{P}_{\texttt{5shot}}$) brings ambivalent results, increasing Llama 3.3 scores but doing the opposite for DeepSeek-R1. This observation is aligned with DeepSeek-AI (2025), who note that few-shot prompting degrades the performance of DeepSeek-R1. Given these considerations, we decided to use $\mathcal{P}_{\texttt{base}}$ for further experiments.

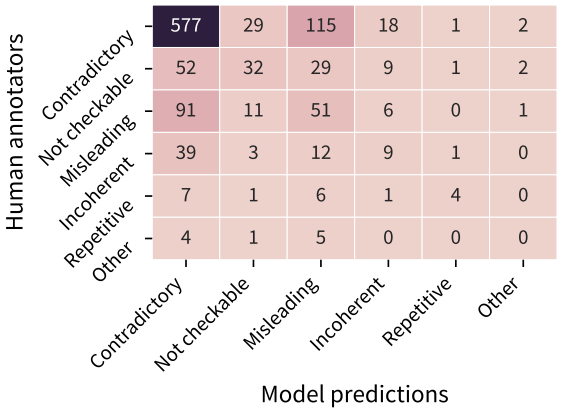


Figure 3: Confusion matrix for D2T-EVAL (*Contradictory*, *Not checkable*, *Misleading*, *Incoherent*, *Repetitive*, *Other*), averaged across models (see Table 5 for category descriptions).

5.2 LLM vs. Human Annotations

Next, we compare LLM and human annotations using the metrics described in Section 3.3. The overall results for all tasks are given in Figure 2. We provide detailed results for individual tasks in the Appendix F.

Reasoning models outperform instruction-tuned models. DeepSeek-R1 generally outperforms the same-sized Llama 3.3.[5] Its superiority is most pronounced on D2T-EVAL (F_1-score of 0.23 vs. 0.18, γ score of 0.19 vs. 0.11). The same observation applies to OpenAI models, where o3-mini outperforms GPT-4o. A notable exception to this trend is the non-reasoning Claude 3.7 Sonnet, which scores mostly on par with o3-mini and excels at MT-EVAL.

[5]The models are comparable as the 70B distilled variant of DeepSeek-R1 is based on Llama 3.3 70B. See Section A.1 for details.

Model	Cost ($/1k out)	Time (s/out)
crowdworkers	500	129.1
Llama 3.3	-	21.6
DeepSeek-R1	-	227.5
Claude 3.7 Sonnet	10.5	9.0
o3-mini	3.6	21.8

Table 3: Estimate of costs and time requirements on D2T-EVAL: crowdworkers on Prolific, open models (Llama 3.3, DeepSeek-R1), and proprietary models (Claude 3.7 Sonnet, o3-mini).

LLMs reach human IAA on D2T-EVAL, PROPAGANDA is harder. For D2T-EVAL, we compare model results with an average IAA on a subset of examples annotated by two human annotators. Here, o3-mini, Claude 3.7 and Gemini 2.0 mostly reach or surpass human agreement. For PROPAGANDA, the upper bound of IAA is the result of Da San Martino et al. (2019), who report $\gamma = 0.31$ for annotators before consolidation. This score is substantially higher than LLMs (the best LLM score being $\gamma = 0.16$ for Claude 3.7 Sonnet). However, this task used expert annotators and has the largest number of categories. The latter property is reflected in the large difference between the soft and hard F_1 scores.[6]

Models confuse related categories. Confusion matrices (see Figure 3 and Appendix F) suggest that the models tend mainly to confuse related categories, which may be related to ambiguity or subjective understanding of category definitions. The models also use a less diverse distribution of categories than human annotators.

LLMs are more cost- and time-efficient than human annotators. An important factor when comparing LLMs and human annotators is efficiency with respect to cost and time per output. For D2T-EVAL, crowdsourced annotation for 1k outputs costs approximately $500, while annotating the same amount of outputs with the high-performance o3-mini LLM costs $3.60 (see Table 3). In terms of time, the crowdworkers take 129.1 seconds per output on average, which is better than DeepSeek-R1 70B running on our local infrastructure, but an order of magnitude slower than the API-based models.[7] Therefore, LLMs are a more efficient alternative in terms of costs and time.

5.3 Manual Analysis of LLM Annotations

To gain more insights into the qualitative aspects of LLM annotations, we manually analyzed the quality of LLM annotations on 216 samples from D2T-EVAL and PROPAGANDA.[8] For each model, we sampled three annotations per category in D2T-EVAL and one annotation per category in PROPAGANDA. Without access to the annotation source, we classified the annotations and their explanations as *Correct*, *Partially correct*, *Wrong category*, *Incorrect*, and *Undecidable*.

We show the results in Figure 4 and Tables 19 and 20. In total, we marked 49.5% of LLM-generated annotations and 50.5% of reasons as correct (with 9.2% of annotations and 12.5% of reasons additionally marked as partially correct). Reasoning models perform better, with 56.4% of their annotations and 58.3% reasons marked as completely correct. The most accurate annotations on D2T-EVAL were those made by Gemini 2.0 and DeepSeek-R1. o3-mini performed well on both tasks, although PROPAGANDA proved challenging for all models.

What are the sources of model errors? We find that the models often select wrong error categories despite identifying real issues (e.g., labeling *Contradictory* statements as *Incoherent*). Models also tend to be overly attentive, flagging noise in the data (e.g., markup or off-topic content in PROPAGANDA) as errors, or marking slight numerical variations (such as rounded values) as misleading. All of these cases could be tackled by more descriptive guidelines or additional examples. However, in some cases, the models also misread or misinterpreted the data (e.g., claiming wind speed measurements do not exist when they do), which hints at deeper issues with understanding the data. Incorrect explanations vary from incomplete explanations (addressing only part of a multi-issue span), irrelevant explanations (e.g., appealing to facts that are "missing" from the text) to incorrectly flagging subjective statements (e.g., "*long-lasting usage*")

[6] We omit the comparison with human IAA in MT-EVAL. While the WMT24 dataset for MT-EVAL contains examples annotated with a pair of annotators, these examples take up only a small fraction and exhibit high variance between language pairs.

[7] Note that we do not ask the crowdworkers to give us a reason r for the annotation, which would arguably make the responses of the crowdworkers slower.

[8] The analysis was split among 7 authors of this paper. While we did not do double annotation due to lack of time, we discussed any unclear cases throughout the process. We do not include MT-EVAL in the manual analysis due to our insufficient expertise in most target languages.

Figure 4: Results of our manual analysis. We analyzed 18 annotations and their explanations for each model and task (216 annotations in total). The color bars show annotations that we classified as *Correct*, *Partially correct*, *Wrong category*, *Incorrect*, and *Undecidable*. Detailed results are provided in Tables 19 and 20.

as factual errors. Occasionally, the model admits that it marked a correct span as an error, such as in *"The description of the game's duration aligns with the data, providing coherent information"*.[9]

How good are human annotations? Concerningly, the LLM annotations that were marked as correct have only 24% hard character-level overlap (51% soft) with human annotations. This fact led us to analyze the quality of human annotations in D2T-EVAL (the task in which we had the necessary domain expertise). Using the same methodology as we used for the LLM annotations, we annotated a limited sample of 108 human annotations. We marked 45.3% of the annotations as *Correct*, which is comparable to the LLM annotations (see Table 4 for the results). These findings suggest that the task is hard even for human annotators, and the quality of annotations from crowdworkers varies, even if they are preselected using a qualification task.

6 Discussion

Here, we summarize our findings and discuss the implications of our results.

Can LLMs substitute human annotators? The IAA between LLMs and human annotators is only moderate, suggesting LLMs cannot straightforwardly replace human annotators. However, using LLMs may be a reasonable option in scenarios based on crowdworkers, where the strongest LLMs reach the average IAA between human annotators themselves. In other cases, when deciding whether to employ LLMs as span annotators, one needs to balance desired output quality with other practical aspects. Here, LLMs provide better flexibility, shorter response times, and lower costs. One should also consider the quality of available human annotators, as even qualified crowdworkers (i.e., those who passed a qualification task) make similar amounts of errors as LLMs. It can be also assumed that LLM-based span annotation will benefit from future increases in LLM capabilities, while crowdworkers may increasingly rely on LLM to complete tasks (Veselovsky et al., 2023). A promising solution seems to be a hybrid approach in which LLMs pre-annotate the text and humans post-edit the annotations (Zouhar et al., 2025).

How to deploy LLMs as span annotators? We recommend providing LLMs with detailed guidelines that describe conventions and how to handle ambiguous cases (cf. Figure 5). In contrast, we do not recommend providing specific examples (cf. Figure 8), as this approach did not bring consistent improvements. Arguably, this is due to the length and complexity of the examples, making them distracting to the model. When using LLMs with custom categories or guidelines, we recommend validating the model's annotations against examples hand-annotated by experts on a sample of the data. In general, reasoning models tend to provide more reliable annotations at the cost of higher response times and token count.

Is the task meaningful despite the low scores? As pointed out by an anonymous reviewer of this paper, the low annotation accuracy – as found by our manual analysis – may indicate a fundamental limitation of the proposed evaluation setup. Span annotation is indeed complex and leaves more room for subjectivity than more straightforward annotation such as simple labels or scores. However, we argue that the detailed actionable feedback gained through span annotation outweighs the increased noise, both in terms of explainability of the outputs and their potential for further processing.

[9] This typically happened to GPT-4o, even though OpenAI API ensures JSON key ordering so the explanation *should* have been generated before the annotation (cf. Section 3.2).

Source	Annotations				
	C	P	W	I	U
Human annotators	49	4	17	31	7

Table 4: Manual evaluation results for human annotators on D2T-EVAL. Categories: C=*Correct*, P=*Partially correct*, W=*Wrong category*, I=*Incorrect*, U=*Undecidable*.

7 Conclusion

We showed that LLMs can serve as span annotators for three span annotation tasks: evaluating data-to-text generation, identifying errors in machine translation, and detecting propaganda in human-written texts. Our experiments show that LLMs achieve moderate agreement with skilled human annotators. The models perform best in D2T-EVAL, where they are comparable to verified crowdworkers who passed a qualification task. Reasoning models consistently outperform their instruction-tuned counterparts, delivering more accurate annotations and providing more valid explanations for their decisions. Automating span annotation with LLMs seems to be a promising alternative to fine-grained human evaluation sourced from crowdsourcing platforms, opening the way towards scalable and actionable automatic NLG evaluation methods.

Limitations

Although we aimed to select a representative sample of models, prompts and tasks, our choice is constrained by our limited time frame and budget. Our estimates of the upper-bound IAA for each task are difficult to establish and depend on many factors, such as the chosen annotation categories, their ambiguity, the annotation guidelines, or the qualification level of human annotators. The estimates are also not readily available for existing datasets and require additional data collection. Due to our insufficient expertise in the target languages, we also do not provide language-specific manual error analysis of results.

As an evaluation method, span annotation is not well-suited for certain NLG evaluation tasks such as annotating omissions or rating the overall text style. In these cases, it is best to combine span annotation with other evaluation methods.

Author Contributions

DG and SB first came up with the idea for the project, with SB further coordinating and overseeing the research process. ZK led the experimental design and execution part, including conducting both preliminary and main experiments, organizing the crowdsourcing campaigns, and processing the collected data. Multiple authors (DG, IK, KO, SB, SM, VZ, ZK) participated in the collection of gold data for D2T-EVAL. Similarly, multiple authors (IK, KO, OD, OP, PS, SB, ZK) were involved in manual evaluation of the model outputs. DG provided financial resources for the Prolific campaigns. SM and SB provided expertise in preparing annotation guidelines and structuring the Prolific campaigns. Data processing and analysis were handled mainly by ZK, VZ, and PS, with VZ providing extra support with the WMT data. The paper was written by ZK, VZ, IK, PS, OD, and SB.

Acknowledgments

This work was funded by the European Union (ERC, NG-NLG, 101039303). It was additionally supported by the National Recovery Plan funded project MPO 60273/24/21300/21000 CEDMO 2.0 NPO and the Charles University Research Centre program No. 24/SSH/009. It used resources of the LINDAT/CLARIAH-CZ Research Infrastructure (Czech Ministry of Education, Youth, and Sports project No. LM2023062). We thank David M. Howcroft for his early input and contributions to the research methodologies adopted in this study.

Ethics Statement

The human evaluation study was approved by the internal ethics committee of our institution. Our human annotators were hired over Prolific and paid the platform-recommended wage of 9 GBP/hour (adjusted to slightly higher rates to account for real annotation times). Annotators were pre-selected on the basis of their primary language (English). All annotators were shown detailed instructions and explanation of the data types, data sources, and the purpose of the research. The domains were selected so that they do not contain sensitive or potentially offensive content. We do not collect demographic data about participants.

References

Marah I Abdin, Sam Ade Jacobs, Ammar Ahmad Awan, Jyoti Aneja, Ahmed Awadallah, Hany Awadalla, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Harkirat S. Behl, Alon Benhaim, Misha Bilenko, Johan Bjorck, Sébastien Bubeck, Martin Cai, Caio

César Teodoro Mendes, Weizhu Chen, Vishrav Chaudhary, Parul Chopra, Allie Del Giorno, Gustavo de Rosa, Matthew Dixon, Ronen Eldan, Dan Iter, Amit Garg, Abhishek Goswami, Suriya Gunasekar, Emman Haider, Junheng Hao, Russell J. Hewett, Jamie Huynh, Mojan Javaheripi, Xin Jin, Piero Kauffmann, Nikos Karampatziakis, Dongwoo Kim, Mahoud Khademi, Lev Kurilenko, James R. Lee, Yin Tat Lee, Yuanzhi Li, Chen Liang, Weishung Liu, Eric Lin, Zeqi Lin, Piyush Madan, Arindam Mitra, Hardik Modi, Anh Nguyen, Brandon Norick, Barun Patra, Daniel Perez-Becker, Thomas Portet, Reid Pryzant, Heyang Qin, Marko Radmilac, Corby Rosset, Sambudha Roy, Olatunji Ruwase, Olli Saarikivi, Amin Saied, Adil Salim, Michael Santacroce, Shital Shah, Ning Shang, Hiteshi Sharma, Xia Song, Masahiro Tanaka, Xin Wang, Rachel Ward, Guanhua Wang, Philipp A. Witte, Michael Wyatt, Can Xu, Jiahang Xu, Sonali Yadav, Fan Yang, Ziyi Yang, Donghan Yu, Chengruidong Zhang, Cyril Zhang, Jianwen Zhang, Li Lyna Zhang, Yi Zhang, Yue Zhang, Yunan Zhang, and Xiren Zhou. 2024. Phi-3 Technical Report: A Highly Capable Language Model Locally on Your Phone. *CoRR*, abs/2404.14219.

Anthropic. 2025. Claude 3.7 Sonnet.

Simone Balloccu, Patrícia Schmidtová, Mateusz Lango, and Ondrej Dusek. 2024. Leak, Cheat, Repeat: Data Contamination and Evaluation Malpractices in Closed-source LLMs. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2024 - Volume 1: Long Papers*, pages 67–93, St. Julian's, Malta.

Anna Bavaresco, Raffaella Bernardi, Leonardo Bertolazzi, Desmond Elliott, Raquel Fernández, Albert Gatt, Esam Ghaleb, Mario Giulianelli, Michael Hanna, Alexander Koller, André F. T. Martins, Philipp Mondorf, Vera Neplenbroek, Sandro Pezzelle, Barbara Plank, David Schlangen, Alessandro Suglia, Aditya K. Surikuchi, Ece Takmaz, and Alberto Testoni. 2025. LLMs instead of Human Judges? A Large Scale Empirical Study across 20 NLP Evaluation Tasks. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), ACL 2025*, pages 238–255, Vienna, Austria.

Dylan Castillo. 2024. Structured Outputs: Don't Put the Cart Before the Horse.

Asli Celikyilmaz, Elizabeth Clark, and Jianfeng Gao. 2020. Evaluation of Text Generation: A Survey. *CoRR*, abs/2006.14799.

Yapei Chang, Kyle Lo, Tanya Goyal, and Mohit Iyyer. 2024. BooookScore: A systematic exploration of book-length summarization in the era of LLMs. In *The Twelfth International Conference on Learning Representations, ICLR 2024*, Vienna, Austria.

David Cheng-Han Chiang and Hung-yi Lee. 2023. Can Large Language Models Be an Alternative to Human Evaluations? In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023*, pages 15607–15631, Toronto, Canada.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Giovanni Da San Martino, Seunghak Yu, Alberto Barrón-Cedeño, Rostislav Petrov, and Preslav Nakov. 2019. Fine-grained analysis of propaganda in news article. 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 5636–5646, Hong Kong, China. Association for Computational Linguistics.

Google Deepmind. 2025. Gemini 2.0 Flash Thinking.

DeepSeek-AI. 2025. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. *CoRR*, abs/2501.12948.

Yihong Dong, Xue Jiang, Huanyu Liu, Zhi Jin, Bin Gu, Mengfei Yang, and Ge Li. 2024. Generalization or Memorization: Data Contamination and Trustworthy Evaluation for Large Language Models. In *Findings of the Association for Computational Linguistics, ACL 2024*, pages 12039–12050, Bangkok, Thailand.

Patrick Fernandes, Daniel Deutsch, Mara Finkelstein, Parker Riley, André F. T. Martins, Graham Neubig, Ankush Garg, Jonathan H. Clark, Markus Freitag, and Orhan Firat. 2023. The Devil Is in the Errors: Leveraging Large Language Models for Fine-grained Machine Translation Evaluation. In *Proceedings of the Eighth Conference on Machine Translation, WMT 2023*, pages 1066–1083, Singapore.

Markus Freitag, George F. Foster, David Grangier, Viresh Ratnakar, Qijun Tan, and Wolfgang Macherey. 2021. Experts, Errors, and Context: A Large-scale Study of Human Evaluation for Machine Translation. *Trans. Assoc. Comput. Linguistics*, 9:1460–1474.

Mingqi Gao, Xinyu Hu, Jie Ruan, Xiao Pu, and Xiaojun Wan. 2024. LLM-based NLG Evaluation: Current Status and Challenges. *CoRR*, abs/2402.01383.

Sebastian Gehrmann, Elizabeth Clark, and Thibault Sellam. 2023. Repairing the Cracked Foundation: A Survey of Obstacles in Evaluation Practices for Generated Text. *J. Artif. Intell. Res.*, 77:103–166.

Dimitra Gkatzia and Saad Mahamood. 2015. A Snapshot of NLG Evaluation Practices 2005 - 2014. In *ENLG 2015 - Proceedings of the 15th European Workshop on Natural Language Generation, 10-11 September 2015, University of Brighton*, pages 57–60, Brighton, UK.

Pierpaolo Goffredo, Mariana Espinoza, Serena Villata, and Elena Cabrio. 2023. Argument-based Detection and Classification of Fallacies in Political Debates.

In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023*, pages 11101–11112, Singapore.

Tanya Goyal, Junyi Jessy Li, and Greg Durrett. 2022. SNaC: Coherence Error Detection for Narrative Summarization. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 444–463, Abu Dhabi, United Arab Emirates.

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*.

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, Yuanzhuo Wang, and Jian Guo. 2024. A Survey on LLM-as-a-judge. *CoRR*, abs/2411.15594.

Nuno Miguel Guerreiro, Ricardo Rei, Daan van Stigt, Luísa Coheur, Pierre Colombo, and André F. T. Martins. 2024. xcomet : Transparent Machine Translation Evaluation through Fine-grained Error Detection. *Trans. Assoc. Comput. Linguistics*, 12:979–995.

Maram Hasanain, Fatema Ahmad, and Firoj Alam. 2024. Large Language Models for Propaganda Span Annotation. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 14522–14532, Miami, Florida, USA.

Olivia Huang, Eve Fleisig, and Dan Klein. 2023. Incorporating Worker Perspectives into MTurk Annotation Practices for NLP. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023*, pages 1010–1028, Singapore.

Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Madry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, Alex Nichol, Alex Paino, Alex Renzin, Alex Tachard Passos, Alexander Kirillov, Alexi Christakis, Alexis Conneau, Ali Kamali, Allan Jabri, Allison Moyer, Allison Tam, Amadou Crookes, Amin Tootoonchian, Ananya Kumar, Andrea Vallone, Andrej Karpathy, Andrew Braunstein, Andrew Cann, Andrew Codispoti, Andrew Galu, Andrew Kondrich, Andrew Tulloch, Andrey Mishchenko, Angela Baek, Angela Jiang, Antoine Pelisse, Antonia Woodford, Anuj Gosalia, Arka Dhar, Ashley Pantuliano, Avi Nayak, Avital Oliver, Barret Zoph, Behrooz Ghorbani, Ben Leimberger, Ben Rossen, Ben Sokolowsky, Ben Wang, Benjamin Zweig, Beth Hoover, Blake Samic, Bob McGrew, Bobby Spero, Bogo Giertler, Bowen Cheng, Brad Lightcap, Brandon Walkin, Brendan Quinn, Brian Guarraci, Brian Hsu, Bright Kellogg, Brydon Eastman, Camillo Lugaresi, Carroll L. Wainwright, Cary Bassin, Cary Hudson, Casey Chu, Chad Nelson, Chak Li, Chan Jun Shern, Channing Conger, Charlotte Barette, Chelsea Voss, Chen Ding, Cheng Lu, Chong Zhang, Chris Beaumont, Chris Hallacy, Chris Koch, Christian Gibson, Christina Kim, Christine Choi, Christine McLeavey, Christopher Hesse, Claudia Fischer, Clemens Winter, Coley Czarnecki, Colin Jarvis, Colin Wei, Constantin Koumouzelis, and Dane Sherburn. 2024. GPT-4o System Card. *CoRR*, abs/2410.21276.

Neslihan Iskender, Tim Polzehl, and Sebastian Möller. 2020. Best Practices for Crowd-based Evaluation of German Summarization: Comparing Crowd, Expert and Automatic Evaluation. In *Proceedings of the First Workshop on Evaluation and Comparison of NLP Systems, Eval4NLP 2020*, pages 164–175, Online.

Minhao Jiang, Ken Ziyu Liu, Ming Zhong, Rylan Schaeffer, Siru Ouyang, Jiawei Han, and Sanmi Koyejo. 2024. Investigating Data Contamination for Pre-training Language Models. *CoRR*, abs/2401.06059.

Ivan Kartáč, Mateusz Lango, and Ondrej Dusek. 2025. OpeNLGauge: An Explainable Metric for NLG Evaluation with Open-weights LLMs. *CoRR*, abs/2503.11858.

Zdeněk Kasner and Ondrej Dušek. 2024. Beyond Traditional Benchmarks: Analyzing Behaviors of Open LLMs on Data-to-text Generation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12045–12072, Bangkok, Thailand.

Zdenek Kasner, Simon Mille, and Ondrej Dusek. 2021. Text-in-Context: Token-level Error Detection for Table-to-text Generation. In *Proceedings of the 14th International Conference on Natural Language Generation, INLG 2021*, pages 259–265, Aberdeen, Scotland, UK.

Zdenek Kasner, Ondrej Plátek, Patrícia Schmidtová, Simone Balloccu, and Ondrej Dusek. 2024. factgenie: A Framework for Span-based Evaluation of Generated Texts. In *Proceedings of the 17th International Natural Language Generation Conference, INLG 2024 - System Demonstrations*, pages 13–15, Tokyo, Japan.

Seungone Kim, Jamin Shin, Yejin Choi, Joel Jang, Shayne Longpre, Hwaran Lee, Sangdoo Yun, Seongjin Shin, Sungdong Kim, James Thorne, and Minjoon Seo. 2024a. Prometheus: Inducing Fine-grained Evaluation Capability in Language Models. In *The Twelfth International Conference on Learning Representations, ICLR 2024*, Vienna, Austria.

Seungone Kim, Juyoung Suk, Shayne Longpre, Bill Yuchen Lin, Jamin Shin, Sean Welleck, Graham Neubig, Moontae Lee, Kyungjae Lee, and Minjoon Seo. 2024b. Prometheus 2: An Open Source Language Model Specialized in Evaluating Other Language Models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language*

Processing, EMNLP 2024, Miami, FL, pages 4334–4353, USA.

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, Martin Popel, Maja Popović, Mariya Shmatova, Steinthór Steingrímsson, and Vilém Zouhar. 2024a. 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, USA.

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, Martin Popel, Maja Popović, Mariya Shmatova, Steinthór Steingrímsson, and Vilém Zouhar. 2024b. 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, USA.

Tom Kocmi and Christian Federmann. 2023. GEMBA-MQM: Detecting Translation Quality Error Spans with GPT-4. In *Proceedings of the Eighth Conference on Machine Translation, WMT 2023*, pages 768–775, Singapore.

Tom Kocmi, Vilém Zouhar, Eleftherios Avramidis, Roman Grundkiewicz, Marzena Karpinska, Maja Popovic, Mrinmaya Sachan, and Mariya Shmatova. 2024c. Error Span Annotation: A Balanced Approach for Human Evaluation of Machine Translation. In *Proceedings of the Ninth Conference on Machine Translation, WMT 2024, Miami, FL*, pages 1440–1453, USA.

Ryan Koo, Minhwa Lee, Vipul Raheja, Jong Inn Park, Zae Myung Kim, and Dongyeop Kang. 2024. Benchmarking Cognitive Biases in Large Language Models as Evaluators. In *Findings of the Association for Computational Linguistics, ACL 2024*, pages 517–545, Bangkok, Thailand.

Klaus Krippendorff. 1980. *Content Analysis: An Introduction to Its Methodology*. SAGE Publications, Beverly Hills, CA.

Christoph Leiter, Juri Opitz, Daniel Deutsch, Yang Gao, Rotem Dror, and Steffen Eger. 2023. The Eval4NLP 2023 Shared Task on Prompting Large Language Models as Explainable Metrics. In *Proceedings of the 4th Workshop on Evaluation and Comparison of NLP Systems, Eval4NLP 2023*, pages 117–138, Bali, Indonesia.

Junlong Li, Shichao Sun, Weizhe Yuan, Run-Ze Fan, Hai Zhao, and Pengfei Liu. 2024. Generative Judge for Evaluating Alignment. In *The Twelfth International Conference on Learning Representations, ICLR 2024*, Vienna, Austria.

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, EMNLP 2023*, pages 2511–2522, Singapore.

Yuxuan Liu, Tianchi Yang, Shaohan Huang, Zihan Zhang, Haizhen Huang, Furu Wei, Weiwei Deng, Feng Sun, and Qi Zhang. 2024. Calibrating LLM-based Evaluator. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation, LREC/COLING 2024, 20-25 May, 2024*, pages 2638–2656, Torino, Italy.

Arle Lommel, Hans Uszkoreit, and Aljoscha Burchardt. 2014. Multidimensional quality metrics (MQM): A framework for declaring and describing translation quality metrics. *Tradumàtica*, 0(12):0455–463.

Valerie R Mariana. 2014. *The Multidimensional Quality Metric (MQM) framework: A new framework for translation quality assessment*. Brigham Young University.

Sara Vera Marjanović, Arkil Patel, Vaibhav Adlakha, Milad Aghajohari, Parishad BehnamGhader, Mehar Bhatia, Aditi Khandelwal, Austin Kraft, Benno Krojer, Xing Han Lù, et al. 2025. DeepSeek-R1 Thoughtology: Let's think about LLM Reasoning. *arXiv preprint arXiv:2504.07128*.

G. Da San Martino, A. Barrón-Cedeño, H. Wachsmuth, R. Petrov, and P. Nakov. 2020. SemEval-2020 Task 11: Detection of Propaganda Techniques in News Articles.

Yann Mathet, Antoine Widlöcher, and Jean-Philippe Métivier. 2015. The Unified and Holistic Method Gamma (γ) for Inter-annotator Agreement Measure and Alignment. *Comput. Linguistics*, 41(3):437–479.

Jekaterina Novikova, Ondrej Dusek, Amanda Cercas Curry, and Verena Rieser. 2017. Why We Need New Evaluation Metrics for NLG. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017*, pages 2241–2252, Copenhagen, Denmark.

OpenAI. 2025. OpenAI o3-mini System Card.

Jakub Piskorski, Nicolas Stefanovitch, Nikolaos Nikolaidis, Giovanni Da San Martino, and Preslav Nakov. 2023. Multilingual Multifaceted Understanding of Online News in Terms of Genre, Framing, and Persuasion Techniques. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023*, pages 3001–3022, Toronto, Canada.

Maja Popovic. 2020. Informative Manual Evaluation of Machine Translation Output. In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020*, pages 5059–5069, Barcelona, Spain (Online.

Alan Ramponi, Agnese Daffara, and Sara Tonelli. 2025. Fine-grained Fallacy Detection with Human Label Variation. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2025 - Volume 1: Long Papers, Albuquerque*, pages 762–784, New Mexico, USA.

Ehud Reiter. 2018. A Structured Review of the Validity of BLEU. *Comput. Linguistics*, 44(3).

Ananya B. Sai, Akash Kumar Mohankumar, and Mitesh M. Khapra. 2023. A Survey of Evaluation Metrics Used for NLG Systems. *ACM Comput. Surv.*, 55(2):26:1–26:39.

Patrícia Schmidtová, Saad Mahamood, Simone Balloccu, Ondrej Dusek, Albert Gatt, Dimitra Gkatzia, David M. Howcroft, Ondrej Plátek, and Adarsa Sivaprasad. 2024. Automatic Metrics in Natural Language Generation: A survey of Current Evaluation Practices. In *Proceedings of the 17th International Natural Language Generation Conference, INLG 2024*, pages 557–583, Tokyo, Japan.

Danil Semin, Ondřej Dušek, and Zdeněk Kasner. 2026. Strategies for span labeling with large language models.

Mandar Sharma, Ajay Kumar Gogineni, and Naren Ramakrishnan. 2022. Innovations in Neural Data-to-text Generation. *CoRR*, abs/2207.12571.

Andrea Sottana, Bin Liang, Kai Zou, and Zheng Yuan. 2023. Evaluation Metrics in the Era of GPT-4: Reliably Evaluating Large Language Models on Sequence to Sequence Tasks. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023*, pages 8776–8788, Singapore.

Rickard Stureborg, Dimitris Alikaniotis, and Yoshi Suhara. 2024. Large Language Models are Inconsistent and Biased Evaluators. *CoRR*, abs/2405.01724.

Melanie Subbiah, Sean Zhang, Lydia B. Chilton, and Kathleen R. McKeown. 2024. Reading Subtext: Evaluating Large Language Models on Short Story Summarization with Writers. *Trans. Assoc. Comput. Linguistics*, 12:1290–1310.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.

Craig Thomson and Ehud Reiter. 2020. A Gold Standard Methodology for Evaluating Accuracy in Data-To-text Systems. In *Proceedings of the 13th International Conference on Natural Language Generation, INLG 2020*, pages 158–168, Dublin, Ireland.

Hadrien Titeux and Rachid Riad. 2021. pygamma-agreement: Gamma γ measure for inter/intra-annotator agreement in Python. *Journal of Open Source Software*, 6(62):2989.

Tina Tseng, Amanda Stent, and Domenic Maida. 2020. Best Practices for Managing Data Annotation Projects. *CoRR*, abs/2009.11654.

Veniamin Veselovsky, Manoel Horta Ribeiro, and Robert West. 2023. Artificial Artificial Artificial Intelligence: Crowd Workers Widely Use Large Language Models for Text Production Tasks. *CoRR*, abs/2306.07899.

Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Lingpeng Kong, Qi Liu, Tianyu Liu, and Zhifang Sui. 2024. Large Language Models are not Fair Evaluators. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024*, pages 9440–9450, Bangkok, Thailand.

Lining Zhang, Simon Mille, Yufang Hou, Daniel Deutsch, Elizabeth Clark, Yixin Liu, Saad Mahamood, Sebastian Gehrmann, Miruna Clinciu, Khyathi Raghavi Chandu, and João Sedoc. 2023. A Needle in a Haystack: An Analysis of High-agreement Workers on MTurk for Summarization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14944–14982, Toronto, Canada.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. 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 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023*, New Orleans, LA, USA.

Vilém Zouhar, Tom Kocmi, and Mrinmaya Sachan. 2025. AI-assisted Human Evaluation of Machine Translation. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2025 - Volume 1: Long Papers, Albuquerque*, pages 4936–4950, New Mexico, USA.

A Implementation Details

A.1 Open Models

We run the local models using the `ollama` framework in 4-bit quantization. Specifically, we use `llama3.3:70b` and `deepseek-r1:70b` (which is based on Llama 3.3 70B) for span annotations. We also use `gemma2:2b` and `phi3.5:3.8b` for generating texts in D2T-EVAL.

For better reproducibility, we set the seed to 42 and the temperature to 0 for the local models. We do not use these parameters for proprietary models as these parameters are generally not supported.

We run the models using several GPU variants, including NVIDIA H100 NVL (95G), AMD MI210 (64G), and NVIDIA RTX 3090 (24G).

A.2 Proprietary Models

We use the following proprietary model versions:

- GPT-4o: `gpt-4o-2024-11-20`
- Claude 3.7 Sonnet: `claude-3-7-sonnet-20250219`
- o3-mini: `o3-mini-2025-01-31`
- Gemini 2.0 Flash Thinking: `gemini-2.0-flash-thinking-exp-01-21`

A.3 Web Interface

We implement our span annotation process using `factgenie` (Kasner et al., 2024): a tool that supports both collecting span annotations from humans via a web interface and from LLMs via API calls.

Figure 12 shows samples of our annotation interface implemented in `factgenie` for human annotators, including data visualizations from the `football` and `openweather` domains.

B Annotating D2T-EVAL

B.1 Generating Outputs

For generating the outputs for the structured inputs we collected, we use two larger models – Llama 3.3 70B (Grattafiori et al., 2024) and GPT-4o (Hurst et al., 2024) – and two smaller models – Gemma 2 2B (Team et al., 2024) and Phi-3.5 3.8B (Abdin et al., 2024). See more details on the models in Appendix A and prompts in Appendix D.

B.2 Collecting annotations

Annotation guidelines For the annotation guidelines, we went through an iterative process to establish the annotation guidelines $\mathcal{G}$ and the annotation categories C. We started with a preliminary version of the guidelines and annotation categories, drawing inspiration from the guidelines in previous works (Kasner and Dušek, 2024; Thomson and Reiter, 2020). We settled on the following annotation categories (see Table 5 for details): semantic accuracy errors due to information *Contradictory* to the input, *Not Checkable*, or *Misleading*; any *Incoherent* and *Repetitive* content, and any *Other* errors.

Gold annotations With the annotation guidelines established, we proceeded to collect our own internally annotated gold data: $\mathcal{D}_{\texttt{dev}}$, which contains 84 examples annotated individually by one of 7 annotators (12 examples per annotator) and $\mathcal{D}_{\texttt{iaa}}$, which contains 12 examples annotated commonly by all annotators.[10] The purpose of $\mathcal{D}_{\texttt{dev}}$ is to create a high-quality development set for the model prompting study, while the purpose of $\mathcal{D}_{\texttt{iaa}}$ is to pre-select skilled crowdworkers and quantify the performance of crowdworkers during data collection. Our average IAA on $\mathcal{D}_{\texttt{iaa}}$ was $F_1 = 0.444$ and $\gamma = 0.399$.

Crowdsourcing annotations We gather span annotations for $\mathcal{D}_{\texttt{test}}$ with crowdworkers from Prolific.com. Our process of collecting annotations proceeded in two stages, following the setup of Zhang et al. (2023): (1) a *qualification task* for pre-selecting skilled annotators, and (2) the *main task* for collecting the annotations.

- **Qualification task**: For the qualification task, we pre-selected workers whose first language is English, with >99% approval rate and more than 100 submissions. The workers were presented with a detailed tutorial with annotation guidelines and examples of individual errors. After the tutorial, we tested the worker performance on five manually pre-selected examples from $\mathcal{D}_{\texttt{iaa}}$. We invited annotators with the F_1 score higher than 0.5 w.r.t. our internal annotations for the main task.
- **Main task**: Of the 230 annotators who participated in the qualification task, 50 annotators

[10] We selected an example for each of the 4 domains and 3 models.

(21.7%) qualified. Of these, 45 annotators accepted (=90% turnover rate). For annotating the data in $\mathcal{D}_{\texttt{test}}$, we presented each annotator with a batch of 32 examples: 25 examples from $\mathcal{D}_{\texttt{test}}$ and 7 remaining examples from $\mathcal{D}_{\texttt{iaa}}$ (that is, the examples that we did not use for the qualification task). All the 1200 outputs in $\mathcal{D}_{\texttt{test}}$ were annotated by at least one annotator. Furthermore, 475 outputs (39.6%) were annotated by an additional annotator.[11]

For the qualification task, we paid all the annotators an average reward of 9.58 GBP / hour regardless of the qualification outcome. For the main task, we pay all the annotators an average reward of 10.70 GBP / hour.

C Annotation Categories

Tables 5 to 7 show an overview of the annotation span categories that we used for our tasks along with their descriptions.

Category Name	Description
Contradictory	The fact contradicts the data.
Not checkable	The fact cannot be verified from the data.
Misleading	The fact is technically true, but leaves out important information or otherwise distorts the context.
Incoherent	The text uses unnatural phrasing or does not fit the discourse.
Repetitive	The fact has been already mentioned earlier in the text.
Other	The text is problematic for another reason.

Table 5: Annotation categories for the D2T-EVAL task.

Category Name	Description
Major	An error that disrupts the flow and makes the understandability of text difficult or impossible.
Minor	An error that does not disrupt the flow significantly and what the text is trying to say is still understandable.

Table 6: Annotation categories for the MT-EVAL task.

D Prompts

Here, we provide the model prompts:

- Figures 5 to 8 show the prompts for the D2T-EVAL that we use for the experiments in Section 5.1.
- Figure 9 shows the base prompt we used for MT-EVAL.
- Figure 10 shows the base prompt we used for PROPAGANDA.
- Figure 11 shows the prompt we used for *generating* the outputs for D2T-EVAL.

E Examples

In Tables 8 to 10, we show examples of the annotated outputs for our tasks. Figure 12 shows our annotation interface.

F Results

Here, we provide detailed results of our experiments:

- **Main results**: Table 11 (D2T-EVAL), Tables 12 and 13 (MT-EVAL), Table 14 (PROPAGANDA)
- **Extra statistics**: (D2T-EVAL) Tables 13 and 17 (MT-EVAL), Table 18 (PROPAGANDA).
- **Confusion matrices**: Figure 13 (MT-EVAL) and Figure 14 (PROPAGANDA).
- **Manual evaluation**: Table 19 (D2T-EVAL), Table 20 (PROPAGANDA), Table 4 (human annotators).

[11] We use examples with two annotators to compute the average IAA for D2T-EVAL in Section 5.2. For other experiments, we use only the outputs from the first annotator as reference data.

Category Name	Description
Appeal to Authority	Stating that a claim is true simply because a valid authority or expert on the issue said it was true, without any other supporting evidence offered. We consider the special case in which the reference is not an authority or an expert in this technique, altough it is referred to as Testimonial in literature
Appeal to fear-prejudice	Seeking to build support for an idea by instilling anxiety and/or panic in the population towards an alternative. In some cases the support is built based on preconceived judgements
Bandwagon	Attempting to persuade the target audience to join in and take the course of action because "everyone else is taking the same action"
Black-and-White Fallacy	Presenting two alternative options as the only possibilities, when in fact more possibilities exist. As an the extreme case, tell the audience exactly what actions to take, eliminating any other possible choices (Dictatorship)
Causal Oversimplification	Assuming a single cause or reason when there are actually multiple causes for an issue. It includes transferring blame to one person or group of people without investigating the complexities of the issue
Doubt	Questioning the credibility of someone or something
Exaggeration, Minimisation	Either representing something in an excessive manner: making things larger, better, worse (e.g., "the best of the best", "quality guaranteed") or making something seem less important or smaller than it really is (e.g., saying that an insult was just a joke)
Flag-Waving	Playing on strong national feeling (or to any group; e.g., race, gender, political preference) to justify or promote an action or idea
Loaded Language	Using specific words and phrases with strong emotional implications (either positive or negative) to influence an audience
Name Calling, Labeling	Labeling the object of the propaganda campaign as either something the target audience fears, hates, finds undesirable or loves, praises
Obfuscation, Intentional Vagueness, Confusion	Using words which are deliberately not clear so that the audience may have its own interpretations. For example when an unclear phrase with multiple definitions is used within the argument and, therefore, it does not support the conclusion
Red Herring	Introducing irrelevant material to the issue being discussed, so that everyone‟s attention is diverted away from the points made
Reductio ad hitlerum	Persuading an audience to disapprove an action or idea by suggesting that the idea is popular with groups hated in contempt by the target audience. It can refer to any person or concept with a negative connotation
Repetition	epeating the same message over and over again so that the audience will eventually accept it
Slogans	A brief and striking phrase that may include labeling and stereotyping. Slogans tend to act as emotional appeals
Straw Men	When an opponent‟s proposition is substituted with a similar one which is then refuted in place of the original proposition
Thought-terminating Cliches	Words or phrases that discourage critical thought and meaningful discussion about a given topic. They are typically short, generic sentences that offer seemingly simple answers to complex questions or that distract attention away from other lines of thought
Whataboutism	A technique that attempts to discredit an opponent‟s position by charging them with hypocrisy without directly disproving their argument

Table 7: Annotation categories for the PROPAGANDA task. The categories are adopted from Da San Martino et al. (2019).

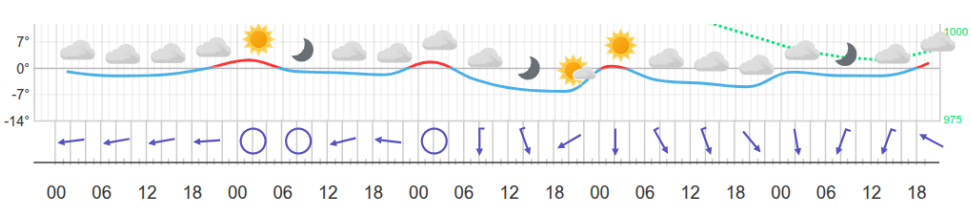
Source (part of the visualization)	
Annotated by Llama 3	Bear Creek will experience mostly overcast skies for the next five days, with **a chance of occasional showers on Wednesday and Thursday**[not check.]. Temperatures will remain relatively cool, hovering around freezing point or slightly below. Winds will be light to moderate throughout the period, with **some gusts expected on Friday**[not check.]. The overall weather pattern suggests a generally cloudy and potentially wet week.
Annotated by o3-mini	Bear Creek will experience mostly overcast skies for the next five days, with a **chance of occasional showers on Wednesday and Thursday**[cont.]. Temperatures will remain relatively cool, hovering around freezing point or slightly below. Winds will be light to moderate throughout the period, **with some gusts expected on Friday**[cont.]. The overall weather pattern suggests a generally cloudy and **potentially wet week**[cont.].
Annotated by human	Bear Creek will experience mostly overcast skies for the next five days, with a chance **of occasional showers on Wednesday and Thursday**[cont.]. Temperatures will remain relatively cool, hovering around freezing point or slightly below. Winds will be light to moderate throughout the period, **with some gusts expected on Friday**[mislead.]. The overall weather pattern suggests a generally cloudy and **potentially**[cont.] **wet**[cont.] **week**[cont.].

Table 8: Example for span annotation of D2T-EVAL in the `weather` domain with **contradictory**[cont], **misleading**[mislead.], and **not-checkeable**[not check.] as error span categories. In the annotation interface, the visualization is interactive. The top part (not visible here) shows the place name and date timeline.

Source	"It's your birthday soon, isn't it?" Ivory asked, remembering that the princess' birthday was in a few days. Kari nodded, a sad glint in her light blue eyes.
Annotated by Llama 3	"¿Es su cumpleaños pronto, **no lo es?"**[major] **Ivory le preguntó,**[minor] recordando que el cumpleaños de la princesa era en unos días. Kari nodó, **un deslumbramiento triste**[major] en sus ojos azules claros.
Annotated by o3-mini	"¿Es su cumpleaños pronto, **no lo es?"**[minor] Ivory le preguntó, recordando que el cumpleaños de la princesa era en unos días. Kari nodó, un **deslumbramiento triste**[major] en sus ojos azules claros.
Annotated by human	"¿Es su cumpleaños pronto, **no lo es?"**[minor] Ivory le preguntó, recordando que el cumpleaños de la princesa era en unos días. Kari **nodó**[major], un deslumbramiento triste en sus ojos azules claros.

Table 9: An example for span annotation of MT-EVAL outputs (English→Spanish) with **minor** and **major** as error span categories.

Annotated by Llama 3	**When the left made Linda Sarsour into its role model, it climbed into bed with**[fear] **the worst of the worst**[labelling]. The father of a missing 4-year-old Georgia boy was training children at **a filthy New Mexico compound**[loaded] to commit school shootings, prosecutors alleged in court documents Wednesday.
Annotated by o3-mini	**When the left made Linda Sarsour into its role model, it climbed into bed with the worst of the worst.**[loaded] The father of a missing 4-year-old Georgia boy was training children at a **filthy New Mexico compound**[loaded] to commit school shootings, prosecutors alleged in court documents Wednesday.
Annotated by human	When the left made Linda Sarsour into **its role model**[labelling], **it climbed into bed**[loaded] **with the worst of the worst.**[exag.] The father of a missing 4-year-old Georgia boy was training children at **a filthy New Mexico compound**[labelling] to commit school shootings, prosecutors alleged in court documents Wednesday.

Table 10: Two examples for span annotation of PROPAGANDA outputs with **appeal-to-fear**[fear], **name-calling-labelling**[labelling], **loaded-language**[loaded], and **exaggeration**[exag.] as span categories.

Model	ρ	Precision		Recall		F1		Δ	γ	$S_\emptyset$
		Hard	Soft	Hard	Soft	Hard	Soft			
Llama 3.3	0.307	0.176	0.365	0.187	0.388	0.181	0.377	0.196	0.109	0.418
GPT-4o	0.346	0.233	0.391	0.184	0.308	0.206	0.345	0.139	0.130	0.429
Claude 3.7	**0.512**	0.294	0.442	**0.304**	**0.457**	0.299	0.449	0.150	0.203	0.592
DeepS. R1	0.453	0.317	0.532	0.185	0.310	0.233	0.392	0.159	0.185	**0.645**
o3-mini	0.505	**0.392**	**0.542**	0.285	0.395	**0.330**	0.457	0.127	**0.273**	0.637
Gem. 2-FT	0.458	0.293	0.488	0.263	0.438	0.277	**0.462**	0.185	0.209	0.612

Table 11: Evaluation of human and LLM annotations using $\mathcal{P}_{\text{base}}$ on D2T-EVAL. See Figure 2 for visualizaton of this table.

Model	ρ	Precision		Recall		F1		Δ	γ	$S_\emptyset$
		Hard	Soft	Hard	Soft	Hard	Soft			
Llama 3.3	0.182	0.121	0.200	0.229	0.378	0.155	0.257	0.102	0.014	0.392
GPT-4o	0.158	0.141	0.240	0.195	0.327	0.156	0.266	0.110	0.076	0.428
Claude 3.7	**0.301**	**0.226**	**0.325**	**0.335**	**0.469**	**0.262**	**0.373**	0.111	**0.131**	0.628
DeepS. R1	0.177	0.169	0.268	0.183	0.280	0.168	0.262	0.094	0.058	0.631
o3-mini	0.197	0.169	0.291	0.161	0.275	0.160	0.275	0.115	0.100	0.646
Gem. 2-FT	0.257	0.184	0.312	0.180	0.339	0.173	0.304	0.130	0.066	**0.710**

Table 12: Evaluation of human and LLM annotations using $\mathcal{P}_{\text{base}}$ on the MT-EVAL – average across languages.

Model	ρ	Precision		Recall		F1		Δ	γ	$S_\emptyset$
		Hard	Soft	Hard	Soft	Hard	Soft			
en-cs	0.303	0.144	0.268	0.180	0.326	0.156	0.286	0.130	0.084	0.582
en-es	0.171	0.161	0.243	0.236	0.362	0.190	0.288	0.098	0.080	**0.631**
en-hi	0.170	0.173	0.265	0.208	0.327	0.173	0.269	0.096	-0.0	0.552
en-is	**0.347**	0.136	0.246	0.187	0.361	0.145	0.269	0.124	0.108	0.493
en-ja	0.127	0.193	0.302	0.249	0.363	**0.209**	**0.318**	0.109	0.063	0.569
en-ru	0.225	0.178	0.256	**0.273**	**0.386**	0.208	0.298	0.090	**0.162**	0.588
en-uk	0.192	0.166	0.254	0.214	0.339	0.184	0.286	0.102	0.031	0.542
en-zh	0.163	**0.196**	**0.346**	0.163	0.294	0.169	0.302	0.133	0.075	0.623

Table 13: Evaluation of human and LLM annotations using $\mathcal{P}_{\text{base}}$ on the MT-EVAL separately for each language (average across models).

Model	ρ	Precision		Recall		F1		Δ	γ	$S_\emptyset$
		Hard	Soft	Hard	Soft	Hard	Soft			
Llama 3.3	0.336	0.070	0.243	0.063	0.219	0.066	0.230	0.164	0.092	0.343
GPT-4o	0.344	0.095	0.293	0.038	0.115	0.054	0.166	0.112	0.066	0.234
Claude 3.7	0.460	0.110	0.274	0.096	0.239	0.103	0.255	0.152	0.155	0.113
DeepS. R1	0.354	0.083	0.246	0.062	0.182	0.071	0.209	0.138	0.091	0.476
o3-mini	0.418	**0.152**	**0.411**	0.066	0.179	0.092	0.249	0.157	0.154	**0.517**
Gem. 2-FT	**0.560**	0.106	0.268	**0.190**	**0.477**	**0.136**	**0.343**	0.207	**0.202**	0.493

Table 14: Evaluation of human and LLM annotations using $\mathcal{P}_{\text{base}}$ on the PROPAGANDA.

Annotator	Ann	Ann/Ex	w/o%	Char/Ann
Human	2981	2.5	28.8	50.3
Llama 3.3	3214	2.7	7.4	65.5
GPT-4o	2284	1.9	4.8	66.3
Claude 3.7	2865	2.4	22.5	57.2
DeepS. R1	1387	1.2	44.2	56.8
o3-mini	1836	1.5	35.6	58.0
Gem. 2-FT	2517	2.1	28.9	54.3

Table 15: Statistics of models and human annotators using $\mathcal{P}_{\text{base}}$ on D2T-EVAL. Ann=# of annotations, Ann/Ex=ann. per example. w/o=% ex. without annotations, Char/Ann=# chars per ann.

Annotator	Ann.	Ann/Ex	w/o%	Char/Ann
Human	2090	0.7	66.0	14.5
Llama 3.3	6361	2.3	6.2	17.4
GPT-4o	4866	1.7	7.0	15.9
Claude 3.7	3782	1.4	30.6	15.9
DeepS. R1	2586	0.9	36.3	15.1
o3-mini	3039	1.1	35.8	13.8
Gem. 2-FT	2181	0.8	50.0	15.2

Table 16: Statistics of models and human annotators using $\mathcal{P}_{\text{base}}$ on MT-EVAL. See Table 15 for the legend.

Lang.	Annot.	Ann.	Ann/Ex	w/o%	Char/Ann
en-cs	Model	600	1.4	27.0	16.6
	Human	399	0.7	66.1	13.0
en-es	Model	417	1.1	38.9	18.8
	Human	248	0.6	70.3	10.3
en-hi	Model	396	1.3	26.2	19.0
	Human	222	0.5	71.2	10.7
en-is	Model	563	1.9	14.3	15.7
	Human	752	2.5	18.3	16.6
en-ja	Model	471	1.3	28.7	11.1
	Human	118	0.2	87.5	14.8
en-ru	Model	500	1.3	25.9	18.2
	Human	287	0.7	58.7	19.4
en-uk	Model	436	1.5	25.4	17.8
	Human	208	0.7	64.3	12.3
en-zh	Model	420	1.2	34.6	7.2
	Human	171	0.2	85.1	6.6

Table 17: Statistics of models (averaged) and human annotators using $\mathcal{P}_{\text{base}}$ on the MT-EVAL separately for each language. See Table 15 for the legend.

Annotator	Ann.	Ann/Ex	w/o%	Char/Ann
Human	1439	14.2	4.0	40.2
Llama 3.3	574	5.7	3.0	92.0
GPT-4o	246	2.4	8.9	91.1
Claude 3.7	803	8.0	7.9	58.5
DeepS. R1	459	4.5	9.9	89.3
o3-mini	376	3.7	10.9	65.3
Gem. 2-FT	1864	18.5	3.0	54.1

Table 18: Statistics of models and human annotators using $\mathcal{P}_{\text{base}}$ on the PROPAGANDA. See Table 15 for the legend.

	Annotations					Explanations				
Model	C	P	W	I	U	C	P	W	I	U
Llama 3.3	7	1	2	8	0	6	0	1	10	1
GPT-4o	5	2	1	10	0	6	1	0	11	0
Claude 3.7	7	2	3	6	0	9	2	0	7	0
DeepSeek	11	2	4	1	0	12	3	0	3	0
o3-mini	10	3	2	0	3	8	7	0	0	3
Gemini 2 F-T	12	4	1	1	0	12	5	1	0	0
Total	52	14	13	26	3	53	18	2	31	4

Table 19: Manual evaluation results for D2T-Eval domain. Categories for annotation and reason: C=Correct, P=Partially correct, W=Wrong category, I=Incorrect, U=Undecidable.

	Annotations					Explanations				
Model	C	P	W	I	U	C	P	W	I	U
Llama 3.3	9	1	2	5	1	8	2	1	6	1
GPT-4o	9	0	3	6	0	8	2	1	7	0
Claude 3.7	9	2	3	4	0	9	3	3	3	0
DeepSeek	6	0	4	8	0	7	0	3	8	0
o3-mini	15	0	0	2	1	15	0	0	2	1
Gemini 2 F-T	7	3	2	6	0	9	2	1	6	0
Total	55	6	14	31	2	56	9	9	32	2

Table 20: Manual evaluation results for Propaganda domain. See Table 19 for the legend.

````
Your task is to identify errors in the text and classify them.

Output the errors as a JSON object with a single key "annotations". The value of "annotations" is a list in which each object contains fields "reason", "text", and "annotation_type". The value of "reason" is the short sentence justifying the annotation. The value of "text" is the literal value of the identified span (we will later identify the span using string matching). The value of "annotation_type" is an integer index of the error based on the following list:

{categories}

Examples:
- Contradictory: The lowest temperature does not drop below 4°C, but the text mentions 2°C.
- Not checkable: The text mentions that "both teams display aggressive play", which cannot be checked from the data.
- Misleading: The tone of the text suggests there are many sensors out of which just a few are listed here. However, according to the data, the device has exactly these four sensors.
- Incoherent: The text states that both teams had "equal chances until the first half ended scoreless." While this is technically true, the expression sounds unnatural for a sport summary.
- Repetitive: The output text unnecessarily re-states information about a smartphone battery that was mentioned earlier.
- Other: Use this as a last resort when you notice something else not covered by the above categories.

Hints:
- Always try to annotate the longest continuous span (i.e., the whole fact instead of a single word).
- Annotate only the spans that you are sure about. If you are not sure about an annotation, skip it.
- Ignore subjective statements: for example "a lightweight smartphone" highly depends on the context: you should not annotate these statements.
- Numerical conventions: For weather forecasts, we accept both precise numbers (e.g. 10.71°C) and the rounded ones (e.g. 11°C) as long as they agree with the data.
- Annotate only according to your own knowledge. If you are considering using an external tool (Google, ChatGPT etc.), just skip that specific fact.
If there is nothing to annotate in the text, "annotations" will be an empty list.

Given the data:
```
{data}
```
annotate the errors in the corresponding text generated from the data:
```
{text}
```
````

Figure 5: The prompt $\mathcal{P}_{\texttt{base}}$ for D2T-EVAL.

````
Your task is to identify errors in the text and classify them.

Output the errors as a JSON object with a single key "annotations". The value of "annotations" is a list in which each object contains fields "reason", "text", and "annotation_type". The value of "reason" is the short sentence justifying the annotation. The value of "text" is the literal value of the identified span (we will later identify the span using string matching). The value of "annotation_type" is an integer index of the error based on the following list:

{categories}

Given the data:
```
{data}
```
annotate the errors in the corresponding text generated from the data:
```
{text}
```
````

Figure 6: The prompt $\mathcal{P}_{\texttt{noguide}}$ for D2T-EVAL.

````
Think about it step-by-step. You should enclose your chain of thoughts between the <think> and </think> tags. Once you are ready, output the JSON object in the required format.

Example:
```
<think> ... chain of thoughts ... </think>  ...
JSON object ...
```
````

Figure 7: The additional text added for $\mathcal{P}_{\texttt{cot}}$.

````
In order to help you with the task, we provide you with five examples of inputs, outputs and annotations:

Example #1:
data:
```
{data}
```
text:
```
{text}
```
output:
```
{annotations}
```
(...)
````

Figure 8: The additional text added for $\mathcal{P}_{\texttt{5shot}}$.

```
Your task is to identify errors in the translation and classify them.

Output the errors as a JSON object with a single key "annotations". The value of "annotations" is a list in which each object contains fields "reason", "text", and "annotation_type". The value of "reason" is the short sentence justifying the annotation. The value of "text" is the literal value of the identified span (we will later identify the span using string matching). The value of "annotation_type" is an integer index of the error based on the following list:

{categories}

Error spans can include parts of the words, whole words, or multi-word phrases.
Hint: errors are usually accuracy-related (addition, mistranslation, omission, untranslated text), fluency-related (character encoding, grammar, inconsistency, punctuation, register, spelling), style-related (awkward), terminology (inappropriate for context, inconsistent use).

Make sure that the annotations are not overlapping. If there is nothing to annotate in the text, "annotations" will be an empty list.

Given the source:
```
{source}
```
annotate its translation:
```
{text}
```
```

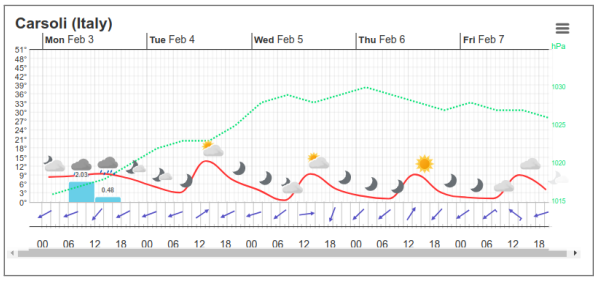
Figure 9: The prompt $\mathcal{P}_{\text{base}}$ for MT-EVAL.

```
Your task is to identify spans of text that employ propaganda techniques.

Output the errors as a JSON object with a single key "annotations". The value of "annotations" is a list in which each object contains fields "reason", "text", and "annotation_type". The value of "reason" is the short sentence justifying the annotation. The value of "text" is the literal value of the identified span (we will later identify the span using string matching). The value of "annotation_type" is an integer index of the error based on the following list:

{categories}

Now annotate the following text:
```
{text}
```
```

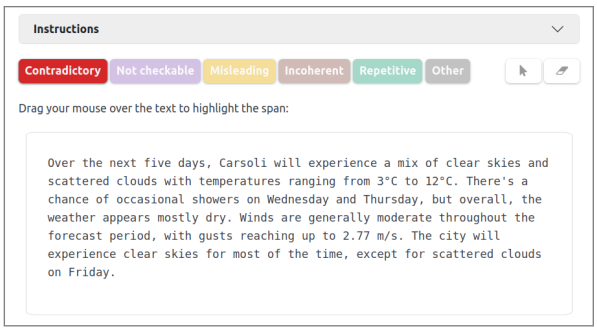
Figure 10: The prompt $\mathcal{P}_{\text{base}}$ for PROPAGANDA.

```
Given the structured summary of a football game:
```
{data}
```
Generate a match summary using approximately five sentences. The summary should sound natural, reporting on the important moments of the game. Avoid subjective statements, keep the tone of the summary neutral. Do not fabricate any facts that are not explicitly stated in the data.
```

Figure 11: The prompt used for generating outputs in the `football` domain for D2T-EVAL. The prompts for the other domains are analogical. For more robust parsing, we initialize the model response with 'Sure, here is the summary: "' .



(a) Data visualization – `openweather`



(b) Interface for highlighting spans

Figure 12: Samples from the `factgenie` annotation interface used for collecting span annotations.

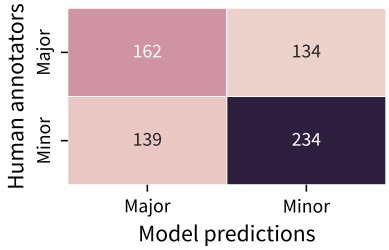


Figure 13: Confusion matrix for MT-EVAL, averaged across models (see Table 6 for category descriptions).

Figure 14: Confusion matrix comparing human annotations (rows) with model predictions (columns) for PROPAGANDA, averaged across models. (see Table 7 for the description of categories).