# Understanding Factual Errors in Summarization: Errors, Summarizers, Datasets, Error Detectors

**Liyan Tang**[◇], **Tanya Goyal**[◇], **Alexander R. Fabbri**[♠], **Philippe Laban**[♠], **Jiacheng Xu**[◇,♠], **Semih Yavuz**[♠], **Wojciech Kryściński**[♠], **Justin F. Rousseau**[◇], **Greg Durrett**[◇]
[◇]The University of Texas at Austin [♠]Salesforce AI Research
lytang@utexas.edu

## Abstract

The propensity of abstractive summarization models to make factual errors has been studied extensively, including design of metrics to detect factual errors and annotation of errors in current systems' outputs. However, the ever-evolving nature of summarization systems, metrics, and annotated benchmarks makes factuality evaluation a moving target, and drawing clear comparisons among metrics has become increasingly difficult. In this work, we aggregate factuality error annotations from nine existing datasets and stratify them according to the underlying summarization model. We compare performance of state-of-the-art factuality metrics, including recent ChatGPT-based metrics, on this stratified benchmark and show that their performance varies significantly across different types of summarization models. Critically, our analysis shows that much of the recent improvement in the factuality detection space has been on summaries from older (pre-Transformer) models instead of more relevant recent summarization models. We further perform a finer-grained analysis per error-type and find similar performance variance across error types for different factuality metrics. Our results show that no one metric is superior in all settings or for all error types, and we provide recommendations for best practices given these insights.[1]

## 1 Introduction

Although abstractive summarization systems (Liu and Lapata, 2019; Lewis et al., 2020; Raffel et al., 2020; Zhang et al., 2020) have improved dramatically in recent years, these models still often include factual errors in generated summaries (Kryscinski et al., 2020; Maynez et al., 2020). A number of metrics have emerged to detect factuality errors, including methods based on sentence entailment (Kryscinski et al., 2020), finer-grained entailment (Goyal and Durrett, 2020; Zhao et al., 2020), question generation and answering (Wang et al., 2020; Durmus et al., 2020; Scialom et al., 2021), and discrimination of synthetically-constructed error instances (Cao and Wang, 2021). Despite recent analyses (Pagnoni et al., 2021; Laban et al., 2022), reliably comparing these metrics remains difficult.

In this paper, we provide a new benchmark that allows for finer-grained comparison between different factuality systems. We aggregate 9 existing annotated factuality datasets to create our benchmark **AGGREFACT**. We stratify it according to the underlying summarization model, categorized into FTSOTA, EXFORMER and OLD based on their development timeline (see Section 2). First, we ask: **do factuality metrics perform equally well at identifying errors from state-of-the-art summarization models and from earlier models?** For nine recent factuality metrics, including recent ChatGPT-based metrics, we show that metric performance varies substantially between different categories of summarization models. Most importantly, we found that the standard way of reporting improvements on category-agnostic benchmarks can be misleading, as most of these gains are on the OLD or EXFORMER subset of the data which are less important to detect. On summaries generated by FTSOTA models, we found that there is no single metric that is superior in evaluating summaries from both the CNN/DM (Hermann et al., 2015) and XSum (Narayan et al., 2018) datasets.

To better understand their behavior, we next analyze **what error types are different factuality metrics capable of identifying** (Section 4). To do this, we leverage datasets from our benchmark that have fine-grained error annotations and unify these into a single taxonomy. We find that the error type distribution changes over time and even differs between annotations of the same summarization models across factuality datasets. Analysis of the factuality metrics shows that metrics claim-

[1]Code and data are available at https://github.com/Liyan06/AggreFact.



*Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*
*Volume 1: Long Papers*, pages 11626–11644
July 9-14, 2023 

| Dataset | Annotators | Kappa | Gran | Annotation Scheme |
|---|---|---|---|---|
| FactCC (Kryscinski et al., 2020) | 2 authors | - | summ | binary consistency label (consistent/inconsistent) |
| Wang'20 (Wang et al., 2020) | 3 crowd-sourced annotators | 0.34/0.51 | sent | binary consistency label (consistent/inconsistent) |
| SummEval (Fabbri et al., 2021b) | 5 crowd-sourced annotators and 3 authors | 0.70 | summ | 5-point Likert scale |
| Polytope (Huang et al., 2020) | 3 trained annotators | - | span | {addition, ommision, inaccuracy intrinsic, inaccuracy extrinsic, positive-negative aspect} |
| Cao'22 (Cao et al., 2022) | 2 authors and 3 graduate students | 0.81 | entity | {Non-hallucinated, Non-factual Hallucination, Intrinsic Hallucination, Factual Hallucination} |
| XSumFaith (Maynez et al., 2020) | 3 trained annotators | 0.80 | span | {intrinsic, extrinsic} |
| FRANK (Pagnoni et al., 2021) | 3 crowd-sourced annotators | 0.53 | sent | {RelE, EntE, CircE, OutE, GramE, LinkE, CorefE, OtherE, NoE} |
| Goyal'21 (Goyal and Durrett, 2021) | 2 authors | - | span | {intrinsic, extrinsic} $\times$ {entity, event, noun phrase, others} |
| CLIFF (Cao and Wang, 2021) | 2 experts | 0.35/0.45 | word | {intrinsic, extrinsic, world knowledge, correct} |

Table 1: Metadata of datasets in AGGREFACT. We report the annotator source, inter-annotator agreement, annotation granularity, and scheme for each dataset. Wang'20 and CLIFF reported kappa scores for XSum/CNNDM separately.

ing SOTA performance can identify each error type better in general, but all metrics differ significantly in how they perform on the same error types across CNN/DM and XSum.

We conclude with the following recommendations for best practices in this area:

1. **Evaluate factuality metrics on summaries generated by the state-of-the-art summarization models.** We found generally worse performance when evaluating factuality systems on summaries generated by FTSOTA models instead of less recent models (Section 3). We release AGGREFACT to support this, which combines existing benchmarks and stratifies them according to the base summarization model, summarization dataset and error types. We suggest future work to augment our benchmark with LLM-generated summaries, e.g. from ChatGPT, which is beyond the scope of this paper.

2. **Choose an appropriate factuality metric for your downstream task at hand.** No one metric is superior across all settings (Section 4). Fine-grained insights offered by our benchmark can be useful to compare strengths of different factuality metrics and make this choice.

3. **Annotate error types consistently with prior work for better comparability.** We found that error type boundaries in existing works are not clear and are not easy to leverage for cross-dataset metric comparisons (Section 4).

We hope that our analysis can shed light on what comparisons practitioners should focus on, how to understand the pros and cons of different metrics, and where metrics should go next. Further, we hope that future work would extend this to incorporate diverse summarization domains such as dialogue summarization (Tang et al., 2022; Fabbri et al., 2021a; Zhang et al., 2021) and medical evidence summarization (Tang et al., 2023). These would have different error distributions, and annotated datasets are needed to perform a more comprehensive comparison and design domain-invariant factuality metrics.

## 2 Benchmark

### 2.1 Benchmark Standardization

Current factuality metrics are evaluated without considering the types of summarization models used to generate the annotated summaries. In these annotated datasets, a large proportion of summaries are generated by older models, such as a pointer-generator network (See et al., 2017), that often make obvious errors that recent models do not make. **We hypothesize that current factuality systems primarily make progress in identifying**

| | OLD | | EXFORMER | | FTSOTA | |
|---|---|---|---|---|---|---|
| | val | test | val | test | val | test |
| -CNN | 2297 | 2166 | 275 | 375 | 459 | 559 |
| -XSUM | 500 | 430 | 500 | 423 | 777 | 558 |

Table 2: Statistics of AGGREFACT-CNN and AGGREFACT-XSUM. Details of individual annotated datasets can be found in Appendix Table 6 and 7.

**factuality inconsistencies in summaries generated by out-of-date summarization models.** If this hypothesis is correct, comparing factuality systems on such datasets provide us less useful information on how these metrics perform on modern summarization systems.

**Summarization datasets splits** We introduce a new benchmark **AGGREFACT** built on top of SummaC from Laban et al. (2022). The benchmark **Aggre**gates nine publicly available datasets (see Table 1) that consist of human evaluations of **Fact**ual consistency on model generated summaries. We focus particularly on incorporating recent datasets annotated on top of state-of-the-art pre-trained Transformer models.

All datasets contain summaries generated from articles in CNN/DM and XSum. Given the unique characteristics of CNN/DM and XSum, our proposed benchmark includes two subsets, AGGREFACT-CNN and AGGREFACT-XSUM, that evaluate the performance of factuality metrics on these two datasets separately (Table 2; see also Table 6 and 7 in the Appendix). This facilitates a more fine-grained and rigorous analysis of the metric performance.

Our benchmark formulates factual consistency evaluation as a binary classification task, following Laban et al. (2022). The binary factuality labels for the summaries are determined by human evaluations on the annotated datasets (Section 2.2).

**Summarization model splits** To validate our hypothesis and make a careful comparison of factuality metrics, we further divide models that were used to generated summaries in the benchmark into three distinct categories: $C = \{$ FTSOTA, EXFORMER, OLD $\}$, as seen in Table 2. FTSOTA represents state-of-the-art fine-tuned summarization models, including BART (Lewis et al., 2020), PEGASUS (Zhang et al., 2020) and T5 (Raffel et al., 2020). EXFORMER is a collection of early Transformer-based summarization models. Typical models that fit into this category include BERTSum (Liu and Lapata, 2019), and GPT-2 (Radford et al., 2019). The remaining models, such as Pointer-Generator (See et al., 2017) and BottomUp (Gehrmann et al., 2018), are instances of OLD. A full description of the models in each category is found in Appendix A.

### 2.2 Benchmark Datasets

The SUMMAC benchmark (Laban et al., 2022) includes six annotated datasets for factual consistency evaluation. We directly include XSumFaith (Maynez et al., 2020), FactCC (Kryscinski et al., 2020), SummEval (Fabbri et al., 2021b), and FRANK (Pagnoni et al., 2021) from SUMMAC in our benchmark. We do not include the CoGenSumm (Falke et al., 2019) dataset as the original task is ranking pairs of generated summaries instead of detecting factually consistent summaries, and pairs of summaries can be both factually consistent or inconsistent. We modify the Polytope (Huang et al., 2020) dataset in SUMMAC where we view summaries annotated with *addition*, *omission* or *duplication* errors as factually consistent since these three error types are not related to factual consistency. We use the validation and test splits from SUMMAC for the above mentioned datasets.

In addition to modifying SUMMAC, we further include four annotated datasets. For Wang'20 (Wang et al., 2020), CLIFF (Cao and Wang, 2021) and Goyal'21 (Goyal and Durrett, 2021), we create data splits based on the parity of indices, following SUMMAC. For Cao'22 (Cao et al., 2022), we use the existing splits from the original work.

**Deduplication and label disagreement correction** Some examples may be labeled for errors in multiple datasets. We removed all duplicates so that each instance appears only once in our benchmark. During this deduplication process, we detected 100 instances of the same summaries that are annotated in different datasets with *different* factual consistency labels. 98 of them are between FRANK and XSumFaith, and 2 of them are between FRANK and SummEval. The authors of this work manually corrected the labels for these examples based on our judgment.

### 2.3 Benchmark Evaluation Metrics

We use balanced accuracy to evaluate the performance of factuality metrics due to the imbalance of factually consistent and inconsistent summaries. We refer readers to Laban et al. (2022) for further

justification of balanced accuracy as the evaluation metric. In each dataset, a factuality metric selects a threshold for FTSOTA, EXFORMER and OLD, respectively, based on the performance on the corresponding validation set. The chosen thresholds convert raw scores from metrics into binary labels for balanced accuracy evaluation. We provide a weighted average of performance across all datasets in the benchmark (see Table 3).

## 3 Comparison of Factuality Metrics

First, we evaluate several SOTA factual consistency metrics on our benchmark, namely **DAE** (Goyal and Durrett, 2020, 2021), **QuestEval** (Scialom et al., 2021), **SummaC-ZS**, **SummaC-Conv** (Laban et al., 2022) and **QAFactEval** (Fabbri et al., 2021c).[2] We also benchmark recent ChatGPT-based evaluation metics from Luo et al. (2023) and Wang et al. (2023). **ChatGPT-ZS** and **ChatGPT-CoT** (Luo et al., 2023) prompt LLMs to directly output a binary factuality decision. On the other hand, **ChatGPT-DA** and **ChatGPT-Star** (Wang et al., 2023) ask LLMs to score the factuality of generated summaries on a scale of 0-100 and 1-5 respectively. More details about these metrics, including exact prompts are included in Appendix B.

**Unifying these metrics** We consider each metric as a function $f(d, s) \rightarrow y$, mapping each (document, summary) pair to a score $y \in \mathbb{R}$. We convert each method into a binary classifier $f'(d, s) \rightarrow \{0, 1\}$ by picking a threshold $t$ such that we predict 1 if $f(d, s) > t$ and 0 otherwise.[3]

All thresholds are set separately for each metric. We consider two ways of setting the threshold for a metric: **threshold-per-dataset** and **single-threshold**. The first setting has thresholds $\{t^m_{d,c}\}$ within each metric for every dataset we consider, where $d, c$ and $m$ are any dataset in $D$, any model category from $C$, and any factuality metric, respectively. This allows one to choose the right metric for the task at hand. The **single-threshold** setting defines one threshold $\{t^m\}$ per metric.

**Threshold Analysis** We analyze scores from factuality metrics using chosen thresholds $\{t^m_{d,c}\}$ from the validation sets. Specifically, for each factuality metric, we average the values of thresholds for each of SOTA, EXFORMER and OLD across all datasets (Figure 1). For all facuality metrics, the average threshold values for AGGREFACT-CNN are greater than those for AGGREFACT-XSUM across all categories. **This discrepancy of threshold values shows that evaluating on both of these datasets with a single threshold is a difficult balancing act and may lead to poor results on at least one dataset.**

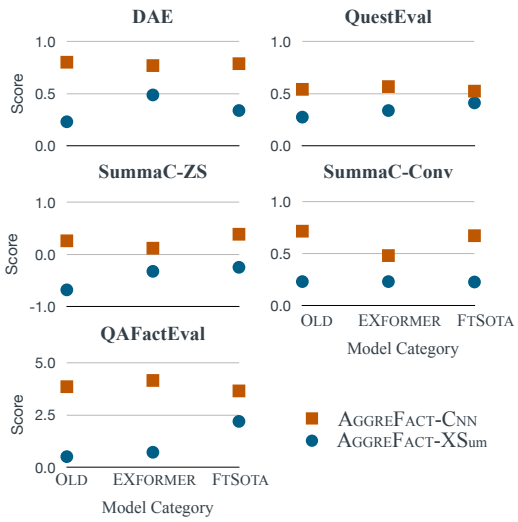


Figure 1: Average threshold values on AGGREFACT-CNN and AGGREFACT-XSUM.

The higher threshold values on CNN/DM are connected to both the nature of the errors involved and overall extractiveness of the summaries XSum summaries are more abstractive and tend to contain a larger number of errors, making it harder for the metrics to verify the consistency of summaries with respect to the source text and resulting in lower scores in general, even for factual cases. For CNN/DM, smaller deviations from the source may indicate non-factuality.

**Binary Classification Results** A weighted average of performance in terms of balanced accuracy for AGGREFACT-CNN and AGGREFACT-XSUM is shown in Table 3.[4] It shows results using both trained metrics (upper half) and ChatGPT-based metrics (bottom half).

Our results show that for AGGREFACT-CNN, both trained and ChatGPT-based factuality metrics achieve the best performance in evaluating the summaries in OLD. This result is intuitive: the sum-

[2] We do not consider other common metrics like ROUGE (Lin, 2004), BLEU (Papineni et al., 2002) or BERTScore (Zhang* et al., 2020) as prior work (Fabbri et al., 2021c) has shown that they have low correlation with factuality.

[3] CHATGPT-ZS and CHATGPT-COT do not require thresholds as they directly predict factual consistency labels.

[4] Dataset-wise comparison between factuality metrics is shown in Appendix Table 8.

| | AGGREFACT-CNN | | | AGGREFACT-XSUM | | |
|---|---|---|---|---|---|---|
| | FTSOTA | EXF | OLD | FTSOTA | EXF | OLD |
| Baseline | 50.0 | 50.0 | 50.0 | 50.0 | 50.0 | 50.0 |
| DAE* | 59.4 | 67.9 | **69.7** | 73.1 | - | - |
| QuestEval | 63.7 | 64.3 | **65.2** | 61.6 | 60.1 | 59.7 |
| SummaC-ZS | 63.3 | **76.5** | 76.3 | **56.1** | 51.4 | 53.3 |
| SummaC-Cv | 70.3 | 69.8 | **78.9** | 67.0 | 64.6 | **67.5** |
| QAFactEval | 61.6 | 69.1 | **80.3** | **65.9** | 59.6 | 60.5 |
| ChatGPT-ZS | 66.2 | 64.5 | **74.3** | 62.6 | **69.2** | 60.1 |
| ChatGPT-CoT | 49.7 | 60.4 | **66.7** | 56.0 | **60.9** | 50.1 |
| ChatGPT-DA | 48.0 | 63.6 | **71.0** | 53.6 | **65.6** | 61.5 |
| ChatGPT-Star | 55.8 | 65.8 | **71.2** | 57.7 | **70.6** | 53.8 |

Table 3: Balanced accuracy on AGGREFACT-CNN and AGGREFACT-XSUM across factuality metrics (threshold-per-dataset setting). A trivial baseline that predicts all examples as factually (in)consistent reports a balanced accuracy of 50%. Since DAE was trained on the human-annotated XSumFaith data (Goyal and Durrett, 2021) including EXFORMER (EXF in table) and OLD summaries, we exclude those results for a fair comparison. Best performing metric across all three categories is highlighted in **bold**, and underlined if it is significantly better than the second best metric according to a paired bootstrap test.

maries in OLD contain obvious errors, such as repetition, that can be more easily detected compared to more nuanced errors made by more recent models. From Table 2, the majority of annotated summaries are generated by models from OLD, so category agnostic performance evaluation will weight these more heavily. **There is a significant performance drop when evaluating the CNN/DM summaries generated by models from EXFORMER or FTSOTA instead.** Approximately a 10% balanced accuracy decrease on average occurs from OLD to FTSOTA. Evaluating on entire datasets, as is standard in prior work, gives us limited information of how these metrics perform on the FTSOTA summaries that are of more interest.

We observe more mixed results for AGGREFACT-XSUM. Here, the trained and ChatGPT-based metrics perform best on FTSOTA and EXFORMER respectively. In fact, the ChatGPT-ZS and ChatGPT-Star metrics report new state-of-the-art results for the EXFORMER category.[5] In the case of AGGREFACT-XSUM also, we advocate for comparing metrics according to such a category-wise view as it provides more information on the most suitable metric to use while evaluating a given category of models.

[5]We found that using different prompts can substantially vary the performance of ChatGPT metrics on both datasets. In our work, we use the exact same prompts as the original papers. Check the prompts in Appendix B.

| | AGGREFACT-CNN-FTSOTA | AGGREFACT-XSUM-FTSOTA |
|---|---|---|
| DAE | $65.4 \pm 4.4$ | $\mathbf{70.2 \pm 2.3}$ |
| QuestEval | $\mathbf{70.2 \pm 3.2}$ | $59.5 \pm 2.7$ |
| SummaC-ZS | $64.0 \pm 3.8$ | $56.4 \pm 1.2$ |
| SummaC-Conv | $61.0 \pm 3.9$ | $65.0 \pm 2.2$ |
| QAFactEval | $67.8 \pm 4.1$ | $63.9 \pm 2.4$ |
| ChatGPT-ZS | $56.3 \pm 2.9$ | $62.7 \pm 1.7$ |
| ChatGPT-COT | $52.5 \pm 3.3$ | $55.9 \pm 2.1$ |
| ChatGPT-DA | $53.7 \pm 3.5$ | $54.9 \pm 1.9$ |
| ChatGPT-Star | $56.3 \pm 3.1$ | $57.8 \pm 0.2$ |

Table 4: Balanced binary accuracy using a single threshold on the FTSOTA subset (single-threshold setting). We show 95% confidence intervals. Highest performance is highlighted in **bold**.

**Binary Classification: FTSOTA** To encourage comparison of factuality metrics on FTSOTA summaries, we provide a separate benchmark which consists of two subsets AGGREFACT-CNN-FTSOTA and AGGREFACT-XSUM-FTSOTA that only consider summaries generated by FTSOTA models. This benchmark consists of validation and test splits from the FTSOTA subsets of the two datasets. This setting allows for comparisons of metrics to be made using only a single threshold.

We show metric comparisons on the FTSOTA subset in Table 4. Note that the ranking of factuality metric here (single-threshold setting) is slightly different from the ranking in Table 3 (threshold-per-dataset setting). For AGGREFACT-CNN-FTSOTA, QuestEval achieves the best performance amongst all metrics. We did not observe a statistically significant improvement over other trained evaluation metrics; however, its improvement over ChatGPT-based metrics is statistically significant. For AGGREFACT-XSUM-FTSOTA, the DAE metric is significantly better than all other metrics.

Interestingly, metrics such as SummaC-Conv, QAFactEval and the recent ChatGPT metrics were all proposed as improved factuality evaluation on the category-agnostic SummaC benchmark (different from the SummaC metric). However, our stratified analysis provides a much clearer picture and shows that **metrics which claim improved performance on SUMMAC do not show similar gains when evaluated on FTSOTA summaries.** We recommend that future work similarly focuses on the SOTA category of generated summaries when comparing factuality metrics.

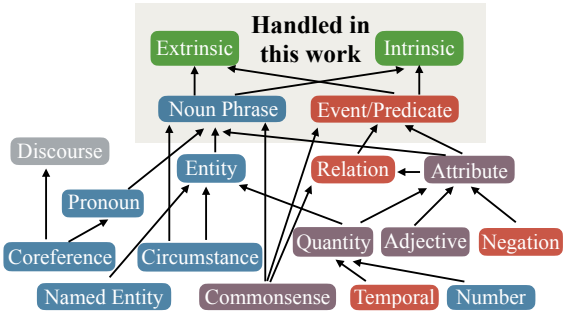


Figure 2: Taxonomy of factual consistency errors. We use unique colors to represent entity- and predicate-related errors, as well as the mix of two. See Appendix C for citations of papers that use each error type.

## 4 Finer-grained Error Analysis

Having established differences among factuality metrics across underlying summarization models, we now explore differences in metrics according to factuality error types. To do this, we need a way to unify error types across datasets in our benchmark and map them into a shared taxonomy.

### 4.1 A Taxonomy of Error Types

We surveyed existing error type taxonomies in prior work and unified the types of factual errors among them into a hierarchical taxonomy in Figure 2. Arrows relate more specific error types to more general "parent" errors. The prior works that make use of each error type can be found in Appendix C. As shown in the figure, most error types related to factual consistency fall under the subset *{intrinsic, extrinsic} × {noun phrase, predicate}* if we consider the coarsest level of the hierarchy. We discard discourse errors as these are uncommon and not present in most of our datasets. Therefore, we consolidate all unique error type taxonomies from all four datasets we consider here into this error type subset (shown in the gray box in Figure 2). Descriptions and examples for these error types are in Table 9. Further, we introduce two additional error categories *{intrinsic-entire sent., extrinsic-entire sent.}* if an entire sentence is annotated as erroneous.

We are able to map four of the datasets (see Section 4.2) in AGGREFACT that contain fine-grained annotations to our unified taxonomy. For all four datasets, if there are multiple annotators, we assign an error type to a summary if the error is annotated by more than one annotator. We allow one summary to have multiple error types. We call the annotated subset related to CNN/DM and XSum as AGGREFACT-CNN-UNIFIED and AGGREFACT-XSUM-UNIFIED, respectively.

### 4.2 Error Mapping

**XSumFaith** XSumFaith consists of 500 summaries each from human reference, two models in OLD, and two models in EXFORMER. All summaries are annotated with intrinsic and extrinsic errors, but no finer categories are distinguished. For error type mapping, we automatically detect predicates in a summary and assign each error span intrinsic- or extrinsic-predicate error if it contains a predicate. We map the remaining error spans to intrinsic- or extrinsic-noun phrase error.

**FRANK** The CNN/DM subset of FRANK consists of three models in OLD, and one model each in both EXFORMER and FTSOTA. The XSum portion of FRANK has two models each in OLD and EXFORMER. Each model contains 250 summaries in the dataset. We mapped Entity error and Out of Article error to extrinsic-noun phrase error; Predicate error and Grammatical error to extrinsic-predicate error; Circumstance error and Coreference error to intrinsic-noun phrase error; and other errors to intrinsic-predicate error.

**Goyal'21** Authors of the original dataset manually identified all hallucinated text spans for each summary and classified hallucination types into {intrinsic, extrinsic} × {entity, event, noun phrase, others}. The dataset consists of summaries for both CNN/DM and XSum. For the CNN/DM susbset, the authors directly annotated 50 summaries from FactCC, where summaries were generated by OLD models. The XSum subset consists of summaries from FTSOTA models. We map entity-related and noun phrase-related errors to noun phrase errors, event errors to predicate errors and others to entire sentence errors.

**CLIFF** This dataset consists of 150 summaries each for both CNN/DM and XSum from two models in FTSOTA. We use the same approach for error mapping as we do for XSumFaith by only considering words labeled as extrinsic or intrinsic errors.

We evaluate the accuracy of our error type mapping via manual inspection. Specifically, the authors of this work inspect 30 factually inconsistent examples each for XSumFaith, FRANK and CLIFF. Those examples cover summaries generated by all

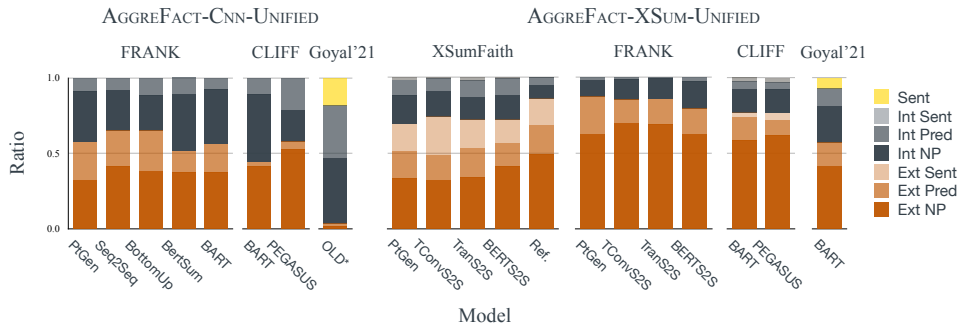


Figure 3: Error types of summaries from AGGREFACT-CNN-UNIFIED and AGGREFACT-XSUM-UNIFIED. Ref. is annotated reference summary from XSumFaith. Since Goyal'21 in AGGREFACT-CNN-UNIFIED annotated summaries from FactCC, we use OLD* to denote summaries generated from OLD models.

models used in the datasets. Results of the manual inspection show that the accuracy of our error type mapping is over 90%.

A common discrepancy noticed by annotators was that in several cases the examples were originally annotated as intrinsic/extrinsic but we believe those errors are extrinsic/intrinsic. These cases are not a result of error in our mapping, but instead disagreement or error in the original annotation itself. For error mapping, we found out mapping of FRANK to be least accurate among all 4 datasets. For example, we found that the entity error (EntE) can be either intrinsic or extrinsic even though FRANK explicitly defines an extrinsic error type, i.e. "out of article" error. For Goyal'21, we manually correct any mapping errors that occur in the 150 examples. Corrections mostly happen for the event-related error defined in Goyal'21 which can be either noun phrase- or predicate-related.

### 4.3 Distribution Shift of Error Types

Next, we explore how the number of errors in specific groups of models from FTSOTA, EXFORMER, and OLD has changed with the progress in the field. Specifically, for each of the FRANK, XSumFaith, Goyal'21, and CLIFF datasets, we calculate the ratio of error types from factually inconsistent summaries generated by each model. We then study any distribution shift of error types in AGGREFACT-CNN-UNIFIED and AGGREFACT-XSUM-UNIFIED under FTSOTA, EXFORMER, and OLD.

**Summaries generated by the same models consist of different error distributions over different annotated datasets.** As shown in AGGREFACT-XSUM-UNIFIED (Figure 3), BART summaries are annotated by both Goyal'21 and CLIFF. However, it is interesting that BART summaries were annotated as making more intrinsic-noun phrase and intrinsic-predicate errors in Goyal'21 but more extrinsic-noun phrase errors in CLIFF. Similar observations can be found in AGGREFACT-CNN-UNIFIED, where BART summaries have a higher proportion of extrinsic-predicate error in FRANK and more intrinsic-noun phrase error in CLIFF.

In addition, although XSumFaith and FRANK annotate the same set of model generated summaries in AGGREFACT-XSUM-UNIFIED, the distribution of error types looks dramatically different. The main discrepancy lies in the proportion of extrinsic-noun phrase and intrinsic-predicate errors. There are two possible reasons for such discrepancy. First, FRANK does not have "entire sent." errors based on our conversation of its annotation schema to the unified taxonomy (Section 4.2). Second, and more important, it is not easy to map error types from FRANK directly to our unified error types in spite of our validation. For example, the "out of article error" in FRANK is defined as an error where some statements in the summary do not show up in the source text. We found this error can be mapped to either an extrinsic-noun phrase error or extrinsic-predicate error. These observations indicate that **previous work disagrees about where the individual error class boundaries are, even when aligned with our taxonomy**.

**A combined meta-analysis shows shifts in error distributions.** Figure 3 shows that error type distribution can vary among models from the same category. For example, summaries from BART contain a higher ratio of intrinsic-noun phrase errors than PEGASUS in AGGREFACT-CNN-UNIFIED. We now combine all datasets together from AGGREFACT-CNN-UNIFIED and AGGREFACT-XSUM-UNIFIED and show the uni-

| | AGGREFACT-CNN-ERROR | | | | AGGREFACT-XSUM-ERROR | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Intrinsic | | Extrinsic | | Intrinsic | | | Extrinsic | | |
| | NP (183) | Pred. (60) | NP (220) | Pred. (129) | NP (196) | Pred. (113) | Sent (17) | NP (434) | Pred. (181) | Sent (197) |
| DAE* | 59.6 | 53.3 | 67.7 | 62.8 | - | - | - | - | - | - |
| QuestEval | 62.8 | 50.0 | 72.3 | 68.2 | 33.2 | 44.2 | 64.7 | 40.6 | 50.3 | 69.0 |
| SummacZS | 66.1 | **71.7** | **81.8** | 72.1 | 50.0 | 57.5 | 76.5 | 48.6 | 47.5 | 36.0 |
| SummacConv | 62.8 | **65.0** | 76.4 | 59.7 | 54.1 | 62.8 | 29.4 | 64.5 | 60.8 | 70.6 |
| QAFactEval | 56.3 | 51.7 | **79.1** | 63.6 | 66.8 | 75.2 | **88.2** | 55.1 | 70.2 | 79.2 |
| ChatGPT-ZS | 56.3 | 45.0 | 63.2 | 52.7 | **83.2** | 85.8 | **94.1** | **74.2** | 83.4 | **93.9** |
| ChatGPT-COT | 54.1 | 60.0 | 61.8 | 52.7 | **83.2** | **91.2** | **94.1** | **77.2** | **89.5** | **91.9** |
| ChatGPT-DA | 65.0 | **73.3** | 71.8 | 67.4 | 55.6 | 67.3 | **94.1** | 53.7 | 65.7 | 67.5 |
| ChatGPT-Star | 65.0 | 68.2 | 68.2 | 56.6 | 66.8 | 73.5 | **94.1** | 64.7 | 74.6 | 75.1 |

Table 5: Recall of factually incorrect summaries that contain certain error types (number of such summaries shown in parenthesis). Binary labels are directly obtained from AGGREFACT-CNN and AGGREFACT-XSUM. We obtain 95% confidence intervals and numbers in **bold** indicates that models have significantly higher recall of identifing certain error types compared to the rest of of the metrics. Since DAE is trained with human annotated data from XSumFaith, we remove DAE for a fair comparison in XSum error types.

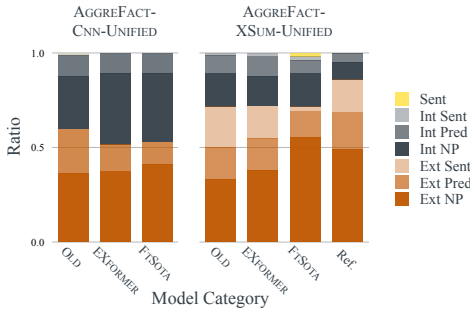


Figure 4: Distribution shift of error types on AGGREFACT-CNN-UNIFIED and AGGREFACT-XSUM-UNIFIED. Ref. is human reference from XSumFaith.

fied error distributions over three model categories.[6] As shown in Figure 4, models make approximately 50% extrinsic errors in CNN/DM, with a slightly decrease from OLD to more recent models. For XSum, the proportion of extrinsic errors remains unchanged and is at 70%.

### 4.4 Error Detection by Type

In this section, we analyze how factuality metrics perform on summaries that contain certain error types. Specifically, we collect subsets of examples from four annotated datasets and group them into AGGREFACT-CNN-ERROR and AGGREFACT-XSUM-ERROR.[7] Every subset contains summaries that **include only one error type** defined in Section 4.1. Each factuality metric assigns a binary label to an instance obtained directly from AGGREFACT-CNN and AGGREFACT-XSUM. Note that each subset only consists of test set examples from our benchmark since examples from the validation set were used to choose the optimal thresholds (Section 3). Since there are limited annotations for each model category after only considering examples from the test set of the benchmark, we decide not to split data by model categories in this part of the analysis. We calculate the recall of identifying error types from those subsets and show the results in Table 5. Summaries in AGGREFACT-CNN-ERROR and AGGREFACT-XSUM-ERROR primarily come from non-FTSOTA models (89.6% and 92.1%, respectively). On AGGREFACT-CNN-ERROR, where 79.0% of summaries were generated from OLD, there are more extrinsic errors (349) than intrinsic errors (243). This agrees with our above analysis that also shows that errors in generated summaries from less recent models are more likely to be extrinsic (Figure 4).

Across both AGGREFACT-CNN-ERROR and AGGREFACT-XSUM-ERROR, we found that recent metrics like SummaC-Conv, QAFactEval and ChatGPT-based achieve higher recall for most error types. This indicates that **more recent factuality metrics are better at capturing obvious errors generated by less recent models.** This mirrors our earlier finding in Table 3 (column EXFORMER and OLD). Interestingly, we find that **summarization datasets (CNN/DM and XSum) have a non-negligible effect on the metrics' capabilities of**

[6]For AGGREFACT-XSUM-UNIFIED, since XSumFaith and FRANK annotated the same set of summaries, we only use the annotation results from XSumFaith since our error mapping is more accurate on the span-level annotations.

[7]We exclude FRANK for this analysis for the same reason as in Section 4.3.

**detecting certain error types, even in the cases of out-of-date errors.** For example, the recall of identifying extrinsic-noun phrase error drops 10-30% across all trained factuality metrics when evaluated on XSUM, compared to CNN/DM. Similarly, ChatGPT metrics report 20-30% higher recall on CNN/DM, compared to its XSUM counterparts.

Another observation is that although DAE is trained using annotations from XSumFaith, which provides supervision for multiple error types, it does not identify errors as well in AGGREFACT-CNN-ERROR. These findings indicate that **summarization models make fundamentally different errors for each error type, and current factuality metrics cannot be uniformly good at identifying certain error types across datasets.** We believe this conclusion still holds when evaluating metrics on summaries generated from FTSOTA models since they generate less obvious errors.

## 5 Recommendations

**Evaluate factuality models on modern summarization systems** We have seen that FTSOTA yields significantly different results than EXFORMER or OLD. Because of the prevalence of these systems, we believe that any new work should prefer evaluating on these SOTA summaries.

Particularly for factuality metrics that are either based on latest LLMs or on pre-trained models, evaluating on modern summarization systems is needed to see if these metrics are actually improving from the current state-of-the-art or merely patching errors in outdated systems that have already been fixed by other advances.

**Annotate factual consistency errors from summaries generated by LLMs** Recent work (Goyal et al., 2022) shows that LLMs like GPT-3 are capable of generating summaries that are preferred over FTSOTA summaries by human annotators. Furthermore, they show that existing factuality metrics cannot reliably detect errors in summaries from GPT-3 models as these latter summaries differ substantially from existing benchmarks and training sets. We encourage future work to annotate errors from LLM-generated summaries and evaluate new factual consistency metrics on this set as well in addition to the FTSOTA set. As such, we believe that future work should construct "living" benchmarks for factuality evaluation that are consistently updated as more powerful summarization systems are introduced.

**Choose the right metric for the job** We note that there is no one clear winner among the metrics evaluated here (Section 3). Depending on the downstream application, different methods may be more or less appropriate, as our analysis shows. Moreover, none of current factuality metrics can identify certain error types across datasets equally well. As QG/QA and NLI models get better, we expect all of these methods to improve further. Alternatively, although recent ChatGPT-based metrics (Luo et al., 2023; Wang et al., 2023) do not perform well on modern summarization systems, they can be a starting point for leveraging LLMs to perform factual consistency evaluation.

**Use more consistent error types** With our taxonomy, we have mapped error types annotated in previous work. It is relatively easier and more accurate to map errors from XSumFaith, Goyal'21, and CLIFF to our unified error types as they have annotation granularity finer than sentence-level. We encourage future work to follow this taxonomy where possible and leverage definitions in prior work to make *cross-dataset* comparisons possible. Here also, we encourage future work to prioritize annotation and evaluation of SOTA summaries.

**Annotate and evaluate on non-news datasets** Most of current annotated datasets are within the news domain and factuality metrics are evaluated on news summaries accordingly. As there is a rising interest in other domains such as dialogue summarization (Tang et al., 2022; Fabbri et al., 2021a; Zhang et al., 2021), and medical evidence summarization (Tang et al., 2023), future work could annotate and analyze errors made by SOTA models there. We encourage future work to develop factuality metrics that have superior performance over cross-domain evaluation.

## 6 Conclusion

In this work, we analyzed several factuality metrics across a large meta-benchmark assembled from existing datasets. We find state-of-the-art fine-tuned summarization models still present challenges for detecting factual errors, and the performance of error detectors is often overestimated due to the reliance on older datasets. Furthermore, we unify existing datasets into a common taxonomy and use this to highlight differences between datasets and summarization models, as well as the complexity of unifying concepts in this problem space.

## Limitations

There are a few limitations of our work. First, we focus on evaluating state-of-the-art factuality metrics on English newswire datasets. This setting restricts us to English-language data, a formal style of text, and topics consisting of what is discussed in US and UK-centric news sources. Moreover, other summarization domains such as dialogue summarization have different common error types such as *wrong reference error* (Tang et al., 2022), which are not fully evaluated under current metrics. As settings like this are studied in future work, we believe that the kinds of analysis we do here can be extended to these settings as well.

Second, since our work is built on top of previous work, some analysis such as the error type mapping is limited by the quality and annotation agreement from previous work. We chose not to undertake large-scale reannotation to avoid causing confusion in the literature with multiple versions of datasets reflecting divergent annotator opinions. In spite of these limitations, we believe that our re-evaluation of these metrics and the analysis of error types under newswire data can bring insights for future works in choosing, designing and evaluating factuality metrics.

## Acknowledgments

The UT Austin team on this work was supported by a gift from Salesforce Inc., NSF Grant IIS-1814522, and a gift from Amazon.

## References

Meng Cao, Yue Dong, and Jackie Cheung. 2022. Hallucinated but factual! inspecting the factuality of hallucinations in abstractive summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3340–3354, Dublin, Ireland. Association for Computational Linguistics.

Meng Cao, Yue Dong, Jiapeng Wu, and Jackie Chi Kit Cheung. 2020. Factual error correction for abstractive summarization models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6251–6258, Online. Association for Computational Linguistics.

Shuyang Cao and Lu Wang. 2021. CLIFF: Contrastive learning for improving faithfulness and factuality in abstractive summarization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6633–6649, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Sihao Chen, Fan Zhang, Kazoo Sone, and Dan Roth. 2021. Improving faithfulness in abstractive summarization with contrast candidate generation and selection. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5935–5941, Online. Association for Computational Linguistics.

Yen-Chun Chen and Mohit Bansal. 2018. Fast abstractive summarization with reinforce-selected sentence rewriting. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 675–686, Melbourne, Australia. Association for Computational Linguistics.

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, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Yue Dong, Yikang Shen, Eric Crawford, Herke van Hoof, and Jackie Chi Kit Cheung. 2018. BanditSum: Extractive summarization as a contextual bandit. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3739–3748, Brussels, Belgium. Association for Computational Linguistics.

Esin Durmus, He He, and Mona Diab. 2020. FEQA: A question answering evaluation framework for faithfulness assessment in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5055–5070, Online. Association for Computational Linguistics.

Alexander Fabbri, Faiaz Rahman, Imad Rizvi, Borui Wang, Haoran Li, Yashar Mehdad, and Dragomir Radev. 2021a. ConvoSumm: Conversation summarization benchmark and improved abstractive summarization with argument mining. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6866–6880, Online. Association for Computational Linguistics.

Alexander R. Fabbri, Wojciech Kryściński, Bryan McCann, Caiming Xiong, Richard Socher, and Dragomir Radev. 2021b. SummEval: Re-evaluating summarization evaluation. *Transactions of the Association for Computational Linguistics*, 9:391–409.

Alexander R. Fabbri, Chien-Sheng Wu, Wenhao Liu, and Caiming Xiong. 2021c. Qafacteval: Improved qa-based factual consistency evaluation for summarization.

Tobias Falke, Leonardo F. R. Ribeiro, Prasetya Ajie Utama, Ido Dagan, and Iryna Gurevych. 2019. Ranking generated summaries by correctness: An interesting but challenging application for natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2214–2220, Florence, Italy. Association for Computational Linguistics.

Sebastian Gehrmann, Yuntian Deng, and Alexander Rush. 2018. Bottom-up abstractive summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4098–4109, Brussels, Belgium. Association for Computational Linguistics.

Tanya Goyal and Greg Durrett. 2020. Evaluating factuality in generation with dependency-level entailment. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3592–3603, Online. Association for Computational Linguistics.

Tanya Goyal and Greg Durrett. 2021. Annotating and modeling fine-grained factuality in summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1449–1462, Online. Association for Computational Linguistics.

Tanya Goyal, Junyi Jessy Li, and Greg Durrett. 2022. News summarization and evaluation in the era of gpt-3. *arXiv preprint arXiv:2209.12356*.

Han Guo, Ramakanth Pasunuru, and Mohit Bansal. 2018. Soft layer-specific multi-task summarization with entailment and question generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 687–697, Melbourne, Australia. Association for Computational Linguistics.

Karl Moritz Hermann, Tomás Kočiský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching Machines to Read and Comprehend. In *Proceedings of the Conference on Neural Information Processing Systems (NeurIPS)*.

Wan-Ting Hsu, Chieh-Kai Lin, Ming-Ying Lee, Kerui Min, Jing Tang, and Min Sun. 2018. A unified model for extractive and abstractive summarization using inconsistency loss. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 132–141, Melbourne, Australia. Association for Computational Linguistics.

Dandan Huang, Leyang Cui, Sen Yang, Guangsheng Bao, Kun Wang, Jun Xie, and Yue Zhang. 2020. What have we achieved on text summarization? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 446–469, Online. Association for Computational Linguistics.

Yichen Jiang and Mohit Bansal. 2018. Closed-book training to improve summarization encoder memory. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4067–4077, Brussels, Belgium. Association for Computational Linguistics.

Wojciech Kryscinski, Bryan McCann, Caiming Xiong, and Richard Socher. 2020. Evaluating the factual consistency of abstractive text summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9332–9346, Online. Association for Computational Linguistics.

Wojciech Kryściński, Romain Paulus, Caiming Xiong, and Richard Socher. 2018. Improving abstraction in text summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1808–1817, Brussels, Belgium. Association for Computational Linguistics.

Philippe Laban, Tobias Schnabel, Paul N. Bennett, and Marti A. Hearst. 2022. SummaC: Re-Visiting NLI-based Models for Inconsistency Detection in Summarization. *Transactions of the Association for Computational Linguistics*, 10:163–177.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

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

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3730–3740, Hong Kong, China. Association for Computational Linguistics.

Zheheng Luo, Qianqian Xie, and Sophia Ananiadou. 2023. Chatgpt as a factual inconsistency evaluator for abstractive text summarization. *arXiv preprint arXiv:2303.15621*.

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

Rada Mihalcea and Paul Tarau. 2004. TextRank: Bringing order into text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language*

*Processing*, pages 404–411, Barcelona, Spain. Association for Computational Linguistics.

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2017. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, page 3075–3081. AAAI Press.

Feng Nan, Ramesh Nallapati, Zhiguo Wang, Cicero Nogueira dos Santos, Henghui Zhu, Dejiao Zhang, Kathleen McKeown, and Bing Xiang. 2021a. Entity-level factual consistency of abstractive text summarization. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2727–2733, Online. Association for Computational Linguistics.

Feng Nan, Cicero Nogueira dos Santos, Henghui Zhu, Patrick Ng, Kathleen McKeown, Ramesh Nallapati, Dejiao Zhang, Zhiguo Wang, Andrew O. Arnold, and Bing Xiang. 2021b. Improving factual consistency of abstractive summarization via question answering. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6881–6894, Online. 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.

Artidoro Pagnoni, Vidhisha Balachandran, and Yulia Tsvetkov. 2021. Understanding factuality in abstractive summarization with FRANK: A benchmark for factuality metrics. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4812–4829, Online. Association for Computational Linguistics.

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

Ramakanth Pasunuru and Mohit Bansal. 2018. Multi-reward reinforced summarization with saliency and entailment. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 646–653, New Orleans, Louisiana. Association for Computational Linguistics.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language Models are Unsupervised Multitask Learners. *OpenAI Blog*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Thomas Scialom, Paul-Alexis Dray, Sylvain Lamprier, Benjamin Piwowarski, Jacopo Staiano, Alex Wang, and Patrick Gallinari. 2021. QuestEval: Summarization asks for fact-based evaluation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6594–6604, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Liyan Tang, Zhaoyi Sun, Betina Idnay, Jordan G Nestor, Ali Soroush, Pierre A. Elias, Ziyang Xu, Ying Ding, Greg Durrett, Justin Rousseau, Chunhua Weng, and Yifan Peng. 2023. Evaluating large language models on medical evidence summarization.

Xiangru Tang, Arjun Nair, Borui Wang, Bingyao Wang, Jai Desai, Aaron Wade, Haoran Li, Asli Celikyilmaz, Yashar Mehdad, and Dragomir Radev. 2022. CONFIT: Toward faithful dialogue summarization with linguistically-informed contrastive fine-tuning. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5657–5668, Seattle, United States. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Alex Wang, Kyunghyun Cho, and Mike Lewis. 2020. Asking and answering questions to evaluate the factual consistency of summaries. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5008–5020, Online. Association for Computational Linguistics.

Jiaan Wang, Yunlong Liang, Fandong Meng, Haoxiang Shi, Zhixu Li, Jinan Xu, Jianfeng Qu, and Jie Zhou. 2023. Is chatgpt a good nlg evaluator? a preliminary study. *arXiv preprint arXiv:2303.04048*.

Yuxiang Wu and Baotian Hu. 2018. Learning to extract coherent summary via deep reinforcement learning. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'18/IAAI'18/EAAI'18. AAAI Press.

Zhiyuan Zeng, Jiaze Chen, Weiran Xu, and Lei Li. 2021. Gradient-based adversarial factual consistency evaluation for abstractive summarization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4102–4108, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, pages 11328–11339. PMLR.

Shiyue Zhang, Asli Celikyilmaz, Jianfeng Gao, and Mohit Bansal. 2021. EmailSum: Abstractive email thread summarization. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6895–6909, Online. Association for Computational Linguistics.

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

Yuhao Zhang, Derek Merck, Emily Tsai, Christopher D. Manning, and Curtis Langlotz. 2020. Optimizing the factual correctness of a summary: A study of summarizing radiology reports. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5108–5120, Online. Association for Computational Linguistics.

Zheng Zhao, Shay B. Cohen, and Bonnie Webber. 2020. Reducing quantity hallucinations in abstractive summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2237–2249, Online. Association for Computational Linguistics.

Qingyu Zhou, Nan Yang, Furu Wei, Shaohan Huang, Ming Zhou, and Tiejun Zhao. 2018. Neural document summarization by jointly learning to score and select sentences. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 654–663, Melbourne, Australia. Association for Computational Linguistics.

## A Model Categories

In this section, we briefly describe the summarization models we use in this paper.

For FTSOTA, we include Transformer-based pre-trained models like BART (Lewis et al., 2020), T5 (Raffel et al., 2020), and PEGASUS (Zhang et al., 2020). They are pre-trained on massive text corpus and further fine-tuned on summarization datasets.

For EXFORMER, we use BERTSumExt and BERTSumAbs from Liu and Lapata (2019), GPT-2 (Radford et al., 2019), TransS2S (Vaswani et al., 2017), and BERTS2S (Devlin et al., 2019).

For OLD, we include models FastAbsRl (Chen and Bansal, 2018), TConvS2S (Narayan et al., 2018), BottomUp (Gehrmann et al., 2018), PGNet (See et al., 2017), NeuSUM (Zhou et al., 2018), BanditSum (Dong et al., 2018), SummaRuNNer (Nallapati et al., 2017), TextRank (Mihalcea and Tarau, 2004), CBDec (Jiang and Bansal, 2018), RNES (Wu and Hu, 2018), ROUGESal (Pasunuru and Bansal, 2018), ImproveAbs (Kryściński et al., 2018), MultiTask (Guo et al., 2018), and UnifiedExtAbs (Hsu et al., 2018).

## B Factuality Metrics

We show the descriptions of consistency metrics we considered in our benchmark.

**DAE** (Goyal and Durrett, 2020) propose an arc entailment approach that evaluates the factuality $F_a(a, x) = P(\text{entailment} \mid a, x)$ of each dependency arc $a \in \text{Arc}(s)$ of the generated summary $s$ independently with respect to the input article $x$. It then uses their aggregation $\frac{1}{|\text{Arc}(s)|}\sum_{a\in\text{Arc}(s)} F_a(a, x)$ as the overall score. We use the default model and hyperparameters provided by the authors,[8] described in Goyal and Durrett (2021), which is trained on data from XSum-Faith, which we account for later in our comparisons.

**QuestEval** (Scialom et al., 2021) propose a QA-based metric that aggregates answer overlap scores from selected spans $r$ and questions $q_i \in Q_G(x)$ that derived from the input article $x$ and answered $Q_A(s, q_i)$ using the summary $s$ (recall-based); and those derived from the summary $q_i \in Q_G(s)$ and answered $Q_A(x, q_i)$ using the input article $x$ (precision-based). $Q_G$ and $Q_A$ denote question generation and question answering components, respectively. We use the implementation provided by the authors[9] and apply the unweighted version of

[8] https://github.com/tagoyal/factuality-datasets

[9] https://github.com/ThomasScialom/QuestEval

the metric as in Laban et al. (2022).

**SummaC-ZS** (Laban et al., 2022) is a zero-shot entailment metric that computes a sentence-level entailment score $F(s_i, x_j)$ between each summary sentence $s_i$ and input sentence $x_j$ using an NLI model $F$. It first find the maximum entailment score $\text{score}(s_i) = \max_j F(s_i, x_j)$ for each summary sentence $s_i$, and averaging over all summary sentences for the final score $\frac{1}{|s|}\sum_i \text{score}(s_i)$. We use the default model and hyperparameters provided by the authors, which may return a negative score.

**SummaC-Conv** (Laban et al., 2022) extends SummaC-ZS by replacing the max operation with a binning of the entailment scores between each summary sentence $s_i$ and all input sentences $x_j$ to create a histogram $\text{hist}(s_i, x)$. The histogram is then passed through a learned 1-D convolution layer Conv to produce the summary sentence score $\text{score}(s_i) = \text{Conv}(\text{hist}(s_i, x))$. Parameters for the convolution layer are learned on synthetic data from FactCC (Kryscinski et al., 2020).

**QAFactEval** (Fabbri et al., 2021c) is a QA-based metric analogous to the precision-based component of QuestEval and includes optimized question answering, generation, and answer-overlap components. We do not make use of the variation of QAFactEval which combines QA and entailment-based scores into a single metric.

**ChatGPT-ZS** (Luo et al., 2023) uses a zero-shot template and directly asks for a binary label of summary factuality.

*Decide if the following summary is consistent with the corresponding article. Note that consistency means all information in the summary is supported by the article.*
*Article: [Article]*
*Summary: [Summary]*
*Answer (yes or no):*

**ChatGPT-CoT** (Luo et al., 2023) also uses a zero-shot template but invokes chain-of-thought (CoT) style reasoning in its prompt. Similar to ChatGPT-ZS, it directly asks for a binary factuality label for a given summary.

*Decide if the following summary is consistent with the corresponding article. Note that consistency means all information in the summary is supported by the article.*
*Article: [Article]*
*Summary: [Summary]*
*Explain your reasoning step by step then answer (yes or no) the question:*

**ChatGPT-DA** (Wang et al., 2023) uses a direct assessment (DA) prompt template that asks to assign a factual consistency score to a summary on a continuous scale from 0 to 100.

*Score the following news summarization given the corresponding news with respect to consistency on a continuous scale from 0 to 100, where a score of zero means "inconsistency" and score of one hundred means "perfect consistency". Note that consistency measures whether the facts in the summary are consistent with the facts in the original article. Consider whether the summary does reproduce all facts accurately and does not make up untrue information.*
*Article: [Article]*
*Summary: [Summary]*
*Scores:*

**ChatGPT-Star** (Wang et al., 2023) is an alternative version of ChatGPT-DA that asks LLMs to score summaries on a scale of one-to-five.

*Score the following news summarization given the corresponding news with respect to consistency with one to five stars, where one star means "inconsistency" and five stars means "perfect consistency". Note that consistency measures whether the facts in the summary are consistent with the facts in the original article. Consider whether the summary does reproduce all facts accurately and does not make up untrue information.*
*Article: [Article]*
*Summary: [Summary]*
*Stars:*

## C Surveyed Error Types

Here are our surveyed error types that are related to factual inconsistency.

**Negation Error** (Zhang et al., 2020; Kryscinski et al., 2020; Huang et al., 2020; Zeng et al., 2021)

**Adjective Error** (Zhang et al., 2020)

**Coreference Error** (Zhang et al., 2020; Kryscinski et al., 2020; Pagnoni et al., 2021; Nan et al., 2021b)

**Number error** (Kryscinski et al., 2020; Nan et al., 2021b; Chen et al., 2021; Cao et al., 2020)

**Entity error** (Kryscinski et al., 2020; Pagnoni et al., 2021; Zeng et al., 2021; Wang et al., 2020; Nan et al., 2021b,a; Chen et al., 2021; Cao et al., 2020)

**Attribute error** (Pagnoni et al., 2021; Huang et al., 2020)

**Pronoun error** (Kryscinski et al., 2020; Zeng et al., 2021; Cao et al., 2020)

**Commonsense error** (Kryscinski et al., 2020)

**Temporal error** (Kryscinski et al., 2020; Cao et al., 2020)

**Predicate error** (Pagnoni et al., 2021)

**Discourse link Error** (Pagnoni et al., 2021)

**Relation error** (Nan et al., 2021a,b)

**Quantity error** (Zhao et al., 2020)

**Event error** (Goyal and Durrett, 2021),

**Noun phrase error** (Wang et al., 2020; Goyal and Durrett, 2021),

**Circumstance error** (Pagnoni et al., 2021)

| | | Polytope | FactCC | SummEval | FRANK | Wang'20 | CLIFF | Goyal'21 | Total |
|---|---|---|---|---|---|---|---|---|---|
| OLD | val | 450 | 931 | 550 | 223 | 118 | - | 25 | 2297 |
| | test | 450 | 503 | 548 | 523 | 117 | - | 25 | 2166 |
| XFORMER | val | 150 | - | 50 | 75 | - | - | - | 275 |
| | test | 150 | - | 50 | 175 | - | - | - | 375 |
| SOTA | val | 34 | - | 200 | 75 | - | 150 | - | 459 |
| | test | 34 | - | 200 | 175 | - | 150 | - | 559 |

Table 6: Statistics of AGGREFACT-CNN. Each dataset is stratified into three categories OLD, EXFORMER, and FTSOTA.

| | | XsumFaith | Wang'20 | CLIFF | Goyal'21 | Cao'22 | Total |
|---|---|---|---|---|---|---|---|
| OLD | val | 500 | - | - | - | - | 500 |
| | test | 430 | - | - | - | - | 430 |
| XFORMER | val | 500 | - | - | - | - | 500 |
| | test | 423 | - | - | - | - | 423 |
| SOTA | val | - | 120 | 150 | 50 | 457 | 777 |
| | test | - | 119 | 150 | 50 | 239 | 558 |

Table 7: Statistics of AGGREFACT-XSUM.

| | | | | **Factuality Metric** | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | SummaC | | | ChatGPT | | | |
| | **Dataset** | **Category** | **Count** | DAE | QuestEval | ZS | Conv | QAFactEval | ZS | COT | DA | Star |
| CNN /DM | FactCC | OLD | 503 | 0.704 | 0.655 | 0.835 | **0.891** | 0.843 | 0.793 | 0.697 | 0.686 | 0.743 |
| | Wang'20 | OLD | 117 | 0.586 | 0.552 | 0.655 | 0.672 | **0.754** | **0.758** | 0.599 | 0.695 | 0.652 |
| | SummEval | OLD | 548 | 0.661 | 0.649 | 0.773 | 0.801 | **0.814** | 0.735 | 0.680 | 0.735 | 0.713 |
| | | EXFORMER | 50 | 0.760 | 0.680 | 0.620 | 0.580 | 0.740 | 0.720 | 0.740 | **0.820** | 0.760 |
| | | FTSOTA | 200 | 0.452 | 0.649 | 0.622 | **0.827** | 0.652 | 0.783 | 0.401 | 0.453 | 0.568 |
| | Polytope | OLD | 450 | 0.779 | 0.687 | 0.802 | 0.791 | **0.824** | 0.768 | 0.695 | 0.741 | 0.752 |
| | | EXFORMER | 150 | 0.774 | 0.733 | **0.970** | 0.811 | 0.726 | 0.693 | 0.632 | 0.713 | 0.740 |
| | | FTSOTA | 34 | 0.294 | 0.176 | **0.971** | 0.735 | 0.324 | **0.941** | 0.735 | 0.206 | 0.412 |
| | FRANK | OLD | 523 | 0.704 | 0.669 | 0.692 | 0.728 | **0.773** | 0.694 | 0.628 | 0.695 | 0.672 |
| | | EXFORMER | 175 | 0.574 | 0.556 | 0.631 | 0.634 | **0.646** | 0.583 | 0.540 | 0.517 | 0.558 |
| | | FTSOTA | 175 | **0.699** | 0.626 | 0.570 | 0.601 | 0.547 | 0.519 | 0.514 | 0.523 | 0.531 |
| | Goyal'21 | OLD | 25 | 0.188 | 0.146 | 0.375 | 0.354 | 0.271 | 0.375 | 0.417 | **0.500** | **0.479** |
| | CLIFF | FTSOTA | 150 | 0.730 | **0.740** | 0.646 | 0.649 | 0.716 | 0.603 | 0.550 | 0.528 | 0.612 |
| XSum | Wang'20 | FTSOTA | 119 | **0.756** | 0.560 | 0.698 | 0.721 | **0.756** | 0.608 | 0.514 | 0.533 | 0.620 |
| | Cao'22 | FTSOTA | 239 | **0.723** | 0.601 | 0.490 | 0.668 | 0.613 | 0.643 | 0.576 | 0.502 | 0.530 |
| | XSumFaith | OLD | 430 | - | 0.597 | 0.533 | **0.675** | 0.605 | 0.601 | 0.501 | 0.615 | 0.538 |
| | | EXFORMER | 423 | - | 0.601 | 0.514 | 0.646 | 0.596 | **0.692** | 0.609 | 0.656 | 0.706 |
| | Goyal'21 | FTSOTA | 50 | 0.644 | **0.814** | 0.466 | 0.552 | 0.754 | 0.581 | 0.585 | 0.597 | 0.666 |
| | CLIFF | FTSOTA | 150 | **0.754** | 0.619 | 0.596 | 0.668 | 0.613 | 0.643 | 0.576 | 0.502 | 0.530 |

Table 8: Dataset-wise comparsion between factuality metrics. Since DAE is trained with human annotated data from XsumFaith, we remove DAE for a fair comparison. The best performance is highlighted in **bold** for each row.

| Error Type | Definition | Example of Generated Summaries |
|---|---|---|
| Intrinsic-Noun Phrase | A model misrepresents word(s) from the source text that function(s) in a summary as subject, object, or prepositional object. | The world's first subsea power hub which uses a lithium-based drive system to generate electricity is being tested off the west coast of orkney. |
| Intrinsic-Predicate | A model misrepresents word(s) from the source text that function(s) in a summary as the main content verb or content like adverbs that closely relate to the verb. | A conservative mp has resigned from his constituency as part of an investigation into a # 10.25 m loan to a football club. |
| Extrinsic-Noun Phrase | A model introduces word(s) not from the source text that function(s) in a summary as subject, object, or prepositional object but cannot be verified from the source. | Shale gas drilling in lancashire has been suspended after a magnitude-7.5 earthquake struck. |
| Extrinsic-Predicate | A model introduces word(s) not from the source text that function(s) in a summary as the main content verb or content like adverbs that closely relate to the verb, but which cannot be verified from the source. | Folate - also known as folic acid - should be added to flour in the uk, according to a new study. |

Table 9: Definition and examples of unified error types. Factually inconsistent spans are highlighted in red.

## ACL 2023 Responsible NLP Checklist

### A For every submission:

☐ A1. Did you describe the limitations of your work?
*Left blank.*

☐ A2. Did you discuss any potential risks of your work?
*Left blank.*

☐ A3. Do the abstract and introduction summarize the paper's main claims?
*Left blank.*

☐ A4. Have you used AI writing assistants when working on this paper?
*Left blank.*

### B ☐ Did you use or create scientific artifacts?

*Left blank.*

☐ B1. Did you cite the creators of artifacts you used?
*Left blank.*

☐ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
*Left blank.*

☐ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
*Left blank.*

☐ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
*Left blank.*

☐ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
*Left blank.*

☐ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
*Left blank.*

### C ☐ Did you run computational experiments?

*Left blank.*

☐ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
*Left blank.*

---

*The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.*

☐ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
*Left blank.*

☐ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
*Left blank.*

☐ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
*Left blank.*

**D ☐ Did you use human annotators (e.g., crowdworkers) or research with human participants?**

*Left blank.*

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
*Left blank.*

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
*Left blank.*

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
*Left blank.*

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
*Left blank.*

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
*Left blank.*