Article

Metric Ensembles for Hallucination Detection

Grant C. Forbes[1], **Parth Katlana** [2] **and Zeydy Ortiz** [3]

[1] North Carolina State University; gforbes@ncsu.edu
[2] North Carolina State University; pkatlan@ncsu.edu
[3] DataCrunch Lab, LLC; zortiz@datacrunchlab.com

Abstract: Abstractive text summarization has garnered increased interest as of late, in part due to the proliferation of large language models (LLMs). One of the most pressing problems related to generation of abstractive summaries is the need to reduce "hallucinations," information that was not included in the document being summarized, and which may be wholly incorrect. Due to this need, a wide array of metrics estimating consistency with the text being summarized have been proposed. We examine in particular a suite of unsupervised metrics for summary consistency, and measure their correlations with each other and with human evaluation scores in the *wiki_bio_gpt3_hallucination* dataset. We then compare these evaluations to models made from a simple linear ensemble of these metrics. We find that LLM-based methods outperform other unsupervised metrics for hallucination detection. We also find that ensemble methods can improve these scores even further, provided that the metrics in the ensemble have sufficiently similar and uncorrelated error rates. Finally, we present an ensemble method for LLM-based evaluations that we show improves over this previous SOTA.

Keywords: Large Language Models; Text summarization; Hallucination Detection; Ensemble methods

2. Introduction

Text summarization is a rapidly changing and advancing field, due in no small part to the advent of Large Language Models (LLMs) such as GPT [1] and LaMDA [2]. Many summarization methods, however, struggle with "hallucinating:" inserting false, misleading and/or nonrepresentative material into the summaries. As such, many automatic methods for hallucination detection have been proposed in the literature for both evaluation and iterative improvement of text summarization methods. This diversity of methods, while indicative of rapid progress, has also led to a situation where there is no one clear standard evaluative metric for hallucinations in text summarization. With this in mind, we test a suite of hallucination detection from prior literature on the *wiki_bio_gpt3_hallucination* dataset [3,4], and examine their correlations with both each other and with a human evaluation baseline. We also, drawing on prior work in ensemble methods, test these against a linear ensemble of the sampled methods, and found that this ensemble outperforms most individual evaluation metrics. We found evaluation methods based on directly querying LLMs themselves to be most closely correlated with human evaluation, outperforming all non-LLM metrics and the ensemble. With this in mind, we constructed a new ensemble of LLM evaluations with a range of temperatures, with the expectation that perturbations to the metric that didn't correlate with the "true" value of what was being measured would cancel out in aggregate (we elaborate on this expectation in Section 3.3). We found that our LLM ensemble outperformed even the best LLM-based single evaluation, indicating our method to be the most accurate and effective hallucination detection metric to date for our chosen dataset.

3. Metrics Evaluated and Related Work

Here, we discuss prior work, and describe the metrics we've chosen to represent the suite of prior methods that exist. We also discuss ensemble methods, the theoretical

Citation: Forbes, C. G.; Katlana, P.; Ortiz, Z. Title. *Journal Not Specified* **2023**, *1*, 0. https://doi.org/

Received:
Revised:
Accepted:
Published:

justifications for their use in this domain, and the conditions generally required for them to be effective.

3.1. Text Summarization

Traditionally, text summarization has been categorized as either extractive or abstractive. Extractive text summarizers, such as OCCAMS [5], produce summaries by concatenating particularly salient sentences (“extracts") from the document being summarized. On the other hand, abstractive summarizers, such as most methods using LLMs [6], attempt to generate a summary "from scratch," assembling new sentences in an attempt to synthesize the information in a document in a more human-seeming way. Abstractive summaries are often able to be more natural-sounding than extractive summaries, but, as has been noted repeatedly in the literature, have a tendency to hallucinate [7]. It is often challenging to evaluate these models, as has been noted in [8].

3.2. Hallucination Detection Metrics

There are many methods for hallucination detection in prior work: too many to feasibly include them all within this work. We limited the scope of methods that we tested in two key ways: by constraining the broader scope of concern to unsupervised metrics, and by choosing a selection of well-regarded methods meant to cover the breadth of scope within that subfield. When we say we are specifically concerned with unsupervised hallucination detection methods, we mean those which require no input other than the summary and the source text itself. We chose to focus on these metrics as they are the most general, requiring no gold-standard human summaries or other supplementary information, and are thus the most widely deploy-able. More particularly, unsupervised hallucination detection metrics are deploy-able in two important context which exclude any other types of metrics:

1. As an evaluative tool on summarization data (possibly generated continuously, rather than part of a finite set)

2. As an in-the-loop tool for actively curbing hallucinations in a summarization tool at runtime.

The survey [9] identifies four general types of unsupervised summarization metrics: “triple-based," “textual-entailment-based," “QA-based," and “Other." We evaluate representatives from each of these categories (a more detailed analysis of these categories and the intricacies therein can be found in the aforementioned survey). We chose these to both cover the identified breadth of evaluation methods in the literature (i.e., pulling representatives from each of these categories), as well as to find methods with good theoretical/empirical backing and wide use, while still being recently developed and relevant to the SOTA.

3.2.1. FactSumm

FactSumm [10] is a triple-based metric to estimate the factual accuracy of generated text. It builds on prior works in graph-based hallucination detection [11,12], using pre-trained models to extract fact triples (subject, relation, object) from both the source document and the summary, and returns a count of the number of triples extracted from the summary that are included in the extract from the document itself. This serves as a heuristic for the percentage of "facts" in the summary that are contained within the source document.

3.2.2. QAGS

QAGS [13] is a question-answering based metric that automatically generates questions and answers from the source document and summary, then scores the summary based on how many of the derived questions are answered correctly. We used the implementation of QAGS in [10]. For comparison, other notable representatives from this category are QAEval [14] and FEQA [15].

3.2.3. ROUGE

ROUGE [16] is traditionally used as a supervised metric, by calculating the ROUGE score between a generated summary and a gold-standard human summary. However, [17] introduced the idea of using ROUGE as an unsupervised metric, and demonstrated its efficacy in such a capacity. Using ROUGE without supervision ("supervision," in this case, referring to gold standard human summaries that can be compared against) involves taking the ROUGE score between the generated summary and the text itself being summarized: treating the text itself, in other words, as its own "gold standard." The intuition behind this as a heuristic is that hallucinatory passages, on average, are likely to have less similarity (measured by ROUGE) to the source text than those which accurately summarize the source text.

3.2.4. SMART

SMART (Sentence Matching for Rating Text) [18] evaluation works on two principal ideas. Firstly, it treats sentences, rather than tokens, as the basic units of matching between system and reference summaries. Because, then, exactly matching sentences are most likely nonexistent (in abstractive summaries, though they are trivially present in extractive summaries), SMART utilizes soft-matching functions to compare sentences which can vary with respect to the type of SMART that is being used. SMART types utilized for the purposes of our study are:

- SMART1: Unigram-based scoring.
- SMART2: Bigram-based scoring.
- SMARTL: Longest common subsequence-based scoring.

It is also significant to mention that the unit of n-grams used here are chunks of tokens (sentences by default). This is different from the token-level n-grams used in standard ROUGE.
Secondly, SMART allows to compare the candidate system summary with the source document. This is particularly important when evaluating dimensions of summary quality that rely on the source document such as factuality.

3.2.5. SummaC

SummaC [19], similarly to SMART, runs evaluations on a sentence-by-sentence basis, but unlike SMART, is explicitly based on Natural Language Entailment (NLI) evaluations between sentences in the source document and the summary. SummaC first generates a matrix for every sentence pair between the summary and source document. Then the two models we benchmark analyze this matrix to achieve a benchmark.

SUMMAC$_{zs}$ (Zero-Shot) reduces the pair matrix to a one-dimensional vector by taking the maximum value of each column, then computes the mean. This step retains the score for the document sentence that provides the strongest support for each summary sentence. It leverages the intuition that each sentence in the summary document, if non-hallucinatory, should have at least one sentence in the source document which has a high entailment score.

SUMMAC$_{conv}$ (Convolutional) reduces reliance on extreme values and takes the entire distribution of entailment scores for each summary sentence into account. It utilizes a learned convolutional network on the NLI matrix to compute a final score for the respective summary sentence.

3.2.6. SelfCheckGPT

SelfCheckGPT [4] is an unsupervised hallucination detection method that relies on the intuition that factual generated summaries are much more likely to be similar to each other than to those which contain hallucinations, whereas hallucination-containing summaries are not more likely to be similar to each other than to factual summaries. Another way of framing this intuition is that language models which are confident in their knowledge are likely to have much less diverse responses than those which are making things up.

It involves generating multiple summaries for a given source document, then utilizing a variety of distance metrics to check generated summaries against each other for similarity, and shows that higher similarity to other generated summaries is highly correlated with human annotations for textual consistency. Note that we specifically benchmarked their unigram-based approach, as it was the single approach that had the highest correlations with human judgements in [4].

3.2.7. LLM Self-Evaluation

Recent work, such as [20], has explored the possibility of using LLMs themselves as evaluative tools for text data generally, and for abstractive summaries in particular. These recent results are very promising, and may usurp traditional, non-LLM-based evaluative methods in this domain, such as those we have listed thus far. For benchmarking these methods, we reproduced the prompting technique described in [20]. For ease of reference, the prompt is in Listing 1 below.

Listing 1. LLM evaluation prompt

```
Score the following summary given the corresponding article with respect to
consistency from 1 to 10.  Note that consistency measures how much
information included in the summary is present in the source article.  10
points indicate the summary contains only statements that are entailed by
the source document.
Summary:  {summary}
Source Article:  {source}
Marks:
```

We used this prompt in both GPT 3.5 turbo and GPT 4 models as benchmarks for this method. Our results lend evidence to the claim that these methods indeed surpass more traditional hallucination detection methods, and we use these results to further refine these methods into an ensemble approach that outperforms prior work.

3.3. Metric Ensembles

It's been long noted that ensembles of models or metrics, even subpar ones combined naively, are surprisingly efficient and can rival or outperform expert judgement [21]. Ensembles also have been noted to aid in explainability in some contexts, something particularly relevant for analysis work, which is very sensitive to reliability and has a high threshold of required trust [22]. Ensemble methods are thus a promising avenue for improving over a baseline, particularly in a domain such as hallucination detection, where there are a wide variety of disparate metrics, none of which is clearly superior to others in measuring the "true" value.

Ensemble methods operate, fundamentally, by leveraging the ability of the individual metrics' error from the “true" value to cancel each other out in aggregate. As derived in [23], if there is a collection of value-estimating functions f_i, each of which differs from some true function f by some $m_i = f - f_i$, and we assume the errors are uncorrelated, then in expectation we should expect the error m_{sum} of f_{sum} to be

$$m_{sum} = f - \frac{1}{N}\sum_{i=0}^{N-1} f_i = \frac{\bar{m}_i}{N} \tag{1}$$

where $\bar{m}_i$ is the mean value of M_i. Due to this minimization of error by a factor of N, ensembles are a powerful tool to deploy in spaces, such as hallucination detection, where we have many diverse estimates for ground truth, but no (or prohibitively slow and expensive) access to that ground truth itself. There are then two qualities of some collection of metrics f_i that we would want, in order for an ensemble method to be effective:

Condition 1. The metrics must be diverse: that is, their errors should be relatively uncorrelated with each other (this assumption is key for the referenced derivation in [23]).

Condition 2. The metrics must have m_i's that are similar in magnitude. If this condition is not met, then it is possible that the f_i with the lowest error alone would outperform an ensemble model. In other words, the ensemble should only be derived from models that are similarly good estimators of the true function f.

The metrics we're using in this work certainly meet condition 1, as we've selected them to cover the breadth of methods in the literature. It remains to be seen, however, if they meet condition 2, and in fact we shall see that, as selected, they do not yet meet this condition.

Some recent prior work has used ensemble, or ensemble-esque methods to leverage LLMs effectively. These often involve iterative prompting techniques, such as in [24], which prompts agents to "debate" each other before arriving at a final answer. SelfCheckGPT [4] itself could be seen as a variation of an ensemble method, as it involves self-checking the model's responses against other responses it might have given. Similar work is in [25], which incorporates a self-checking ensemble approach into the sampling algorithm for an LLM.

While in this work, we simply use naive similar ensembles of metrics with uniform weights, as has been shown to be effective [21,23], some other work has been done on using unlabelled data to find the best term weights for metrics [26–28]. While this particular line of work is applied specifically to binary classifiers, and we're working with metrics that cannot be generally constricted to outputs $\in \{0,1\}$, we believe something like this approach could be extended to metric ensembles in future work.

4. Method

We used the *wiki_bio_gpt3_hallucination* dataset. This dataset consists of a subset of 238 entries from the original Wikibio dataset, generated in [3], accompanied by GPT3-generated summaries and sentence-by-sentence human evaluations of those summaries, ranking each sentence as "accurate," "minor inaccurate," or "major inaccurate." These additional summaries and human evaluations were generated in [4]. For each summary in the dataset, we evaluate each metric on this summary and the source document from which it was generated. We then compute the Pearson correlation between this metric and the other metrics, including human evaluation, which we treat as our "gold standard" metric, or ground truth. For this ground truth, we translate the human evaluations into a single scalar value by taking their mean, wherein we treat "major inaccurate" as a 0, "accurate" as a 1, and "minor inaccurate" as a 0.5. Note that this is the reverse of the method used in [4]: we chose this in order to align the direction of our gold standard with our other hallucination detection benchmarks, in which higher numbers consistently indicate good summaries, rather than bad.

For our ensemble, we simply took a weighted mean of the other models. Weights were set to a constant of 1 for each model, except for in cases where two models were simply closely-aligned permutations of the same model, in which case they were treated as "sub-models" of the same base model, which would not have the required diversity to meet Condition 1 in Section 3.3. In these cases, each sub-model was weighted by some fraction chosen according to how many sub-models were being used, so that they sum to 1. Going down the rows of Table 1 (excluding the first and last rows), the weights used were $[1, 1, \frac{1}{3}, \frac{1}{3}, \frac{1}{3}, \frac{1}{3}, \frac{1}{3}, \frac{1}{3}, \frac{1}{2}, \frac{1}{2}, 1, \frac{1}{2}, \frac{1}{2}]$.

5. Results

Our results are split into roughly two sections: those comparing all metrics or ensembles of those metrics, and those which focus exclusively on "Non-LLM" metrics. Note that this term, as we're using it, refers to all metrics that do not involve a direct evaluation from an LLM: so SelfCheckGPT [4], while ostensibly running functions over many LLM outputs, is evaluated in only the former of these sections, and not the latter. Section 5.1 covers all

Metrics	Human Eval	FactSumm Tuples	QAGS	ROUGE-1	ROUGE-2	ROUGE-L	SMART-1	SMART-2	SMART-L	SummaC$_{zs}$	SummaC$_{conv}$	SelfCheckGPT	GPT-3.5	GPT-4	Ensemble
Human Eval	1.00	0.50	0.75	0.66	0.64	0.67	0.64	0.61	0.62	0.67	0.50	0.60	0.85	0.89	0.82
FactSumm Tuples	0.50	1.00	0.60	0.45	0.50	0.47	0.54	0.52	0.53	0.49	0.43	0.46	0.44	0.50	0.68
QAGS	0.75	0.60	1.00	0.67	0.77	0.70	0.76	0.74	0.75	0.69	0.67	0.61	0.66	0.74	0.89
ROUGE-1	0.66	0.45	0.67	1.00	0.90	0.99	0.72	0.68	0.72	0.65	0.57	0.63	0.66	0.68	0.82
ROUGE-2	0.64	0.50	0.77	0.90	1.00	0.94	0.86	0.83	0.87	0.68	0.76	0.69	0.58	0.66	0.89
ROUGE-L	0.67	0.47	0.70	0.99	0.94	1.00	0.77	0.73	0.78	0.67	0.64	0.66	0.65	0.67	0.85
SMART-1	0.64	0.54	0.76	0.72	0.86	0.77	1.00	0.99	0.99	0.71	0.83	0.69	0.59	0.64	0.91
SMART-2	0.61	0.52	0.74	0.68	0.83	0.73	0.99	1.00	0.99	0.70	0.84	0.68	0.56	0.62	0.89
SMART-L	0.62	0.53	0.75	0.72	0.87	0.78	0.99	0.99	1.00	0.70	0.83	0.68	0.57	0.63	0.91
SummaC$_{zs}$	0.68	0.49	0.69	0.65	0.68	0.67	0.71	0.69	0.70	1.00	0.68	0.64	0.64	0.67	0.81
SummaC$_{conv}$	0.50	0.43	0.67	0.57	0.76	0.64	0.83	0.84	0.83	0.68	1.00	0.70	0.39	0.49	0.79
SelfCheckGPT	0.60	0.46	0.61	0.63	0.69	0.66	0.69	0.67	0.68	0.64	0.70	1.00	0.48	0.57	0.78
GPT-3.5	0.85	0.44	0.66	0.66	0.58	0.65	0.59	0.56	0.57	0.64	0.39	0.48	1.00	0.85	0.77
GPT-4	0.89	0.50	0.74	0.68	0.66	0.67	0.64	0.62	0.63	0.67	0.49	0.57	0.85	1.00	0.83
Ensemble	0.82	0.68	0.88	0.82	0.89	0.85	0.91	0.89	0.91	0.81	0.79	0.78	0.77	0.83	1.00

Table 1. Pearson correlations between all metrics, our linear ensemble, and human evaluations in the WikiBio hallucination dataset.

metrics, while Section 5.2 covers the further dedicated experiments on solely LLM-based metrics/ensembles.

5.1. Non-LLM Metric Correlations

The correlations between all benchmarks, human evaluations, and our linear ensemble method are recorded in Table 1, and displayed visually as a heatmap in Figure 1. Additionally, a plot showing just the correlations of each method with human evaluations (corresponding to the topmost/leftmost row/column in Figure 1) is shown in Figure 2.

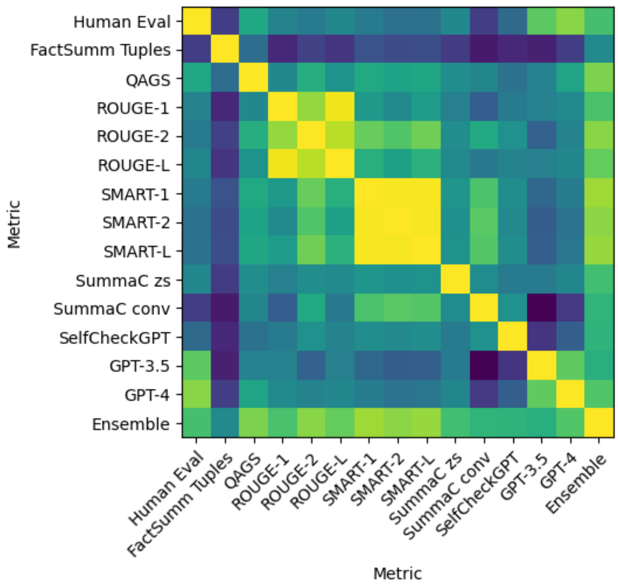

Figure 1. Heatmap of Pearson correlations between all benchmark metrics, our linear ensemble of all benchmarks, and human evaluations.

We note and discuss several of the more salient data from this study.

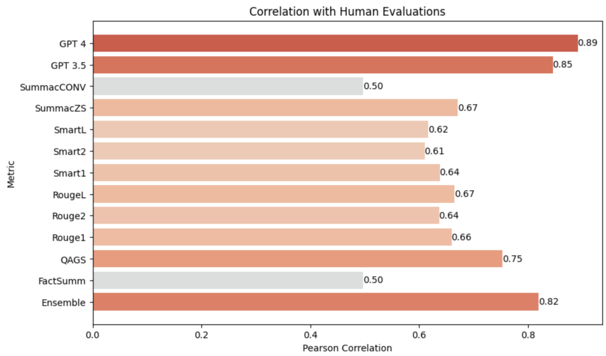

Figure 2. Plot of Pearson correlation with human judgement for each model.

5.1.1. Ensemble Outperforms All Non-LLM-Based Methods

We observe that the LLM-based methods have the highest correlation with human evaluations, but that the ensemble method outperforms all others. [1] The fact that some of the base methods used in the ensemble outperform the ensemble itself suggests that, in this particular case, Condition 2 of Section 3.3 is not satisfied: there are certain methods (GPT-3 and GPT-4, used as in [20]) that outperform the others to a significant enough extent that the gains from the ensemble's diversity are not sufficient to overcome the losses from giving more weight to less accurate models. It is for this reason that we conduct additional experiments in Section 5.2, focusing exclusively on the LLM-based methods.

5.1.2. Methodological Similarities Yield High Correlations

As we would expect, blocks of methods that are methodologically similar yield high Pearson correlations with each other. Particularly, the ROUGE-based methods are all highly correlated with each other, as are the SMART-based methods. These two blocks are also somewhat highly correlated with each other, likely due to the fact that they're doing very similar things under the hood (though SMART is doing this with sentences as the basic blocks of analysis, whereas ROUGE is doing it with tokens).

SummaC and LLM-based methods, alternatively, while slightly higher-correlated with methodologically similar methods, are not as homogeneous as these.

5.1.3. Comparative Performance Measured By Human Evaluation

As noted previously, LLM-based methods and QA-based methods both perform well on this dataset. Our graph-based method performed surprisingly poorly: without further study, however, it cannot be said if this is an indictment of graph-based hallucination detection methods as a whole, or simply of the particular tuple-generating models employed by FactSumm. It is possible that a more complex or robust graph-generating model would considerably boost performance here.

The zero-shot version of SummaC outperformed the convolutional version by a substantial margin, which is surprising, as it goes against the findings in the original SummaC paper. We take this as suggesting that the convolutional model weights might be overfit to the CNN dailymail dataset that the model was trained on, and that a more "commonsense" zero-shot model may actually be more generalizable.

5.2. LLM Metrics Correlations

We compared the LLM across different temperatures for the prompt that stands to yield the most accurate response. [20]

[1] For further context, if the LLM-based models are removed from the ensemble, it outperforms all remaining methods other than QAGS, and if both the LLM-based and QAGS metrics are removed from the ensemble, it outperforms all remaining methods.

Metrics	Human Eval	FactSumm Tuples	QAGS	ROUGE-1	ROUGE-2	ROUGE-L	SMART-1	SMART-2	SMART-L	SummaC$_{zs}$	SummaC$_{conv}$	SelfCheckGPT	GPT-3.5	GPT-4	Ensemble
Human Eval	1.00	0.50	0.75	0.66	0.64	0.67	0.64	0.61	0.62	0.67	0.50	0.60	0.85	0.89	0.82
FactSumm Tuples	0.50	1.00	0.60	0.45	0.50	0.47	0.54	0.52	0.53	0.49	0.43	0.46	0.44	0.50	0.68
FactSumm Tuples	0.75	0.60	1.00	0.67	0.77	0.70	0.76	0.74	0.75	0.69	0.67	0.61	0.66	0.74	0.89
ROUGE-1	0.66	0.45	0.67	1.00	0.90	0.99	0.72	0.68	0.72	0.65	0.57	0.63	0.66	0.68	0.82
ROUGE-2	0.64	0.50	0.77	0.90	1.00	0.94	0.86	0.83	0.87	0.68	0.76	0.69	0.58	0.66	0.89
ROUGE-L	0.67	0.47	0.70	0.99	0.94	1.00	0.77	0.73	0.78	0.67	0.64	0.66	0.65	0.67	0.85
SMART-1	0.64	0.54	0.76	0.72	0.86	0.77	1.00	0.99	0.99	0.71	0.83	0.69	0.59	0.64	0.91
SMART-2	0.61	0.52	0.74	0.68	0.83	0.73	0.99	1.00	0.99	0.70	0.84	0.68	0.56	0.62	0.89
SMART-L	0.62	0.53	0.75	0.72	0.87	0.78	0.99	0.99	1.00	0.70	0.83	0.68	0.57	0.63	0.91
SummaC$_{zs}$	0.68	0.49	0.69	0.65	0.68	0.67	0.71	0.69	0.70	1.00	0.68	0.64	0.64	0.67	0.81
SummaC$_{conv}$	0.50	0.43	0.67	0.57	0.76	0.64	0.83	0.84	0.83	0.68	1.00	0.70	0.39	0.49	0.79
SelfCheckGPT	0.60	0.46	0.61	0.63	0.69	0.66	0.69	0.67	0.68	0.64	0.70	1.00	0.48	0.57	0.78
GPT-3.5	0.85	0.44	0.66	0.66	0.58	0.65	0.59	0.56	0.57	0.64	0.39	0.48	1.00	0.85	0.77
GPT-4	0.89	0.50	0.74	0.68	0.66	0.67	0.64	0.62	0.63	0.67	0.49	0.57	0.85	1.00	0.83
Ensemble	0.82	0.68	0.88	0.82	0.89	0.85	0.91	0.89	0.91	0.81	0.79	0.78	0.77	0.83	1.00

Table 2. Pearson Correlation

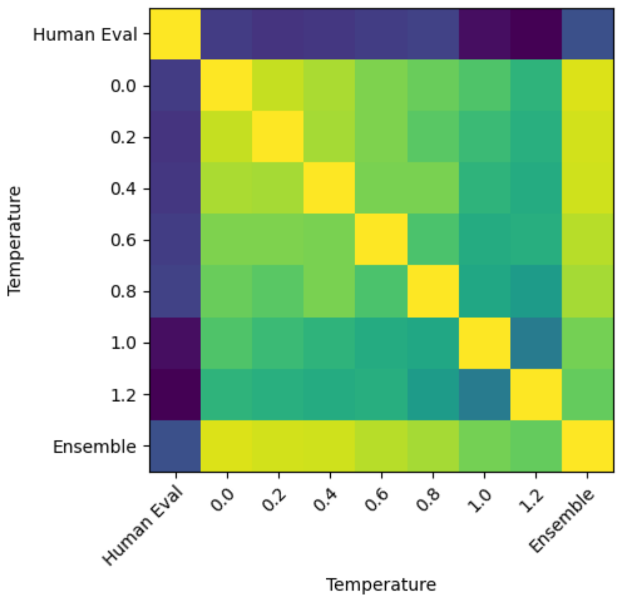

Figure 3. Heatmap of Pearson correlations between GPT-4 evaluations at various temperatures, our linear GPT-4 ensemble, and human evaluations.

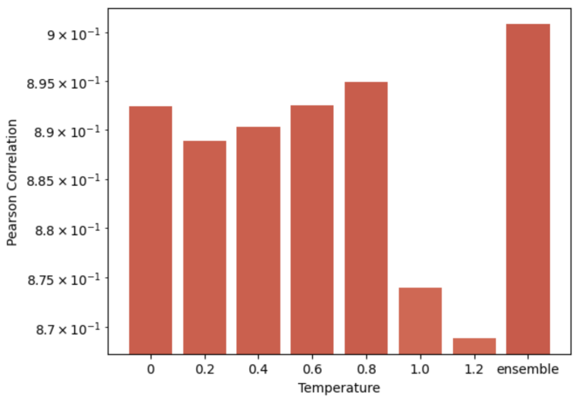

Figure 4

Previous literature has shown that LLMs perform better at lower temperatures and drastically decline in their efficiency at higher temperatures. We observe similar results in context to hallucination evaluations while using GPT 3.5turbo and GPT-4 for the ensemble in our study. The evaluation correlates increasingly from 0.2 to 0.8 before dropping drastically at temperature 1.0.

6. Discussion and Conclusion

Abstractive summaries are prone to hallucinations, meaning they may include statements that lack support from the original text. Some of these statements can be outright false, while others may be unsupported due to insufficient evidence within the source document. To address this issue, prior research has introduced several fact-checking tools that rely on automatic question-answering systems and textual entailment methods.[17,29]

In our study, we conducted a pilot experiment to explore the effectiveness of ensembles in detecting hallucinations. To evaluate their performance, we compared the ensembles using benchmark state-of-the-art metrics commonly employed in this domain. We have presented a simple self-training linear sum ensemble approach which leads to sizeable gains on both unsupervised metrics and LLMs evaluating unlabeled data for hallucination. We piloted the use of ensembles for hallucination detection by comparing them across the benchmark state-of-the-art metrics.

Improvements on the Benchmark. The models we introduced in this paper are just a first step towards harnessing ensemble models for hallucination detection. Future work could explore a number of improvements: measuring the errors for benchmarking with FRANK[30], optimizing weight redistribution to achieve the most optimal level, and creating a ground rule algorithm[28] by utilizing various metrics or combining multiple temperature settings.

Interpretability of model output. If a model has the ability to achieve better correlations with human evaluations or annotations, certain studies have indicated that ensemble models can proficiently quantify those problematic sections in many instances. Additionally, the ensemble can be further fine-tuned with respect to the temperatures along with other LLM models to establish consistency while scoring against other metrics.

Towards Consistent Summarization. Hallucination detection is but a first step in eliminating inconsistencies from summarization. Future work can include more powerful hallucination detectors in the training of next-generation summarizers to both detect and reduce the prevalence of hallucinations in generated text

References

1. Bubeck, S.; Chandrasekaran, V.; Eldan, R.; Gehrke, J.; Horvitz, E.; Kamar, E.; Lee, P.; Lee, Y.T.; Li, Y.; Lundberg, S.; et al. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712* **2023**.
2. Thoppilan, R.; De Freitas, D.; Hall, J.; Shazeer, N.; Kulshreshtha, A.; Cheng, H.T.; Jin, A.; Bos, T.; Baker, L.; Du, Y.; et al. Lamda: Language models for dialog applications. *arXiv preprint arXiv:2201.08239* **2022**.
3. Lebret, R.; Grangier, D.; Auli, M. Neural text generation from structured data with application to the biography domain. *arXiv preprint arXiv:1603.07771* **2016**.
4. Manakul, P.; Liusie, A.; Gales, M.J. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. *arXiv preprint arXiv:2303.08896* **2023**.
5. White, C.T.; Molino, N.P.; Yang, J.S.; Conroy, J.M. occams: A Text Summarization Package. *Analytics* **2023**, *2*, 546–559.
6. Zhang, H.; Liu, X.; Zhang, J. SummIt: Iterative Text Summarization via ChatGPT. *arXiv preprint arXiv:2305.14835* **2023**.
7. Ji, Z.; Lee, N.; Frieske, R.; Yu, T.; Su, D.; Xu, Y.; Ishii, E.; Bang, Y.J.; Madotto, A.; Fung, P. Survey of hallucination in natural language generation. *ACM Computing Surveys* **2023**, *55*, 1–38.
8. Goyal, T.; Li, J.J.; Durrett, G. News summarization and evaluation in the era of gpt-3. *arXiv preprint arXiv:2209.12356* **2022**.
9. Huang, Y.; Feng, X.; Feng, X.; Qin, B. The factual inconsistency problem in abstractive text summarization: A survey. *arXiv preprint arXiv:2104.14839* **2021**.
10. Heo, H. FactSumm: Factual Consistency Scorer for Abstractive Summarization. https://github.com/Huffon/factsumm, 2021.
11. Kryściński, W.; McCann, B.; Xiong, C.; Socher, R. Evaluating the factual consistency of abstractive text summarization. *arXiv preprint arXiv:1910.12840* **2019**.

12. Goodrich, B.; Rao, V.; Liu, P.J.; Saleh, M. Assessing the factual accuracy of generated text. In Proceedings of the proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining, 2019, pp. 166–175.
13. Wang, A.; Cho, K.; Lewis, M. Asking and answering questions to evaluate the factual consistency of summaries. *arXiv preprint arXiv:2004.04228* **2020**.
14. Deutsch, D.; Bedrax-Weiss, T.; Roth, D. Towards question-answering as an automatic metric for evaluating the content quality of a summary. *Transactions of the Association for Computational Linguistics* **2021**, *9*, 774–789.
15. Durmus, E.; He, H.; Diab, M. FEQA: A question answering evaluation framework for faithfulness assessment in abstractive summarization. *arXiv preprint arXiv:2005.03754* **2020**.
16. Lin, C.Y. Rouge: A package for automatic evaluation of summaries. In Proceedings of the Text summarization branches out, 2004, pp. 74–81.
17. Louis, A.; Nenkova, A. Automatically evaluating content selection in summarization without human models **2009**.
18. Amplayo, R.K.; Liu, P.J.; Zhao, Y.; Narayan, S. SMART: sentences as basic units for text evaluation. *arXiv preprint arXiv:2208.01030* **2022**.
19. Laban, P.; Schnabel, T.; Bennett, P.N.; Hearst, M.A. SummaC: Re-visiting NLI-based models for inconsistency detection in summarization. *Transactions of the Association for Computational Linguistics* **2022**, *10*, 163–177.
20. Luo, Z.; Xie, Q.; Ananiadou, S. Chatgpt as a factual inconsistency evaluator for abstractive text summarization. *arXiv preprint arXiv:2303.15621* **2023**.
21. Dawes, R.M. The robust beauty of improper linear models in decision making. *American psychologist* **1979**, *34*, 571.
22. Forbes, G.; Crouser, R.J. Metric Ensembles Aid in Explainability: A Case Study with Wikipedia Data. *Analytics* **2023**, *2*, 315–327.
23. Perrone, M.P.; Cooper, L.N. When networks disagree: Ensemble methods for hybrid neural networks. In *How We Learn; How We Remember: Toward An Understanding Of Brain And Neural Systems: Selected Papers of Leon N Cooper*; World Scientific, 1995; pp. 342–358.
24. Du, Y.; Li, S.; Torralba, A.; Tenenbaum, J.B.; Mordatch, I. Improving Factuality and Reasoning in Language Models through Multiagent Debate. *arXiv preprint arXiv:2305.14325* **2023**.
25. Wang, X.; Wei, J.; Schuurmans, D.; Le, Q.; Chi, E.; Narang, S.; Chowdhery, A.; Zhou, D. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171* **2022**.
26. Platanios, E.A.; Blum, A.; Mitchell, T.M. Estimating Accuracy from Unlabeled Data. In Proceedings of the UAI, 2014, Vol. 14, p. 10.
27. Platanios, E.A.; Dubey, A.; Mitchell, T. Estimating accuracy from unlabeled data: A bayesian approach. In Proceedings of the International Conference on Machine Learning. PMLR, 2016, pp. 1416–1425.
28. Platanios, E.; Poon, H.; Mitchell, T.M.; Horvitz, E.J. Estimating accuracy from unlabeled data: A probabilistic logic approach. *Advances in neural information processing systems* **2017**, *30*.
29. based self-training for abstractive opinion summarization, O.E. Understanding Factuality in Abstractive Summarization with FRANK: A Benchmark for Factuality Metrics. *arXiv preprint arXiv:2212.10791* **2022**.
30. Artidoro Pagnoni, Vidhisha Balachandran, Y.T. Understanding Factuality in Abstractive Summarization with FRANK: A Benchmark for Factuality Metrics. *arXiv preprint arXiv:2104.13346* **2021**.