NU-RU at SemEval-2024 Task 6: Hallucination and Related Observable Overgeneration Mistake Detection Using Hypothesis-Target Similarity and SelfCheckGPT

Thanet Markchom[1] and **Subin Jung**[2] and **Huizhi Liang**[2]
[1]Department of Computer Science, University of Reading, Reading, UK
[2]School of Computing, Newcastle University, Newcastle upon Tyne, UK
t.markchom@pgr.reading.ac.uk, {s.jung4, huizhi.liang}@newcastle.ac.uk

Abstract

One of the key challenges in Natural Language Generation (NLG) is "hallucination", in which the generated output appears fluent and grammatically sound but may contain incorrect information. To address this challenge, "SemEval-2024 Task 6 - SHROOM, a Shared-task on Hallucinations and Related Observable Overgeneration Mistakes" is introduced. This task focuses on detecting overgeneration hallucinations in texts generated from Large Language Models for various NLG tasks. To tackle this task, this paper proposes two methods: (1) hypothesis-target similarity, which measures text similarity between a generated text (hypothesis) and an intended reference text (target), and (2) a SelfCheckGPT-based method to assess hallucinations via predefined prompts designed for different NLG tasks. Experiments were conducted on the dataset provided in this task. The results show that both proposed methods can effectively detect hallucinations in LLM-generated texts.

1 Introduction

Natural Language Generation (NLG) is a field within Natural Language Processing (NLP) that focuses on enabling machines to produce human-like texts. In NLG, one of the challenges is the phenomenon of "hallucination", where the generated output is fluent and grammatically sound but contains incorrect information or extends beyond the provided information. This issue is particularly significant in NLG applications where correctness is crucial, such as machine translation and paraphrasing. It can compromise the quality and reliability of the generated content, resulting in a loss of fidelity to the sources or models from which the content is generated. To address this challenge, "SemEval-2024 Task 6 - SHROOM, a Shared-task on Hallucinations and Related Observable Overgeneration Mistakes" (Mickus et al., 2024) is introduced. This task aims to identify grammatically correct outputs that contain incorrect semantic information or overgenerated content, with or without access to the model that produced the output. The outputs are obtained from various Large Language Models (LLMs) in three distinct NLG tasks: definition modeling (DM), machine translation (MT), and paraphrase generation (PG).

Recent efforts have been made to develop frameworks for detecting hallucinations in LLM-generated texts. One approach involves calculating information overlap and contradictions between generated and reference texts (Dhingra et al., 2019; Shuster et al., 2021). Higher mismatches suggest a greater likelihood of hallucination. Another popular approach is an LLM-based evaluation. This approach focuses on prompting LLMs to assess a machine-generated text and determine the probability of this text being a hallucination (Kadavath et al., 2022; Manakul et al., 2023).

Despite the success of these existing methods, they have mainly focused on detecting factual hallucinations. This paper further explores how information overlap calculation and LLM-based evaluation approaches can be applied to detect overgeneration hallucinations. Specifically, we propose two methods to detect overgeneration hallucinations in SemEval Task 6. The first method is hypothesis-target similarity, which measures text similarity between a generated text (hypothesis) and an intended reference text (target). The second method is an LLM-based evaluation approach that utilizes a state-of-the-art framework called SelfCheckGPT (Manakul et al., 2023). This method assesses hallucinations via distinct predefined prompts tailored for texts generated from different NLG tasks.

2 Related Work

Recently there have been some attempts to develop frameworks for evaluating hallucinations in LLM-

Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024), pages 253–260
June 20-21, 2024 ©2024 Association for Computational Linguistics

generated texts. One approach is to consider lexical features of LLM-generated texts and reference texts and calculate the information overlap and contradictions between the generated and the reference texts. The higher the mismatch counts, the lower the faithfulness and thus the higher the hallucination score. For example, Dhingra et al. (2019) proposed PARENT (Precision And Recall of Entailed n-grams from the Table), which is capable of assessing hallucinations by referencing both the source and target texts. Shuster et al. (2021) introduced a metric for knowledge-grounded dialogue tasks aimed at measuring the alignment between LLM-generated texts and the relevant knowledge judged by humans. Martindale et al. (2019) proposed the Bag-Of-Vectors Sentence Similarity (BVSS) metric for assessing sentence adequacy in machine translation. This metric aids in identifying disparities in information between the output and the translation reference. Despite the simplicity and effectiveness of information overlapping, it has limitations in handling syntactic or semantic variations, which can impact its accuracy in evaluating faithfulness.

Another recent approach is an LLM-based evaluation where an LLM is prompted to evaluate generated texts, e.g., to predict the probability that a generated text is a hallucination. For instance, Kadavath et al. (2022) used LLMs to evaluate the validity of their own claims by asking models to first generate answers and then to evaluate the probability that their answers are correct. Manakul et al. (2023) proposed an approach called SelfCheckGPT with prompts. In their approach, each LLM-generated sentence was compared against multiple generated responses from an LLM. An LLM was asked to assess whether an LLM-generated sentence was supported by the generated responses. If it was consistently supported by multiple responses, then it was likely to not be a hallucination. Friel and Sanyal (2023) proposed the ChainPoll approach where an LLM was asked to decide whether an LLM-generated text contained hallucinations, using a detailed and carefully engineered prompt. However, the majority of existing approaches have primarily focused on detecting factual hallucinations related to incorrect information in texts, rather than overgeneration hallucinations. Thus, there remains a critical need to explore and adapt these approaches for the detection of overgeneration hallucinations.

3 Problem Formulation

The objective of this task is to predict whether the actual model production (generated text) is a hallucination, with or without having access to the model that generated the text. Specifically, each input in this task consists of the following information:

- Task (task): the task for which the model was optimized, which can be either Definition Modeling (DM), Paraphrase Generation (PG), or Machine Translation (MT).
- Source (src): the input provided to the model.
- Target (tgt): the intended reference 'gold' text that the model is expected to generate.
- Hypothesis (hyp): the actual model output.
- Reference (ref): specifies whether the target, source, or both fields contain the semantic information necessary to establish whether the hypothesis is a hallucination.
- Model Checkpoint (model): Identifies the model used to produce the hypothesis (only applicable for model-aware inputs).

For each input, the goal is to predict a label indicating whether the hypothesis is a hallucination, along with the probability of the hypothesis being a hallucination (p(Hallucination)).

In this task, two datasets were provided: **model-aware dataset** and **model-agnostic dataset**. In the model-aware dataset, model checkpoints (available on HuggingFace) were provided for every sample. Conversely, in the model-agnostic dataset, these checkpoints were not included. For each dataset, an unlabeled training set, a validation set (with true labels), and a test set were provided. Also, a trial set was given without categorizing the samples based on whether they were model-aware or model-agnostic. The validation, trial and test sets contain binary annotations provided by at least five different annotators, along with a majority vote gold label.

4 Methods

To achieve the task of detecting overgeneration hallucinations, we propose two methods: (1) hypothesis-target similarity and (2) SelfCheckGPT-based methods. The details of each approach are discussed in the following subsections.

4.1 Hypothesis-Target Similarity Method

The proposed hypothesis-target similarity approach is an intuitive method for evaluating whether a generated text (hypothesis) contains hallucinations by comparing it with an intended reference or gold text (target). Specifically, we compute the text similarity between a hypothesis and a target and use the resulting value to determine whether the hypothesis contains hallucinations. The lower the similarity, the more likely it is that the hypothesis may contain a certain degree of hallucination. To compute text similarity, a text embedding method is first applied to generate embeddings of the generated and intended reference texts. In this work, we adopt *SentenceTransformers*[1] (Reimers and Gurevych, 2019) (*paraphrase-MiniLM-L6-v2*) to generate such embeddings since it has demonstrated success across various applications (Reimers and Gurevych, 2020; Choi et al., 2021; Markchom et al., 2020). Then, a cosine similarity metric is applied to these embeddings to compute the similarity. It is worth noting that other metrics are also applicable. This work selects cosine similarity due to its widespread usage and simplicity (Lin et al., 2014; Zhang et al., 2023).

After obtaining the similarity between a hypothesis and a target, we set a threshold δ to determine whether a hypothesis is a hallucination or not. If the similarity is lower than δ, it means that the hypothesis is different from the target and may contain hallucinations. Mathematically, given a hypothesis h and a target t, let e_h and e_t denote embeddings of the hypothesis and target, respectively. We define a function $f(h, t)$ that outputs the labels "Hallucination" and "Not Hallucination" for a given hypothesis h and target t as follows:

$$f(h,t) = \begin{cases} \text{Hallucination}, & \text{if } s(e_h, e_t) < \delta \\ \text{Not Hallucination}, & \text{otherwise} \end{cases} \tag{1}$$

where $s(e_h, e_t)$ denotes the cosine similarity between the hypothesis h and the target t. Furthermore, we compute $p(\text{Hallucination})$ based on the computed cosine similarity by applying a sigmoid function to the similarity as follows:

$$p(\text{Hallucination}) = \sigma(s(e_h, e_t)) \tag{2}$$

where σ denotes a sigmoid function. This function scales the computed similarity to the $[0, 1]$ interval and treats the resulting value as the probability of the hypothesis being a hallucination. Note that, in the PG task, target texts are unavailable for certain samples. Consequently, we consider source texts as target texts in these instances. In other words, we assess the similarity between a generated (paraphrased) text and its corresponding source text instead.

4.2 SelfCheckGPT-Based Method

In the SelfCheckGPT-based method, we adopt the SelfCheckGPT with Prompt approach in (Manakul et al., 2023) and design prompts to validate hallucinations. Specifically, for each sample, a prompt is crafted to assess whether a hypothesis is supported by a context, which includes a provided source and target (if available). If a hypothesis is not supported by a context, it is considered a hallucination. The prompt formats vary slightly for each task. Table 1 shows the prompt formats for samples from each task, where {src} denotes a source, {tgt} denotes a target, {hypo} denotes a hypothesis, and {term} denotes the term to be defined in a source only for the DM task. As shown in this table, the prompt for the DM task is noticeably different from the others. This is because we would like to semantically use the term as additional information apart from the source and hypothesis.

Task	Prompt format

DM	Context: {src} The term "{term}" means {tgt}
	Sentence: The term {term} means {hypo}
	Is the sentence supported by the context above?
	Answer Yes or No:

PG	Context: {src}
	Sentence: {hypo}
	Is the sentence supported by the context above?
	Answer Yes or No:

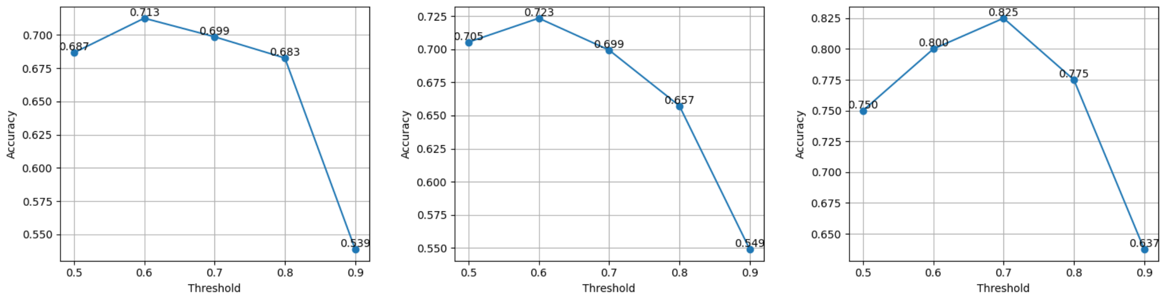
MT	Context: {src} {tgt}
	Sentence: {hypo}
	Is the sentence supported by the context above?
	Answer Yes or No:

Table 1: Prompt formats for samples from each task where {src} represent a source, {tgt} represents a target, {hypo} represents a hypothesis, and {term} represents the term to be defined in a source of the DM task.

Each prompt is run through the GPT-3.5 model (*gpt-3.5-turbo-1106*)[2] (Brown et al., 2020; OpenAI, 2022) N times, and the final label is determined by the majority of these responses. The probability $p(\text{Hallucination})$ of each sample is computed based

[1] https://www.sbert.net/

[2] https://platform.openai.com/docs/models/gpt-3-5



(a) Model-aware validation set

(b) Model-agnostic validation set

(c) Trial set

Figure 1: The accuracy on (a) the model-aware validation set, (b) model-agnostic validation set, and (c) the trial set when different threshold values are applied in the hypothesis-target similarity method

on the corresponding N responses as follows:

$$p(\text{Hallucination}) = \frac{1}{N}\sum_{i=1}^{N} l_i \tag{3}$$

where l_i denotes the ith predicted label (1 for "Hallucination" and 0 for "Not Hallucination") identified from the ith response.

5 Experiments

The experiments were conducted on both **model-aware** and **model-agnostic** datasets. For each of these datasets, we applied the proposed methods to both the validation and test sets to evaluate their performance. Two evaluation metrics were employed: the *accuracy* of binary classification and the *Spearman correlation* of the predicted probabilities (p(Hallucination)) with the proportion of the annotators marking the hypothesis as "Hallucination". We compared the proposed methods with the baseline provided by the task organizers on the test set. This baseline is based on the SelfCheckGPT with Prompt approach, employing an open-source Mistral model (Jiang et al., 2023). In this baseline, for samples from a PG task, only the source text is provided as context to the Mistral model, similar to our SelfCheckGPT-based method. For DM and MT tasks, the baseline utilizes only the target text as context. In contrast, our SelfCheckGPT method incorporates both source and target texts as context. Also, in this baseline, each prompt was run through the Mistral model only once.

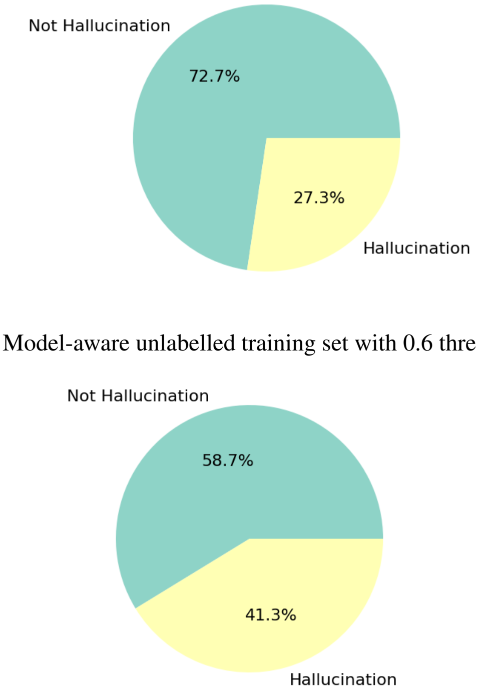
5.1 Hyperparameter Settings

Hypothesis-Target Similarity Method To determine the threshold δ, we conducted an analysis on the validation set to identify the optimal value. We varied the threshold from 0.5 to 0.9, increasing it by 0.1 at each step, and evaluated the accuracy on the validation and trial sets. Figure 1 displays the accuracy on the model-aware validation set, model-agnostic validation set, and the trial set when different threshold values were applied. From this figure, a threshold of 0.6 achieved the highest accuracy on the validation sets and closely approached the highest accuracy on the trial set. Therefore, we selected $\delta = 0.6$ when applying this method to the test set. To further examine the performance of using $\delta = 0.6$, it was applied to determine hallucinations on both the model-aware and model-agnostic training sets. However, since the training set is unlabelled, it is not possible to examine the accuracy. Therefore, our focus shifted to examining the frequency of "Hallucination" and "Not Hallucination" predictions. This was to ensure that using $\delta = 0.6$ would not result in the tendency of exclusively predicting one or the other. As shown in Figure 2, with $\delta = 0.6$, 27.3% and 41.3% of the samples in the model-aware and model-agnostic sets, respectively, were predicted as hallucinations.

SelfCheckGPT-Based Method To select the number of generated responses (N), we varied N from 1 to 5. The accuracy and Spearman correlation results on both model-aware and model-agnostic validation sets, with different values of N, are presented in Figure 3. This figure indicates that as N increased, accuracy generally improved with fluctuations observed in both datasets. However, Spearman correlation consistently increased with no fluctuations as N increased. Therefore, we set N to 5 to obtain five responses for each sample in the test set. All hyperparameters of *gpt-3.5-turbo-1106* were configured with their default values. For any model response that indicated undetermined answers, the corresponding sample was considered

Dataset	Method	Validation set		Test set	
		Accuracy	Spearman correlation	Accuracy	Spearman correlation
Model-aware	Baseline	0.707	0.461	0.745	0.488
	Hypothesis-Target Similarity	0.699	**0.536**	0.734	0.518
	SelfCheckGPT-based	**0.722**	0.510	**0.768**	**0.582**
Model-agnostic	Baseline	0.649	0.380	0.697	0.403
	Hypothesis-Target Similarity	0.699	**0.574**	0.687	0.467
	SelfCheckGPT-based	**0.707**	0.567	**0.728**	**0.595**

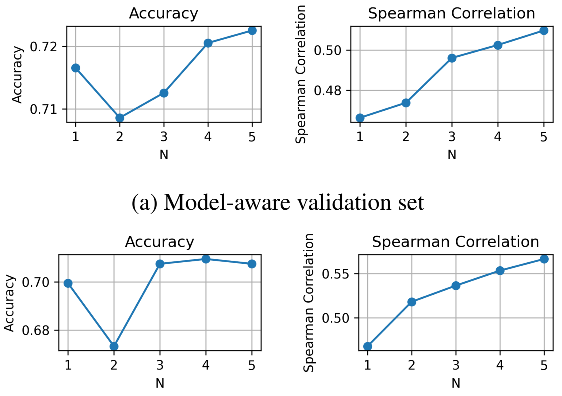
Table 2: Comparative performance of the proposed methods measured by accuracy and Spearman correlation on the validation and test sets, with the highest value in bold



(a) Model-aware unlabelled training set with 0.6 threshold

(b) Model-agnostic unlabelled training set with 0.6 threshold

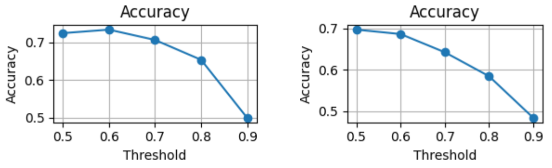
Figure 2: The percentage of “Hallucination” and “Not Hallucination” from the result of the hypothesis-target similarity method



(a) Model-aware validation set

(b) Model-agnostic validation set

Figure 3: Accuracy and Spearman correlation results on both (a) model-aware and (b) model-agnostic validation sets when using different values of N.



(a) Model-aware

(b) Model-agnostic

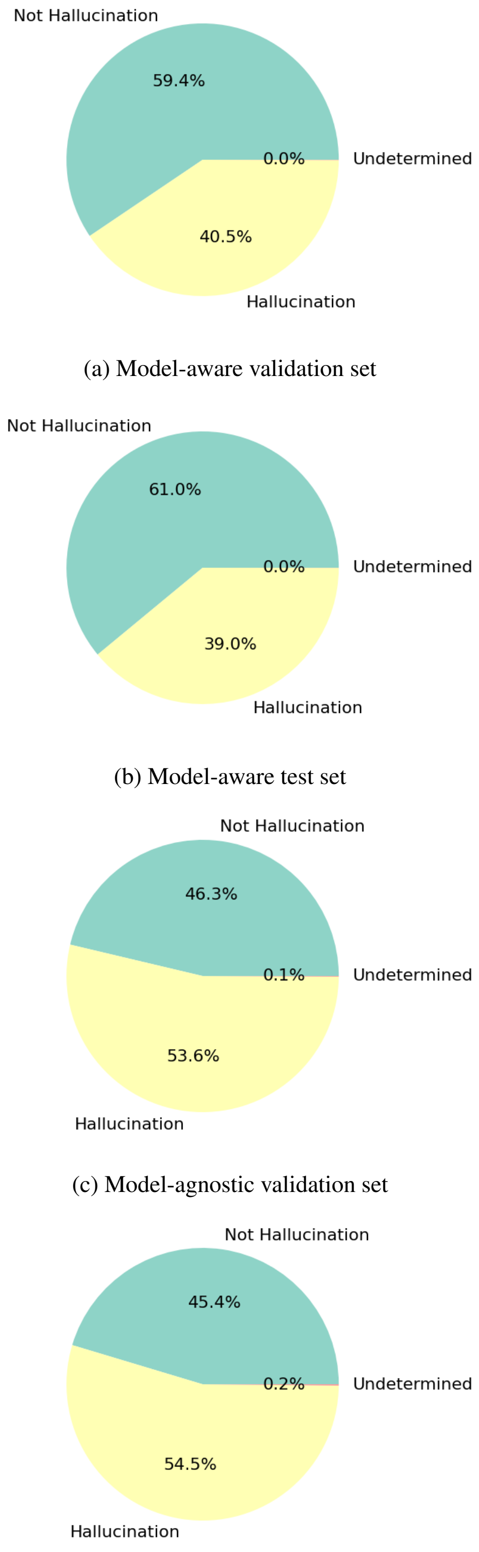
Figure 4: Accuracy results on (a) model-aware and (b) model-agnostic test sets using different thresholds δ.

as “Hallucination”.

5.2 Results and Discussions

Table 2 shows the comparative performance of the proposed methods measured by accuracy and Spearman correlation on the validation and test sets. From this table, the proposed method based on SelfCheckGPT outperformed the baseline in terms of both accuracy and Spearman correlation on both model-aware and model-agnositc datasets. This indicates the effectiveness of using the GPT-3.5 model with prompts that include both source and target as context. Also, it suggests the benefit of running each prompt through an LLM multiple times to obtain a final prediction.

The proposed hypothesis-target similarity method closely approached the performance of the SelfCheckGPT-based approach on the validation sets, showing higher Spearman correlation values. However, on the test sets, the latter surpassed it. The reason could be that the selected threshold might not be precisely suitable for the test sets. Figure 4 shows the accuracy results on model-aware and model-agnostic test sets when different thresholds δ were used. From this figure,



(a) Model-aware validation set

(b) Model-aware test set

(c) Model-agnostic validation set

(d) Model-agnostic test set

Figure 5: The percentages of response types, including "Not Hallucination", "Hallucination", and "Undetermined", obtained from the proposed SelfCheckGPT-based method on (a) model-aware validation set, (b) model-aware test set, (c) model-agnostic validation set, and (d) model-agnostic test set

using $\delta = 0.6$ resulted in the highest accuracy on the model-aware test set. However, on the model-agnostic test set, using $\delta = 0.5$ resulted in better accuracy. This indicates the challenge of selecting an optimal threshold based solely on observed data for generalizing to unseen data.

We investigated the number of undetermined responses in the SelfCheckGPT approach to validate whether this approach can effectively generate definitive answers for this task. Figure 5 shows the percentages of response types, including "Not Hallucination", "Hallucination", and "Undetermined", obtained from the proposed SelfCheckGPT-based method on the model-aware validation set, model-aware test set, model-agnostic validation set, and model-agnostic test set. According to this figure, the proposed SelfCheckGPT approach predicted less than 0.2% of undetermined answers across all datasets. This indicates that the SelfCheckGPT approach is effective in terms of producing definitive answers. Nonetheless, one limitation of this approach is its reliance on the availability of prior knowledge or expected outcomes (which, in this case, are the targets). In real-world situations, such information may not be available.

In the official competition rankings, the top-performing model achieved an accuracy of 0.813 and a Spearman correlation of 0.699 on the model-aware test set, and an accuracy of 0.847 and a Spearman correlation of 0.770 on the model-agnostic test set. Consequently, our SelfCheckGPT model secured the 26th position on the model-aware test set and the 35th position on the model-agnostic tes set.

6 Conclusions

This work proposes two methods for detecting hallucinations and observable overgeneration mistakes in texts generated by LLMs. The first method, the hypothesis-target similarity method, involves calculating the information overlap between a generated text and a reference text. This method utilizes a pre-trained SentenceTransformer model to calculate text embeddings for both the generated and reference texts, and cosine similarity to measure their similarity. The second method employs an LLM-based evaluation approach. It uses the SelfCheckGPT technique with prompts tailored to LLM-generated texts from various NLG tasks. The experimental results highlight the effectiveness of the proposed hypothesis-target similarity method in detecting hallucinations, particularly when the similarity threshold is carefully chosen. Additionally, the findings reveal that the proposed SelfCheckGPT-based method outperformed the baseline, and effectively identified hallucinations in texts generated by LLMs. Moreover, these results underscore the significance of prompt design in evaluating hallucinations using LLMs. However, there is still room for improvement in the performance of our methods.

For future work, other SentenceTransformers models, such as Multi-QA or MSMARCO Passage models (SBERT.net, 2022) or alternative embedding models, such as InferSent (Conneau et al., 2018) or Universal Sentence Encoder (Cer et al., 2018) for the Hypothesis-Target Similarity approach will be considered. As for the SelfCheckGPT-based approach, other LLMs besides GPT-3.5 will also be investigated. Moreover, various prompt formats and the use of few-shot examples in the prompt will be explored.

References

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Daniel Cer, Yinfei Yang, Sheng yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2018. Universal sentence encoder.

Hyunjin Choi, Judong Kim, Seongho Joe, and Youngjune Gwon. 2021. Evaluation of bert and albert sentence embedding performance on downstream nlp tasks. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 5482–5487.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loic Barrault, and Antoine Bordes. 2018. Supervised learning of universal sentence representations from natural language inference data.

Bhuwan Dhingra, Manaal Faruqui, Ankur Parikh, Ming-Wei Chang, Dipanjan Das, and William W Cohen. 2019. Handling divergent reference texts when evaluating table-to-text generation. *arXiv preprint arXiv:1906.01081*.

Robert Friel and Atindriyo Sanyal. 2023. Chainpoll: A high efficacy method for llm hallucination detection. *arXiv preprint arXiv:2310.18344*.

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

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, Deep Ganguli, Danny Hernandez, Josh Jacobson, Jackson Kernion, Shauna Kravec, Liane Lovitt, Kamal Ndousse, Catherine Olsson, Sam Ringer, Dario Amodei, Tom Brown, Jack Clark, Nicholas Joseph, Ben Mann, Sam McCandlish, Chris Olah, and Jared Kaplan. 2022. Language models (mostly) know what they know.

Yung-Shen Lin, Jung-Yi Jiang, and Shie-Jue Lee. 2014. A similarity measure for text classification and clustering. *IEEE Transactions on Knowledge and Data Engineering*, 26(7):1575–1590.

Potsawee Manakul, Adian Liusie, and Mark JF Gales. 2023. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. *arXiv preprint arXiv:2303.08896*.

Thanet Markchom, Bhuvana Dhruva, Chandresh Pravin, and Huizhi Liang. 2020. UoR at SemEval-2020 task 4: Pre-trained sentence transformer models for commonsense validation and explanation. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 430–436, Barcelona (online). International Committee for Computational Linguistics.

Marianna Martindale, Marine Carpuat, Kevin Duh, and Paul McNamee. 2019. Identifying fluently inadequate output in neural and statistical machine translation. In *Proceedings of Machine Translation Summit XVII: Research Track*, pages 233–243, Dublin, Ireland. European Association for Machine Translation.

Timothee Mickus, Elaine Zosa, Raúl Vázquez, Teemu Vahtola, Jörg Tiedemann, Vincent Segonne, Alessandro Raganato, and Marianna Apidianaki. 2024. SemEval-2024 Task 6: SHROOM, a shared-task on hallucinations and related observable overgeneration mistakes. In *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)*, Mexico City, Mexico. Association for Computational Linguistics.

OpenAI. 2022. Gpt-3.5 turbo. `https://platform.openai.com/docs/models/gpt#gpt35-turbo`.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation.

SBERT.net. 2022. Pretrained models. `https://www.sbert.net/docs/pretrained_models.html`.

Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. 2021. Retrieval augmentation reduces hallucination in conversation. *arXiv preprint arXiv:2104.07567*.

Ruichen Zhang, Zeshui Xu, and Xunjie Gou. 2023. Electre ii method based on the cosine similarity to evaluate the performance of financial logistics enterprises under double hierarchy hesitant fuzzy linguistic environment. *Fuzzy Optimization and Decision Making*, 22(1):23–49.