Are LLMs Better than Reported?
Detecting Label Errors and Mitigating Their Effect on Model Performance

Anonymous ACL submission

Abstract

NLP benchmarks rely on standardized datasets for training and evaluating models and are crucial for advancing the field. Traditionally, expert annotations ensure high-quality labels; however, the cost of expert annotation does not scale well with the growing demand for larger datasets required by modern models. While crowd-sourcing provides a more scalable solution, it often comes at the expense of annotation precision and consistency. Recent advancements in large language models (LLMs) offer new opportunities to enhance the annotation process, particularly for detecting label errors in existing datasets. In this work, we consider the recent approach of LLM-as-a-judge, leveraging an ensemble of LLMs to flag potentially mislabeled examples. Through a case study of four datasets from the TRUE benchmark, covering different tasks, we empirically analyze the labeling quality of existing datasets and compare expert, crowd-sourced, and LLM-based annotations in terms of the agreement, label quality, and efficiency, demonstrating the strengths and limitations of each annotation method. Our findings reveal a substantial number of label errors, which, when corrected, induce a significant upward shift in reported model performance. This suggests that many of the LLMs' so-called mistakes are due to label errors rather than genuine model failures. Additionally, we discuss the implications of mislabeled data and propose methods to mitigate them in training to improve performance.

1 Introduction

Natural Language Processing (NLP) benchmarks have long served as a cornerstone for advancing the field, providing standardized datasets for training and evaluating methods and models (Wang et al., 2019; Hendrycks et al., 2021; Srivastava et al., 2023; Calderon et al., 2024). These datasets have been developed over the years for various tasks and scales, annotated using different schemes. Gold labels represent the "true" or ground truth annotations, which are typically established through expensive rigorous processes, including expert consensus and extensive quality control. However, as models have increased in size (Devlin et al., 2019; Brown et al., 2020), the demand for larger datasets has also grown (Kaplan et al., 2020). Since expert annotation is cost-prohibitive, it does not scale well to meet these demands. The demand for large quantities of annotated data quickly and cost-effectively has led researchers to adopt crowd-sourcing, often sacrificing expertise for scale.

That way or another, constructing datasets heavily involves making compromises in annotation, trading off between scale, efficiency and expertise. Even when annotated by experts, datasets can naturally contain labeling errors, arising from factors such as task subjectivity, annotator fatigue, inattention, insufficient guidelines, and more (Rogers et al., 2013; Reiss et al., 2020; Sylolypavan et al., 2023). Mislabeled data is even more pronounced when non-expert annotators are involved (Kennedy et al., 2020; Chong et al., 2022a). Widespread mislabeled data is particularly concerning because both the research community and the industry rely heavily on benchmarks. In training data, label errors harm model quality and hinder generalization, while in test sets, they lead to flawed comparisons, false conclusions, and prevent progress.

Recent advancements in large language models (LLMs) (Ouyang et al., 2022; Chiang and Lee, 2023; Li et al., 2023; Gat et al., 2024) present new opportunities to improve the annotation process, specifically in detecting label errors within existing datasets. Rather than re-annotating entire datasets (e.g., through experts or crowd-workers), we consider the LLM-as-a-judge approach (Zheng et al., 2023), and propose a simple yet effective method by leveraging an ensemble of LLMs to flag a set of potentially mislabeled examples. These can then be sent to experts for re-annotation and correction,

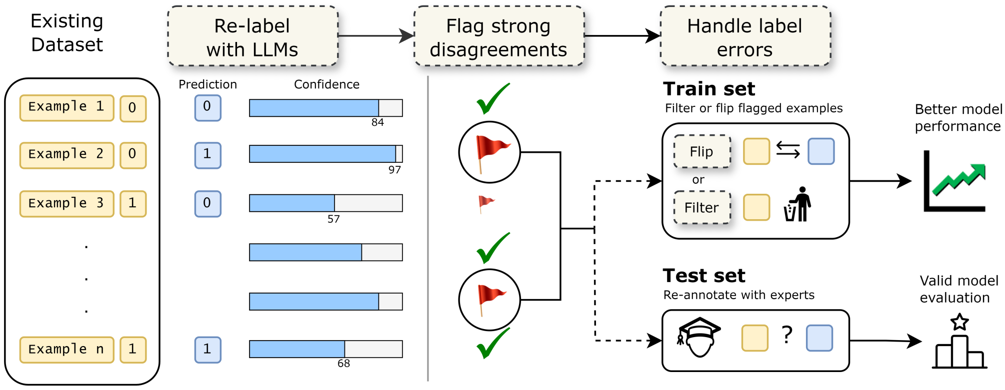


Figure 1: An illustration of our approach for detecting and addressing mislabeled data: (1) Re-label examples from existing datasets using an ensemble of LLMs. (2) Identify *strong disagreements* between the LLM's predictions and the original labels (i.e., high confidence in a different label), flagging examples based on confidence levels. Our findings show that LLMs detect between 6% and 21% of label errors, and higher LLM confidence is strongly associated with improved precision in error detection. (3) In the training set, we either filter or flip flagged examples, leading to an increase of up to 4%. For the test set, flagged examples are re-annotated by experts to make sure the evaluation is accurate. Under accurate evaluation, the performance of LLMs is up to 15% higher.

or even get filtered during training.

Specifically, we construct an ensemble model using multiple LLMs with diverse prompts, gathering both their predicted labels and corresponding confidence scores. These predictions are contrasted with the original labels, and instances where the LLMs *strongly disagree* with the original label (i.e., show high confidence in a different label) are flagged as potential mislabeling cases. Additionally, we not only explore the role of LLMs in detecting errors but also evaluate their performance as annotators, comparing them with expert and crowd-sourced annotations. We assess these approaches in terms of agreement, label quality, and efficiency, highlighting their strengths and limitations.

We aim to answer the following questions through a comprehensive end-to-end study: (1) Do current benchmarks include mislabeled data? (2) Can LLMs detect label errors? (3) How do expert, crowd-sourced, and LLM-based annotations compare in quality and efficiency? and (4) What are the implications of mislabeled data on model performance and can we mitigate their impact?

To this end, we choose the TRUE benchmark (Honovich et al., 2022) – A collection consolidating 11 existing datasets annotated for factual consistency in a unified format – as a case-study and empirically investigate its labeling quality. Specifically, we analyze four datasets from TRUE with binary factual consistency annotation originating from different tasks. To support our claims and results in other setups, we conduct similar experiments on an additional dataset, SummEval (Fabbri et al., 2021), which evaluates generated summaries in four dimensions on a scale of 1 to 5.

Our paper presents both methodological and empirical contributions. We propose a straightforward approach for detecting potential mislabeled examples (as illustrated in Figure 1), revealing a substantial number of label errors in existing datasets, ranging from 6% to 21%. Additionally, we demonstrate that the precision of LLMs in identifying errors improves with their confidence in an incorrect label; when their confidence exceeds 95%, over two-thirds of those labels are human errors. Moreover, we show that LLM-based annotations not only excel in error detection but also perform similarly to, or better than, traditional annotation methods, offering better trade-offs between quality, scale, and efficiency. Finally, we empirically illustrate the negative impact of mislabeled data on model training and evaluation. We propose a simple automated method for addressing label errors, improving the performance of fine-tuned models by up to 4%. In evaluation, we found that mislabeled data can significantly distort reported performance; LLMs may perform up to 15% better. This indicates that many so-called prediction errors are not genuine errors but are instead human annotation mistakes.

2 Related Work

Traditional Human Annotation Approaches Crowdsourcing is widely used for annotating large-scale NLP datasets (Rajpurkar et al., 2016;

Williams et al., 2018; Wang et al., 2022), offering rapid and scalable data collection. However, quality control remains a challenge, with labeling inconsistencies increasing as dataset complexity grows (Lu et al., 2020; Allahbakhsh et al., 2013). Moreover, as LLMs approach near-human performance (Chiang and Lee, 2023; Chen and Ding, 2023), crowd workers increasingly rely on these models for assistance, further complicating annotation quality (Veselovsky et al., 2023b,a). Expert annotation provides more reliable labels for domain-specific and cognitively demanding tasks (e.g., medical or legal domains) but is significantly slower and costlier than crowdsourcing (Snow et al., 2008; Chau et al., 2020). Ensuring inter-annotator agreement among experts adds further complexity and expense (Baledent et al., 2022). Hybrid approaches that combine expert and crowd-sourced annotations help balance cost and quality, though expert oversight remains crucial for high-quality labels (Nguyen et al., 2015). Our study compares expert, crowd-sourced, and LLM-based annotation approaches in terms of quality and efficiency.

LLMs in the Annotation Loop LLMs have been increasingly utilized as annotators in various NLP tasks, offering potential benefits in efficiency and scalability. Several studies have demonstrated that LLMs can effectively generate annotations from scratch, sometimes outperforming human annotators or crowd workers (He et al., 2023; Gilardi et al., 2023; Törnberg, 2023; Calderon and Reichart, 2024). However, LLMs are not flawless and cannot be considered gold-standard annotators when used alone. They may produce incorrect annotations, especially in complex (Chen et al., 2024), social (Ventura et al., 2023; Felkner et al., 2024), emotional (Lissak et al., 2024), or low-resource (Bhat and Varma, 2023) contexts. These studies showed that LLMs can exhibit poor performance and biases, highlighting the necessity of human oversight to ensure quality or fairness. To address this issue, several approaches for collaborative (Kim et al., 2024; Li et al., 2023) or active learning (Zhang et al., 2023; Kholodna et al., 2024) were suggested, where LLMs and humans are both part of the annotation procedure. While most research focuses on annotation from scratch, our work employs an ensemble of LLMs to flag potentially mislabeled data points in existing datasets.

Handling Label Errors Label errors (also referred to as label noise) in training and evaluation datasets can significantly impair NLP model performance and reliability (Frénay and Verleysen, 2014). Previous work mainly focuses on fine-tuned models and typically identifies mislabeled examples based on the model's low confidence or high training loss (Chong et al., 2022b; Hao et al., 2020; Pleiss et al., 2020; Northcutt et al., 2019). For example, Chong et al. (2022b) showed that ranking data points based on the training loss can help detect errors. Once these high-loss or low-confidence examples are flagged, they are typically filtered out (Nguyen et al., 2019; Northcutt et al., 2019), corrected automatically (Pleiss et al., 2020; Hao et al., 2020), or re-labeled by human annotators (Northcutt et al., 2021) to verify and improve dataset quality. Unlike previous works, we use an ensemble of LLMs to flag only high-confidence false predictions. Our results demonstrate that low-confidence examples weakly correlates with errors, but high-confidence in the false predictions strongly do.

3 LLM as an Annotator and Detector

This study aims to evaluate the potential of LLMs in detecting mislabeled examples and compare three annotation approaches: experts, crowdsourcing, and LLMs. To this end, we use an ensemble model that combines multiple LLMs with varied prompts. The motivation for this ensemble is twofold: first, we demonstrate that it enhances error detection and aligns more closely with expert annotations while also decreases the variance; second, it offers a simple approach that avoids the need for complex model selection or extensive prompt engineering, relying instead on the collective strength.

Prediction and Confidence To make a prediction using the ensemble, we first extract class probabilities of each LLM and prompt from the logits of the representing class tokens (e.g., 0 or 1 for the binary TRUE datasets, and 1 to 5 for the ordinal SummEval). The probabilities are then normalized to sum to 1. Next, we compute the average probability for each class across the ensemble and select the class with the highest probability (argmax) as the final prediction. The confidence in the prediction is defined as the corresponding ensemble probability. If the token probabilities are not accessible, they can be approximated via sampling.

Errors Detection We re-label the dataset using the ensemble, keeping both the prediction and confidence for each example. We then flag potentially mislabeled examples where there is *strong* *disagreement* between the ensemble prediction and the original label, specifically when the model exhibits high confidence in a false prediction. In the binary case, we examine only examples where the ensemble prediction differs from the original label. In the ordinal case, we examine examples where the difference between the original label and the ensemble prediction is strictly greater than 1 (e.g., 3 vs. 5, 1 vs. 5, 4 vs. 2, etc.). After examining these examples, only those with confidence exceeding a predefined threshold are flagged as potentially mislabeled. Our experiments show that as confidence in an incorrect prediction increases, the likelihood of the example being mislabeled also rises.

For test sets, flagged examples can be re-examined by human experts to verify their true labels. For training sets, the same approach can be applied, but we also propose an automated method to improve model training: flagged examples can either be removed from the dataset or have their labels corrected based on the ensemble prediction.

4 Experimental Setup

4.1 Data

As a case-study, we choose to explore the extensive and widely used TRUE benchmark (Honovich et al., 2022), which is typically used as an evaluation set (Steen et al., 2023; Gekhman et al., 2023; Wang et al., 2024; Zha et al., 2023). It consists of 11 datasets from various NLP tasks such as summarization and knowledge-grounded dialogue. This benchmark is unique in its approach of bringing multiple datasets and tasks into a unified schema of binary factual consistency labels. Each dataset is transformed from its original structure (e.g., a source document and a summary) into two input texts, *Grounding* and *Generated Text*, and a binary label indicating whether the generated text is factually consistent w.r.t the grounding. This enables us to examine multiple tasks and domains under the same umbrella at once while maintaining a unified binary-label schema. Specifically, we focus on four TRUE datasets, one from each task: MNBM – summarization evaluation (Maynez et al., 2020); BEGIN – grounded dialogue evaluation (Dziri et al., 2022); VitaminC – fact verification (Schuster et al., 2021); and PAWS – paraphrasing evaluation (Zhang et al., 2019). See Appendix E for additional details on these datasets.

For each of the four datasets, we randomly sampled 1000 examples (or the whole dataset if the number of examples is smaller than 1000). These examples are annotated by LLMs. We set an evaluation (i.e., test set) based on 160 randomly sampled examples from each dataset (a total of 640), while the rest remain for training and validation (they will be relevant for subsection 7.1). In addition to the LLM annotations, the evaluation set is also re-annotated by two experts three crowd worker.

SummEval In addition to the TRUE benchmark, we replicate some of the experiments on the full SummEval benchmark (Fabbri et al., 2021). This benchmark includes 1600 generated summaries evaluated on *four dimensions* (relevance, fluency, coherence, consistency) by crowd-workers and experts. The SummEval benchmark is widely used for benchmarking reference-free automatic evaluation methods such as LLM-as-a-judge. In contrast to TRUE, the labeling scheme is *ordinal* on a scale of 1 to 5. For further information on the SummEval data and experimental setting, see Appendix A. Noteworthy, when researchers employ the SummEval benchmark, they use solely the expert annotations. Accordingly, the focus of our experiments conducted on SummEval is (1) to simulate a setup where the original labels are obtained through crowd-sourcing while relying on expert annotations as the gold standard; and (2) to compare the three annotation approaches (crowd-sourcing, experts, and LLMs).

4.2 Annotation Procedure

This subsection outlines the annotation procedures for the various approaches. Refer to Appendix D for additional implementation and technical details not covered here, or Appendix A for the SummEval LLM annotation details.

LLMs We re-annotate the data with four LLMs: GPT-4, (OpenAI, 2023), PaLM2 (Anil et al., 2023), Mistral (7B) (Jiang et al., 2023), Llama 3 (8B) (Dubey et al., 2024), and GPT-4o and Gemini-1.5-Flash for SummEval. Our ensemble model leverages four different prompts which control the variance caused by task descriptions. The prompts are designed as a zero-shot classification task, e.g., for TRUE the requested output is a single token, either `'0'` for factual inconsistency or `'1'` for factual consistency (as described in Figure 12).

Crowd-sourcing Generally, crowd-sourced annotators span a spectrum– from untrained, "common" crowd-workers to carefully selected and trained annotators. Our paper focuses on the lower end of this spectrum. We utilize the platform of Amazon Mechanical Turk (MTurk) to recruit crowd-workers for annotating 100 examples from each TRUE dataset (a total of 400), and to design the interface layout. Examples were randomly assigned to annotators. Each annotated example was manually reviewed. Rejected examples were returned to the pool and re-annotated until each example was annotated by three different annotators. To prevent LLM use, we disabled right-click and `Ctrl+c` in the platform (as suggested by Veselovsky et al., 2023a). To obtain a single label per example, we consider two different aggregations: (1) *Majority* - by majority vote, and (2) *Strict* - if any annotator marks it *inconsistent*, that becomes the label. For SummEval, we use the crowd-sourced annotations provided by Fabbri et al. (2021), aggregated by their median.

Experts All TRUE examples where the prediction differed from the original label, regardless of confidence, were annotated by human experts. The experts are two of the paper's authors, who are fully familiar with the guidelines and task characteristics. Each example was independently annotated by both experts on a scale from 0 (*inconsistent*) to 1 (*consistent*). The examples were shuffled and presented in no specific order, with neither the original nor LLM labels shown. For cases where the experts disagreed, a reconciliation phase followed, during which they discussed and attempted to resolve their differences. For more details on the procedure and annotation platform, see Appendix D.2. After re-annotating all conflicted examples, we define the *gold label* as the original label, if the LLM prediction agrees with it, or the expert resolution, if there was a disagreement. For SummEval, we use the expert annotations provided by Fabbri et al. (2021), aggregated by their median.

5 Label Errors: Analysis and Detection

5.1 Do current benchmarks include mislabeled data?

To address the first research question, we annotate the test-set of TRUE (as described in section 4 using LLMs. We then contrast these annotations with the original labels, to find disagreements. As shown in Table 2, the disagreement rate is significant and can be up to $\sim$ 40% of the examples. An example of such disagreement is presented in Table 1. While this would typically suggest that the LLMs performed poorly, we chose to further investigate these cases and resolve the disagreements. To this end, we asked human experts to re-annotate the examples, allowing us to determine which is more accurate: the original label or the LLMs' prediction.

Dataset: BEGIN
Grounding: Hillary Clinton, the nominee of the Democratic Party for president of the United States in 2016, has taken positions on political issues while serving as First Lady of Arkansas (1979–81; 1983–92), First Lady of the United States (1993–2001);
Generated Text: She is the nominee in 2016.
Original Label: 0 **LLM** p: 0.98 **Gold Label:** 1
Explanation: She (Hillary Clinton) is indeed the nominee in 2016 as specifically stated in the grounding.

Table 1: Example of an annotation error in the original datasets, discovered by LLMs and corrected by experts. In Appendix Table 6 we provide additional examples.

Dataset	Task	% pos	% LLM disagree	% error
MNBM	Summarization	10.6	39.4	16.9 (11.6)
BEGIN	Dialogue	38.7	34.4	21.2 (15.8)
VitaminC	Fact Verification	52.5	17.5	8.1 (4.4)
PAWS	Paraphrasing	44.3	22.5	6.2 (3.0)

Table 2: Summary of LLM disagreement and label error rates across different datasets. %pos is the percentage of positive (i.e., the *consistent* class) examples in the data. % LLM disagree refers to the percentage of examples where the LLM label differs from the original one. % error indicates the error rate in the sampled test set, while the number in parentheses denotes the estimated lower bound of the error rate for the entire dataset.

Our findings show a considerable number of label errors for all examined datasets (see the %error column in Table 2). Based on the experts *gold label* and the sample sizes, we also estimate a lower bound for the total percentage of label errors in the full datasets. We employed the Clopper-Pearson exact method (Clopper and Pearson, 1934) to construct a 95% confidence interval for the binomial proportion, adjusted by a finite population correction (FPC) (see more details in Appendix G.1). We provide the lower bound of these confidence intervals in parentheses in Table 2, under the %error column. The lower bounds range from 3% in the PAWS dataset to 15.8% in the BEGIN dataset.

5.2 Can LLMs Detect Label Errors?

As described in subsection 5.1, we utilize LLMs to flag candidates for mislabeling, and indeed find label errors. In this subsection, we focus on the LLM viewpoint, exploring the effect of LLM confidence,

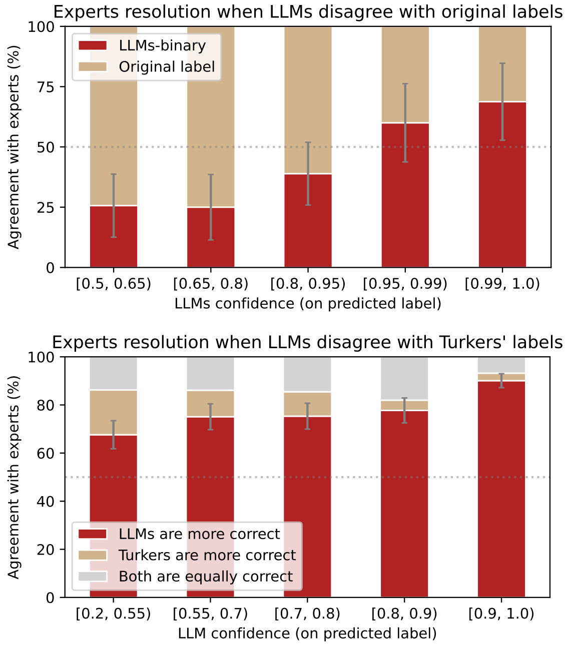


Figure 2: When LLMs disagree with original labels - who is correct? **(Top)** TRUE **(Bottom)** SummEval. As the LLM's confidence grows, so does the precision of identifying an error in the original labels.

and the power of ensemble.

Confidence LLM annotations are valuable for flagging mislabeled data, offering more than just hard labels. By considering LLM confidence scores alongside their predictions, we can improve the precision of automatic error detection. Leveraging confidence can reduce re-annotation efforts by flagging only cases exceeding a predefined threshold. The rationale is that not all flagged examples should be treated equally. Instances flagged with low confidence indicate that the LLM recognizes a potential issue, however, when the LLM is highly confident in a label that contradicts the original one, it provides a stronger signal of a possible error.

Figure 2 shows the rate of the experts' agreement with the LLMs compared to the agreement with original labels, divided into confidence-based bins. Bins are balanced by size, and defined by a confidence interval of 95% based on bootstrap sampling (see Appendix G.2 for further details). The bins reflect increasing levels of LLM confidence in its predicted label (i.e., a stronger disagreement between LLMs and the original labels).

From the top of Figure 2, we observe a clear trend: as LLM confidence increases, so does its precision in detecting label errors in the original dataset. In the highest confidence bin, LLM annotations surpass the original labels in agreement with expert re-labeling, and this difference is statistically significant. This indicates that when the LLM is highly confident in its disagreement with the original label, the labeled example serves as a strong candidate for a labeling error. Note that even in cases where the expert agreement with LLMs was below 50%, mislabeled data was still discovered.

We replicated this analysis on the SummEval dataset (bottom of Figure 2) and observed a similar trend: higher confidence increases the likelihood that the LLM prediction is closer to the expert annotation than the original label. In the SummEval case, we consider the crowd-sourced labels as the original labels. For more details see Appendix A.

Ensemble By varying the size of the LLM ensemble, we examine two key aspects: predictive power (how well predictions align with gold labels, measured by ROC AUC for TRUE and average correlation for SummEval), and error detection power (measured by F1-score, averaging the recall of errors and the precision of correctly identifying a candidate as a true error). The ensemble power analysis is presented in Figure 3, with additional details in Appendix B. Our findings show that incorporating multiple LLMs and prompts in an ensemble is valuable. As the ensemble size increases, both label quality and error detection improve.

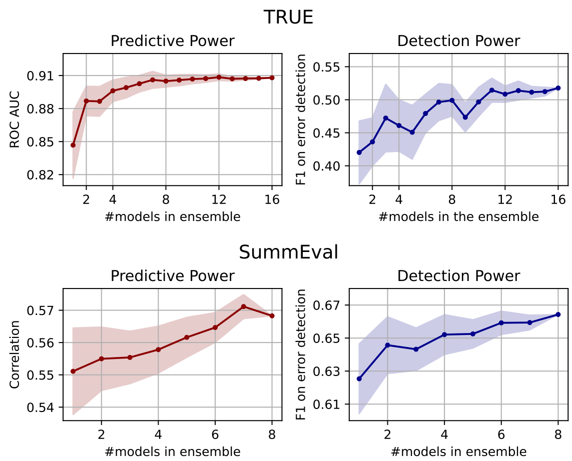


Figure 3: The power of ensemble. **(Top)** TRUE **(Bottom)** SummEval.As the ensemble size increases (**x-axis**), its performance against gold labels (**Left**), and its ability to detect label errors (**Right**) improves.

6 Comparing Annotation Approaches

Our paper discusses three annotation approaches, each with its own benefits and drawbacks, differing in how they balance label quality, scalability, and cost. Due to space limitations, we provide a

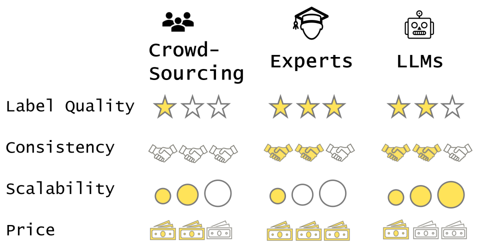


Figure 4: Annotation approaches comparison.

concise summary of our key findings here, with the full analysis available in Appendix C. Figure 4 highlights the main insights.

LLMs exhibit strong agreement with experts and among themselves. Inter-annotator agreement (IAA) among LLMs, as well as their alignment with expert annotations, are significantly higher than that of crowd workers. In contrast, crowd-sourced annotations exhibit larger variability and lower agreement with experts, making them less reliable without additional verification.

Crowd worker quality improves with experience but remains inconsistent. Our analysis shows that experienced crowd workers produce higher-quality annotations. However, even among them, annotation quality and consistency remain lower than LLM-based annotation, which is more reliable.

LLMs provide fast, scalable, and cost-efficient annotation. Compared to expert and crowd-sourced annotation, LLMs require less time and are much more cost-effective per annotation, making them a viable alternative for large-scale annotation while effectively balancing the trade-off.

7 Implications of Mislabeled Data

7.1 Training on Mislabeled Data

Training on mislabeled data can harm model performance and stability, as learning from errors makes it harder to identify consistent patterns. The impact depends on various factors, such as the fraction of mislabeled data and the training procedure. In this subsection, we show that addressing this issue, even heuristically, significantly improves the model's performance on a test set.

Handling Label Errors In order to handle label errors in the training set, and reduce its effect on model performance, we propose two manipulations. For both manipulations, we flag examples where the model strongly disagrees with the original label(i.e., with confidence above a certain threshold).

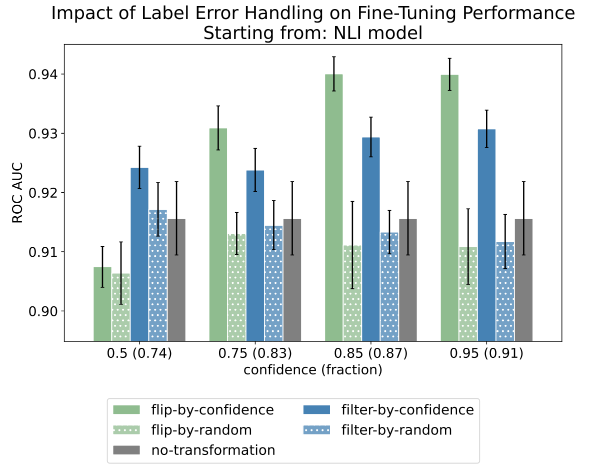


Figure 5: Fine-tuning a model on a transformed dataset. The gray bar is the original dataset - without any changes. The green bars present results for label flipping for a subset of examples, determined by LLMs-confidence (plain), or at random (dotted). The blue bars represent filtering of these examples.

The first manipulation is *filtering* flagged examples out, which maintains a "cleaner" yet smaller training set. The second manipulation is label *flipping* for flagged examples, which maintains the same amount of data, but may also cause harm if flipping too many correct labels.

Experimental Setup We set the training set to be the additional data examples from the datasets (i.e., MNBM, BEGIN, VitaminC, PAWS), which are disjoint from the test set. Note that we posses gold labels for the test set alone, while for the training set we only extract the confidence. The fine-tuning procedure includes splitting the training set into train and validation sets, and fine-tuning on the train set. We report average results of five seeds.

As an ablation study, we also apply these manipulations on a random subset of examples rather than the flagged examples. The ablation study aims to maintain a consistent number of training examples, while the ablation for flipping aims to address the claim that in some cases, a relatively small fraction of label errors may be even considered as a noise that improves model robustness (e.g., as in label perturbation (Zhang et al., 2018) or label smoothing (Szegedy et al., 2016)).

We conducted this experiment starting from two base models: DeBERTa-v3, and a fine-tuned version of it on classic NLI datasets, which we will refer to as the NLI-base model. We chose the NLI-base model as NLI tasks closely resemble factual consistency evaluation (FCE), making it

Model	Rank		ROC AUC		F1 Score		Accuracy	
	Original	Gold	Original	Gold	Original	Gold	Original	Gold
GPT-4	3	1 (+2)	0.81	0.93 (+15%)	0.73	0.83 (+14%)	0.73	0.83 (+14%)
NLI model	1	2 (–1)	0.93	0.91 (–2%)	0.87	0.87 (—)	0.87	0.87 (—)
PaLM2	6	3 (+3)	0.81	0.91 (+12%)	0.71	0.81 (+14%)	0.71	0.81 (+14%)
GPT-4o	4	4 (—)	0.81	0.91 (+12%)	0.74	0.83 (+12%)	0.74	0.83 (+12%)
GPT-4-mini	5	5 (—)	0.81	0.91 (+12%)	0.71	0.79 (+11%)	0.70	0.79 (+13%)
Llama3	7	6 (+1)	0.75	0.86 (+15%)	0.47	0.50 (+6%)	0.52	0.55 (+6%)
Mistral-v0.3	8	7 (+1)	0.75	0.85 (+13%)	0.61	0.68 (+11%)	0.62	0.68 (+10%)
DeBERTa-v3	2	8 (–6)	0.84	0.80 (–5%)	0.76	0.73 (–4%)	0.76	0.73 (–4%)
Mistral-v0.2	9	9 (—)	0.73	0.82 (+12%)	0.66	0.72 (+9%)	0.66	0.72 (+9%)

Table 3: Comparison of Model Performance on Original and Gold Labels. Ranking is defined over ROC AUC.

well-suited for this experiment. Given the similar trends, we present the results for the NLI model here. Additional experiments and implementation details can be found in Appendix F.1.

Results Figure 5 shows the results of our experiments. In our confidence-based approaches, we clearly see the trend that as the confidence threshold—according to which our manipulations are applied—grows, our manipulation results in improved ROC AUC for both models. This trend eventually (i.e., for high enough LLM confidence) brings these approaches to significantly outperform the baseline. In contrast, when we applied our manipulations on random subsets, we generally see a diminishing effect of manipulation, converging to the no-manipulation baseline.

Comparing between the handling approaches, it appears that flipping is better than filtering for high confidence. We hypothesize that this stems from the amount of data that remains after flipping (i.e., the same amount as before the flipping) compared to the filtering approach, combined with the high error rate in these datasets. Note that this is contrary to the random case where filtering is better than flipping, as flipping a subset with low error-rate brings more damage than value.

7.2 Evaluating on Mislabeled Data

In this subsection, we examine the impact of mislabeled data in evaluation sets and its potential to distort results. Labeling errors can mislead the evaluation process, resulting in inaccurate performance metrics and, in some cases, flawed model comparisons that lead to incorrect conclusions.

Experimental Setup To test this assumption, we evaluate the performance of nine models, mostly state-of-the-art LLMs, on the test datasets. We compare their performance between the *original* labels, and the *gold* labels. For LLMs, we used zero-shot prediction as described in section 3, and averaged over prompts. For DeBERTa-based models, we used the fine-tuned models from subsection 7.1, and averaged over seeds.

Results Prior to this work, an evaluation of these models would induce the values and ranking as in Table 3 under the *Original* sub-columns. However, as shown before, these datasets include labeling errors, and therefore do not support fair evaluation. Considering the new gold labels, based on expert intervention (as described in subsection 4.2), we obtain different results, shown in the *Gold* sub-columns. The first observed discrepancy is the ranking of models. For example, DeBERTa-v3 has shifted from being the second-best to the second-worst. Beyond the change in ranking, all metrics' (i.e., ROC AUC, F1-score, and accuracy) range has shifted upward, indicating that LLMs perform better on this task than what was previously thought, likely due to label errors. If this phenomenon extends to other tasks and datasets beyond those examined in this study, it could suggest that LLMs are better than currently perceived.

8 Discussion

Labeling errors are a persistent issue in NLP datasets, negatively affecting model fine-tuning and evaluation. Our findings demonstrate that LLMs, particularly when highly confident, can effectively detect these errors, outperforming crowd workers in accuracy, consistency, and cost-efficiency. As LLM capabilities advance, their role in refining data quality will become central to improving NLP benchmarks. Future work could explore applying LLM-based error detection to a broader range of datasets and tasks, as well as refining methods for optimizing label correction strategies. We encourage researchers to adopt our methods and critically evaluate existing datasets to drive more robust, reliable results in the field.

Limitations

While our study provides valuable insights into the role of LLMs in identifying label errors and improving dataset quality, several limitations should be considered. First, crowd workers encompass a broad range of annotators with varying expertise and training. Our analysis, focuses on the "common" crowd worker, typically an annotator selected with minimal qualifications, such as an approved task completion rate, and without specialized training. However, some datasets implement additional measures, such as requiring prior experience or task-specific instruction, which can influence annotation quality. Importantly, we did not take crowd-worker annotations at face value; we applied filtering (based on the explanation crowd workers were asked to write for each example) to remove a substantial number of low-quality assignments, such as clearly invalid responses, in addition to enforcing minimal qualification criteria.

Second, our analysis does not account for potential data contamination, where LLMs may have been trained on the datasets we evaluate. However, since our analysis focuses on identifying and correcting label errors within these datasets, contamination would likely hinder rather than enhance our findings. If an LLM had memorized these datasets, it would be more likely to reproduce existing errors rather than detect and correct them, making contamination a potential limitation only for certain aspects of evaluation but not for our core claims.

Third, LLM-based annotations can vary depending on the choice of prompting strategies and ensemble methods. In this work, we use zero-shot prompting and simple averaging for ensembling. Still, alternative approaches – such as few-shot prompting, chain-of-thought reasoning (Wei et al., 2022), or self-refine (Madaan et al., 2023) – could improve annotation accuracy and consistency. Likewise, for ensembling, more advanced methods—such as percentile-based aggregation (Sherratt et al., 2023), error-aware weighting (Freund and Schapire, 1997), confidence-aware methods (Lee, 2010; Lu et al., 2024), or even LLM-based aggregation strategies like debate variants (Liang et al., 2023; Du et al., 2024) – may yield more reliable consensus labels. We leave the exploration of these strategies for future work and hope our study encourages such further research.

References

Mohammad Allahbakhsh, Boualem Benatallah, Aleksandar Ignjatovic, Hamid Reza Motahari-Nezhad, Elisa Bertino, and Schahram Dustdar. 2013. Quality control in crowdsourcing systems: Issues and directions. *IEEE Internet Computing*, 17(2):76–81.

Rohan Anil, Andrew M. Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, Eric Chu, Jonathan H. Clark, Laurent El Shafey, Yanping Huang, Kathy Meier-Hellstern, Gaurav Mishra, Erica Moreira, Mark Omernick, Kevin Robinson, Sebastian Ruder, Yi Tay, Kefan Xiao, Yuanzhong Xu, Yujing Zhang, Gustavo Hernández Ábrego, Junwhan Ahn, Jacob Austin, Paul Barham, Jan A. Botha, James Bradbury, Siddhartha Brahma, Kevin Brooks, Michele Catasta, Yong Cheng, Colin Cherry, Christopher A. Choquette-Choo, Aakanksha Chowdhery, Clément Crepy, Shachi Dave, Mostafa Dehghani, Sunipa Dev, Jacob Devlin, Mark Díaz, Nan Du, Ethan Dyer, Vladimir Feinberg, Fangxiaoyu Feng, Vlad Fienber, Markus Freitag, Xavier Garcia, Sebastian Gehrmann, Lucas Gonzalez, and et al. 2023. Palm 2 technical report. *CoRR*, abs/2305.10403.

Anaëlle Baledent, Yann Mathet, Antoine Widlöcher, Christophe Couronne, and Jean-Luc Manguin. 2022. Validity, agreement, consensuality and annotated data quality. In *International Conference on Language Resources and Evaluation*.

Savita Bhat and Vasudeva Varma. 2023. Large language models as annotators: A preliminary evaluation for annotating low-resource language content. In *Proceedings of the 4th Workshop on Evaluation and Comparison of NLP Systems*, pages 100–107, Bali, Indonesia. Association for Computational Linguistics.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Nitay Calderon, Naveh Porat, Eyal Ben-David, Alexander Chapanin, Zorik Gekhman, Nadav Oved, Vitaly Shalumov, and Roi Reichart. 2024. Measuring the robustness of nlp models to domain shifts. *arXiv preprint arXiv:2306.00168*.

Nitay Calderon and Roi Reichart. 2024. On behalf of the stakeholders: Trends in NLP model interpretability in the era of llms. *CoRR*, abs/2407.19200.

Hung Chau, Saeid Balaneshin, Kai Liu, and Ondrej Linda. 2020. Understanding the tradeoff between cost and quality of expert annotations for keyphrase extraction. In *Law*.

Honghua Chen and Nai Ding. 2023. Probing the "creativity" of large language models: Can models produce divergent semantic association? In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 12881–12888, Singapore. Association for Computational Linguistics.

Ruirui Chen, Chengwei Qin, Weifeng Jiang, and Dongkyu Choi. 2024. Is a large language model a good annotator for event extraction? In *AAAI Conference on Artificial Intelligence*.

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)*, pages 15607–15631, Toronto, Canada. Association for Computational Linguistics.

Michael Chmielewski and Sarah C. Kucker. 2019. An mturk crisis? shifts in data quality and the impact on study results. *Social Psychological and Personality Science*, 11:464 – 473.

Derek Chong, Jenny Hong, and Christopher D. Manning. 2022a. Detecting label errors by using pre-trained language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 9074–9091. Association for Computational Linguistics.

Derek Chong, Jenny Hong, and Christopher D. Manning. 2022b. Detecting label errors by using pre-trained language models. In *Conference on Empirical Methods in Natural Language Processing*.

C. J. Clopper and E. S. Pearson. 1934. The use of confidence or fiducial limits illustrated in the case of the binomial. *Biometrika*, 26(4):404–413.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Thomas G. Dietterich. 2007. Ensemble methods in machine learning.

Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2019. Wizard of wikipedia: Knowledge-powered conversational agents. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B. Tenenbaum, and Igor Mordatch. 2024. Improving factuality and reasoning in language models through multiagent debate. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, and et al. 2024. The llama 3 herd of models. *CoRR*, abs/2407.21783.

Nouha Dziri, Hannah Rashkin, Tal Linzen, and David Reitter. 2022. Evaluating attribution in dialogue systems: The BEGIN benchmark. *Transactions of the Association for Computational Linguistics*, 10:1066–1083.

Alexander R. Fabbri, Wojciech Kryscinski, Bryan McCann, Caiming Xiong, Richard Socher, and Dragomir R. Radev. 2021. Summeval: Re-evaluating summarization evaluation. *Trans. Assoc. Comput. Linguistics*, 9:391–409.

Virginia K. Felkner, Jennifer A. Thompson, and Jonathan May. 2024. Gpt is not an annotator: The necessity of human annotation in fairness benchmark construction. *ArXiv*, abs/2405.15760.

Joseph L. Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76:378–382.

Benoît Frénay and Michel Verleysen. 2014. Classification in the presence of label noise: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 25:845–869.

Yoav Freund and Robert E Schapire. 1997. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139.

Yair Ori Gat, Nitay Calderon, Amir Feder, Alexander Chapanin, Amit Sharma, and Roi Reichart. 2024. Faithful explanations of black-box NLP models using llm-generated counterfactuals. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Zorik Gekhman, Jonathan Herzig, Roee Aharoni, Chen Elkind, and Idan Szpektor. 2023. TrueTeacher: Learning factual consistency evaluation with large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2053–2070, Singapore. Association for Computational Linguistics.

Fabrizio Gilardi, Meysam Alizadeh, and Maël Kubli. 2023. Chatgpt outperforms crowd workers for text-annotation tasks. *Proceedings of the National Academy of Sciences of the United States of America*, 120.

Degan Hao, Lei Zhang, Jules H. Sumkin, Aly A. Mohamed, and Shandong Wu. 2020. Inaccurate labels in weakly-supervised deep learning: Automatic identification and correction and their impact on classification performance. *IEEE Journal of Biomedical and Health Informatics*, 24:2701–2710.

David N. Hauser, Aaron J. Moss, Cheskie Rosenzweig, Shalom N. Jaffe, Jonathan Robinson, and Leib Litman. 2021. Evaluating cloudresearch's approved group as a solution for problematic data quality on mturk. *Behavior Research Methods*, 55:3953 – 3964.

Xingwei He, Zheng-Wen Lin, Yeyun Gong, Alex Jin, Hang Zhang, Chen Lin, Jian Jiao, Siu Ming Yiu, Nan Duan, and Weizhu Chen. 2023. Annollm: Making large language models to be better crowdsourced annotators. In *North American Chapter of the Association for Computational Linguistics*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Or Honovich, Roee Aharoni, Jonathan Herzig, Hagai Taitelbaum, Doron Kukliansky, Vered Cohen, Thomas Scialom, Idan Szpektor, Avinatan Hassidim, and Yossi Matias. 2022. TRUE: re-evaluating factual consistency evaluation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pages 3905–3920. Association for Computational Linguistics.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *CoRR*, abs/2310.06825.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *CoRR*, abs/2001.08361.

Gabriella Kazai, Jaap Kamps, and Natasa Milic-Frayling. 2013. An analysis of human factors and label accuracy in crowdsourcing relevance judgments. *Inf. Retr.*, 16(2):138–178.

Ryan Kennedy, Scott Clifford, Tyler Burleigh, Philip D Waggoner, Ryan Jewell, and Nicholas JG Winter. 2020. The shape of and solutions to the mturk quality crisis. *Political Science Research and Methods*, 8(4):614–629.

Nataliia Kholodna, Sahib Julka, Mohammad Khodadadi, Muhammed Nurullah Gumus, and Michael Granitzer. 2024. Llms in the loop: Leveraging large language model annotations for active learning in low-resource languages. *ArXiv*, abs/2404.02261.

Han Jun Kim, Kushan Mitra, Rafael Li Chen, Sajjadur Rahman, and Dan Zhang. 2024. Meganno+: A human-llm collaborative annotation system. In *Conference of the European Chapter of the Association for Computational Linguistics*.

Klaus Krippendorff. 1970. Estimating the reliability, systematic error, and random error of interval data. *Educational and Psychological Measurement*, 30(1):61–70.

Chi-Hoon Lee. 2010. Learning to combine discriminative classifiers: confidence based. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, July 25-28, 2010*, pages 743–752. ACM.

Minzhi Li, Taiwei Shi, Caleb Ziems, Min-Yen Kan, Nancy F. Chen, Zhengyuan Liu, and Diyi Yang. 2023. Coannotating: Uncertainty-guided work allocation between human and large language models for data annotation. *ArXiv*, abs/2310.15638.

Tian Liang, Zhiwei He, Wenxiang Jiao, Xing Wang, Yan Wang, Rui Wang, Yujiu Yang, Zhaopeng Tu, and Shuming Shi. 2023. Encouraging divergent thinking in large language models through multi-agent debate. *CoRR*, abs/2305.19118.

Shir Lissak, Nitay Calderon, Geva Shenkman, Yaakov Ophir, Eyal Fruchter, Anat Brunstein Klomek, and Roi Reichart. 2024. The colorful future of llms: Evaluating and improving llms as emotional supporters for queer youth. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), NAACL 2024, Mexico City, Mexico, June 16-21, 2024*, pages 2040–2079. Association for Computational Linguistics.

Jian Lu, Wei Li, Qingren Wang, and Yiwen Zhang. 2020. Research on data quality control of crowdsourcing annotation: A survey. In *2020 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)*, pages 201–208.

Zhihe Lu, Jiawang Bai, Xin Li, Zeyu Xiao, and Xinchao Wang. 2024. Beyond sole strength: Customized ensembles for generalized vision-language models. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.

Bill MacCartney and Christopher D. Manning. 2009. An extended model of natural logic. In *Proceedings of the Eight International Conference on Computational Semantics*, pages 140–156, Tilburg, The Netherlands. Association for Computational Linguistics.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan T. McDonald. 2020. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 1906–1919. Association for Computational Linguistics.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807, Brussels, Belgium. Association for Computational Linguistics.

An Thanh Nguyen, Byron C. Wallace, and Matthew Lease. 2015. Combining crowd and expert labels using decision theoretic active learning. In *AAAI Conference on Human Computation & Crowdsourcing*.

Duc Tam Nguyen, Chaithanya Kumar Mummadi, Thi-Phuong-Nhung Ngo, Thi Hoai Phuong Nguyen, Laura Beggel, and Thomas Brox. 2019. Self: Learning to filter noisy labels with self-ensembling. *ArXiv*, abs/1910.01842.

Curtis G. Northcutt, Anish Athalye, and Jonas W. Mueller. 2021. Pervasive label errors in test sets destabilize machine learning benchmarks. *ArXiv*, abs/2103.14749.

Curtis G. Northcutt, Lu Jiang, and Isaac L. Chuang. 2019. Confident learning: Estimating uncertainty in dataset labels. *J. Artif. Intell. Res.*, 70:1373–1411.

OpenAI. 2023. GPT-4 technical report. *CoRR*, abs/2303.08774.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Geoff Pleiss, Tianyi Zhang, Ethan R. Elenberg, and Kilian Q. Weinberger. 2020. Identifying mislabeled data using the area under the margin ranking. *ArXiv*, abs/2001.10528.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Frederick Reiss, Hong Xu, Bryan Cutler, Karthik Muthuraman, and Zachary Eichenberger. 2020. Identifying incorrect labels in the conll-2003 corpus. In *Proceedings of the 24th Conference on Computational Natural Language Learning, CoNLL 2020, Online, November 19-20, 2020*, pages 215–226. Association for Computational Linguistics.

Simon Rogers, Derek H. Sleeman, and John Kinsella. 2013. Investigating the disagreement between clinicians' ratings of patients in icus. *IEEE J. Biomed. Health Informatics*, 17(4):843–852.

Tal Schuster, Adam Fisch, and Regina Barzilay. 2021. Get your vitamin C! robust fact verification with contrastive evidence. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 624–643, Online. Association for Computational Linguistics.

Katharine Sherratt, Hugo Gruson, Rok Grah, Helen Johnson, Rene Niehus, Bastian Prasse, and et al. 2023. Predictive performance of multi-model ensemble forecasts of covid-19 across european nations. *eLife*, 12:e81916.

Rion Snow, Brendan T. O'Connor, Dan Jurafsky, and A. Ng. 2008. Cheap and fast – but is it good? evaluating non-expert annotations for natural language tasks. In *Conference on Empirical Methods in Natural Language Processing*.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R. Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, Agnieszka Kluska, Aitor Lewkowycz, Akshat Agarwal, Alethea Power, Alex Ray, Alex Warstadt, Alexander W. Kocurek, Ali Safaya, Ali Tazarv, Alice Xiang, Alicia Parrish, Allen Nie, Aman Hussain, Amanda Askell, Amanda Dsouza, Ambrose Slone, Ameet Rahane, Anantharaman S. Iyer, Anders Andreassen, and et al. 2023. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Trans. Mach. Learn. Res.*, 2023.

Julius Steen, Juri Opitz, Anette Frank, and Katja Markert. 2023. With a little push, NLI models can robustly and efficiently predict faithfulness. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 914–924, Toronto, Canada. Association for Computational Linguistics.

Aneeta Sylolypavan, Derek H. Sleeman, Honghan Wu, and Malcolm Sim. 2023. The impact of inconsistent human annotations on AI driven clinical decision making. *npj Digit. Medicine*, 6.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 2818–2826. IEEE Computer Society.

Derek Tam, Anisha Mascarenhas, Shiyue Zhang, Sarah Kwan, Mohit Bansal, and Colin Raffel. 2023. Evaluating the factual consistency of large language models through news summarization. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 5220–5255, Toronto, Canada. Association for Computational Linguistics.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and VERification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 809–819, New Orleans, Louisiana. Association for Computational Linguistics.

Petter Törnberg. 2023. Chatgpt-4 outperforms experts and crowd workers in annotating political twitter messages with zero-shot learning. *ArXiv*, abs/2304.06588.

Alexandra Uma, Tommaso Fornaciari, Dirk Hovy, Silviu Paun, Barbara Plank, and Massimo Poesio. 2021. Learning from disagreement: A survey. *J. Artif. Intell. Res.*, 72:1385–1470.

Mor Ventura, Eyal Ben-David, Anna Korhonen, and Roi Reichart. 2023. Navigating cultural chasms: Exploring and unlocking the cultural POV of text-to-image models. *CoRR*, abs/2310.01929.

Veniamin Veselovsky, Manoel Horta Ribeiro, Philip Cozzolino, Andrew Gordon, David Rothschild, and Robert West. 2023a. Prevalence and prevention of large language model use in crowd work. *CoRR*, abs/2310.15683.

Veniamin Veselovsky, Manoel Horta Ribeiro, and Robert West. 2023b. Artificial artificial artificial intelligence: Crowd workers widely use large language models for text production tasks. *CoRR*, abs/2306.07899.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Tong Wang, Ninad Kulkarni, and Yanjun Qi. 2024. Less is more for improving automatic evaluation of factual consistency. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 6: Industry Track)*, pages 324–334, Mexico City, Mexico. Association for Computational Linguistics.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Atharva Naik, Arjun Ashok, Arut Selvan Dhanasekaran, Anjana Arunkumar, David Stap, Eshaan Pathak, Giannis Karamanolakis, Haizhi Lai, Ishan Purohit, Ishani Mondal, Jacob Anderson, Kirby Kuznia, Krima Doshi, Kuntal Kumar Pal, Maitreya Patel, Mehrad Moradshahi, Mihir Parmar, Mirali Purohit, Neeraj Varshney, Phani Rohitha Kaza, Pulkit Verma, Ravsehaj Singh Puri, Rushang Karia, Savan Doshi, Shailaja Keyur Sampat, Siddhartha Mishra, Sujan Reddy A, Sumanta Patro, Tanay Dixit, and Xudong Shen. 2022. Super-NaturalInstructions: Generalization via declarative instructions on 1600+ NLP tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5085–5109, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122. Association for Computational Linguistics.

Jianguo Xia, David I. Broadhurst, Michael Wilson, and David Scott Wishart. 2012. Translational biomarker discovery in clinical metabolomics: an introductory tutorial. *Metabolomics*, 9:280 – 299.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. Alignscore: Evaluating factual consistency with a unified alignment function. In *Annual Meeting of the Association for Computational Linguistics*.

Hongyi Zhang, Moustapha Cissé, Yann N. Dauphin, and David Lopez-Paz. 2018. mixup: Beyond empirical risk minimization. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Ruoyu Zhang, Yanzeng Li, Yongliang Ma, Ming Zhou, and Lei Zou. 2023. Llmaaa: Making large language models as active annotators. *ArXiv*, abs/2310.19596.

Yuan Zhang, Jason Baldridge, and Luheng He. 2019. PAWS: Paraphrase adversaries from word scrambling. 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 1298–1308, Minneapolis, Minnesota. Association for Computational Linguistics.

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, December 10 - 16, 2023*.

Appendix

A Additional Experiments - SummEval

In addition to the datasets from the TRUE benchmark, we replicate our experiments on another dataset with a different objective and a different labeling scheme, to strengthen our results and conclusions.

A.1 Data

SummEval (Fabbri et al., 2021) is an extensive and commonly used summarization benchmark, evaluating the quality of multiple model-generated summarization outputs compared to a source CNN/DailyMail sources on four dimensions: coherence, relevance, consistency, and fluency. Each summarization is labeled on each dimension with five crowd-workers and three experts, enabling us to replicate some of the experiments without additional crowd-worker or expert annotation costs. The labeling schema is ordinal on a scale of 1 to 5 (higher is better). Note that this dataset does not have a singular gold-standard label per summarization, but rather a collection of annotations from experts and crowd-workers. Therefore, we will not claim to find label errors in this benchmark, but rather showcase our methodology as if the crowd-sourced annotations are the original labels for the dataset, and we have access to experts' annotations for gold-standard reference, to determine if the LLM was correct when flagging examples.

A.2 Definitions

To apply our methods for error detection via LLMs ensemble, we first define the following:

Labels We aggregate crowd-sourced annotations by their median, to construct a single original label on a scale of 1 to 5. Similarly, we take the median of the experts' annotations to be a single gold-standard label.

A disagreement We say that the LLM annotation *disagrees* with the original label if there is a difference of more than 1 between the scores. The idea is that we can confidently say the LLM and the original label "disagree", as if the difference is 1 or less, this is a weak disagreement we will probably not flag for.

A.3 Experimental Setting

Similar to the description in subsection 4.2, we utilize two LLMs– GPT-4o (`gpt-4o-2024-11-20`) and Gemini 1.5 Flash (`gemini-1.5-flash-002`). We constructed four prompts, differing by phrasing and compatible with the four prompt template structures used for the TRUE benchmark experiments. The answer to each query was a JSON format with 'Relevance', 'Coherence', 'Consistency', and 'Fluency' as its keys. The scores are integers on a scale of 1 to 5, as are the ratings in the SummEval dataset. We extract the probability of each score possible through the log-probs for each score token. Finally, we average all models' probabilities, to obtain an ensemble of LLMs, with p being the distribution over the five possible scores.

A.4 Experiments and Results

A.4.1 Can LLMs Detect Label Errors?

We replicate the experiment described in subsection 5.2 with the appropriate adjustment for the SummEval dataset, based on the definitions above. The result is shown in Figure 2 (bottom). The plot presents the subset of examples where there was a disagreement between the crowd-sourced annotation and the LLMs' annotation. Each bin represents the confidence of the LLMs in their predicted label. As there are five ordinal categories, even if there was a disagreement between two annotations, they both might be "wrong", where the expert's answer is a third option. Therefore, to show clearer results, we do not resolve by experts "who is correct", but rather "who is more correct?". For completeness, we also provide the "both equally correct" option, for the case the expert's label is exactly in the middle, and none is "more correct" than the other. The bins are relatively balanced in terms of the amount of examples per bin. Note that in contrast to the TRUE binary labeling scheme, where confidence 0.5 is the minimal threshold for an answer, here we start from 0.2.

From the results, we see a clear dominance of the LLM over the crowd-sourced annotations, for all confidence bins. This suggests that the LLMs not only *detect* error by flagging possibly mislabeled data points, but also provide better answers, which can account for error *correction*. Similar to the result on the TRUE benchmark, we observe a trend where as the LLMs' confidence increases, they are more correct, indicating that they find label errors with higher precision. However, in this dataset, the difference from the original labels (in this case, the MTurk labels) is even more apparent, and the LLMs are correct even when with lower confidence.

A.4.2 The Power of Ensemble

We analyze the importance of utilizing more than a single model and a single prompt on two dimensions - performance compared to the gold labels (the quality of the annotations we utilize), and error detection (the ability to identify errors more accurately). For performance evaluation on the ordinal labels, we report Pearson correlation; for error detection evaluation, we report the F1-score based on binary error/not-error classification. See results in Figure 3 and discussion in Appendix B.

A.4.3 Annotation Approaches Comparison

In Appendix C, we thoroughly discuss the comparison between the different annotation approaches. For SummEval, experts and crowd-sourced annotations are provided. Together with our LLM-ensemble annotations (as described in subsection A.3), we analyze and compare the annotation approaches in terms of quality (see Figure 6 (bottom)) and consistency (see Table 5). To account for ordinal labels, we measure IAA via Krippendorff's α (Krippendorff, 1970).

B The Power of Ensemble

As mentioned in subsection 4.2, we treat the LLM annotations as an ensemble of 2 models combined with 4 different prompts, in order to ensure greater stability in the results. Where one LLM may succeed, the other may fail, and averaging all their probabilities enables us to have more confidence in the final answer. In this subsection, we further analyzed the performance of LLMs by varying the size of the LLM ensemble, examining how this affects the model performance. We evaluate two aspects of model performance. First, we assess how closely the ensemble's annotations match the gold labels– essentially, how much we can trust the LLM annotations. We measure this aspect of label quality using the ROC AUC compared to the gold labels. The second aspect is the ensemble's ability to detect label errors. For this, we compute the F1-score by averaging the recall of errors and the precision of correctly identifying a candidate as a true error.

Results are shown in Figure 3 (top). For both aspects, we see a clear trend. As we increase the number of models in the ensemble, the performance increases. In terms of ROC AUC w.r.t the gold labels (left plot), this suggests better annotation quality, while the right plot, a higher F1 score indicates a stronger error detector, either by recalling more errors or improving precision, or through a balance of both. Additionally, for both measures, the variance decreases as the ensemble size grows, which indicates more stable and consistent annotations and error detections. Similarly, Figure 3 (bottom) shows the power of LLM ensemble on the same aspects on the SummEval datasets, aggregated over four summarization dimensions (see experiment details on Appendix A.4.2). Trends of diminishing variance and increased performance and error detection are observed here as well.

Although not yet discussed in the context of error detection with LLMs, these results align with previous work showing the power of ensemble (Dietterich, 2007). These observations justify our choice to use an ensemble of models rather than a single one.

C Comparing Annotation Approaches

Our paper discusses three annotation approaches, each with its own benefits and drawbacks. These approaches differ in how they manage the trade-offs between label quality, scalability, and cost. In the following section, we discuss and compare their characteristics. A summary of this comparison is given in Figure 4.

C.1 Annotation Quality

When annotating or validating a dataset, one of our main concerns is the quality of the labels, or in other words, establishing a reliable gold standard. However, each annotation approach produces different labels. To estimate the quality of these approaches, we measure the agreement between different annotations using the weighted F1-score (which accounts for both classes). Note that this metric is not symmetric, meaning that treating one annotation as the *true* label and the other as the *prediction*, or vice versa, can result in different scores.

Figure 6 (top) presents the F1-score between each pair of annotation approaches. As the figure shows, LLMs have disagreements with the *original* labels (0.72). Yet, as discussed in subsection 5.1, the original labels themselves contain mistakes, so this disagreement does not necessarily indicate poor performance of the LLMs. When considering the *Gold* as the true label, LLM performance increases to 0.83. This suggests that LLMs, despite their discrepancies with the original labels, perform closer to the truth than initially reported. The *Gold* label, obtained by experts, has high agreement with both the *Original* and *LLM* labels. On the other hand, the *MTurk-Majority* approach performs poorly, with near-random F1-scores compared to both the original and gold labels, and even when compared to its stricter variant, *MTurk-Strict*. The results indicate that basic crowd-sourcing, without additional training to enhance crowd-workers into specialized sub-experts, performs significantly worse compared to other approaches, including LLM-based methods. On the SummEval dataset

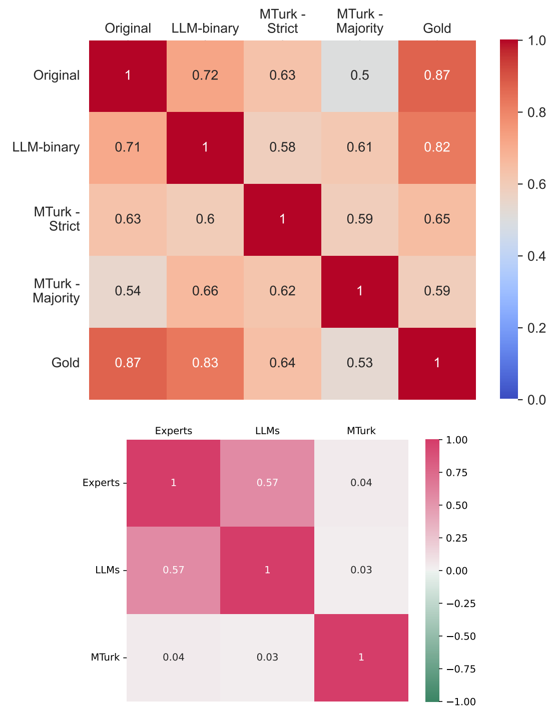


Figure 6: Comparison between all annotation methods: **(Top)** on the TRUE benchmark, measured by the weighted-F1-score. Rows represent the *"true"* label and columns represent the *"prediction"*. For instance, the score of *LLMs* compared to the *Original* label is 0.72. **(Bottom)** Comparison on the SummEval benchmark, measured by Pearson correlation (results are averaged over all dimensions).

(bottom of Figure 6 bottom) we observe similar results, where the LLMs are more correlated with the Experts rather than the crowd-workers, which in turn have almost-no-correlation with LLMs or experts' annotations– this implies poor quality of the annotations obtained from crowd-source.

Crowd-sourcing For crowd-sourcing, the reported F1-score does not provide the complete picture. When we focus on individual annotators, we see that those who annotate more examples generally deliver higher-quality annotations, achieving greater accuracy when compared to both the original and gold labels (see Figure 7). This phenomenon can be explained by two hypotheses: (1) a learning process– as the annotators see more examples, they improve at the task, or (2) users who dedicate time to annotating multiple examples are likely those who either read the guidelines carefully and strive to perform the task to the best of their ability, or are naturally proficient at the task and therefore continue annotating. Even though annotators who label more instances tend to provide higher-quality

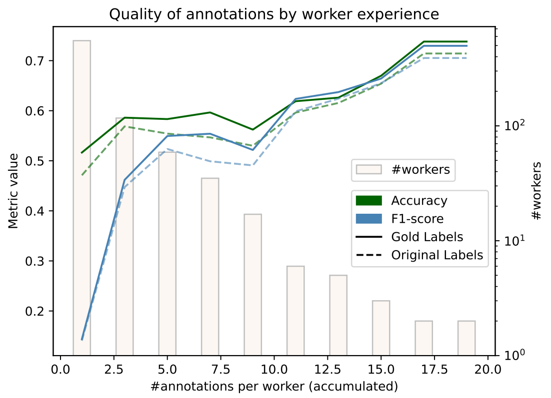


Figure 7: **(x-axis)** at list x annotations per annotator. **(Right y-axis)** The number of annotators with at least x annotations (bins). **(Left y-axis)** the average F1-score or accuracy for all user annotations with at least x annotations.

annotations, they are less common—most annotators tend to stop after only a few examples. This distribution of annotators results in overall insufficient annotation quality. Pre-qualification tests are often used to shift this distribution from the "average worker" towards more experienced or dedicated annotators; however, this requires a significantly larger budget and greater micro-management involvement from the researcher.

C.2 Consistency

Usually, when annotating a dataset, more than one annotator is involved. This applies to crowd-workers, experts, and even LLMs— in this study, we use an ensemble of different LLMs and prompts. The use of multiple annotators, similar to an ensemble, is meant to overcome the variance between individuals, which can arise from the subjective nature of NLP tasks, different interpretations of instructions, lack of experience, task difficulty, and cognitive bias (Uma et al., 2021).

As such, a common practice in the NLP community is to report Inter Annotator Agreement (IAA)— a set of statistical measures used to evaluate the agreement between individuals. Typically, IAA can be viewed as an adjustment of the proportion of pairwise agreements, where 0.0 indicates random agreement. We focus on Fleiss's κ (Fleiss, 1971), as it accounts for label imbalance and multiple (> 2) annotators. High IAA, or low variance between independent annotators, is considered an indicator of high-quality annotation. In Table Table 4, we report the agreement between annotators across different approaches. For LLMs, we report two variants: (1) same model, different prompts; and (2) different models, where each model's result is the aggregation across prompts. For reference, we also include the IAA from the original annotations, as reported in the original papers: *MNBM* reported an average Fleiss's κ of 0.696 for the hallucination annotation task; *BEGIN* reported Krippendorff's α (a generalization of Fleiss's κ) of 0.7; *VitaminC* reported Fleiss's κ of 0.7065 on a sample of 2,000 examples; and *PAWS* reported a 94.7% agreement between a single annotator's label and the majority vote on the Wikipedia subset used in TRUE.

Experts While it's true that reconciliation naturally leads to increased agreement, the significant improvement in IAA we observed highlights its importance. Though this phase is less common in practice, it is crucial not only for increasing agreement but also for improving the overall quality of annotations and ensuring more reliable outcomes. Interestingly, label changes in this phase were not symmetric, as most changes (69.3%) were in the direction of *consistent* $\rightarrow$ *inconsistent*, where one annotator found an inconsistency that the other did not (see all change details in Figure 11). It is important to note that the κ obtained by the experts (both before and after reconciliation) was calculated on a more challenging subset, where the original label differed from the LLM prediction, and should be interpreted with this context in mind. This is reflected in the decrease in κ observed for all other annotator groups on this subset.

LLMs GPT-4 and PaLM2, the better-performing LLMs on this task, show high IAA, with $\kappa = 0.706$ and $\kappa = 0.75$, respectively, which is similar to the experts' reported κ. This suggests a comparable level of variance and quality in annotation, providing further empirical evidence for considering LLMs as annotators. This property adds to previous studies showing LLMs' quality as surrogates for human preferences (Zheng et al., 2023) or evaluations (Chiang and Lee, 2023).

Crowd-Sourcing. Crowd workers showed near-random agreement, indicating relatively poor-

[*] Multiple MTurk workers have participated in annotation, yet exactly 3 annotations per example were obtained. Annotators independence assumption was made to calculate Fleiss's κ as with 3 annotators.

[†] These MTurk annotators were chosen with stricter pre-qualification criteria than those in the TRUE dataset and do not correspond to the MTurk line in the TRUE table.

Annotator group	Fleiss's κ	%agreement	#examples	Fleiss's κ (disagree. subset)	#annotators
Experts			222		2
Before reconciliation	0.486	75.7		0.486	
After reconciliation	0.851	93.2		0.851	
MTurk	0.074	60.5	400	-0.004	3*
LLM (different prompts)			640		4
GPT-4	0.706	85.3		0.571	
PaLM2	0.750	87.7		0.696	
LLaMA3	0.219	71.7		0.078	
Mistral	0.459	73.2		0.314	
LLMs (different models)	0.521	77.5	640	0.389	4

Table 4: Inter-Annotator Agreement in different annotator groups. %agreement is the proportion of pairwise annotator comparison. Fleiss's κ (disagree. subset) refers to the κ over the subset of disagreement between LLM and the original label.

Annotator group	Krippendorff's α	%agreement	#annotators
Experts	0.584	60.4	3
MTurk[†]	0.496	65.6	5
LLM (different prompts)			4
GPT-4o	0.760	63.6	
Gemini 1.5 Flash	0.733	79.7	
LLMs (different models)	0.576	62.9	2

Table 5: Inter-Annotator Agreement in different annotator groups on the SummEval benchmark. %agreement is the proportion of pairwise annotator comparisons.

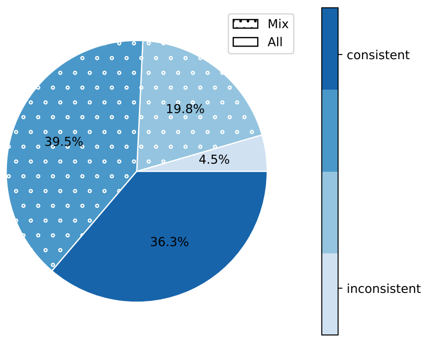


Figure 8: Distribution of crowd-source annotators. Each example was annotated by 3 workers. Plain segments are unanimous annotation, while dotted segments indicate examples where some annotators labeled as *inconsistent*, and other as *consistent*. For example, 19.8% of the examples had two *inconsistent* annotation, and one *consistent* annotation.

quality annotations. Figure 8 describes the distribution of annotations by MTurk workers. Only 40.8% of the examples were labeled unanimously, whereas the rest included annotations from both classes. In addition, if aggregating by majority vote, we get that 75.8% of the examples are labeled as *consistent*, which is far from the original distribution of classes. As mentioned before, even experts may miss a small inconsistency nuance, and finding it requires attention. Even from the subset of examples unanimously labeled as *consistent*, 37.9% have a label of *inconsistent* in both original and gold labels, which points to a lack of attention and thoroughness.

SummEval. Table 5 shows the IAA analysis on the SummEval benchmark. We report Krippendorff's α (Krippendorff, 1970), a generalization of κ to account for ordinal labeling. LLMs exhibit high IAA (compared to experts' IAA) of $\alpha = 0.57$ and 62.9% agreement between models, with high consistency across prompts for the same model. Crowd-workers obtain decent results (maybe due to stricter pre-qualification criteria of 10,000 approved HITs), yet they still fall short compared to experts or LLMs.

C.3 Cost and Scalability

In MTurk platform, a total of $400 \times 3 = 1200$ annotations cost 572$, including 2 small pilot experiments. All annotations were prepared within a few hours. However, it demanded an additional and significant time for review, after which rejected examples returned to the pool. This annotation-review cycle was conducted for ~ 5 iterations. Inference via OpenAI's API on GPT-4 cost $\sim 4.5\$$ per prompt. Inference via VertexAI's API on `PaLM2` cost $\sim 0.15\$$ per prompt. Both took ~ 8 minutes per prompt. Inference on `Mistral` and `Llama3` was via the HuggingFace API, and its cost is estimated by the cost of using a suitable Virtual Machine (VM) on Google Cloud Platform (GCP) for the time of inference (1 minute per model)- $\sim 0.1\$$ per prompt.

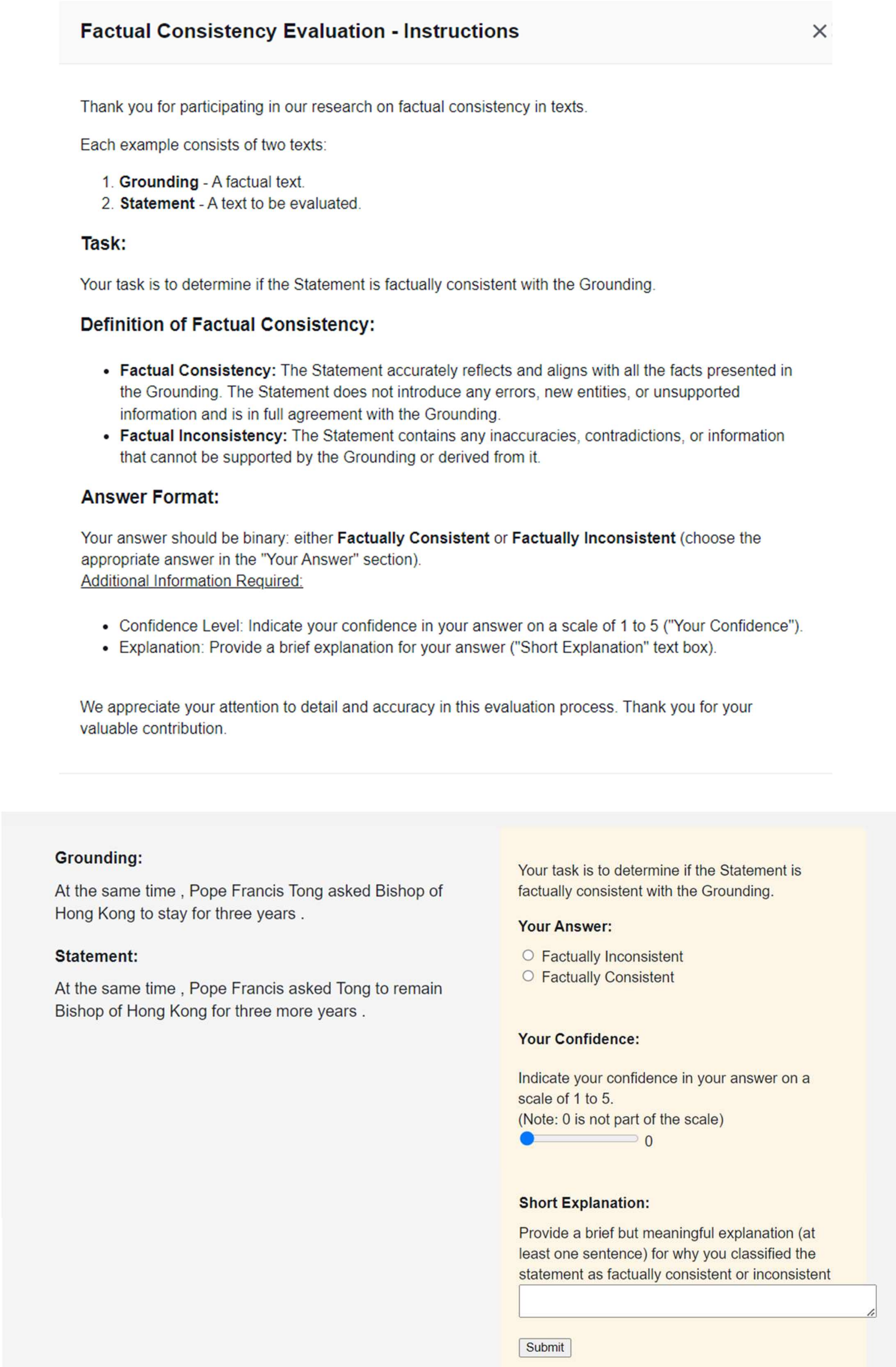
LLM-based annotation is significantly cheaper and faster than crowd-sourcing platforms like MTurk, especially when considering the additional time required for human review cycles. It is estimated to be 100 to 1,000 times more cost-effective than using human annotators, including experts. This scalability and speed make LLMs a highly efficient alternative for large-scale annotation tasks.

D Annotation

D.1 Crowd-source

Each example was annotated by three annotators, who in addition to the binary label were requested to provide their confidence in their answer, and also write a short explanation for why they chose this label. Pre-qualifications included 50+ approved HITs and 97%+ approval rate, which are at standard scale for the MTurk platform (Kazai et al., 2013; Hauser et al., 2021; Chmielewski and Kucker, 2019). Also, locations were limited to [USA, UK, Australia], which are all English-speaker countries. We disabled the possibility of right-click and `Ctrl+c` in the platform (as suggested by (Veselovsky et al., 2023a)), to prevent (as much as possible) the case where generative-AI (e.g., ChatGPT) will be applied to solve the task instead of humans solving it themselves (as shown by (Veselovsky et al., 2023b)). The maximum time allowed per HIT was 6 minutes, while the actual average execution time was 2:20 minutes for all assignments, and 3 minutes for approved assignments. The guidelines provided to annotators and the annotation platform layout are presented in Figure 9.

Each annotation was manually reviewed and was rejected if the answers were not in line with the instructions, or if it was obvious that the task was not done honestly. Overall, this task suffered from a high rejection rate of 49.2% (1163 rejected, 1200 approved). The main rejection reasons were: lack of meaningful explanation, obvious copy-paste annotations across different examples, explanations contradicting the label annotation, and cases where the explanation was a copy-paste of either the grounding or the statement.



Factual Consistency Evaluation - Instructions

Thank you for participating in our research on factual consistency in texts.

Each example consists of two texts:

1. **Grounding** - A factual text.
2. **Statement** - A text to be evaluated.

Task:

Your task is to determine if the Statement is factually consistent with the Grounding.

Definition of Factual Consistency:

- **Factual Consistency:** The Statement accurately reflects and aligns with all the facts presented in the Grounding. The Statement does not introduce any errors, new entities, or unsupported information and is in full agreement with the Grounding.
- **Factual Inconsistency:** The Statement contains any inaccuracies, contradictions, or information that cannot be supported by the Grounding or derived from it.

Answer Format:

Your answer should be binary: either **Factually Consistent** or **Factually Inconsistent** (choose the appropriate answer in the "Your Answer" section).
Additional Information Required:

- Confidence Level: Indicate your confidence in your answer on a scale of 1 to 5 ("Your Confidence").
- Explanation: Provide a brief explanation for your answer ("Short Explanation" text box).

We appreciate your attention to detail and accuracy in this evaluation process. Thank you for your valuable contribution.

Grounding:

At the same time , Pope Francis Tong asked Bishop of Hong Kong to stay for three years .

Statement:

At the same time , Pope Francis asked Tong to remain Bishop of Hong Kong for three more years .

Your task is to determine if the Statement is factually consistent with the Grounding.

Your Answer:

- Factually Inconsistent
- Factually Consistent

Your Confidence:

Indicate your confidence in your answer on a scale of 1 to 5.
(Note: 0 is not part of the scale)
0

Short Explanation:

Provide a brief but meaningful explanation (at least one sentence) for why you classified the statement as factually consistent or inconsistent

Submit

Figure 9: Platform for crowd-sourcing annotation in Amazon Mechanical Turk (MTurk). **(Top)** Guidelines for the task and definitions. **(Bottom)** Annotation layout for a single instance.

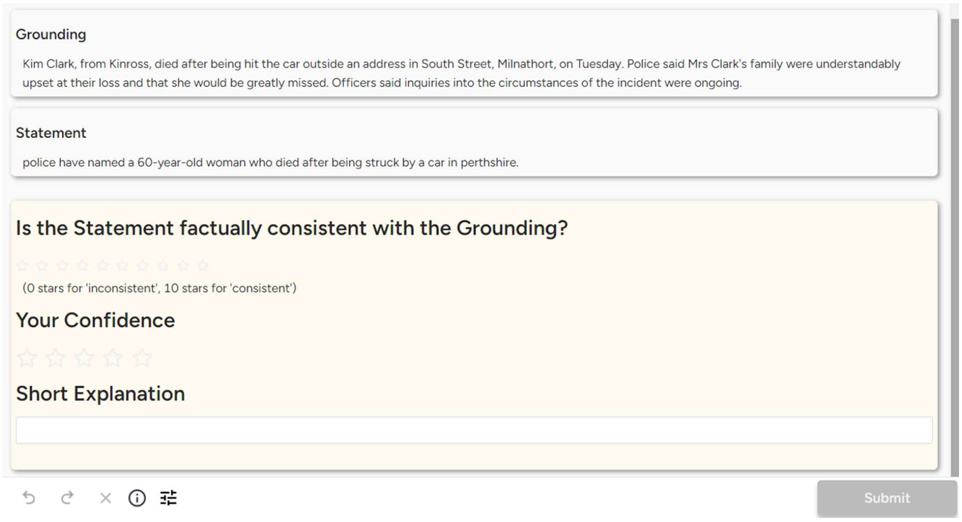


Figure 10: Annotation platform on Label-Studio for experts

D.2 Experts

Experts annotation was using the platform of Label Studio. [1] Layout design is presented in Figure 10. Examples were presented in random order, and neither the LLM prediction nor the original label were presented during the annotation. In the first stage, each example was annotated independently by both experts. Afterward, the human experts began in a second phase of a reconciliation, where a discussion was made over examples they disagreed over. This reconciliation phase ended up with a much higher agreement and higher-quality labels.

In the reconciliation phase, we observed that most changes (69.3%) were from label 1 to label 0, indicating that contradictions might be hard to find, and not all annotators catch them at first. For the full distribution of label change in the reconciliation phase, see Figure 11.

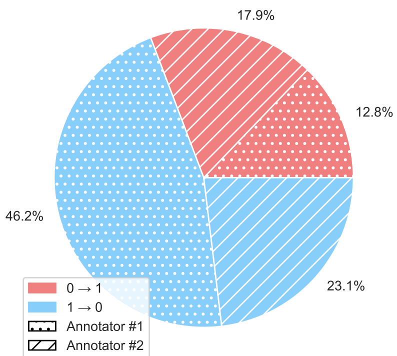


Figure 11: How experts' annotations have changed after the reconciliation phase. Most changes occur from 1 (*consistent*) to 0 (*inconsistent*).

[1] https://labelstud.io/

D.3 LLMs

To annotate a total of $160 \times 4 = 640$ examples from four different datasets, we used four LLMs: GPT-4 (`gpt-4-1106-preview`) (OpenAI, 2023), PaLM2 (`text-bison@002`) (Anil et al., 2023), Mistral (7B)[2] (Jiang et al., 2023) and Llama 3 (8B)[3] (Dubey et al., 2024).

Each model was run with four different prompts (see full prompts in Figure 12). We used a variety of terminology, as this task appears to have different framings in different studies. For example, the premise-hypothesis terminology from classic NLI (MacCartney and Manning, 2009), or document-statement used in (Tam et al., 2023).

For API models (GPT-4, PaLM2), we set `temperature=0.0` and extracted the logit of the generated token (functionality provided by both APIs), if the generated token was either `'0'` or `'1'` as expected. This logit was then transformed into a probability $p_t = P(y = t|x)$ via exponent corresponding the generated token t, and $1 - p_t$ for the other label. To address the case where the first generated token was an unrelated token such as `' '`, `'\n'`, we set `max_tokens=2` and took the first appearance of either `'0'` or `'1'`. For all models, prompts and examples, `'0'` or `'1'` were in the first two generated tokens. Rest of parameters were set according to their default values.

For models available through the HuggingFace API (e.g., Mistral, Llama 3), we can load the model

[2] https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.2

[3] https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct

parameters and make inference locally. In that case, we get access to logits for all tokens, instead of just for the generated ones. Therefore, we applied a similar procedure, where we seek for the first appearance of either `'0'` or `'1'` to be the most probable token to be generated, and then directly extracted the logits of the `'0'` and `'1'` tokens. These logits were transformed into probabilities $(P(y = 0|x), P(y = 1|x))$ via a softmax function.

E Data

For our main experiments, we used the TRUE benchmark for factual consistency. Specifically, we focus on four TRUE datasets, one from each task (summarization, dialogue, fact verification, paraphrasing):

MNBM (Maynez et al., 2020): Summarization. This dataset provides annotations for hallucinations in generated summaries from the XSum dataset (Narayan et al., 2018). *Grounding* refers to the source document that the summary is based on, while *Generated Text* consists of model-generated summaries, which may include hallucinated information not present in the source. Three human annotators, trained for the task through two pilot studies, annotated the dataset for the existence of hallucinations. In TRUE, the binary annotations were determined by majority vote.

BEGIN (Dziri et al., 2022): Dialogue. This dataset evaluates groundedness in knowledge-grounded dialogue systems, where responses are expected to align with an external *Grounding* source, typically a span from Wikipedia. *Generated Text* refers to model-generated dialogue responses that were fine-tuned on datasets like Wizard of Wikipedia (Dinan et al., 2019). Data was annotated into entailment/neutral/contradiction labels, by three human annotators, trained for the task through two pilot studies, aggregated by majority vote. In TRUE, binary annotations were then determined by the entailment/not-entailment partition.

VitaminC (Schuster et al., 2021): Fact Verification. This dataset is based on factual revisions of Wikipedia. The evidence, or *Grounding*, consists of Wikipedia sentences, either before or after these revisions. Most human involvement came from creating *Generated Text* rather than the annotation process, with annotators writing claim/evidence pairs derived from Wikipedia revisions, inherently generating labeled data for fact verification. Synthetic examples from the FEVER dataset (Thorne et al., 2018) were also included. Additionally, three annotators reviewed 2,000 examples, presumably to ensure data quality.

PAWS (Zhang et al., 2019): Paraphrasing. This dataset consists of paraphrase and non-paraphrase pairs. *Grounding* refers to source sentences drawn from Quora and Wikipedia, while *Generated Text* was automatically generated through controlled word swapping and back-translation. Five human annotators annotated the dataset with binary labels w.r.t paraphrasing correctness. The dataset includes both high- and low-agreement annotations.

```
prompt1
Here are two texts:
TEXT 1. <..PREMISE..>.
TEXT 2. <..HYPOTHESIS..>.

Is TEXT 2 contradictory or is it factually inconsistent with TEXT 1? If yes answer 0.
Is TEXT 2 entailed or is it factually consistent with TEXT 1? If yes answer 1.
Refer only to the two texts above, and not any other external knowledge or context.
Answer only 0 or 1
Answer only with one token: 0 or 1

Answer:
```

```
prompt2
DOCUMENT: <..PREMISE..>.

QUESTION: Is the following STATEMENT factually consistent with the above document?

STATEMENT: <..HYPOTHESIS..>.

ANSWER FORMAT: 0 for No, 1 for Yes

Answer only with one token: 0 or 1

Answer:
```

```
prompt3
You are given the two following texts:
TEXT 1. <..PREMISE..>.
TEXT 2. <..HYPOTHESIS..>.

TEXT 1 is a fact. TEXT 2 is a statement. Is TEXT 2 factually consistent with TEXT 1?
Answer 0 for No, 1 for Yes.
Answer only with one token: 0 or 1

Answer:
```

```
prompt4
Given the following texts:
<PREMISE> : <..PREMISE..>.
<HYPOTHESIS> : <..HYPOTHESIS..>.

Please assess the factual consistency of <HYPOTHESIS> with respect to <PREMISE>.
If the content of <HYPOTHESIS> aligns with the information provided in <PREMISE>, assign a label of 1.
If there are factual inconsistencies between <HYPOTHESIS> and <PREMISE>, assign a label of 0.

Target Format: either 0 (for Factual Inconsistency) or 1 (for Factual Consistency).
Answer only with one token: 0 or 1

Answer:
```

Figure 12: Four different prompt input templates to LLMs for obtaining binary labels

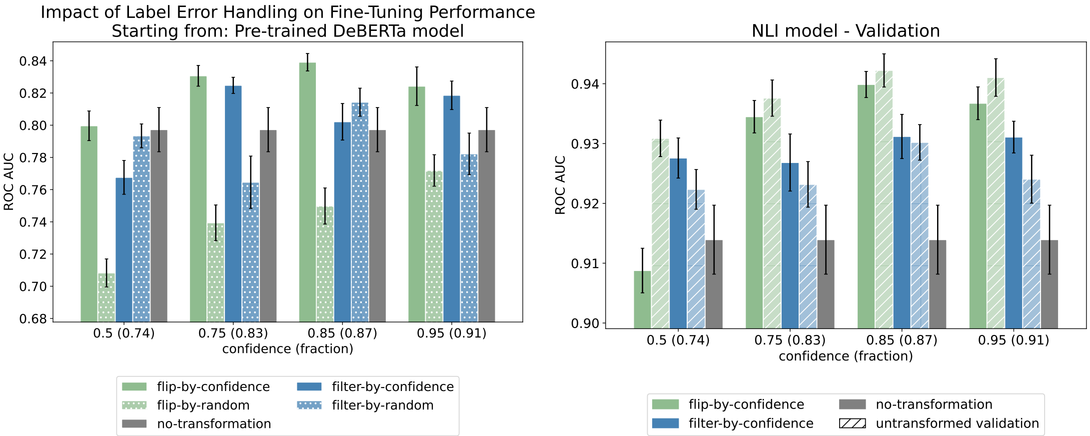


Figure 13: Similar experiments to the one in Figure 5, with small alterations. **(Left)** Starting from a different base model - pre-trained `DeBERTa-v3-base`. **(Right)** Dashed columns present results for when flipping or filtering methods were applied only on the training set, but not the validation.

F Mislabeled Data Implications

F.1 Fine-tuning

Hardware. For the finetuning of DeBERTa models, both the base pre-trained model, and the NLI model which is in the same size, in subsection 7.1, we used 2 Quadro RTX6000 (24GB) GPUs.

Implementation. We finetuned starting from two base models: DeBERTa-v3 [4], and a fine-tuned version of it on classic NLI datasets [5]. We used HuggingFace trainer with early stopping of 4 epochs. The finetuning procedure includes splitting the training set into train and validation sets (where validation size is 25% and train 75%), fine-tuning on the train set, and choosing the best checkpoint based on the validation ROC AUC. We ran all experiments on five different seeds, affecting also the train-validation split and the random set chosen for ablation. We fine-tuned all variants with the same hyperparameters, determined by the best performing on the no-manipulation baseline. This includes 30 epochs at most, batch size of 16, learning rate of `5e-5` and weight-decay of 0.03. The rest were set as the trainer and model default.

Additional Experiments. The left plot in Figure 13 presents the same experiment discussed in subsection 7.1, but starting from the pre-trained `DeBERTa-v3-base`. Same trends applies here, where our LLM-confidence-based manipulations of either flipping or filtering flagged examples outperforms the baselines.

The right plot in Figure 13 compares the performance of these methods (starting from the NLI model) when applied to both the training and validation sets (solid bars) or only the training set (dashed bars). The results are consistent, with no statistically significant differences between the two settings. Importantly, all variations outperform the baseline, underscoring the critical role of a well-curated training set in enhancing the model's ability to generalize effectively.

F.2 Model Evaluation

In subsection 7.2 we evaluated the following models: GPT-4, PaLM2 (`text-bison@002`), Mistral-v0.2 (7B), and Llama3 (8B), which are covered in subsection 4.2; DeBERTa-v3 and NLI-model, which is a fine-tuned version of it on NLI datasets, as discussed in subsection 7.1; and GPT-4o, GPT-4o-mini, Mistral-v0.3,[6] which share the same implementation as GPT-4 or Mistral-v0.2.

G Statistical Analysis

G.1 Clopper-Pearson

As mentioned in subsection 5.1, we employed the Clopper-Pearson exact method (Clopper and Pearson, 1934) to construct a 95% confidence interval for the binomial proportion, adjusted by a finite population correction (FPC). As we only have a subset of examples we re-annotated by LLMs or experts, we can not precisely determine what is the error rate in the full dataset, but only construct a confidence interval based on the re-annotated

[4] `microsoft/deberta-v3-base`
[5] `MoritzLaurer/DeBERTa-v3-base-mnli-fever-anli`

[6] `https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.3`

subset. The Clopper-Pearson method provides an exact confidence interval for a binomial proportion, which means it gives a reliable estimate even with small sample sizes. By applying FPC, we adjust the interval because our sample is drawn from a limited population. This adjustment helps refine the estimate by taking into account the size of the overall dataset compared to the sample.

G.2 Bootstrap sampling

In subsection 5.1, we use bootstrap sampling to provide confidence intervals for each bin. While not necessarily the first to introduce it, (Xia et al., 2012) explored bootstrap confidence intervals on ROC AUC. Unlike the method in Appendix G.1, we do not make claims about the entire dataset, but rather focus on the re-annotated subset we possess. To achieve this, we perform 100 bootstrap samples from the empirical distribution of each bin, sampling with replacement. We then measure the agreement between the experts' resolutions and the LLM annotations, compared to its agreement with the original label.

H Label Errors

Table 6 demonstrates one example per dataset, in which the original label is, in fact, an error, the LLM prediction marked it as a candidate, and the expert annotators determined the correct gold label.

Dataset: VITC

Grounding: The British Government and NHS have set up a Coronavirus isolation facility at Arrowe Park Hospital in The Wirral for British People coming back on a special flight from Wuhan. Evacuation of foreign diplomats and citizens from Wuhan. Due to the effective lockdown of public transport in Wuhan and Hubei province , several countries have started to evacuate their citizens and/or diplomatic staff from the area , primarily through chartered flights of the home nation that have been provided clearance by Chinese authorities.

Generated Text: There is a Coronavirus isolation facility at Arrowe Park Hospital that was set up by the NHS and the British Government

Original Label: 0 **LLM p:** 0.99 **Gold Label:** 1

Explanation: Rephrasing of the first sentence, without any contradiction.

Dataset: BEGIN

Grounding: Hillary Clinton, the nominee of the Democratic Party for president of the United States in 2016, has taken positions on political issues while serving as First Lady of Arkansas (1979–81; 1983–92), First Lady of the United States (1993–2001);

Generated Text: She is the nominee in 2016.

Original Label: 0 **LLM p:** 0.98 **Gold Label:** 1

Explanation: She (Hillary Clinton) is indeed the nominee in 2016 as specifically stated in the grounding.

Dataset: PAWS

Grounding: David was born in Coventry on 21 September 1933 , with his twin Charles and Jessamine Robbins , the eighth and ninth children of twelve by Robbins.

Generated Text: David was born on September 21 , 1933 in Coventry with his twin father Charles and Jessamine Robbins , the eighth and ninth child of twelve of Robbins

Original Label: 1 **LLM p:** 0.04 **Gold Label:** 0

Explanation: The generated text incorrectly states "twin father" instead of "twin" which is not the same, and does not even make much sense in English.

Dataset: MNBM

Grounding: The John Deere tractor was pulled over by officers in the village of Ripley and had two other males on board. The vehicle had been seen in nearby Harrogate at about 05:00 GMT with no headlights on. Police said the driver had no licence, was not insured and did not have permission from the tractor's owner. The vehicle was seized, with the three due to be interviewed by officers. Posting on Twitter, Insp Chris Galley said: "A strange end to a night shift. 15-year-old lad driving a tractor as a taxi for his drunk mates."

Generated Text: a 15-year-old boy has been stopped by police after being seen driving a taxi on a night taxi.

Original Label: 1 **LLM p:** 0.19 **Gold Label:** 0

Explanation: The generated text claims that the 15-year-old boy was "driving a taxi on a night taxi", contradicting the grounding in which it was claimed that the boy was driving a tractor as a taxi

Table 6: Annotation errors in the original datasets, discovered by LLMs and corrected by experts.