Verify with Caution: The Pitfalls of Relying on Imperfect Factuality Metrics

Ameya Godbole and **Robin Jia**
University of Southern California
{ameyagod,robinjia}@usc.edu

Abstract

Improvements in large language models have led to increasing optimism that they can serve as reliable evaluators of natural language generation outputs. In this paper, we challenge this optimism in regards to factuality evaluation. We re-evaluate five state-of-the-art factuality metrics on a collection of 11 datasets for summarization, retrieval-augmented generation, and question answering. We find that these evaluators are inconsistent with each other and often misestimate the factual accuracy of NLG systems, both of which can lead to a variety of pitfalls. We further show that these metrics exhibit biases against highly paraphrased outputs and outputs that draw upon faraway parts of the source documents. We urge users of factuality metrics to proceed with caution and manually validate the reliability of these metrics in their domain of interest.[1]

1 Introduction

Building automated evaluation metrics that match human judgment is difficult, ongoing research (Lambert et al., 2024). Past work has highlighted the flaws of automated evaluators in several NLP research domains, particularly machine translation (Mathur et al., 2020; Kocmi et al., 2021, inter alia). Nonetheless, automated evaluation metrics are perennially appealing because they allow NLG system designers to bypass slower and costlier human evaluation. Most recently, LLM-based metrics have led to optimism that NLG evaluation can be automated (Kim et al., 2024; Vu et al., 2024, inter alia). In particular, there is a growing demand for automated attribution evaluators, as LLMs are increasingly used for tasks in which factual reliability is crucial, such as summarization, retrieval-augmented generation, and open-ended chat (Gao et al., 2023; Chen et al., 2023a).

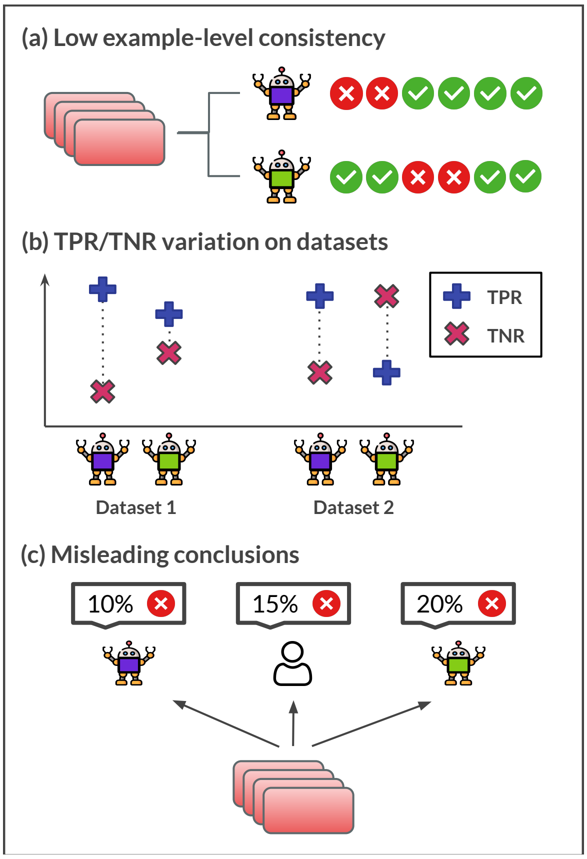
(a) Low example-level consistency
(b) TPR/TNR variation on datasets
TPR
TNR
Dataset 1
Dataset 2
(c) Misleading conclusions
10%
15%
20%

Figure 1: Selecting an AutoAIS evaluator based solely on balanced accuracy (BAcc) hides several underlying inconsistencies. Consider gpt-4-turbo and Bespoke-7B with comparable BAcc on LLM-AGGREFACT. The two evaluators have (a) low instance-level labeling consistency and (b) different true positive and true negative error rate trade-offs. (c) This results in different system-level evaluations when the evaluators are used downstream to evaluate the factuality of NLG systems. In several cases, one evaluator underestimates the human-labeled error rate while the other overestimates it.

In this work, we investigate automated metrics for evaluating "Attribution to Identified Sources" (AutoAIS; Rashkin et al., 2023), i.e., judging whether a source document fully supports a claim. We perform a comprehensive re-evaluation of 5 state-of-the-art AutoAIS evaluators (2 proprietary

[1] Details of our code and data are available at github.com/ameyagodbole/verify-with-caution

Findings of the Association for Computational Linguistics: ACL 2025, pages 22889–22912
July 27 - August 1, 2025 ©2025 Association for Computational Linguistics

and 3 open-source) on the **LLM-AGGREFACT** benchmark (Tang et al., 2024a), a collection of 11 datasets of claim-document pairs that are annotated for attributability.

We find several reasons to be cautious when using AutoAIS evaluators. First, state-of-the-art (SotA) AutoAIS evaluators with comparable leaderboard scores have large differences in predictions. SotA evaluators have low agreement on an instance level (§3.1); error analysis based on different evaluators may yield different conclusions. Evaluators can achieve comparable balanced accuracy by trading off true positive and true negative rates in different ways on different datasets (§3.2); evaluators cannot be relied on without verification on new datasets. Second, evaluators also often give poor estimates of system-level error rate: AutoAIS metrics on some datasets overestimate and on others underestimate how frequently unattributable claims are generated by NLG systems (§3.3). This can lead to misestimation of the scope for improvement on generation tasks (§3.4); new system design ideas (such as new LMs, new decoding algorithms, etc) may be incorrectly cast aside based on imperfect automated metrics. We highlight that current benchmarks may be insufficient to test whether evaluators can correctly rank NLG systems (§3.5).

We identify two biases in the current SotA AutoAIS metrics. In many domains, AutoAIS metrics struggle to detect unattributable claims with a high surface-level similarity with the document (§4.1). We also show that the common practice of chunking long source documents to fit the evaluator's context window inherently limits their performance because certain claims may become unverifiable (§4.2). Both these properties—paraphrasing without directly copying and synthesizing information from different parts of a long input document—are desirable in an NLG system and may be penalized if not appropriately addressed by evaluators.

In § 5, we compare approaches to reduce the bias/discrepancy between the labeled and predicted (estimated) system error rates. Threshold tuning to minimize the absolute bias on a calibration set is a consistent method for achieving low absolute bias. For AutoAIS evaluators that do not have a tunable threshold, posthoc adjustment of the estimated error rate (González et al., 2017) can reduce the absolute estimation bias (with certain caveats).

Finally, in §6, we discuss the impact of these findings on downstream users of the AutoAIS metrics, such as dataset developers and researchers studying how to improve the factuality of NLG systems. Since metrics do not yet transfer consistently to new datasets, we urge users of these metrics to first perform human validation of metric predictions in new data domains and on new systems. We advise developers of new AutoAIS metrics to report a breakdown of metric behavior on the different error types across different bias axes of the evaluation data and with an evaluation of system-level error quantification.

2 Problem Setup

2.1 Notation

Given a claim c and a document d[2], the role of the AutoAIS evaluator $\mathcal{A}$ is to judge whether all the information in c is fully supported by the document d.[3] Following Tang et al. (2024a), we threshold the output of the evaluator at 0.5 and predict a label 0 (unattributable) or 1 (attributable). We will discuss the impact of tuning the threshold for downstream applications in § 5.

$$\mathcal{A}(d, c) \to \{0, 1\}$$

Certain AutoAIS evaluators may have input length limits, in which case the document d is segmented into chunks (of complete sentences) of a certain length $\{d^{(1)}, d^{(2)}, .., d^{(K)}\}$. Then the prediction:

$$\mathcal{A}(d, c) = \max_{k \in [1,K]} \mathcal{A}(d^{(k)}, c) \to \{0, 1\}$$

Our analysis will focus on the validation set of the LLM-AGGREFACT benchmark (Tang et al., 2024b); a collection of 11 datasets with human-annotated attributability annotations. We further split the examples from the RAGTruth dataset (Niu et al., 2024) in the benchmark into the 4 original subsets since they have qualitatively different inputs and task types. This results in a benchmark with 14 datasets.

Except for Wice and FactCheck-GPT, 12 of the 14 datasets contain generations from multiple systems. We use this to analyze the system-level error estimation and ranking of the different AutoAIS evaluators. Appendix A.1 provides a detailed breakdown of the datasets in the benchmark.

The benchmark assumes that each sentence[4] is a

[2] The document may be a composite of multiple evidence passages e.g. LFQA (Chen et al., 2023b).

[3] This is part of the definition of AIS given by Rashkin et al. (2023). Most AutoAIS systems assume decontextualization as a separate preprocessing step.

[4] AggreFact-CNN treats the entire summary (avg of 3.2 sentences) as the claim due to a lack of sentence-level labels.

standalone claim[5]. Except for AggreFact-CNN and AggreFact-XSum, 10 of the 12 datasets (with generations from multiple systems) originally contained multi-sentence responses that have been broken down into sentence-level examples in the benchmark. We evaluate the response-level performance of the AutoAIS evaluators by mapping individual claims back to the original complete response. We obtain a response-level factuality label by aggregating the claim-level labels. We adopt the strict definition of an attributable response (Tang et al., 2024c): a response is attributable if ALL claims in the response are attributable.

2.2 AutoAIS Evaluators

The LLM-AGGREFACT benchmark ranks metrics based on average balanced accuracy (BAcc) across all data sets. BAcc of an evaluator is defined as the average of its True Positive Rate (TPR) and True Negative Rate (TNR) on a dataset, i.e. it measures the average performance of detecting the attributable and unattributable examples.

In this work, we study five evaluators from the LLM-AGGREFACT leaderboard. We choose 2 closed, API-based evaluators: gpt-4-turbo (OpenAI et al., 2024)(in particular, `gpt-4-0125-preview`) and gpt-3.5-turbo (in particular, `gpt-3.5-turbo-0125`), and 3 open-weight models from the MiniCheck series (Tang et al., 2024b): Bespoke-Minicheck-7B (Bespoke-7B), MiniCheck-FlanT5-Large (MiniCheck-FT5) and MiniCheck-RoBERTa-Large (MiniCheck-Rbta). Bespoke-7B and gpt-4-turbo were the top evaluators on the leaderboard at the time of release. Similarly, MiniCheck-FT5, MiniCheck-Rbta, and gpt-3.5-turbo have very similar performances regarding average balanced accuracy across the datasets.

Evaluators with input length constraints (e.g. MiniCheck-FT5, MiniCheck-Rbta, TRUE (Honovich et al., 2022), inter alia.) need to chunk the input documents to fit their max context window. To isolate the effect of chunking, we evaluate the Bespoke-7B metric with chunked documents and compare the predictions against the original predictions without document chunking. In particular, we run the Bespoke-7B metric as if it had a context window of 500 document tokens (same as MiniCheck-FT5). We will refer to this setting as 'Bespoke-7B (cs=500)'.

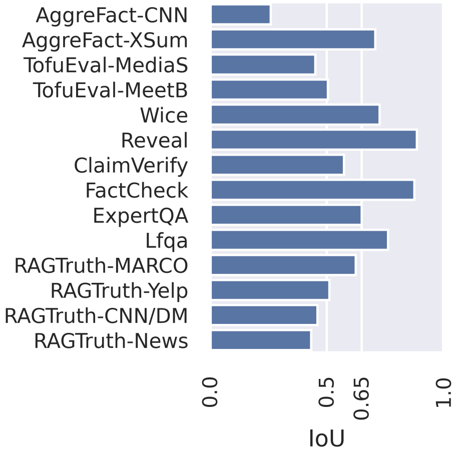

Figure 2: **IoU of "unattributable" predictions by gpt-4-turbo and Bespoke-7B.** Intersection-over-Union less than 50% on 5 of 14 datasets shows that the top-performing models (with very similar balanced accuracy of 76.2% and 77.4% respectively) have low consistency on what examples they predict as "unattributable".

3 Re-Evaluating Factuality Metrics

We begin by comparing the predictions of factuality metrics at the example and system levels. We highlight how inconsistencies in the metrics affect their downstream use for facilitating benchmarks and comparing NLG systems.

3.1 Metrics have low consistency

To study consistency between evaluation metrics, we measure the intersection-over-union (IoU) of the set of examples predicted as "unattributable" by the evaluators. In Figure 2, we see that for the two top-performing evaluators with similar balanced accuracy, Bespoke-7B (Avg BAcc=77.4%) and gpt-4-turbo (Avg BAcc=76.2%), the IoU is less than 50% on 5 of the 14 datasets and less than 65% on 9 of 14 datasets. The consistency is worse on the nine datasets where "unattributable" is the minority class (less than 25% of the dataset). Refer to Appendix A.8 for the pairwise inconsistency of all 5 evaluation metrics studied.

This inconsistency has several implications. When scoring NLG systems, different evaluators may rank NLG systems differently and for different subsets of system predictions. We discuss this further in the next few sections. When conducting error analysis for NLG system development, different evaluators will highlight different "erroneous" unattributable examples. Using a single evaluator may highlight a biased subset of errors. We discuss this further in § 6.2.

[5] Tang et al. (2024b) found that decontextualization and decomposition showed little improvement in the performance of the AutoAIS evaluators.

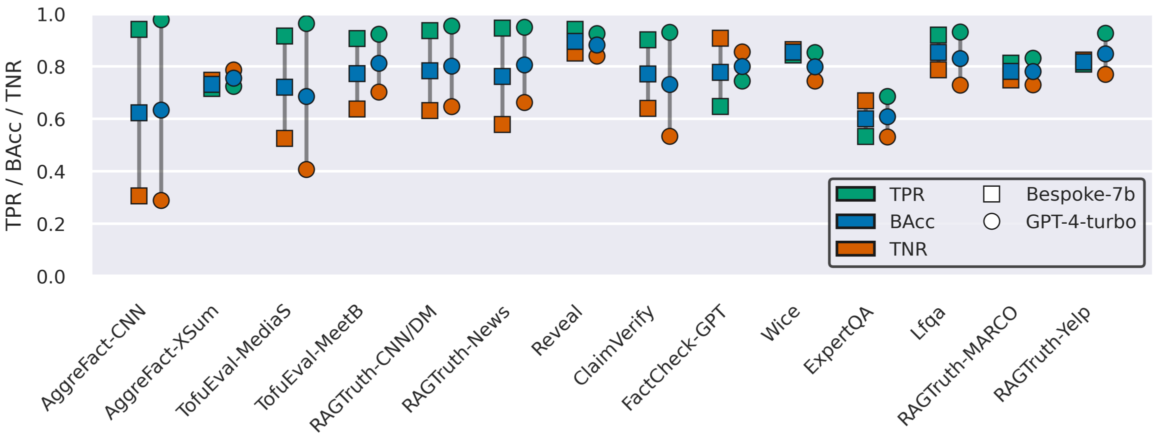

Figure 3: **TPR/TNR/BAcc of evaluators across datasets.** Visualizing the breakdown of BAcc shows that AutoAIS evaluators can have a large gap between TPR and TNR. Moreover, evaluators with the same BAcc can have different TPR and TNR trade-offs. In the extreme case of ExpertQA, GPT-4-turbo has a TPR of 68% and TNR of 53%, while Bespoke-7B has nearly the opposite performance.

3.2 BAcc hides TPR/TNR trade-off

Using balanced accuracy to evaluate AutoAIS metrics hides the underlying trade-off between true-positive and true-negative rates. From Figure 3, we see that the true positive and true negative rates for each evaluator vary widely across the datasets.[6] The gap between TPR and TNR is greater than 20% on 7 of 14 datasets for Bespoke-7B and gpt-4-turbo. By trading off TPR for TNR differently, different evaluators can achieve the same balanced accuracy. For example, on the FactCheck-GPT dataset, Bespoke-7B achieves a BAcc of 77.7% with a difference between TNR and TPR of 26%. gpt-4-turbo achieves a comparable BAcc of 80% but with only an 11% gap between TNR and TPR. Similarly and more surprisingly, Bespoke-7B and gpt-4-turbo achieve the same BAcc on the ExpertQA dataset but with inverted values of TPR and TNR.

The trade-off between TPR and TNR has different implications for downstream users of the metric, where the cost of type I and type II errors differs. We recommend that metric designers report a breakdown of error rates for informed model selection.[7] Similarly, for the metric developers, the breakdown highlights that TNR lags behind TPR by more than 10% on 9 of 14 datasets; improving the ability of metrics to detect unattributable claims is a challenge.

3.3 AutoAIS metrics incorrectly estimate the system error rate

Since the goal of AutoAIS evaluation metrics is to compare NLG systems, we study how accurate the automated metrics are in estimating the true (human-labeled) hallucination rate of the NLG systems. For 12 datasets that contain generations from different systems, we group claims based on the system (S). For each system S, we report the *bias* (González et al., 2017) of the AutoAIS metrics, which is the difference between the labeled error rate (percentage of claims labeled as unattributable) and the predicted error rate (percentage of claims predicted as “unattributable” by the AutoAIS metric). Additionally, on 10 of the 14 datasets where the claims are part of a longer response, we compute a response-level *bias* as the difference between the response-level ground-truth error rate and the response-level predicted error rate.

In Figure 4, we highlight the bias of the metric on TofuEval-MediaSum and TofuEval-MeetingBank. From the claim-level error rates, we see that some metrics under-estimate the error rates of all the systems (gpt-4-turbo, gpt-3.5-turbo, and Bespoke-7B) while others over-estimate the error rate (MiniCheck-FT5 and MiniCheck-Rbta). All 5 metrics have a balanced accuracy of 68-72% on TofuEval-MediaS. Claim-level misclassification and inconsistencies compound when we compute response-level quantification error. On TofuEval-MediaS, the response-level biases (-29.8% at worst for gpt-4-turbo) are about twice the claim-level biases (-12.9% at worst for gpt-4-turbo). Similarly, in Figure 7 (Appendix A.5), we see that the metrics

[6] We report the false positive and false negative rates of the larger set of evaluation metrics in Tables 4 and 5.

[7] Tang et al. (2024c) provides a similar argument in favor of reporting error breakdown.

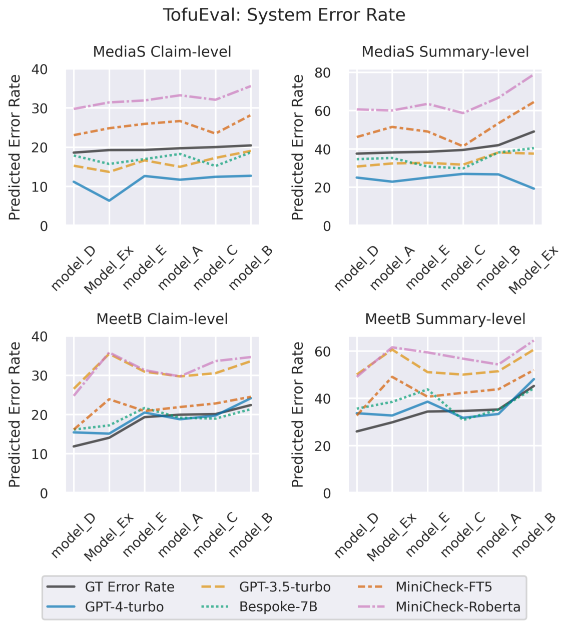

Figure 4: **Predicted system-level error rate on TofuEval.** Imperfect evaluators lead to differences in the ground truth and predicted error rate for different NLG systems. Claim-level misclassification leads to even greater quantification discrepancies in the summary-level attribution error rate.

consistently overestimate the system error rates on the RAGTruth dataset. Moreover, the magnitude of quantification error varies widely across 4 subsets of RAGTruth. We report the claim-level and response-level bias of the AutoAIS metric on the 12 datasets in Appendix A.4.

Thus, the metrics sometimes overestimate and sometimes underestimate the error rate of the systems on different datasets. **This means that we can't know beforehand if a metric will assign reliable system-level scores on a new dataset.**

3.4 Finding 4: Misleading conclusions about headroom

Benchmarks are useful for development if there is room for improvement with future systems. If we want to replace human evaluation with automated metrics on new benchmarks, then the metrics must provide a reliable estimate of this "headroom". From Figure 4 and Table 11, we see that gpt-4-turbo underestimates the headroom on TofuEval-MediaS by 12.3% while MiniCheck-Rbta overestimates the headroom by 11.2% despite both metrics having the same BAcc on the dataset. At the response level, this headroom estimation error grows in magnitude to -18.3% for gpt-4-turbo and +21.2% for MiniCheck-Rbta.

Further, from Table 27, we see that the headroom estimation is worse on smaller systems (7B params) than on larger systems (gpt-3.5-turbo and gpt-4). For example, on RAGTruth-News, the gpt-4-turbo evaluator misestimates headroom on the small systems by +7.3% and on the large systems by +0.8%. Thus, evaluators may unfairly score generations from smaller models leading to an inflated headroom. **When creating a new benchmark, the evaluator must be validated to ensure that it correctly reflects the scope for improvement.**

3.5 Finding 5: Misleading system rankings

The most important reason for using automated metrics is that they enable fast comparison of systems. A reliable metric ranks systems in the same order as the ranking determined by human labeling. Following Mathur et al. (2020), we identify which pairs of systems have statistically indistinguishable/distinguishable error rates. We then compare whether the AutoAIS evaluator correctly predicts significant /insignificant differences between the system pairs. On 8 of 14 datasets (see Tables 12,15,22,25,28) with generations of at least 6 systems, gpt-4-turbo orders 26% system pairs incorrectly on average while Bespoke-7B orders 20% of pairs [8] incorrectly on average. The limited number of examples per system in the dataset and the choice of systems with similar capabilities result in most of the system pairs being indistinguishable.

Further, unlike machine translation, datasets for factuality evaluation do not contain generations from a large number of systems; the datasets in LLM-AGGREFACT contain generations from at most 6 NLG systems, while Mathur et al. (2020) compare between 10-16 machine translation systems. This makes correlation analysis unreliable. For completeness, we report Kendall's τ and Pearson ρ correlation coefficients in Appendix A.3. **Future AIS benchmarks should annotate more samples on the same input prompt (for statistically significant system differences) from more systems (for less noise in ranking correlation).**

[8] An ordering is incorrect if an indistinguishable system pair is predicted to have significant difference and vice versa. For the pairs where system A is significantly better than system B according to the annotated labels, none of the AutoAIS metrics predict a ranking in the opposite direction (system B is significantly better than system A).

4 Analysis of metric biases

We identify two concerning biases that may affect evaluator predictions: (1) dependence on surface-level matches and (2) constraints due to context-window limitations. These biases may cause the evaluators to penalize desirable system outputs.

4.1 Bias towards surface-level similarity

Evaluators heavily rely on surface-level matches between the claim and the document when making predictions. We demonstrate this by studying the behavior of the AutoAIS evaluators as the similarity between the claim and the document varies.

We measure similarity with ROUGE-2 precision (Lin, 2004); this measures the fraction of claim bigrams that appear in the document. Following Vu et al. (2024), we partition the examples into 5 groups based on the task in the source dataset: (1) summarization tasks ('AggreFact-CNN', 'AggreFact-XSum', 'TofuEval-MediaS', 'TofuEval-MeetB', 'RAGTruth-CNN/DM', 'RAGTruth-News'), (2) LLM response verification ('Reveal', 'ClaimVerify', 'FactCheck-GPT'), (3) Wikipedia verification ('Wice'), (4) Long-form QA ('ExpertQA', 'Lfqa', 'RAGTruth-MARCO'), and (5) Data2Text ('RAGTruth-Yelp'). Within each task group, we group examples into 5 bins based on percentiles of ROUGE-2 precision.

From Figure 5, we see that the evaluators mislabel unattributable examples (have low TNR) on the high ROUGE examples. This trend is especially strong in the summarization and long-form QA groups, where the evaluators can detect unattributable claims with high ROUGE only half the time. This highlights that evaluators may struggle to correctly identify small inconsistencies in otherwise heavily copied text. Simultaneously, all evaluators have a trend of a low true positive rate on low ROUGE attributable claims; AutoAIS evaluators penalize heavily paraphrased responses. This is a concern as the evaluators may penalize modern NLG systems for desirable behaviors such as avoiding verbatim copying and drawing valid conclusions. **Overall, the trends demonstrate that word overlap may be a significant component of the metric behavior.**

4.2 Bias from context-size limitations

AutoAIS evaluators with short context windows struggle when the claim connects different document parts. When using AutoAIS evaluators, it is assumed that either (1) the metric has a sufficiently long context window to fit the document and the claim or (2) the metric chunks the document so as to fit it in the input length limit. As NLG systems improve at processing long documents and manipulating facts spread across a source document, it becomes more important for evaluation metrics to handle long evidence documents consistently.

To isolate the effect of chunking on AutoAIS predictions, we study how the predictions of the Bespoke-7B evaluator change when it is forced to evaluate factuality of a claim given document chunks instead of the full document[9]. We set the size of the chunks to 500 tokens (matching the chunk length used by MiniCheck-FT5). For this analysis, we drop examples where the document size was less than 500 words. From Table 30, we see that Bespoke-7B with chunked documents obtains a lower TPR and a higher TNR than the evaluator without chunking. This trade-off can be explained by the decrease in the fraction of examples predicted as "attributable"; Bespoke-7B with chunked documents predicts the "attributable" label 6% less frequently on average than the Bespoke-7B evaluator with the full input.

To identify the examples where the evaluator predictions are most likely to be affected by chunking, we compute a score for every example that measures whether chunking reduces surface-level matches between the document chunk and the claim. In particular, borrowing notation from § 2.1,

$$\texttt{R2-diff} = \texttt{ROUGE-2}_{\texttt{prec}}(d, c) - \max_{k \in [1,K]} \texttt{ROUGE-2}_{\texttt{prec}}(d^{(k)}, c)$$

where $\texttt{ROUGE-2}_{\texttt{prec}}$ is the fraction of claim (c) bigrams that appear in the document (d). We expect examples with a nonzero value of `R2-diff` reference words that do not all appear in one chunk, and thus, the claim is less likely to be verifiable on any single chunk. When the claim becomes unverifiable due to chunking, we expect the evaluator with chunked inputs to predict the label 'unattributable' (0) more often than the evaluator with the full input.

In Figure 6, we plot how the original Bespoke-7B metric predictions change when chunking is enforced. We see that when `R2-diff` is positive (> 0), there is a marked increase in the predictions

[9] Note that the Bespoke-7B evaluator has a sufficient max input length to process all the examples in LLM-AGGREFACT without needing to chunk source documents.

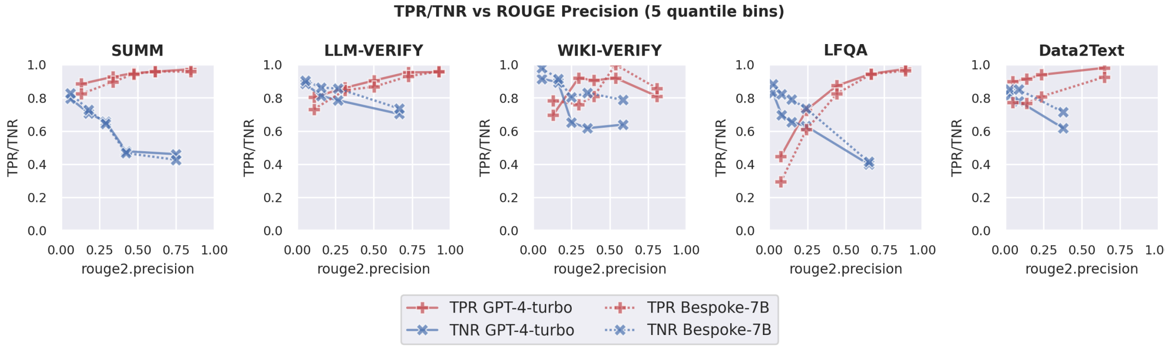

Figure 5: **TPR/TNR vs ROUGE-2 precision of AutoAIS evaluators:** ROUGE-2-precision (fraction of claim bigrams that appear in the document) is (anti-)correlated with true (negative)positive rate, i.e., metrics mislabel attributable generations with low ROUGE-2 precision and unattributable generations with high ROUGE-2 precision.

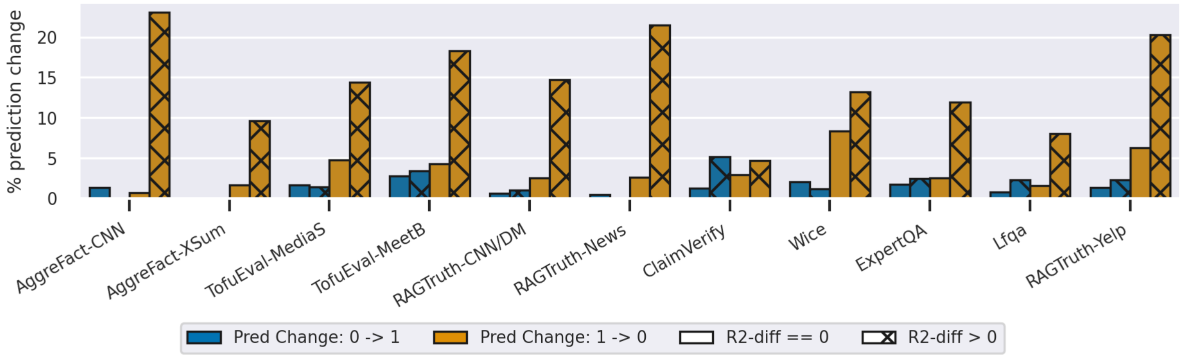

Figure 6: **`R2-diff` vs rate of change in prediction with chunking:** The figure shows the change in predictions of the Bespoke-7B evaluator to the same evaluator with documents chunked to 500 tokens. When chunking causes the overlap between the claim and the document to decrease (`R2-diff` > 0), the evaluator with chunking predicts the label ‘0’ (unattributable) more frequently than the evaluator without chunking.

of the label ‘unattributable’ (0). The rate is greater than 10% on 8 of 11 datasets. The opposite change, a prediction ‘0’ with full context changing to label ‘1’ with chunking, is consistently less than 5%. On the other hand, when `R2-diff` $==0$, the rate of change in prediction is less than 10% on 9 of 11 datasets. This could be attributed to noise in the metric predictions. **Thus, evaluators that chunk their inputs are inherently disadvantaged when verifying attributable claims that reference distant parts of the input document.**

5 Metric Adjustment

As discussed in § 3.3, the AutoAIS evaluators have a high bias in estimating the true error rate of NLG systems. We experiment with methods to reduce this bias and make the metrics more reliable in downstream applications. We assume a scenario where some human-labeled claim document pairs are available for calibration[10]. For these experiments, we use the predictions and scores assigned by the Bespoke-7B evaluator on examples from the RAGTruth datasets. We study three methods for reducing quantification bias (details in Appendix A.7): (1) post-hoc adjustment (Forman, 2006) that changes the predicted error rate based on the known TPR and FPR of the evaluator, (2) threshold tuning to minimize the absolute bias, and (3) threshold tuning to maximize BAcc.

Table 1 reports the results of different methods for adjusting the predicted system error rate. We perform adjustment by using examples from one system for tuning the threshold / computing TPR and FPR, and then computing the mean/worst absolute bias (magnitude) over all the remaining 5 systems. We report both the cross-validated mean and worst-case absolute bias.**We find that tuning to minimize the absolute bias on available annotated data consistently reduces the bias on held-out systems on all four subsets of the RAGTruth dataset.** However, tuning to maximize BAcc leads to an increase in mean and worst-case bias.

[10] This is a reasonable assumption when the metric is used to organize a new benchmark.

Model for Calibration	Source	No Adjustment	Adjusted Counts	Thres. tuning for zero bias	Thres. tuning for ↑BAcc
Cross-Validated	CNN/DM	3.9 (6.1)	15.3 (21.5)	1.9 (4.0)	14.8 (19.6)
	MARCO	14.4 (22.3)	7.2 (12.1)	3.4 (6.7)	20.2 (27.9)
	Recent News	3.0 (5.9)	10.5 (18.8)	2.3 (5.0)	10.2 (18.8)
	Yelp	15.4 (29.7)	26.9 (43.7)	6.0 (13.1)	19.3 (31.2)
`gpt-3.5-turbo-0613`	CNN/DM	4.5 (6.1)	65.6 (86.5)	2.3 (4.7)	37.0 (42.9)
	Recent News	3.3 (6.0)	11.2 (19.3)	1.7 (3.4)	7.7 (15.4)
	MARCO	16.0 (22.6)	22.9 (32.6)	3.7 (7.2)	30.0 (36.1)
	Yelp	17.6 (31.2)	52.8 (80.5)	6.7 (16.2)	32.9 (46.7)

Table 1: **Comparison of adjustment methods on RAGTruth:** We report the bias in estimating the ground-truth system error (hallucination) rates using three adjustment methods. In the upper section, we report cross-validated mean absolute bias by using one system for calibration and calculating the mean absolute bias over the remaining systems. Numbers in parentheses indicate the cross-validated worst-case bias. Green cells indicate a decrease in bias relative to no adjustment. Tuning the evaluator threshold consistently reduces the bias in estimation over the held-out systems. In the lower section, we report the mean absolute bias using the gpt-3.5-turbo model for calibration (this is the model with the least ground-truth error rate). See Tab 31 for the full table.

The "adjusted counts" approach is appealing to use if the AutoAIS evaluator does not provide a scalar score and directly returns a label. The method shows an inconsistent reduction in the absolute bias. In particular, we find that using the best-performing system (gpt-3.5-turbo for RAGTruth datasets) based on labeled error rate leads to poor estimation of the TPR and FPR of the evaluator. This is due to the low prevalence of the label ‘0’ in examples of the best system. Simply adjusting counts based on these bad estimates leads to high bias on the remaining systems (see Table 31 for the full table). When using the adjusted counts approach, we **advise against** using the system with the lowest error rate for calibration.

6 Discussion

6.1 For AutoAIS Metric Developers

Based on our findings in § 3, we urge developers of AutoAIS evaluators to study and compare new approaches holistically. Our findings show that balanced accuracy can hide differences in the underlying behavior of different evaluators. (1) We advise that evaluator performance should be judged on the breakdown of true positive and negative rate (among evaluators with comparable balanced accuracy). AutoAIS metrics should be evaluated on the stability of TPR/TNR across datasets. (2) Quantification bias between predicted and ground-truth unattributable generation rate at the dataset and system levels should be reported. (3) Evaluators should report the rank correlation of NLG systems on the underlying dataset if available. These qualities establish how readily the evaluator can be applied to new domains and be used as a reliable stand-in for human annotations (though metric predictions should still be validated in new domains).

Since chunking long documents can make attributable claims unverifiable, when possible, emphasis should be placed on developing metrics that can process the entire evidence document without chunking. However, use cases such as Wan et al. (2024) require judgment against long reference documents, and chunking becomes necessary. Thus, there is scope and reason to improve the ability of evaluators to correctly handle document chunking.

6.2 For Benchmark Developers

When benchmark curators use automated metrics for evaluation, it is necessary to validate the evaluators’ performance against a human-annotated dataset. Based on the biases (§ 4) and findings (§ 3), we encourage benchmark curators to:

1. Study evaluator behavior by strategically sampling examples from different buckets of the ROUGE precision distribution
2. Validate the choice of using an evaluator that requires input document chunking by testing metric behavior on claims that require long-document reasoning. We highlight `R2-diff` as an easy way to identify these claims.
3. Validate the quantification bias of the evaluator on the human-annotated set. This allows for a better estimation of the actual headroom for improvement on the task.
4. Validate the ranking and quantification bias on predictions from different NLG systems on the benchmark. Threshold tuning can be applied to reduce the bias at the system level.

6.3 For Hallucination Mitigation Research

Based on our findings regarding error quantification bias at the system level, researchers working on hallucination mitigation should not use the absolute error rates predicted by AutoAIS evaluators as the sole support for their research findings. Claims such as “system A hallucinates less than system B" need to be paired with a validation of the evaluator predictions on claims from both systems. The quantification bias also highlights that automated evaluators alone are not an indicator of whether a dataset/task is solved/unsolved. Automated evaluators may under- or over-predict the system error rates. These issues necessitate manual inspection of the evaluator’s predictions to back claims based on automated metrics.

7 Related Work

Meta-Analysis of Automated Evaluation. Nimah et al. (2023) suggest that NLG evaluator (fluency, coherence, consistency, relevance, etc) research should move beyond just measuring the correlation between human preferences and evaluator scores. They study the reliability of evaluators under domain shift and consistency with system rankings. Similar meta-analyses beyond correlation have been studied extensively in machine translation (Mathur et al., 2020; Kocmi et al., 2021). Sai et al. (2021) extend the checklist framework (Ribeiro et al., 2020) to define consistency tests for NLG evaluators. In our work, we find that AutoAIS evaluators are not yet reliable in certain downstream uses out-of-the-box and push for a holistic set of metrics for comparing evaluators.

Meta-Analysis of AutoAIS Evaluators. Similar to LLM-AGGREFACT, AttributionBench (Li et al., 2024) also aggregates datasets into an attribution evaluation benchmark. Error analysis by Yue et al. (2023); Li et al. (2024) also highlights the inability of AutoAIS evaluators in judging nuanced claims. Corroborating our findings about evaluator biases, concurrent work by Ramprasad and Wallace (2024) finds evidence that evaluators may be relying heavily on surface-level syntactic features. They find that evaluators can be “gamed” by making meaning-preserving edits to the claims. Similarly, Opitz (2024) argue that threshold-free metrics such as AUC are insufficient for metric evaluation.

Limitations

Our analysis assumes that the datasets underlying LLM-AGGREFACT have highly accurate human annotations with little ambiguity. There is a potential confounder in our analysis that the human annotations may not be accurate or have significant room for ambiguity(Krishna et al., 2023; Subbiah et al., 2024; Li et al., 2024). In particular, Li et al. (2024) highlight imbalances in the information accessible to humans vs AutoAIS evaluators as a major source of error in evaluator predictions. We leave the reevaluation of this confounder for future work. We believe that a strong metric can be used in-the-loop to identify examples where the metric disagrees with the human label. These disagreements can help narrow down the set of examples with potentially ambiguous labels.

Our analysis is limited to the verification of claims against a single document. Complex claim verification might require multi-document verification (Chen et al., 2024), which is currently out of the scope of this work.

Our analysis of system-level ranking is limited by the number of systems in the underlying dataset. In order to evaluate metrics on the consistency of system-level ranking, we need to collect responses from multiple, diverse NLG systems on a set of generation tasks and collect annotations of attributability. In prior work, the availability of predictions from multiple machine translation systems on a common evaluation set has allowed the machine translation community to study the reliability of automated metrics in ranking (Mathur et al., 2020).

Metrics make inconsistent misestimations on system-level factual accuracy. We do not propose any methods to fix these inconsistencies. A metric with perfect prediction accuracy will automatically solve the problem; however, the community needs a way to make reliable claims based on imperfect metrics in the interim.

Acknowledgements

This work was supported in part by a gift from the USC-Amazon Center on Secure and Trusted Machine Learning, as well as by the National Science Foundation under Grant No. IIS-2403436. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. Feedback from Johnny Tian-Zheng Wei, Liyan Tang, and Gustavo Adolpho Lucas de Carvalho helped shape the project.

References

Anthony Chen, Panupong Pasupat, Sameer Singh, Hongrae Lee, and Kelvin Guu. 2023a. Purr: Efficiently editing language model hallucinations by denoising language model corruptions. *Preprint*, arXiv:2305.14908.

Hung-Ting Chen, Fangyuan Xu, Shane Arora, and Eunsol Choi. 2023b. Understanding retrieval augmentation for long-form question answering. *Preprint*, arXiv:2310.12150.

Jifan Chen, Grace Kim, Aniruddh Sriram, Greg Durrett, and Eunsol Choi. 2024. Complex claim verification with evidence retrieved in the wild. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 3569–3587, Mexico City, Mexico. Association for Computational Linguistics.

George Forman. 2006. Quantifying trends accurately despite classifier error and class imbalance. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '06, page 157–166, New York, NY, USA. Association for Computing Machinery.

Luyu Gao, Zhuyun Dai, Panupong Pasupat, Anthony Chen, Arun Tejasvi Chaganty, Yicheng Fan, Vincent Zhao, Ni Lao, Hongrae Lee, Da-Cheng Juan, and Kelvin Guu. 2023. RARR: Researching and revising what language models say, using language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16477–16508, Toronto, Canada. Association for Computational Linguistics.

Pablo González, Alberto Castaño, Nitesh V. Chawla, and Juan José Del Coz. 2017. A review on quantification learning. *ACM Comput. Surv.*, 50(5).

Or Honovich, Roee Aharoni, Jonathan Herzig, Hagai Taitelbaum, Doron Kukliansy, Vered Cohen, Thomas Scialom, Idan Szpektor, Avinatan Hassidim, and Yossi Matias. 2022. TRUE: Re-evaluating factual consistency evaluation. In *Proceedings of the Second DialDoc Workshop on Document-grounded Dialogue and Conversational Question Answering*, pages 161–175, Dublin, Ireland. Association for Computational Linguistics.

Seungone Kim, Juyoung Suk, Shayne Longpre, Bill Yuchen Lin, Jamin Shin, Sean Welleck, Graham Neubig, Moontae Lee, Kyungjae Lee, and Minjoon Seo. 2024. Prometheus 2: An open source language model specialized in evaluating other language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 4334–4353, Miami, Florida, USA. Association for Computational Linguistics.

Tom Kocmi, Christian Federmann, Roman Grundkiewicz, Marcin Junczys-Dowmunt, Hitokazu Matsushita, and Arul Menezes. 2021. To ship or not to ship: An extensive evaluation of automatic metrics for machine translation. In *Proceedings of the Sixth Conference on Machine Translation*, pages 478–494, Online. Association for Computational Linguistics.

Kalpesh Krishna, Erin Bransom, Bailey Kuehl, Mohit Iyyer, Pradeep Dasigi, Arman Cohan, and Kyle Lo. 2023. LongEval: Guidelines for human evaluation of faithfulness in long-form summarization. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1650–1669, Dubrovnik, Croatia. Association for Computational Linguistics.

Nathan Lambert, Valentina Pyatkin, Jacob Morrison, LJ Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, Noah A. Smith, and Hannaneh Hajishirzi. 2024. Rewardbench: Evaluating reward models for language modeling. *Preprint*, arXiv:2403.13787.

Yifei Li, Xiang Yue, Zeyi Liao, and Huan Sun. 2024. AttributionBench: How hard is automatic attribution evaluation? In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 14919–14935, Bangkok, Thailand. Association for Computational Linguistics.

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

Nitika Mathur, Timothy Baldwin, and Trevor Cohn. 2020. Tangled up in BLEU: Reevaluating the evaluation of automatic machine translation evaluation metrics. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4984–4997, Online. Association for Computational Linguistics.

Iftitahu Nimah, Meng Fang, Vlado Menkovski, and Mykola Pechenizkiy. 2023. NLG evaluation metrics beyond correlation analysis: An empirical metric preference checklist. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1240–1266, Toronto, Canada. Association for Computational Linguistics.

Cheng Niu, Yuanhao Wu, Juno Zhu, Siliang Xu, KaShun Shum, Randy Zhong, Juntong Song, and Tong Zhang. 2024. RAGTruth: A hallucination corpus for developing trustworthy retrieval-augmented language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10862–10878, Bangkok, Thailand. Association for Computational Linguistics.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Juri Opitz. 2024. Schroedinger‘s threshold: When the AUC doesn‘t predict accuracy. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 14400–14406, Torino, Italia. ELRA and ICCL.

Sanjana Ramprasad and Byron C. Wallace. 2024. Do automatic factuality metrics measure factuality? a critical evaluation. *Preprint*, arXiv:2411.16638.

Hannah Rashkin, Vitaly Nikolaev, Matthew Lamm, Lora Aroyo, Michael Collins, Dipanjan Das, Slav Petrov, Gaurav Singh Tomar, Iulia Turc, and David Reitter. 2023. Measuring attribution in natural language generation models. *Computational Linguistics*, 49(4):777–840.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4902–4912, Online. Association for Computational Linguistics.

Ananya B. Sai, Tanay Dixit, Dev Yashpal Sheth, Sreyas Mohan, and Mitesh M. Khapra. 2021. Perturbation CheckLists for evaluating NLG evaluation metrics. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7219–7234, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Melanie Subbiah, Faisal Ladhak, Akankshya Mishra, Griffin Thomas Adams, Lydia Chilton, and Kathleen McKeown. 2024. STORYSUMM: Evaluating faithfulness in story summarization. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 9988–10005, Miami, Florida, USA. Association for Computational Linguistics.

Liyan Tang, Philippe Laban, and Greg Durrett. 2024a. Minicheck: Efficient fact-checking of llms on grounding documents. *Preprint*, arXiv:2404.10774.

Liyan Tang, Philippe Laban, and Greg Durrett. 2024b. MiniCheck: Efficient fact-checking of LLMs on grounding documents. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8818–8847, Miami, Florida, USA. Association for Computational Linguistics.

Liyan Tang, Igor Shalyminov, Amy Wong, Jon Burnsky, Jake Vincent, Yu'an Yang, Siffi Singh, Song Feng, Hwanjun Song, Hang Su, Lijia Sun, Yi Zhang, Saab Mansour, and Kathleen McKeown. 2024c. TofuEval: Evaluating hallucinations of LLMs on topic-focused dialogue summarization. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4455–4480, Mexico City, Mexico. Association for Computational Linguistics.

Tu Vu, Kalpesh Krishna, Salaheddin Alzubi, Chris Tar, Manaal Faruqui, and Yun-Hsuan Sung. 2024. Foundational autoraters: Taming large language models for better automatic evaluation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17086–17105, Miami, Florida, USA. Association for Computational Linguistics.

David Wan, Jesse Vig, Mohit Bansal, and Shafiq Joty. 2024. On positional bias of faithfulness for long-form summarization. *Preprint*, arXiv:2410.23609.

Xiang Yue, Boshi Wang, Ziru Chen, Kai Zhang, Yu Su, and Huan Sun. 2023. Automatic evaluation of attribution by large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 4615–4635, Singapore. Association for Computational Linguistics.

A Appendix

List of Tables

A.1 LLM-AGGREFACT Dataset Details

Table 2 provides details of the 14 sub-datasets in LLM-AGGREFACT. Unlike Tang et al. (2024b), we keep the 4 subsets of RAGTruth

(Niu et al., 2024) separate to highlight the underlying differences. The subsets are RAGTruth-CNN/DM, RAGTruth-Recent_News (referred to as RAGTruth-News to save space), RAGTruth-MARCO, and RAGTruth-Yelp. We approximately follow Vu et al. (2024) in the definition of the task groups. We mark the datasets where the claims are sourced from a longer response.

A.2 Recomputed Metric Performance

Since we sub-divide RAGTruth into its component datasets, we report the recomputed balanced accuracy (BAcc) of the top AutoAIS evaluators in Table 3. We report the breakdown by FPR in Table 4 and FNR in Table 5.

A.3 Evaluator Ranking Performance

On 6 of 14 datasets with generations from at least 6 systems and where the ground truth error rates aren't very close, we further measure the Kendall's τ rank correlation between the predicted ranking of systems by the evaluator and the human-labeled ranking (computed from the system error rates). From Table 6, we see that there is at least one rank inversion in the ranking produced by the top metrics. Bespoke-7B evaluator has up to 4 rank inversions (in ranking 6 systems) on two datasets. We see similar trends in rank correlation when labels are aggregated at the summary level (see Table 7).

However, to make the correlation coefficient useful, a benchmark needs to be built with a larger number of systems with a wide range of ground truth error rates. The machine translation research community (Mathur et al., 2020) has built such resources by running annual shared tasks. Thus, for our main analysis, we count the number of ranking errors where an insignificant ground truth difference between systems becomes significant with automated evaluators and vice versa.

A.4 Evaluator Quantification Bias

In Tables 8-29, we report the system-level predicted error rate and quantification bias (claim and response level), and system-level ranking errors for the AutoAIS metrics on the 14 LLM-AGGREFACT datasets.

A.5 Visualization of System-level Quantification Bias on RAGTruth

In Figure 7, we highlight the bias of the metrics in predicting the claim-level error rate on the RAGTruth dataset. We see that the bias of the top AutoAIS metrics is consistently poor on the MS-MARCO subset, especially on the systems with a higher ground-truth hallucination rate (e.g. the bias is 15-20% for Bespoke-7B). On the Yelp subset, we see that all metrics besides gpt-4-turbo show poor ground truth error estimation; the bias of gpt-4-turbo is 3.6% (in magnitude) on average as opposed to 13.8% (in magnitude) for Bespoke-7B. This is especially glaring since balanced accuracy does not indicate a large difference between gpt-4-turbo and Bespoke-7B (84.7% BAcc vs 81.6% BAcc). On the summarization subsets of RAGTruth (CNN-DM and Recent News), we see that the metrics predict large differences between systems when the ground-truth annotation does not and vice versa. For example, while ground truth annotations predict that Llama-2-13B-chat makes much fewer grounding errors than Mistral-7B-Instruct (9.6% vs 13.5%), Bespoke-7B predicts Mistral-7B-Instruct to be on par with Llama-2-13B-chat. Thus, results indicate several inconsistencies between predicted and ground-truth system error rates. We report trends for response-level bias of the metrics in Figure 8.

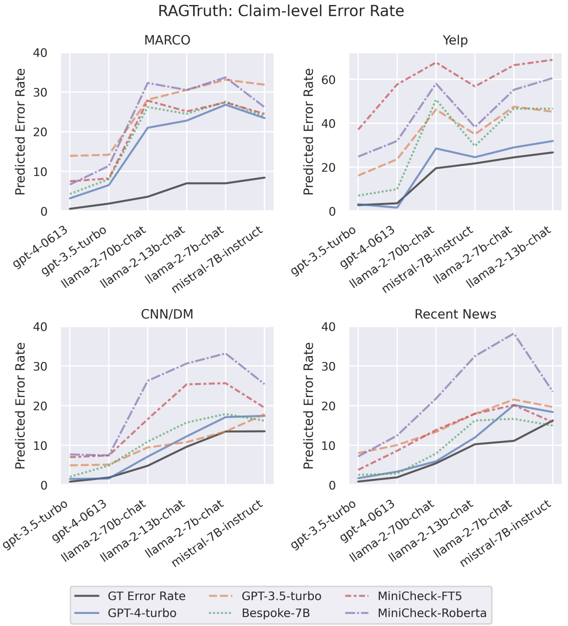

Figure 7: **Predicted system-level error rate on RAGTruth (claim-level).** Inconsistent predictions between different metrics lead to discrepancies in the quantification of the system error rate.

A.6 Effect of Chunking on Evaluator

In Table 30, we report the performance of the Bespoke-7B evaluator without and with chunking

Task	Dataset	Claim Source	Has Long Response?
Summarization	AggreFact-CNN AggreFact-XSum	BART, T5, PEGASUS	N N
	TofuEval-MediaSum TofuEval-MeetingBank	GPT-3.5-Turbo, Vicuna-7B, WizardLM-7B/13B/30B	Y Y
	RAGTruth-CNN/DM RAGTruth-Recent News	GPT-3.5-turbo, GPT-4, Mistral-7b-Instruct, Llama-2-{7B,13B,70B}-chat	Y Y
LLM Response Verification	Reveal	Flan-PaLM-540B, text-davinci-003, Flan-UL2-20B	Y
	ClaimVerify	Bing Chat, NeevaAI, perplexity.ai, YouChat	Y
	FactCheckGPT	ChatGPT	N
Wikipedia Verification	Wice	Human-written	N
Long-form QA	ExpertQA	GPT4, Bing Chat	Y
	LFQA	WebGPT, GPT-3.5, Alpaca-7b	Y
	RAGTruth-MARCO	GPT-3.5-turbo, GPT-4, Mistral-7b-Instruct, Llama-2-{7B,13B,70B}-chat	Y
Data2Text	RAGTruth-Yelp	GPT-3.5-turbo, GPT-4, Mistral-7b-Instruct, Llama-2-{7B,13B,70B}-chat	Y

Table 2: **Description of the task types and claim sources in LLM-AGGREFACT**

Dataset	Avg	AGGREFACT		TOFUEVAL		WICE	REVEAL	CLAIM VERIFY	FACT CHECK	EXPERT QA	LFQA	RAGTRUTH			
		CNN	XSUM	MEDIAS	MEETB							MARCO	YELP	CNN	NEWS
GPT-4-Turbo	76.9	63.3	75.5	68.5	81.2	79.8	88.2	73.1	80.0	60.8	83.0	78.0	84.7	80.0	80.6
Bespoke-7B	76.7	62.3	73.1	72.1	77.1	85.3	89.5	77.1	77.7	60.0	85.2	78.0	81.6	78.3	76.2
+ chunk(500)	75.9	64.5	72.6	72.0	75.8	77.3	89.5	77.1	77.7	59.8	85.0	77.9	78.7	78.4	76.1
MCheck-RBTA	73.3	59.6	66.6	68.8	72.3	66.8	88.6	78.1	75.9	56.7	84.3	79.2	72.1	77.6	79.1
MCheck-FT5	72.8	65.3	68.4	68.4	71.5	70.7	87.4	75.9	74.9	58.7	82.4	76.0	70.2	75.4	73.8
GPT-3.5-turbo	72.2	64.8	71.0	66.3	74.8	70.5	85.1	72.1	74.6	58.3	77.8	70.2	77.4	70.8	76.7
AlignScore	70.5	52.6	65.0	65.7	72.9	67.3	86.8	72.0	75.7	56.8	81.7	73.5	66.7	75.9	75.1
FactKB	56.9	58.5	64.4	51.6	53.1	55.3	71.2	56.8	58.6	53.1	57.9	56.9	50.6	50.4	57.7

Table 3: **Balanced Accuracy of metrics on the dev set of LLM-AGGREFACT**

Dataset	Avg	AGGREFACT		TOFUEVAL		WICE	REVEAL	CLAIM VERIFY	FACT CHECK	EXPERT QA	LFQA	RAGTRUTH			
		CNN	XSUM	MEDIAS	MEETB							MARCO	YELP	CNN	NEWS
GPT-4-turbo	34.1	71.2	21.4	59.3	29.8	25.6	16.1	46.7	14.5	46.9	27.2	27.0	23.1	35.3	33.7
Bespoke-7B	30.6	69.5	25.3	47.5	36.3	13.7	15.0	36.0	9.3	33.2	21.4	25.2	17.6	36.9	42.2
+ chunk(500)	27.7	59.3	24.3	39.8	30.4	9.8	15.0	35.7	9.3	32.8	21.3	25.2	14.7	30.5	39.8
MCheck-RBTA	24.4	66.1	26.4	37.3	31.5	9.0	12.9	25.7	8.8	21.4	11.5	17.7	20.2	28.3	25.3
MCheck-FT5	30.6	59.3	36.6	44.9	42.9	9.8	14.6	36.0	12.6	32.2	22.4	27.9	9.1	36.9	43.4
GPT-3.5-turbo	34.7	59.3	25.8	57.6	28.0	27.8	14.5	41.2	11.8	48.3	28.8	34.5	20.2	51.3	36.1
AlignScore	37.3	93.2	44.9	55.9	34.3	14.1	16.2	46.0	10.2	32.9	24.2	35.4	34.8	37.4	42.2
FactKB	64.8	78.0	17.0	91.2	80.6	59.8	15.6	78.3	32.3	74.5	77.7	44.2	90.8	96.3	71.1

Table 4: **False positive rate (FPR) of metrics on the dev set of LLM-AGGREFACT**

Dataset	Avg	AGGREFACT		TOFUEVAL		WICE	REVEAL	CLAIM VERIFY	FACT CHECK	EXPERT QA	LFQA	RAGTRUTH			
		CNN	XSUM	MEDIAS	MEETB							MARCO	YELP	CNN	NEWS
GPT-4-turbo	12.1	2.2	27.7	3.7	7.8	14.8	7.5	7.1	25.6	31.5	6.9	16.9	7.4	4.7	5.1
Bespoke-7B	16.0	6.0	28.4	8.4	9.4	15.7	6.0	9.9	35.3	46.8	8.2	18.8	19.2	6.5	5.5
+ chunk(500)	20.5	11.8	30.5	16.2	18.0	35.7	6.0	10.2	35.3	47.7	8.7	18.9	27.9	12.7	8.0
MCheck-RBTA	29.0	14.8	40.4	25.1	23.8	57.4	9.9	18.0	39.5	65.1	19.9	23.9	35.6	16.5	16.4
MCheck-FT5	23.8	10.0	26.6	18.3	14.1	48.7	10.6	12.2	37.6	50.3	12.7	20.1	50.5	12.2	8.9
GPT-3.5-turbo	21.0	11.0	32.2	9.7	22.5	31.3	15.3	14.6	39.1	35.1	15.7	25.1	25.1	7.1	10.6
AlignScore	21.6	1.5	25.1	12.7	20.0	51.3	10.1	10.1	38.3	53.5	12.3	17.6	31.8	10.8	7.6
FactKB	21.5	5.0	54.3	5.5	13.2	29.6	42.1	8.2	50.4	19.2	6.5	41.9	8.1	3.0	13.6

Table 5: **False negative rate (FNR) of metrics on the dev set of LLM-AGGREFACT**

		GPT-4-turbo	GPT-3.5-turbo	Bespoke-7B	Bespoke-7B (cs=500)	MiniCheck-FT5	MiniCheck-Roberta	AlignScore
corr type	source							
Kendall's τ	ExpertQA	0.73	0.60	0.47	0.60	0.60	0.87	0.73
	Lfqa	0.87	0.87	0.87	0.87	0.87	0.87	0.87
	RAGTruth-CNN/DM	1.00	1.00	0.87	0.73	0.73	0.47	0.73
	RAGTruth-News	0.87	0.87	0.73	0.47	0.73	0.73	0.87
	RAGTruth-MARCO	0.87	0.87	0.47	0.47	0.33	0.47	0.47
	RAGTruth-Yelp	0.73	0.60	0.60	0.60	0.60	0.73	0.60
	Average	0.84	0.80	0.67	0.62	0.64	0.69	0.71
Pearson ρ	ExpertQA	0.76	0.75	0.77	0.75	0.77	0.88	0.85
	Lfqa	0.99	0.97	1.00	1.00	0.99	0.99	0.99
	RAGTruth-CNN/DM	1.00	0.94	0.96	0.89	0.88	0.85	0.93
	RAGTruth-News	0.93	0.92	0.91	0.93	0.81	0.73	0.94
	RAGTruth-MARCO	0.90	0.92	0.83	0.84	0.80	0.78	0.83
	RAGTruth-Yelp	0.98	0.92	0.91	0.92	0.78	0.87	0.85
	Average	0.93	0.91	0.90	0.89	0.84	0.85	0.90

Table 6: **System ranking correlation (claim-level labels).** For 6 LLM-AGGREFACT datasets, we report the correlations between system rankings based on human-labeled error rate and predicted error rate by AutoAIS evaluators. Each dataset has generations from 6 NLG systems. While the Pearson correlation coefficient is high for the top evaluators, Kendall's τ is lower. The value of τ indicates that the evaluators make one-three ranking errors in each ranking of the 6 systems.

		GPT-4-turbo	GPT-3.5-turbo	Bespoke-7B	Bespoke-7B (cs=500)	MiniCheck-FT5	MiniCheck-Roberta	AlignScore
corr type	source							
Kendall's τ	ExpertQA	0.47	0.60	0.60	0.60	0.60	0.73	0.73
	Lfqa	0.97	1.00	1.00	1.00	0.86	0.71	0.97
	RAGTruth-CNN/DM	0.87	0.73	0.87	0.73	0.60	0.33	0.73
	RAGTruth-News	1.00	1.00	1.00	0.87	0.87	0.73	1.00
	RAGTruth-MARCO	1.00	0.73	0.73	0.73	0.60	0.73	0.73
	RAGTruth-Yelp	0.60	0.47	0.47	0.33	0.47	0.47	0.33
	Average	0.82	0.76	0.78	0.71	0.67	0.62	0.75
Pearson ρ	ExpertQA	0.71	0.80	0.74	0.69	0.76	0.82	0.89
	Lfqa	0.99	0.97	0.98	0.97	0.97	0.92	0.99
	RAGTruth-CNN/DM	1.00	0.93	0.96	0.89	0.89	0.81	0.93
	RAGTruth-News	0.91	0.92	0.88	0.91	0.80	0.69	0.94
	RAGTruth-MARCO	0.92	0.94	0.88	0.88	0.86	0.83	0.87
	RAGTruth-Yelp	0.98	0.94	0.92	0.91	0.75	0.83	0.72
	Average	0.92	0.92	0.89	0.88	0.84	0.82	0.89

Table 7: **System ranking correlation (response-level labels).** For 6 LLM-AGGREFACT datasets, we report the correlations between system rankings based on human-labeled error rate and predicted error rate by AutoAIS evaluators. The labels are aggregated at the response-level. Each dataset has generations from 6 NLG systems. While the Pearson correlation coefficient is high for the top evaluators, Kendall's τ is lower indicating errors in system ranking.

	label	GPT-4-turbo	GPT-3.5-turbo	Bespoke-7B	Bespoke-7B (cs=500)	MiniCheck-FT5	MiniCheck-Rbta
dataset							
Wice	67.0 (0.0)	54.7 (-12.3)	58.7 (-8.3)	63.0 (-4.0)	72.2 (5.2)	76.5 (9.5)	79.9 (12.9)
FactCheck-GPT	82.7 (0.0)	75.1 (-7.5)	79.7 (-3.0)	81.1 (-1.6)	81.1 (-1.6)	78.8 (-3.9)	82.2 (-0.5)

Table 8: **Wice and FactCheck:** Quantification bias of metrics

	label	GPT-4-turbo	GPT-3.5-turbo	Bespoke-7B	Bespoke-7B (cs=500)	MiniCheck-FT5	MiniCheck-Rbta
System Name							
BART	17.9 (0.0)	6.4 (-11.5)	14.5 (-3.4)	9.8 (-8.1)	17.5 (-0.4)	15.8 (-2.1)	18.8 (0.9)
Pegasus	9.6 (0.0)	4.0 (-5.6)	16.0 (6.4)	8.8 (-0.8)	16.0 (6.4)	12.8 (3.2)	20.8 (11.2)
PegasusDynamic	6.0 (0.0)	4.0 (-2.0)	20.0 (14.0)	6.0 (0.0)	10.0 (4.0)	8.0 (2.0)	6.0 (0.0)
T5	4.0 (0.0)	8.0 (4.0)	8.0 (4.0)	10.0 (6.0)	10.0 (6.0)	14.0 (10.0)	12.0 (8.0)
Headroom	4.0 (0.0)	4.0 (0.0)	8.0 (4.0)	6.0 (2.0)	10.0 (6.0)	8.0 (4.0)	6.0 (2.0)

Table 9: **AggreFact-CNN: Predicted instance-level error rates for systems.** Quantification bias in parentheses.

	label	GPT-4-turbo	GPT-3.5-turbo	Bespoke-7B	Bespoke-7B (cs=500)	MiniCheck-FT5	MiniCheck-Roberta
System Name							
BART	49.0 (0.0)	53.3 (4.3)	53.1 (4.1)	52.8 (3.8)	54.4 (5.4)	45.4 (-3.6)	57.7 (8.7)
Pegasus	52.0 (0.0)	48.0 (-4.0)	50.7 (-1.3)	36.0 (-16.0)	37.3 (-14.7)	38.7 (-13.3)	48.0 (-4.0)
Headroom	49.0 (0.0)	48.0 (-1.0)	50.7 (1.7)	36.0 (-13.0)	37.3 (-11.7)	38.7 (-10.3)	48.0 (-1.0)

Table 10: **AggreFact-XSum: Predicted instance-level error rates for systems.** Quantification bias in parentheses.

	GT Label	GPT-4-turbo	GPT-3.5-turbo	Bespoke-7B	Bespoke-7B (cs=500)	MiniCheck-FT5	MiniCheck-Roberta
System Name							
Model-Extra	19.2 (0.0)	6.3 (-12.9)	13.7 (-5.6)	15.7 (-3.5)	21.3 (2.0)	24.8 (5.6)	31.4 (12.2)
model_A	19.7 (0.0)	11.7 (-8.0)	15.0 (-4.7)	18.2 (-1.5)	24.1 (4.4)	26.6 (6.9)	33.2 (13.5)
model_B	20.4 (0.0)	12.7 (-7.7)	19.0 (-1.4)	18.7 (-1.8)	29.9 (9.5)	28.2 (7.7)	35.6 (15.1)
model_C	20.0 (0.0)	12.4 (-7.6)	17.2 (-2.8)	15.2 (-4.8)	24.5 (4.5)	23.4 (3.4)	32.1 (12.1)
model_D	18.6 (0.0)	11.2 (-7.4)	15.2 (-3.3)	17.8 (-0.7)	24.9 (6.3)	23.0 (4.5)	29.7 (11.2)
model_E	19.3 (0.0)	12.6 (-6.6)	16.6 (-2.7)	16.9 (-2.3)	24.6 (5.3)	25.9 (6.6)	31.9 (12.6)
Headroom	18.6 (0.0)	6.3 (-12.3)	13.7 (-4.9)	15.2 (-3.4)	21.3 (2.7)	23.0 (4.5)	29.7 (11.2)

Table 11: **TofuEval-MediaSum: Predicted claim-level error rates for systems.** Quantification bias in parentheses.

GT Order	GPT-4-turbo			GPT-3.5-turbo			Bespoke-7B			Bespoke-7B (cs=500)			MiniCheck-FT5			MiniCheck-Roberta			AlignScore		
	>	=	<	>	=	<	>	=	<	>	=	<	>	=	<	>	=	<	>	=	<
=	1	10	4	0	15	0	0	15	0	0	14	1	0	15	0	0	15	0	0	15	0
<	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
%Err	33.3			0.0			0.0			6.7			0.0			0.0			0.0		
%Maj. Err	0.0			0.0			0.0			0.0			0.0			0.0			0.0		

Table 12: **TofuEval-MediaSum: Inconsistency in system-pair ranking based on claim-level error rates for systems.** We report a confusion matrix of pairwise system ranking decisions. We measure inconsistencies between the ranking based on the labeled error rate and the ranking based on the predicted error rate. For a system pair (s1, s2), '=' indicates no significant difference between s1 and s2, '<' indicates s1 has a lower error rate than s2, and '>' indicates s1 has a higher error rate than s2. When a metric predicts a significant but opposite ranking between a pair, we count it as a Major Error. Significance is computed with the two-proportion z-test and p_value < 0.05.

	label	GPT-4-turbo	GPT-3.5-turbo	Bespoke-7B	Bespoke-7B (cs=500)	MiniCheck-FT5	MiniCheck-Roberta
System Name							
Model-Extra	49.0 (0.0)	19.2 (-29.8)	37.5 (-11.5)	40.4 (-8.7)	55.8 (6.7)	64.4 (15.4)	78.8 (29.8)
model_A	38.1 (0.0)	22.9 (-15.2)	32.4 (-5.7)	35.2 (-2.9)	44.8 (6.7)	51.4 (13.3)	60.0 (21.9)
model_B	41.9 (0.0)	26.7 (-15.2)	38.1 (-3.8)	38.1 (-3.8)	56.2 (14.3)	53.3 (11.4)	66.7 (24.8)
model_C	39.4 (0.0)	26.9 (-12.5)	31.7 (-7.7)	29.8 (-9.6)	48.1 (8.7)	41.3 (1.9)	58.7 (19.2)
model_D	37.5 (0.0)	25.0 (-12.5)	30.8 (-6.7)	34.6 (-2.9)	45.2 (7.7)	46.2 (8.7)	60.6 (23.1)
model_E	38.5 (0.0)	25.0 (-13.5)	32.7 (-5.8)	30.8 (-7.7)	46.2 (7.7)	49.0 (10.6)	63.5 (25.0)
Headroom	37.5 (0.0)	19.2 (-18.3)	30.8 (-6.7)	29.8 (-7.7)	44.8 (7.3)	41.3 (3.8)	58.7 (21.2)

Table 13: **TofuEval-MediaSum: Predicted summary-level error rates for systems.** Quantification bias in parentheses.

	GT Error Rate	GPT-4-turbo	GPT-3.5-turbo	Bespoke-7B	Bespoke-7B (cs=500)	MiniCheck-FT5	MiniCheck-Roberta
System							
Model-Extra	14.0 (0.0)	15.1 (1.1)	35.4 (21.4)	17.2 (3.2)	29.1 (15.1)	23.9 (9.8)	35.8 (21.8)
model-A	19.9 (0.0)	18.8 (-1.2)	29.7 (9.8)	19.1 (-0.8)	28.5 (8.6)	21.9 (2.0)	29.7 (9.8)
model-B	22.4 (0.0)	24.1 (1.7)	33.6 (11.2)	21.3 (-1.0)	27.6 (5.2)	24.5 (2.1)	34.6 (12.2)
model-C	20.1 (0.0)	19.7 (-0.4)	30.5 (10.4)	18.9 (-1.2)	27.4 (7.3)	22.8 (2.7)	33.6 (13.5)
model-D	11.8 (0.0)	15.4 (3.6)	26.5 (14.7)	16.1 (4.3)	22.2 (10.4)	16.1 (4.3)	24.7 (12.9)
model-E	19.3 (0.0)	20.5 (1.2)	30.9 (11.6)	21.6 (2.3)	27.4 (8.1)	20.8 (1.5)	31.3 (12.0)
Headroom	11.8 (0.0)	15.1 (3.3)	26.5 (14.7)	16.1 (4.3)	22.2 (10.4)	16.1 (4.3)	24.7 (12.9)

Table 14: **TofuEval-MeetingBank: Predicted claim-level error rates for systems.** Quantification bias in parentheses.

GT Order	GPT-4-turbo			GPT-3.5-turbo			Bespoke-7B			Bespoke-7B (cs=500)			MiniCheck-FT5			MiniCheck-Roberta			AlignScore		
	>	=	<	>	=	<	>	=	<	>	=	<	>	=	<	>	=	<	>	=	<
=	0	10	0	0	9	1	0	10	0	0	10	0	0	9	1	0	9	1	1	8	1
<	0	3	2	0	5	0	0	5	0	0	5	0	0	4	1	0	3	2	0	2	3
%Err	20.0			40.0			33.3			33.3			33.3			26.7			26.7		
%Maj. Err	0.0			0.0			0.0			0.0			0.0			0.0			0.0		

Table 15: **TofuEval-MeetingBank: Inconsistency in system-pair ranking based on claim-level error rates for systems.** We report a confusion matrix of pairwise system ranking decisions. We measure inconsistencies between the ranking based on the labeled error rate and the ranking based on the predicted error rate. For a system pair (s1, s2), '=' indicates no significant difference between s1 and s2, '<' indicates s1 has a lower error rate than s2, and '>' indicates s1 has a higher error rate than s2. When a metric predicts a significant but opposite ranking between a pair, we count it as a Major Error. Significance is computed with the two-proportion z-test and p_value < 0.05.

	label	GPT-4-turbo	GPT-3.5-turbo	Bespoke-7B	Bespoke-7B (cs=500)	MiniCheck-FT5	MiniCheck-Roberta
System Name							
Model-Extra	29.8 (0.0)	32.7 (2.9)	60.6 (30.8)	38.5 (8.7)	57.7 (27.9)	49.0 (19.2)	61.5 (31.7)
model_A	35.2 (0.0)	33.3 (-1.9)	51.4 (16.2)	35.2 (0.0)	52.4 (17.1)	43.8 (8.6)	54.3 (19.0)
model_B	45.2 (0.0)	48.1 (2.9)	60.6 (15.4)	44.2 (-1.0)	53.8 (8.7)	51.9 (6.7)	64.4 (19.2)
model_C	34.6 (0.0)	31.7 (-2.9)	50.0 (15.4)	30.8 (-3.8)	46.2 (11.5)	42.3 (7.7)	56.7 (22.1)
model_D	26.0 (0.0)	33.7 (7.7)	50.0 (24.0)	35.6 (9.6)	48.1 (22.1)	32.7 (6.7)	49.0 (23.1)
model_E	34.4 (0.0)	38.5 (4.2)	51.0 (16.7)	43.8 (9.4)	53.1 (18.8)	40.6 (6.2)	59.4 (25.0)
Headroom	26.0 (0.0)	31.7 (5.8)	50.0 (24.0)	30.8 (4.8)	46.2 (20.2)	32.7 (6.7)	49.0 (23.1)

Table 16: **TofuEval-MeetingBank: Predicted summary-level error rates for systems.** Quantification bias in parentheses.

	label	GPT-4-turbo	GPT-3.5-turbo	Bespoke-7B	Bespoke-7B (cs=500)	MiniCheck-FT5	MiniCheck-Roberta
System							
Flan-PaLM-540B	67.4 (0.0)	59.7 (-7.7)	61.0 (-6.3)	58.5 (-8.8)	58.5 (-8.8)	61.2 (-6.1)	57.8 (-9.6)
Flan-UL2-20B	79.4 (0.0)	68.8 (-10.6)	71.4 (-8.0)	68.7 (-10.7)	68.7 (-10.7)	67.8 (-11.6)	70.0 (-9.4)
GPT-3	76.2 (0.0)	63.0 (-13.2)	68.6 (-7.6)	65.6 (-10.6)	65.6 (-10.6)	68.3 (-7.9)	72.5 (-3.7)
Headroom	67.4 (0.0)	59.7 (-7.7)	61.0 (-6.3)	58.5 (-8.8)	58.5 (-8.8)	61.2 (-6.1)	57.8 (-9.6)

Table 17: **Reveal: Predicted instance-level error rates for systems.** Quantification bias in parentheses.

	label	GPT-4-turbo	GPT-3.5-turbo	Bespoke-7B	Bespoke-7B (cs=500)	MiniCheck-FT5	MiniCheck-Roberta
System Name							
Flan-PaLM-540B	74.7 (0.0)	72.1 (-2.6)	74.7 (0.0)	69.5 (-5.2)	69.5 (-5.2)	69.5 (-5.2)	69.5 (-5.2)
Flan-UL2-20B	84.2 (0.0)	76.6 (-7.6)	79.5 (-4.7)	76.6 (-7.6)	76.6 (-7.6)	77.2 (-7.0)	80.1 (-4.1)
GPT-3	78.9 (0.0)	66.3 (-12.6)	74.2 (-4.7)	68.4 (-10.5)	68.4 (-10.5)	72.1 (-6.8)	77.4 (-1.6)
Headroom	74.7 (0.0)	66.3 (-8.4)	74.2 (-0.5)	68.4 (-6.3)	68.4 (-6.3)	69.5 (-5.2)	69.5 (-5.2)

Table 18: **Reveal: Predicted summary-level error rates for systems.** Quantification bias in parentheses.

	label	GPT-4-turbo	GPT-3.5-turbo	Bespoke-7B	Bespoke-7B (cs=500)	MiniCheck-FT5	MiniCheck-Roberta
System							
bing_chat	9.4	6.1 (-3.3)	10.7 (1.2)	10.7 (1.2)	9.8 (0.4)	11.5 (2.0)	13.1 (3.7)
neeva	27.3	23.4 (-3.9)	31.2 (3.9)	26.3 (-1.0)	28.6 (1.3)	29.3 (2.0)	34.5 (7.2)
perplexity	30.7	19.6 (-11.2)	29.4 (-1.4)	26.6 (-4.1)	26.8 (-3.9)	28.0 (-2.7)	37.2 (6.5)
you	31.3	34.3 (3.0)	32.8 (1.5)	28.4 (-3.0)	25.4 (-6.0)	29.9 (-1.5)	47.8 (16.4)
Headroom	9.4 (0.0)	6.1 (-3.3)	10.7 (1.2)	10.7 (1.2)	9.8 (0.4)	11.5 (2.0)	13.1 (3.7)

Table 19: **ClaimVerify: Predicted instance-level error rates for systems.** Quantification bias in parentheses.

	label	GPT-4-turbo	GPT-3.5-turbo	Bespoke-7B	Bespoke-7B (cs=500)	MiniCheck-FT5	MiniCheck-Roberta
System Name							
bing_chat	16.3 (0.0)	11.4 (-4.9)	18.7 (2.4)	18.7 (2.4)	16.3 (0.0)	20.3 (4.1)	19.5 (3.3)
neeva	51.9 (0.0)	45.3 (-6.6)	56.6 (4.7)	53.8 (1.9)	56.6 (4.7)	59.4 (7.5)	61.3 (9.4)
perplexity	53.6 (0.0)	38.6 (-15.0)	55.7 (2.1)	52.9 (-0.7)	50.7 (-2.9)	54.3 (0.7)	64.3 (10.7)
you	38.6 (0.0)	45.5 (6.8)	40.9 (2.3)	38.6 (0.0)	36.4 (-2.3)	40.9 (2.3)	61.4 (22.7)
Headroom	16.3 (0.0)	11.4 (-4.9)	18.7 (2.4)	18.7 (2.4)	16.3 (0.0)	20.3 (4.1)	19.5 (3.3)

Table 20: **ClaimVerify: Predicted summary-level error rates for systems.** Quantification bias in parentheses.

System Name	label	GPT-4-turbo	GPT-3.5-turbo	Bespoke-7B	Bespoke-7B (cs=500)	MiniCheck-FT5	MiniCheck-Roberta
bing_chat	16.7 (0.0)	34.1 (17.4)	40.2 (23.5)	44.0 (27.3)	49.4 (32.7)	49.7 (33.0)	57.1 (40.4)
gpt4	27.4 (0.0)	62.1 (34.7)	54.7 (27.4)	73.7 (46.3)	75.8 (48.4)	78.9 (51.6)	90.5 (63.2)
post_hoc_gs_gpt4	22.1 (0.0)	52.8 (30.7)	54.1 (32.0)	74.8 (52.7)	74.8 (52.7)	73.8 (51.7)	86.3 (64.2)
post_hoc_sphere_gpt4	33.5 (0.0)	53.8 (20.4)	53.8 (20.4)	72.8 (39.3)	72.8 (39.3)	71.7 (38.3)	92.6 (59.2)
rr_gs_gpt4	11.7 (0.0)	8.7 (-3.0)	11.8 (0.1)	16.7 (5.1)	16.8 (5.2)	23.3 (11.7)	31.7 (20.1)
rr_sphere_gpt4	20.3 (0.0)	9.8 (-10.4)	17.1 (-3.1)	18.7 (-1.6)	18.9 (-1.4)	28.5 (8.3)	46.9 (26.6)
Headroom	11.7 (0.0)	8.7 (-3.0)	11.8 (0.1)	16.7 (5.1)	16.8 (5.2)	23.3 (11.7)	31.7 (20.1)

Table 21: **ExpertQA: Predicted claim-level error rates for systems.** Quantification bias in parentheses.

GT Order	GPT-4-turbo			GPT-3.5-turbo			Bespoke-7B			Bespoke-7B (cs=500)			MiniCheck-FT5			MiniCheck-Roberta			AlignScore		
	>	=	<	>	=	<	>	=	<	>	=	<	>	=	<	>	=	<	>	=	<
=	1	2	2	1	2	2	1	2	2	1	2	2	1	2	2	1	2	2	0	3	2
<	0	2	8	0	1	9	0	2	8	0	2	8	0	1	9	0	0	10	0	1	9
%Err	33.3			26.7			33.3			33.3			26.7			20.0			20.0		
%Maj. Err	0.0			0.0			0.0			0.0			0.0			0.0			0.0		

Table 22: **ExpertQA: Inconsistency in system-pair ranking based on claim-level error rates for systems.** We report a confusion matrix of pairwise system ranking decisions. We measure inconsistencies between the ranking based on the labeled error rate and the ranking based on the predicted error rate. For a system pair (s1, s2), '=' indicates no significant difference between s1 and s2, '<' indicates s1 has a lower error rate than s2, and '>' indicates s1 has a higher error rate than s2. When a metric predicts a significant but opposite ranking between a pair, we count it as a Major Error. Significance is computed with the two-proportion z-test and p_value < 0.05.

System Name	label	GPT-4-turbo	GPT-3.5-turbo	Bespoke-7B	Bespoke-7B (cs=500)	MiniCheck-FT5	MiniCheck-Roberta
bing_chat	29.0 (0.0)	54.4 (25.4)	60.4 (31.4)	63.3 (34.3)	71.6 (42.6)	73.4 (44.4)	81.7 (52.7)
gpt4	39.2 (0.0)	74.5 (35.3)	70.6 (31.4)	86.3 (47.1)	82.4 (43.1)	88.2 (49.0)	94.1 (54.9)
post_hoc_gs_gpt4	52.0 (0.0)	91.3 (39.3)	92.9 (40.8)	98.0 (45.9)	98.0 (45.9)	98.0 (45.9)	98.5 (46.4)
post_hoc_sphere_gpt4	60.5 (0.0)	86.8 (26.3)	87.9 (27.4)	94.2 (33.7)	94.2 (33.7)	94.7 (34.2)	98.9 (38.4)
rr_gs_gpt4	26.6 (0.0)	27.1 (0.5)	33.0 (6.4)	44.3 (17.7)	44.8 (18.2)	56.2 (29.6)	63.1 (36.5)
rr_sphere_gpt4	42.1 (0.0)	26.4 (-15.7)	44.3 (2.1)	45.0 (2.9)	45.0 (2.9)	61.4 (19.3)	79.3 (37.1)
Headroom	26.6 (0.0)	26.4 (-0.2)	33.0 (6.4)	44.3 (17.7)	44.8 (18.2)	56.2 (29.6)	63.1 (36.5)

Table 23: **ExpertQA: Predicted summary-level error rates for systems.** Quantification bias in parentheses.

System Name	label	GPT-4-turbo	GPT-3.5-turbo	Bespoke-7B	Bespoke-7B (cs=500)	MiniCheck-FT5	MiniCheck-Roberta
alpaca	76.0 (0.0)	68.5 (-7.5)	71.5 (-4.5)	70.8 (-5.2)	70.8 (-5.2)	75.3 (-0.7)	83.5 (7.5)
alpaca_wdoc	41.8 (0.0)	34.7 (-7.0)	44.6 (2.8)	36.8 (-4.9)	37.5 (-4.2)	38.2 (-3.5)	48.8 (7.0)
gpt3	78.9 (0.0)	62.7 (-16.2)	60.5 (-18.4)	69.1 (-9.8)	68.9 (-10.0)	66.4 (-12.5)	81.1 (2.3)
gpt3_wdoc	18.1 (0.0)	15.8 (-2.3)	22.3 (4.3)	17.2 (-0.9)	18.3 (0.3)	22.6 (4.6)	28.7 (10.6)
gpt3_whudoc	28.8 (0.0)	20.5 (-8.3)	25.1 (-3.7)	25.1 (-3.7)	25.6 (-3.1)	30.5 (1.7)	38.2 (9.4)
webgpt	7.4 (0.0)	6.5 (-0.9)	13.9 (6.5)	6.5 (-0.9)	6.5 (-0.9)	7.4 (0.0)	9.6 (2.2)
Headroom	7.4 (0.0)	6.5 (-0.9)	13.9 (6.5)	6.5 (-0.9)	6.5 (-0.9)	7.4 (0.0)	9.6 (2.2)

Table 24: **LFQA: Predicted claim-level error rates for systems.** Quantification bias in parentheses.

GT Order	GPT-4-turbo			GPT-3.5-turbo			Bespoke-7B			Bespoke-7B (cs=500)			MiniCheck-FT5			MiniCheck-Roberta			AlignScore		
	>	=	<	>	=	<	>	=	<	>	=	<	>	=	<	>	=	<	>	=	<
=	0	1	0	1	0	0	0	1	0	0	1	0	1	0	0	0	1	0	0	1	0
<	0	1	13	0	1	13	0	0	14	0	0	14	0	0	14	0	0	14	0	1	13
%Err	6.7			13.3			0.0			0.0			6.7			0.0			6.7		
%Maj. Err	0.0			0.0			0.0			0.0			0.0			0.0			0.0		

Table 25: **LFQA: Inconsistency in system-pair ranking based on claim-level error rates for systems.** We report a confusion matrix of pairwise system ranking decisions. We measure inconsistencies between the ranking based on the labeled error rate and the ranking based on the predicted error rate. For a system pair (s1, s2), '=' indicates no significant difference between s1 and s2, '<' indicates s1 has a lower error rate than s2, and '>' indicates s1 has a higher error rate than s2. When a metric predicts a significant but opposite ranking between a pair, we count it as a Major Error. Significance is computed with the two-proportion z-test and p_value < 0.05.

System Name	label	GPT-4-turbo	GPT-3.5-turbo	Bespoke-7B	Bespoke-7B (cs=500)	MiniCheck-FT5	MiniCheck-Roberta
alpaca	100.0 (0.0)	96.0 (-4.0)	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)
alpaca_wdoc	72.0 (0.0)	68.0 (-4.0)	90.0 (18.0)	80.0 (8.0)	84.0 (12.0)	72.0 (0.0)	76.0 (4.0)
gpt3	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)
gpt3_wdoc	56.0 (0.0)	56.0 (0.0)	68.0 (12.0)	56.0 (0.0)	62.0 (6.0)	68.0 (12.0)	78.0 (22.0)
gpt3_whudoc	68.0 (0.0)	60.0 (-8.0)	72.0 (4.0)	64.0 (-4.0)	64.0 (-4.0)	78.0 (10.0)	86.0 (18.0)
webgpt	36.0 (0.0)	32.0 (-4.0)	52.0 (16.0)	26.0 (-10.0)	26.0 (-10.0)	32.0 (-4.0)	38.0 (2.0)
Headroom	36.0 (0.0)	32.0 (-4.0)	52.0 (16.0)	26.0 (-10.0)	26.0 (-10.0)	32.0 (-4.0)	38.0 (2.0)

Table 26: **LFQA: Predicted summary-level error rates for systems.** Quantification bias in parentheses.

Query Set	System Name	GT Error Rate	GPT-4-turbo	GPT-3.5-turbo	Bespoke-7B	Bespoke-7B (cs=500)	MiniCheck-FT5	MiniCheck-Roberta
CNN/DM	gpt-3.5-turbo-0613	0.8 (0.0)	1.5 (0.7)	4.9 (4.1)	2.1 (1.2)	5.6 (4.8)	7.0 (6.2)	7.7 (6.9)
	gpt-4-0613	1.9 (0.0)	1.6 (-0.2)	5.1 (3.3)	4.9 (3.0)	8.4 (6.5)	7.4 (5.6)	7.4 (5.6)
	llama-2-70b-chat	4.8 (0.0)	7.1 (2.3)	9.5 (4.6)	10.9 (6.1)	18.9 (14.1)	16.6 (11.8)	26.3 (21.4)
	llama-2-13b-chat	9.6 (0.0)	12.2 (2.6)	10.8 (1.2)	15.7 (6.1)	25.9 (16.3)	25.4 (15.7)	30.6 (21.0)
	llama-2-7b-chat	13.5 (0.0)	17.1 (3.6)	13.5 (0.0)	17.9 (4.4)	27.5 (14.0)	25.6 (12.2)	33.2 (19.7)
	mistral-7B-instruct	13.5 (0.0)	17.4 (3.9)	17.8 (4.3)	16.2 (2.7)	21.1 (7.6)	19.5 (5.9)	25.4 (11.9)
	Headroom	0.8 (0.0)	1.5 (0.7)	4.9 (4.1)	2.1 (1.2)	5.6 (4.8)	7.0 (6.2)	7.4 (6.6)
Recent News	gpt-3.5-turbo-0613	0.8 (0.0)	1.7 (0.8)	8.0 (7.2)	2.5 (1.7)	4.6 (3.8)	3.8 (3.0)	7.2 (6.3)
	gpt-4-0613	1.9 (0.0)	3.3 (1.4)	10.0 (8.1)	2.9 (1.0)	4.3 (2.4)	8.6 (6.7)	12.4 (10.5)
	llama-2-70b-chat	5.4 (0.0)	5.9 (0.5)	13.4 (7.9)	7.9 (2.5)	10.9 (5.4)	13.9 (8.4)	21.8 (16.3)
	llama-2-13b-chat	10.3 (0.0)	12.0 (1.7)	17.9 (7.7)	16.2 (6.0)	18.8 (8.5)	17.9 (7.7)	32.5 (22.2)
	llama-2-7b-chat	11.1 (0.0)	20.1 (9.0)	21.5 (10.4)	16.7 (5.6)	18.8 (7.6)	20.1 (9.0)	38.2 (27.1)
	mistral-7B-instruct	16.2 (0.0)	18.4 (2.1)	19.7 (3.4)	15.0 (-1.3)	18.4 (2.1)	15.8 (-0.4)	23.5 (7.3)
	Headroom	0.8 (0.0)	1.7 (0.8)	8.0 (7.2)	2.5 (1.7)	4.3 (3.5)	3.8 (3.0)	7.2 (6.3)
MARCO	gpt-3.5-turbo-0613	1.9 (0.0)	6.5 (4.7)	14.2 (12.3)	8.0 (6.2)	8.2 (6.3)	8.2 (6.3)	11.4 (9.5)
	gpt-4-0613	0.6 (0.0)	3.2 (2.6)	13.9 (13.4)	4.3 (3.8)	4.6 (4.0)	7.5 (7.0)	6.7 (6.1)
	llama-2-70b-chat	3.6 (0.0)	21.0 (17.4)	28.1 (24.5)	26.2 (22.6)	26.0 (22.4)	27.8 (24.2)	32.3 (28.7)
	llama-2-13b-chat	7.0 (0.0)	22.8 (15.8)	30.5 (23.5)	24.5 (17.5)	24.8 (17.8)	25.1 (18.1)	30.5 (23.5)
	llama-2-7b-chat	7.0 (0.0)	26.8 (19.8)	33.1 (26.1)	27.6 (20.6)	27.8 (20.8)	27.4 (20.4)	33.7 (26.7)
	mistral-7B-instruct	8.4 (0.0)	23.4 (15.0)	31.9 (23.4)	23.9 (15.5)	23.9 (15.5)	24.5 (16.1)	26.2 (17.8)
	Headroom	0.6 (0.0)	3.2 (2.6)	13.9 (13.4)	4.3 (3.8)	4.6 (4.0)	7.5 (7.0)	6.7 (6.1)
Yelp	gpt-3.5-turbo-0613	2.7 (0.0)	3.1 (0.4)	16.2 (13.5)	7.0 (4.3)	12.5 (9.8)	37.1 (34.4)	24.8 (22.1)
	gpt-4-0613	3.5 (0.0)	1.5 (-2.0)	23.6 (20.1)	9.9 (6.4)	17.9 (14.4)	57.7 (54.2)	31.9 (28.4)
	llama-2-70b-chat	19.5 (0.0)	28.5 (9.0)	46.2 (26.7)	50.8 (31.2)	58.9 (39.4)	67.7 (48.2)	58.1 (38.6)
	llama-2-13b-chat	26.7 (0.0)	31.9 (5.2)	45.3 (18.6)	46.7 (20.0)	57.0 (30.3)	68.9 (42.2)	60.6 (33.9)
	llama-2-7b-chat	24.5 (0.0)	29.0 (4.5)	47.6 (23.1)	46.7 (22.2)	56.7 (32.3)	66.5 (42.0)	55.3 (30.8)
	mistral-7B-instruct	21.7 (0.0)	24.5 (2.8)	35.0 (13.3)	29.7 (8.0)	37.0 (15.4)	56.8 (35.1)	38.3 (16.6)
	Headroom	2.7 (0.0)	1.5 (-1.1)	16.2 (13.5)	7.0 (4.3)	12.5 (9.8)	37.1 (34.4)	24.8 (22.1)

Table 27: **RAGTruth: Predicted claim-level error rates for systems.** Quantification bias in parentheses.

GT Order	GPT-4-turbo			GPT-3.5-turbo			Bespoke-7B			Bespoke-7B (cs=500)			MiniCheck-FT5			MiniCheck-Roberta			AlignScore		
	>	=	<	>	=	<	>	=	<	>	=	<	>	=	<	>	=	<	>	=	<
RAGTruth-CNN/DM																					
=	0	2	1	0	2	1	0	3	0	1	2	0	2	1	0	1	2	0	0	3	0
<	0	0	3	0	2	1	0	0	3	0	1	2	0	1	2	0	2	1	0	1	2
%Err	16.7			50.0			0.0			33.3			50.0			50.0			16.7		
%Maj. Err	0.0			0.0			0.0			0.0			0.0			0.0			0.0		
RAGTruth-News																					
=	0	4	1	0	4	1	0	3	2	0	3	2	0	5	0	1	2	2	0	4	1
<	0	0	1	0	1	0	0	0	1	0	0	1	0	1	0	0	1	0	0	0	1
%Err	16.7			33.3			33.3			33.3			16.7			66.7			16.7		
%Maj. Err	0.0			0.0			0.0			0.0			0.0			0.0			0.0		
RAGTruth-MARCO																					
=	0	2	1	0	3	0	0	3	0	0	3	0	0	3	0	1	2	0	1	2	0
<	0	2	1	0	2	1	0	3	0	0	3	0	0	3	0	1	2	0	0	3	0
%Err	50.0			33.3			50.0			50.0			50.0			66.7			66.7		
%Maj. Err	0.0			0.0			0.0			0.0			0.0			16.7			0.0		
RAGTruth-Yelp																					
=	2	1	1	1	1	2	1	1	2	1	1	2	1	1	2	1	0	3	1	2	1
<	0	2	9	0	2	9	0	2	9	0	2	9	0	3	8	0	2	9	0	2	9
%Err	33.3			33.3			33.3			33.3			40.0			40.0			26.7		
%Maj. Err	0.0			0.0			0.0			0.0			0.0			0.0			0.0		

Table 28: **RAGTruth: Inconsistency in system-pair ranking based on claim-level error rates for systems.** We report a confusion matrix of pairwise system ranking decisions. We measure inconsistencies between the ranking based on the labeled error rate and the ranking based on the predicted error rate. For a system pair (s1, s2), '=' indicates no significant difference between s1 and s2, '<' indicates s1 has a lower error rate than s2, and '>' indicates s1 has a higher error rate than s2. When a metric predicts a significant but opposite ranking between a pair, we count it as a Major Error. Significance is computed with the two-proportion z-test and p_value < 0.05.

Query Set	System Name	GT Error Rate	GPT-4-turbo	GPT-3.5-turbo	Bespoke-7B	Bespoke-7B (cs=500)	MiniCheck-FT5	MiniCheck-Roberta
CNN/DM	gpt-3.5-turbo-0613	1.5 (0.0)	2.7 (1.2)	8.8 (7.2)	3.5 (2.0)	9.3 (7.8)	12.3 (10.8)	13.5 (12.0)
	gpt-4-0613	2.3 (0.0)	2.3 (0.0)	7.0 (4.7)	6.7 (4.3)	11.7 (9.4)	10.0 (7.7)	10.4 (8.0)
	llama-2-13b-chat	12.0 (0.0)	14.7 (2.6)	13.5 (1.5)	19.2 (7.1)	32.0 (19.9)	31.6 (19.5)	36.8 (24.8)
	llama-2-70b-chat	7.3 (0.0)	10.7 (3.5)	14.2 (6.9)	16.1 (8.8)	26.5 (19.2)	23.0 (15.8)	36.9 (29.7)
	llama-2-7b-chat	18.4 (0.0)	23.5 (5.1)	18.4 (0.0)	24.2 (5.8)	36.8 (18.4)	33.9 (15.5)	43.0 (24.5)
	mistral-7B-instruct	18.8 (0.0)	24.7 (5.9)	24.1 (5.3)	22.5 (3.7)	29.1 (10.3)	27.2 (8.4)	32.5 (13.7)
	Headroom	1.5 (0.0)	2.3 (0.8)	7.0 (5.5)	3.5 (2.0)	9.3 (7.8)	10.0 (8.5)	10.4 (8.9)
Recent News	gpt-3.5-turbo-0613	1.2 (0.0)	2.5 (1.2)	11.8 (10.6)	3.7 (2.5)	6.8 (5.6)	5.6 (4.3)	10.6 (9.3)
	gpt-4-0613	2.6 (0.0)	4.6 (2.0)	13.7 (11.1)	3.9 (1.3)	5.9 (3.3)	11.1 (8.5)	16.3 (13.7)
	llama-2-13b-chat	11.8 (0.0)	12.7 (1.0)	20.6 (8.8)	18.6 (6.9)	21.6 (9.8)	20.6 (8.8)	35.3 (23.5)
	llama-2-70b-chat	7.5 (0.0)	8.2 (0.7)	17.8 (10.3)	11.0 (3.4)	14.4 (6.8)	18.5 (11.0)	27.4 (19.9)
	llama-2-7b-chat	12.8 (0.0)	23.9 (11.1)	25.6 (12.8)	20.5 (7.7)	23.1 (10.3)	23.9 (11.1)	44.4 (31.6)
	mistral-7B-instruct	23.8 (0.0)	25.0 (1.2)	26.2 (2.5)	20.6 (-3.1)	25.0 (1.2)	21.9 (-1.9)	31.9 (8.1)
	Headroom	1.2 (0.0)	2.5 (1.2)	11.8 (10.6)	3.7 (2.5)	5.9 (4.6)	5.6 (4.3)	10.6 (9.3)
MARCO	gpt-3.5-turbo-0613	2.8 (0.0)	8.6 (5.8)	17.7 (14.9)	10.5 (7.7)	10.8 (8.0)	10.2 (7.5)	14.4 (11.6)
	gpt-4-0613	0.8 (0.0)	4.8 (4.0)	18.8 (17.9)	6.5 (5.6)	6.9 (6.0)	10.6 (9.8)	9.6 (8.8)
	llama-2-13b-chat	11.2 (0.0)	30.7 (19.5)	40.2 (29.0)	33.0 (21.8)	33.4 (22.2)	35.1 (23.9)	40.6 (29.4)
	llama-2-70b-chat	6.1 (0.0)	29.7 (23.6)	36.8 (30.8)	34.7 (28.7)	34.5 (28.5)	37.4 (31.4)	45.0 (38.9)
	llama-2-7b-chat	11.3 (0.0)	37.3 (26.0)	43.8 (32.5)	37.2 (25.9)	37.5 (26.2)	38.4 (27.1)	45.8 (34.5)
	mistral-7B-instruct	11.1 (0.0)	30.1 (19.0)	40.3 (29.1)	30.6 (19.4)	30.6 (19.4)	32.5 (21.3)	34.6 (23.5)
	Headroom	0.8 (0.0)	4.8 (4.0)	17.7 (16.8)	6.5 (5.6)	6.9 (6.0)	10.2 (9.4)	9.6 (8.8)
Yelp	gpt-3.5-turbo-0613	5.7 (0.0)	7.1 (1.4)	32.1 (26.4)	15.5 (9.8)	26.5 (20.8)	59.7 (54.0)	46.9 (41.1)
	gpt-4-0613	5.8 (0.0)	2.6 (-3.2)	35.1 (29.3)	16.3 (10.5)	27.4 (21.6)	72.3 (66.5)	46.6 (40.9)
	llama-2-13b-chat	37.9 (0.0)	44.7 (6.8)	59.0 (21.1)	59.0 (21.1)	68.7 (30.8)	77.6 (39.6)	72.9 (35.0)
	llama-2-70b-chat	29.8 (0.0)	41.5 (11.7)	59.6 (29.8)	64.2 (34.4)	73.5 (43.7)	78.0 (48.2)	69.7 (39.9)
	llama-2-7b-chat	34.1 (0.0)	39.9 (5.7)	58.8 (24.6)	58.2 (24.1)	69.2 (35.0)	76.9 (42.8)	67.8 (33.6)
	mistral-7B-instruct	36.0 (0.0)	39.9 (3.9)	51.9 (15.9)	46.7 (10.7)	54.6 (18.6)	72.9 (36.9)	55.2 (19.2)
	Headroom	5.7 (0.0)	2.6 (-3.1)	32.1 (26.4)	15.5 (9.8)	26.5 (20.8)	59.7 (54.0)	46.6 (40.9)

Table 29: **RAGTruth: Predicted summary-level error rates for systems.** Quantification bias in parentheses.

(chunk size of 500 words). We report the performance on the subset of examples where chunking is applicable, i.e., examples where the document was longer than 500 words.

A.7 Details of Metric Adjustment for Reducing Bias

We compare three ways to reduce the bias of AutoAIS evaluators in estimating the error rates of systems. **Adjusted Counts** (Forman, 2006) uses the TPR and FPR of the evaluator to adjust the predicted system level error rate ($\hat{p}_0$).

$$\hat{p} = \text{clip}(\frac{\hat{p}_0 - FPR}{TPR - FPR}, \text{min} = 0, \text{max} = 1)$$

Under this setup, we are estimating the prevalence (quantification) of hallucinations ($\hat{p}$) by extrapolating from the hallucination rate on a sample (González et al., 2017)). For our experiments, we compute the TPR and FPR of the AutoAIS evaluator on the labeled claim-document pairs generated by one system and use it to adjust the predicted error rate ($\hat{p}_0$) of generations by the other systems. This method is appealing because it does not require the evaluator to produce a scalar score, i.e., it works with the predicted 0/1 labels.

When the evaluator predicts a score instead of directly predicting a label, we can apply threshold tuning. Same as before, we use the labeled claim-document pairs for one system to tune the threshold and then predict labels for the remaining held-out systems using this tuned threshold. We

Dataset	Evaluator	BAcc	PPR	TPR	TNR
AggreFact-CNN	Bespoke-7B + chunk(500)	58.4 60.4	89.7 79.8	92.3 83.0	24.4 37.8
AggreFact-XSum	Bespoke-7B + chunk(500)	69.7 68.8	58.3 52.5	74.4 67.8	65.1 69.9
TofuEval-MediaS	Bespoke-7B + chunk(500)	72.1 72.0	82.9 75.2	91.6 83.8	52.5 60.2
TofuEval-MeetB	Bespoke-7B + chunk(500)	77.1 75.8	80.8 72.7	90.6 82.0	63.7 69.6
RAGTruth-CNN	Bespoke-7B + chunk(500)	77.4 77.8	90.0 82.6	93.4 86.0	61.4 69.7
RAGTruth-News	Bespoke-7B + chunk(500)	78.7 78.4	89.3 84.8	94.0 89.5	63.5 67.3
ClaimVerify	Bespoke-7B + chunk(500)	74.6 74.6	78.4 78.0	90.3 89.9	58.8 59.3
Wice	Bespoke-7B + chunk(500)	85.5 76.9	36.7 27.1	84.4 63.3	86.5 90.6
ExpertQA	Bespoke-7B + chunk(500)	61.9 60.5	61.2 54.9	65.2 58.4	58.7 62.7
Lfqa	Bespoke-7B + chunk(500)	81.6 80.7	67.5 66.0	94.3 92.0	68.9 69.4
RAGTruth-MARCO	Bespoke-7B + chunk(500)	85.9 85.3	83.7 82.6	86.0 84.8	85.7 85.7
RAGTruth-Yelp	Bespoke-7B + chunk(500)	81.8 78.7	71.6 63.1	80.9 71.5	82.7 85.9

Table 30: **Change in Bespoke-7B evaluator predictions with document chunking:** We report the performance of the Bespoke-7B evaluator without and with input document chunking (chunk size of 500 words). These results are calculated on the subset of examples where chunking is applicable. The evaluator with chunking has a lower rate of predicting label "attributable" (PPR = percent of examples predicted as positive/attributable). Correspondingly, the TPR is lower, while TNR is higher.

Source	Calibration Model	GT Error Rate	No Adjustment	Adjusted Counts	Thres. tuning for zero bias	Thres. tuning for ↑BAcc
CNN/DM	gpt-3.5-turbo-0613	0.8	4.5 (6.1)	65.6 (86.5)	2.3 (4.7)	37.0 (42.9)
	gpt-4-0613	1.9	4.1 (6.1)	18.2 (27.5)	1.5 (3.5)	3.1 (5.7)
	llama-2-70b-chat	4.8	3.5 (6.1)	2.1 (3.7)	2.2 (4.7)	11.9 (17.8)
	llama-2-13b-chat	9.6	3.5 (6.1)	1.8 (3.2)	1.6 (3.5)	11.3 (16.3)
	llama-2-7b-chat	13.5	3.8 (6.1)	2.4 (4.8)	1.6 (3.6)	21.0 (27.7)
	mistral-7B-instruct	13.5	4.2 (6.1)	1.8 (3.2)	2.0 (3.8)	4.8 (7.3)
Recent News	gpt-3.5-turbo-0613	0.8	3.3 (6.0)	11.2 (19.3)	1.7 (3.4)	7.7 (15.4)
	gpt-4-0613	1.9	3.4 (6.0)	32.3 (52.0)	2.7 (4.3)	13.1 (24.8)
	llama-2-70b-chat	5.4	3.1 (6.0)	8.4 (16.4)	1.6 (3.4)	7.7 (15.4)
	llama-2-13b-chat	10.3	2.4 (5.6)	3.6 (9.4)	1.8 (5.6)	1.9 (4.2)
	llama-2-7b-chat	11.1	2.5 (6.0)	3.2 (7.7)	1.7 (5.6)	12.3 (24.8)
	mistral-7B-instruct	16.2	3.3 (6.0)	4.3 (8.3)	4.3 (7.7)	18.7 (28.2)
MARCO	gpt-4-0613	0.6	16.5 (22.6)	5.6 (8.4)	6.8 (10.4)	38.4 (44.4)
	gpt-3.5-turbo-0613	1.9	16.0 (22.6)	22.9 (32.6)	3.7 (7.2)	30.0 (36.1)
	llama-2-70b-chat	3.6	12.7 (20.6)	3.7 (8.4)	5.0 (8.4)	17.8 (27.7)
	llama-2-13b-chat	7.0	13.7 (22.6)	3.3 (6.4)	1.4 (4.7)	6.8 (12.4)
	llama-2-7b-chat	7.0	13.1 (22.6)	3.6 (8.4)	1.6 (4.7)	14.3 (24.0)
	mistral-7B-instruct	8.4	14.1 (22.6)	3.9 (8.2)	1.9 (4.7)	14.1 (22.6)
Yelp	gpt-3.5-turbo-0613	2.7	17.6 (31.2)	52.8 (80.5)	6.7 (16.2)	32.9 (46.7)
	gpt-4-0613	3.5	17.1 (31.2)	62.1 (80.5)	8.3 (19.4)	53.5 (66.9)
	llama-2-70b-chat	19.5	12.2 (22.2)	11.4 (21.7)	6.6 (10.7)	4.5 (11.3)
	mistral-7B-instruct	21.7	16.8 (31.2)	17.7 (35.9)	6.6 (16.2)	13.7 (26.7)
	llama-2-7b-chat	24.5	14.0 (31.2)	8.3 (21.7)	4.0 (9.3)	6.1 (19.4)
	llama-2-13b-chat	26.7	14.4 (31.2)	8.9 (21.7)	4.0 (6.7)	5.2 (16.2)

Table 31: **Comparison of adjustment methods on RAGTruth:** We report the bias in estimating the ground-truth system error (hallucination) rates using three adjustment methods. In each section, we report mean absolute bias by using one system for calibration and calculating the mean absolute bias over the remaining systems. Numbers in parentheses indicate the worst-case bias over the remaining systems. Green cells indicate a decrease in bias relative to "No Adjustment". Tuning the evaluator threshold for zero bias consistently reduces the absolute bias in estimation over the held-out systems. Threshold tuning to maximize BAcc worsens the estimation of system-level error. Adjusted counts leads to high mean absolute bias when the ground truth error rate of the system is low.

experiment with two tuning objectives: minimizing the absolute bias towards zero on the labeled calibration data or maximizing the BAcc on the labeled calibration data.

Table 31 provides the resulting mean absolute bias by using each of the 6 systems one by one for calibration and computing bias on the remaining 5 systems. We report the average over all the calibration systems as the cross-validated bias in Table 1. We find that tuning the threshold for zero bias leads to consistent improvements in the held-out systems. Moreover, tuning for higher balanced accuracy hurts the error estimation on the held-out systems. We find that the adjusted counts approach does not provide an improvement over no adjustment if the system used for calibration has a low ground truth error rate. We believe that this is due to a skewed estimation of TPR and FPR when the prevalence of the label 0 is low.

A.8 Claim-level Consistency of Metrics

As discussed in § 3.1, the set of claims labeled as unattributable by two top-performing metrics gpt-4-turbo and Bespoke-7B have low overlap. In Figures 9 and 10, we extend this to all five factuality metrics considered. We show the pairwise consistency (IoU) in predicting the label "attributable" and "unattributable" respectively between the different evaluation metrics on each dataset of LLM-AGGREFACT.

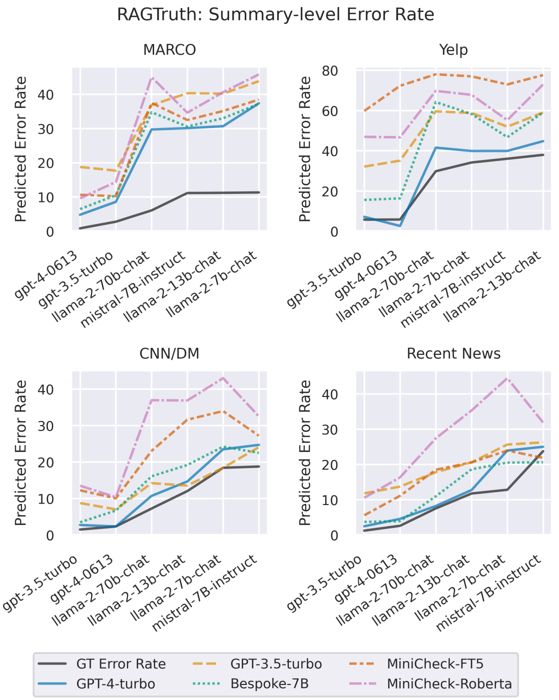

Figure 8: **Predicted system-level error rate on RAGTruth (summary-level).** Claim-level misclassification and metric inconsistency lead to even larger summary-level quantification bias.

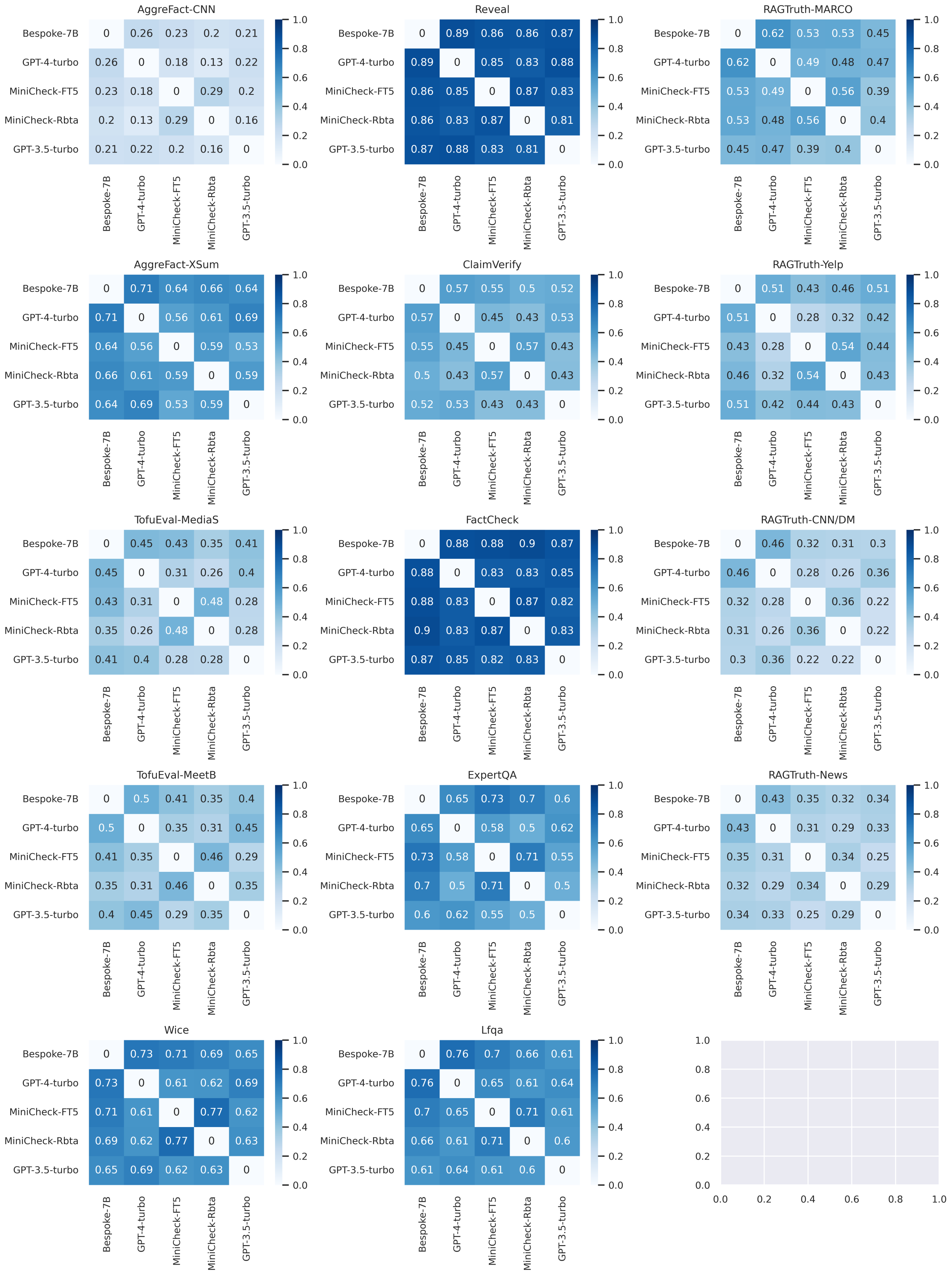

Figure 9: **Pairwise Intersection-over-Union of "unattributable" predictions by AutoAIS metrics.**

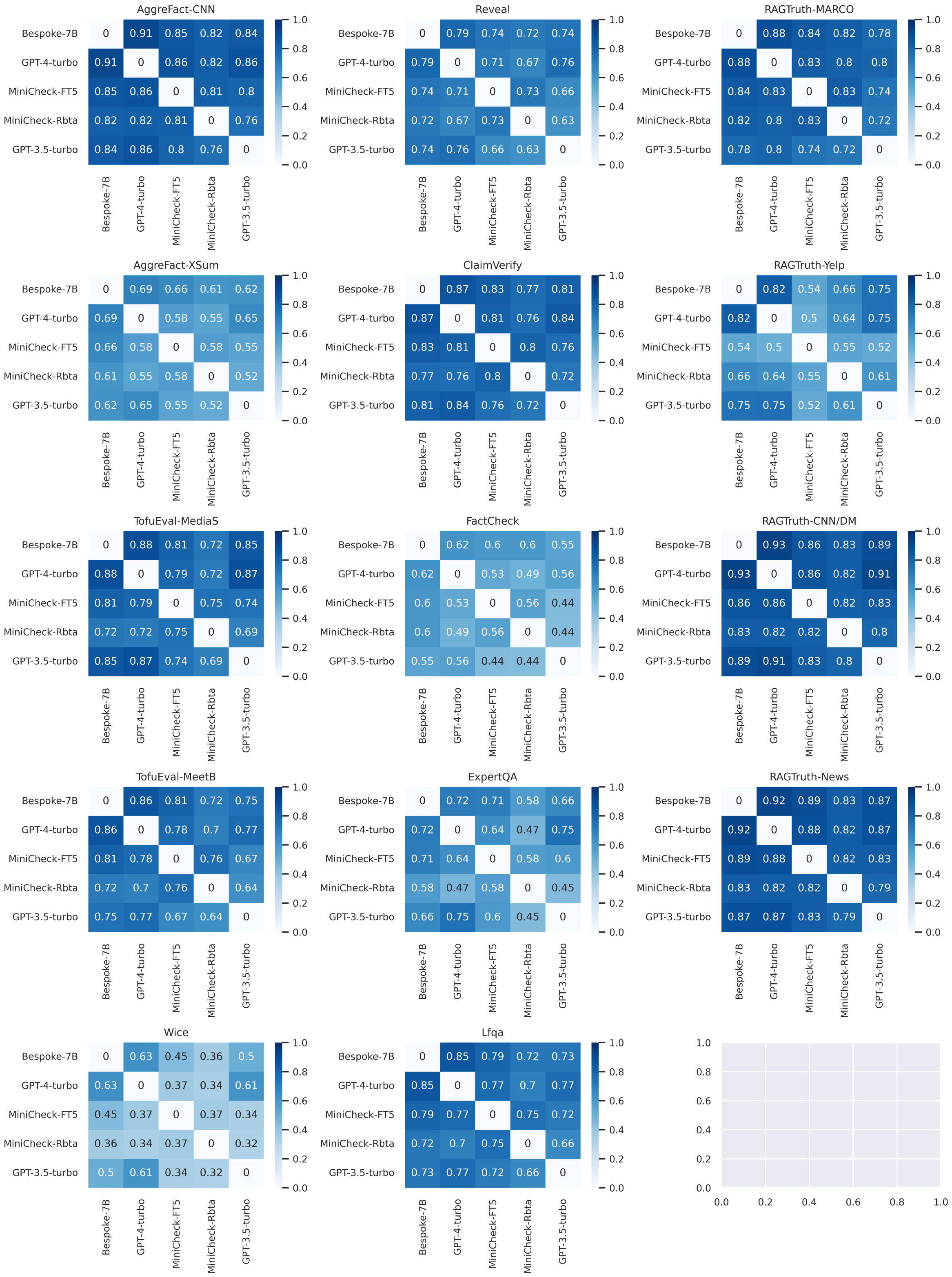

Figure 10: **Pairwise Intersection-over-Union of "attributable" predictions by AutoAIS metrics.**