DUTh at SemEval-2024 Task 6: Comparing Pre-trained Models on Sentence Similarity Evaluation for Detecting of Hallucinations and Related Observable Overgeneration Mistakes

Ioanna Iordanidou **Ioannis Maslaris** **Avi Arampatzis**

Database & Information Retrieval research unit,
Department of Electrical & Computer Engineering,
Democritus University of Thrace, Greece.
{ioaniord1,imaslari,avi}@ee.duth.gr

Abstract

In this paper, we present our approach to SemEval-2024 Task 6: SHROOM, a Shared-task on Hallucinations and Related Observable Overgeneration Mistakes, which aims to determine weather AI generated text is semantically correct or incorrect. This work is a comparative study of Large Language Models (LLMs) in the context of the task, shedding light on their effectiveness and nuances. We present a system that leverages pre-trained LLMs, such as LaBSE, T5, and DistilUSE, for binary classification of given sentences into 'Hallucination' or 'Not Hallucination' classes by evaluating the model's output against the reference correct text. Moreover, beyond utilizing labeled datasets, our methodology integrates synthetic label creation in unlabeled datasets, followed by the prediction of test labels.

1 Introduction

Hallucinations in machine generated text are cases when the model generates output that is partially or fully unrelated to the source sentence. While being a non-frequent phenomenon, it can dramatically impact the user experience and the trust toward the system. Hallucination rates in multiple models vary from 2.8 to 16.2 percent, according to the hallucination leaderboard created by Vectara hosted on HF and GitHub. [1] While the problem of hallucinations is known, it remains challenging, and one aspect of that is the absence of proper datasets. As a result, previous studies relied on scenarios where models are encouraged to hallucinate (Lee et al., 2018; Raunak et al., 2021; Müller et al., 2019; Voita et al., 2020; Zhou et al., 2020). However, it is uncertain if these approaches are effective in more natural, undisturbed environments (Guerreiro et al., 2022).

Recent research conducted in relatively clean settings (Guerreiro et al., 2022) demonstrates that existing hallucinations detection methods fall short. The authors create a natural setting dataset, annotate it for various NMT (Neural Machine Translation) pathologies, and evaluate detection methods. They find most existing detection methods inadequate, with sequence log-probability performing best. Although they demonstrated interesting results, they were limited on detecting hallucinations on Machine Translation (MT) generated text.

SemEval-2024 task 6 (Mickus et al., 2024) goes a step further by providing a human annotated dataset of hallucinated text regarding three different scenarios. Along with Machine Translation (MT), also Definition Modeling (DM) and Paraphrase Generation (PG) cases are considered. This paper describes the system developed by the DUTh team for SemEval-2024 task 6. Our strategy is based on utilizing embeddings to evaluate the similarity between context and hypothesis sentences in order to detect hallucinated text. In our case context sentence is the 'gold' output expected from to the models for generation and hypothesis sentence is the actual model production. For generating the embeddings of the context and hypothesis we are utilizing a pretrained T5 tokenizer (Raffel et al., 2020). Then we measure their similarity by taking the dot product of their corresponding embeddings, followed by summation along axis 1. Finally, using that similarity score, we train an ensembler machine learning model to distinguish hallucinated form non-hallucinated text. We provide our code publicly [2]

2 Background

2.1 Related work

Methods for identifying hallucinations are primarily concentrated on the Machine Translation task and they generally aim to find translations of poor quality that may also satisfy additional constraints. To effectively pinpoint factual inaccuracies in LLM

[1] https://huggingface.co/spaces/vectara/leaderboard

[2] https://github.com/DataMas/ai-hallutinations-detection

Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024), pages 1064–1070

outputs, one straightforward strategy involves comparing the model generated output information from an external knowledge source. Relevant research, starting from traditional fact checking (Augenstein et al., 2019) tries to expand the capabilities of such systems by incorporating various web sources (Chen et al., 2023) and evaluating their truthfulness (Galitsky, 2023). Recently, there is a significant emphasis on enhancing the process of retrieving information from external sources. FACTSCORE introduced by (Min et al., 2023), is a metric specifically for long-text generation. The LLM output is decomposed into atomic facts and each one is validated by reliable external knowledge sources. Furthermore, (Huo et al., 2023), enhanced the retrieving process by augmenting the query to the external sources with the input to and the output of the LLM.

When utilizing external sources, previous research mostly focused on evaluating models output based on a pool of third party knowledge. However, implementation of such systems could be complicated. Similarity between the source and the target estimated via embeddings, has been proved to be a good indicator for hallucinations in Machine Translation scenarios (Dale et al., 2022). In this manner, we are experimenting with this strategy on detecting hallucinations on machine generated text regarding Definition Modeling and Paraphrase Generation. We hypothesize that hallucinations can have a great impact on the conceptual content of the generated text, enough to be detected through sentence similarity evaluation.

2.2 Dataset

The task provided three datasets for each track: the train and test sets comprised unlabeled datapoints, and the validation set contained labeled datapoints enriched with additional features. Each datapoint in the labeled set encompasses the following attributes. The 'model' attribute is included only in the model-aware datasets and the items without bold annotation are not featured in the test datasets.

- **"id":** The datapoint's ID
- **"task":** The model's optimization objective (DM, PG, MT).
- **"model":** The model used for text generation.
- **"tgt":** The intended reference text for model generation.
- **"src":** The input presented to the models for generation.
- **"hyp":** The actual model output.
- **"ref":** Indicates whether the 'tgt' or 'src' fields, or both, are the context that contains the requisite semantic information to discern the datapoint as a hallucination.
- "labels": A set of per-annotator labels gauging whether each annotator perceives the datapoint as a hallucination.
- "label": The majority-based gold-label derived from the per-annotator labels.
- "p(Hallucination)": The probability assigned to the datapoint being a hallucination based on the proportion of annotators considering it as such.

In the model-aware segment, we dived deeper into our data by visually representing (Figure 1) the distribution of three distinct models across data points. In both the validation and test sets, two of the three models exhibit an equal distribution, while the third one, tuner007/pegasus_paraphrase, is utilized less, accounting for roughly 33 percent. The training set shows an equal distribution of all three models.

3 System Overview

3.1 Hallucination Detection

Hallucination detection methods are a developing field in modern NLP. Given input information and parameters can vary and subsequently the methods applied for detection are subject to change. SemEval-2024 Task 6: SHROOM - a Shared-task on Hallucinations and Related Observable Overgeneration Mistakes was divided into two tracks. The first sub-task, Model-Aware, involves determining whether the model produced a hallucination, given information about which model was employed. The second sub-task, Model-Agnostic, pertains to scenarios where the model used is unknown.

3.2 System

We approached the task as a binary text classification problem, implementing our system leveraging the HuggingFace Transformer library. Concisely, our methodology aligns with the conventional approach to addressing text classification problems — training a model with a large labeled dataset and employing it to predict labels for the test set. We first opted for the labeled dataset provided by (Guerreiro et al., 2022) for the training process. Then

we tried the unlabeled training dataset supplied by the task organizers, conducting experiments to automatically generate synthetic labels for its utilization into the training process.

In summary, our system comprised distinct steps, including extraction of embeddings for 'hyp' and 'context', calculation of cosine similarity between the two, generation of synthetic labels for the training set through clustering, and prediction of the test set using ensembled classifiers.

3.2.1 Sentence Embeddings

Sentence embeddings are a potent technique utilizing deep learning models, specifically transformers, to encode words, or in our context, sentences, into vectors. These vectors capture the semantic meaning and contextual information of the input text. This encoding is valuable as vectors provide a robust representation of the semantic content embedded in sentences and are more efficiently compared or handled in any way for various NLP tasks. Our approach employed pre-trained sentence transformers sourced from the HuggingFace library (v. 2.2.2) for extracting these embeddings. From our dataset, the hypothesis ('hyp') was compared to the context sentence provided by the semantic reference ('ref'). The cosine similarity between the vectors resulting from this comparison, along with a probability measure of hallucination, was subsequently incorporated into our system. Formally, the similarity score, denoted as sims, is calculated as

$$\text{sims} = \sum_{i=1}^{n} (\text{emb\_con}_i \cdot \text{emb\_hyp}_i)$$

where emb_con_i and emb_hyp_i represent the embeddings for the i-th context and hypothesis, respectively. In this numerical measure of similarity, higher values indicate greater similarity between the encoded representations of hypotheses and contexts. The probability is computed as $1 - \text{sims}$ and is subsequently appended to the dataset.

3.2.2 Synthetic Labels Creation

Cluster analysis is a technique used in data mining and machine learning to group similar objects into clusters. k-means clustering is a popular unsupervised machine learning algorithm with vector inputs that allocates every data point to the nearest cluster. Synthetic data creation has become a widely adopted methodology within the NLP field, notably used for the purpose of label generation (Zhou et al., 2020). In our pursuit of generating synthetic labels for the unlabeled dataset, we employed the k-means algorithm with $k = 2$, signifying two centroids, to extract 'Hallucination' and 'Not Hallucination' labels. The parameters provided for clustering were the cosine similarity and the probability derived from sentence embeddings within the model-agnostic sub-task. Additionally, for the model-aware subtask, we incorporated the one-hot encoded representation of the utilized model as an additional parameter. We additionally tested the efficacy of our label extraction mechanism on the provided labeled datasets, achieving a notable accuracy rate of 75 percent.

3.2.3 Label Prediction

Following the training phase, we engaged in an ensemble approach, combining several widely recognized classification algorithms to forecast the labels of the test set and identify instances of hallucination. In text classification, each data point is allocated a label, with binary classification typically involving two labels (e.g., 0 and 1). Model ensembling aims to utilize the collective strength of various classifiers to maximize overall performance. We employed the similarity extracted from the embeddings and integrated it as a feature alongside the probability in the training of our classifiers. In our ensembling, we incorporated the following classifiers: Logistic Regression, Random Forest, Gradient Boosting, K Nearest Neighbours, XGBoost and Decision Tree. By employing this methodology, we got labeled test sets as the final outputs.

3.3 Models

Central to our system are pre-trained models from the sentence transformers library of HuggingFace (v. 2.2.2). The models we distinguished were DistilUSE (Reimers and Gurevych, 2019), LaBSE (Feng et al., 2020) and T5 (Raffel et al., 2020). The 'distiluse-base-multilingual-cased-v2' model excels at mapping sentences to a 512-dimensional dense vector space, making it ideal for tasks like clustering and semantic search. With its multilingual capabilities and nuanced representation of case information, it proves valuable across various languages for applications requiring semantic understanding and similarity assessment. LaBSE, Language-agnostic BERT sentence embedding, supports 109 languages and adopts a dual-encoder approach based on pretrained transformers. It has been fine-tuned for translation ranking with an ad-

ditive margin softmax loss. T5, or Text-To-Text Transfer Transformer, is adept at mapping sentences to a 768-dimensional dense vector space. This model particularly excels in tasks related to sentence similarity. This selection of models, ranging from BERT to LaBSE and T5, offers a diverse toolkit for our system. These pre-trained models, with their distinct architectures and capabilities, contribute to the robustness and versatility of the implemented system across a spectrum of natural language processing tasks.

Model	Model-Agnostic		Model-Aware	
	Score 1	Score 2	Score 1	Score 2
LaBSE	0.7366	0.7366	0.7440	0.7440
T5	0.7440	0.7440	0.7553	0.7553
DistilUSE	0.7066	0.7367	0.6867	0.7440

Table 1: Accuracy for all models. Score 1 is using synthetic labeled train set and score 2 is using the Guerreiro set.

Model	Model-Agnostic		Model-Aware	
	Score 1	Score 2	Score 1	Score 2
LaBSE	0.4298	0.4298	0.4277	0.4277
T5	0.4748	0.5224	0.5285	0.5255
DistilUSE	0.3051	0.3576	0.2988	0.3269

Table 2: Spearman Correlation for all models. Score 1 is using synthetic labeled train set and score 2 is using the Guerreiro set.

4 Experimental setup

4.1 Preprocessing

Prior to any NLP problem solving , performing text preprocessing is necessary. The nature of text preprocessing varies depending on the methodology to be employed, encompassing various steps. In the context of our binary classification problem utilizing the provided dataset, we conducted thorough feature extraction and preprocessing on the raw textual data. Specifically, we opted for the English language model available in SpaCy's trained pipelines (v. 3.7.2). This choice was particularly informed by the necessity to preprocess the 'hyp' and 'context' features, being aware that the context in the Machine Translation (MT) task was in English. Across all tasks, our text preprocessing included text lowercase conversion, punctuation removal, and lemmatization, where custom lemmas were incorporated. For the Definition Modeling task, we extracted the word to define from the context. In the model-aware track of the task, we introduced one-hot encoding representation of the model used for all datapoints. Similar techniques were used for the (Guerreiro et al., 2022) dataset adapting to the corresponding feature names.

4.2 Experiments

The conducted experiments incorporated the entirety of available datasets, the training, development, and test sets. Our initial experiment, as outlined in the system section, involved the utilization of the synthetically labeled train and test sets in conjunction with the DistilUSE, LaBSE, and T5 models. In the next experiment, we only utilized T5 and skipped the synthetic labeling phase from our methodology. In this iteration, the training process was conducted by utilizing the (Guerreiro et al., 2022) dataset. This adjustment was motivated by the labeled nature of this set, making it conducive to predicting labels for the test set in both tracks of the task.

4.3 Evaluation

The evaluation measures employed in both tracks of the task were consistent. The initial metric pertained to a general accuracy score, derived from the test reference data provided by the task organizers, applied to our binary classification results. Subsequently, the evaluation for the model-agnostic track extended to include the Spearman's correlation coefficient, a statistical measure of the strength of a monotonic relationship between the output probabilities of the systems and the proportion of annotators marking an item as overgenerating. The Spearman correlation assesses the degree to which the systems' output probabilities align with the consensus among annotators, offering a nuanced evaluation of the models' performance in capturing the observed trends in overgeneration perception. Both metrics have a maximum value of 1.

5 Results

The comprehensive scores of our system across the three utilized models are presented in Tables 1 and 2, for accuracy and correlation, respectively. The highest accuracy score was T5's 0.7553 in model-aware which ranked 25th out of 38 and 0.7440 in model-agnostic which ranked 27th out of 41 while both passed the baseline scores in accuracy and correlation. There was no difference in the dataset used for the training process. The baseline score was obtained through using an instruction-finetuned Mistral model tasked with classifying the sentences as contextual or not, answering with

Algorithm	Model-Agnostic		Model-Aware	
	Score 1	Score 2	Score 1	Score 2
Logistic Regression	0.6874	0.7066	0.7086	0.7160
Random Forest	0.6873	0.7420	0.7380	0.7347
Gradient Boosting Classifier	0.6874	0.7327	0.7067	0.7393
K Nearest Neighbours	0.6867	0.6740	0.7067	0.6687
Decision Tree	0.7067	0.7447	0.7367	0.7493
XGBoost	0.7067	0.6787	0.7407	0.6887
Ensembling	**0.7440**	0.7447	**0.7553**	0.7373

Table 3: Evaluation Metrics for Seven Machine Learning Algorithms. Score 1 is using synthetic labeled train set and score 2 is using the Guerreiro set.

a yes or no. The accuracy score it achieved was 0.697 in the model-agnostic track and 0.745 in the model-aware track. If ranked by correlation, T5 scores highest with a moderate correlation of 0.5285 for the aware track using the synthetically labeled dataset and 0.5224 in the agnostic track using the Guerreiro dataset, also surpassing the baseline system which scored 0.488 and 0.403 respectively. Consequently, after careful consideration, T5 was selected for integration into our final system. For the label prediction part we tried multiple Machine Learning classification algorithms which are shown in Table 3. In both tracks using the synthetic labeled dataset, distinctions, ranging from subtle in some cases to more pronounced in others, were observed among individual algorithms. However, the ensemble strategy consistently surpassed their individual performances, scoring 0.7440 in model-agnostic and 0.7553 in model-aware. When applying the Guerreiro dataset, Decision Tree outperformed the ensemble strategy in the model-aware track, consistently staying below 0.7553. However, this did not hold in the model-agnostic track, where both Decision Tree and the ensemble of all seven algorithms achieved a score of 0.7447, closely mirroring Score 1. Based on the previously mentioned outcomes, the score obtained through the ensemble of classifiers using the synthetically labeled set was ultimately submitted to the tasks leaderboard.

6 Conclusion

Through these experiments, we found that the pretrained T5 model exhibits optimal performance in the detection of hallucinated text in the domain of artificial intelligence. Furthermore, we successfully employed an ensemble of multiple popular top-tier classifiers to augment the predictive capabilities of our system and investigated the implications of synthetically labeling unlabeled data, presenting it as a novel approach to hallucination detection.

The next step could involve an extended comparison of various language models to identify the most powerful one, as well as exploring the option of training on diverse and larger datasets. Additionally, for further exploration, we recommend fine-tuning a Language Model (LLM) to extract enhanced embeddings, thereby improving the accuracy of sentence similarity assessments and consequently bolstering the overall system performance.

References

Isabelle Augenstein, Christina Lioma, Dongsheng Wang, Lucas Chaves Lima, Casper Hansen, Christian Hansen, and Jakob Grue Simonsen. 2019. Multifc: A real-world multi-domain dataset for evidence-based fact checking of claims. *arXiv preprint arXiv:1909.03242*.

Jifan Chen, Grace Kim, Aniruddh Sriram, Greg Durrett, and Eunsol Choi. 2023. Complex claim verification with evidence retrieved in the wild. *arXiv preprint arXiv:2305.11859*.

David Dale, Elena Voita, Loïc Barrault, and Marta R Costa-jussà. 2022. Detecting and mitigating hallucinations in machine translation: Model internal workings alone do well, sentence similarity even better. *arXiv preprint arXiv:2212.08597*.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2020. Language-agnostic bert sentence embedding. *arXiv preprint arXiv:2007.01852*.

Boris A Galitsky. 2023. Truth-o-meter: Collaborating with llm in fighting its hallucinations.

Nuno M Guerreiro, Elena Voita, and André FT Martins. 2022. Looking for a needle in a haystack: A comprehensive study of hallucinations in neural machine translation. *arXiv preprint arXiv:2208.05309*.

Siqing Huo, Negar Arabzadeh, and Charles LA Clarke. 2023. Retrieving supporting evidence for llms generated answers. *arXiv preprint arXiv:2306.13781*.

Katherine Lee, Orhan Firat, Ashish Agarwal, Clara Fannjiang, and David Sussillo. 2018. Hallucinations in neural machine translation.

Timothee Mickus, Elaine Zosa, Raúl Vázquez, Teemu Vahtola, Jörg Tiedemann, Vincent Segonne, Alessandro Raganato, and Marianna Apidianaki. 2024. Semeval-2024 shared 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)*, pages 1980–1994, Mexico City, Mexico. Association for Computational Linguistics.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. *arXiv preprint arXiv:2305.14251*.

Mathias Müller, Annette Rios, and Rico Sennrich. 2019. Domain robustness in neural machine translation. *arXiv preprint arXiv:1911.03109*.

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. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Vikas Raunak, Arul Menezes, and Marcin Junczys-Dowmunt. 2021. The curious case of hallucinations in neural machine translation. *arXiv preprint arXiv:2104.06683*.

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

Elena Voita, Rico Sennrich, and Ivan Titov. 2020. Analyzing the source and target contributions to predictions in neural machine translation. *arXiv preprint arXiv:2010.10907*.

Chunting Zhou, Graham Neubig, Jiatao Gu, Mona Diab, Paco Guzman, Luke Zettlemoyer, and Marjan Ghazvininejad. 2020. Detecting hallucinated content in conditional neural sequence generation. *arXiv preprint arXiv:2011.02593*.

A Appendix

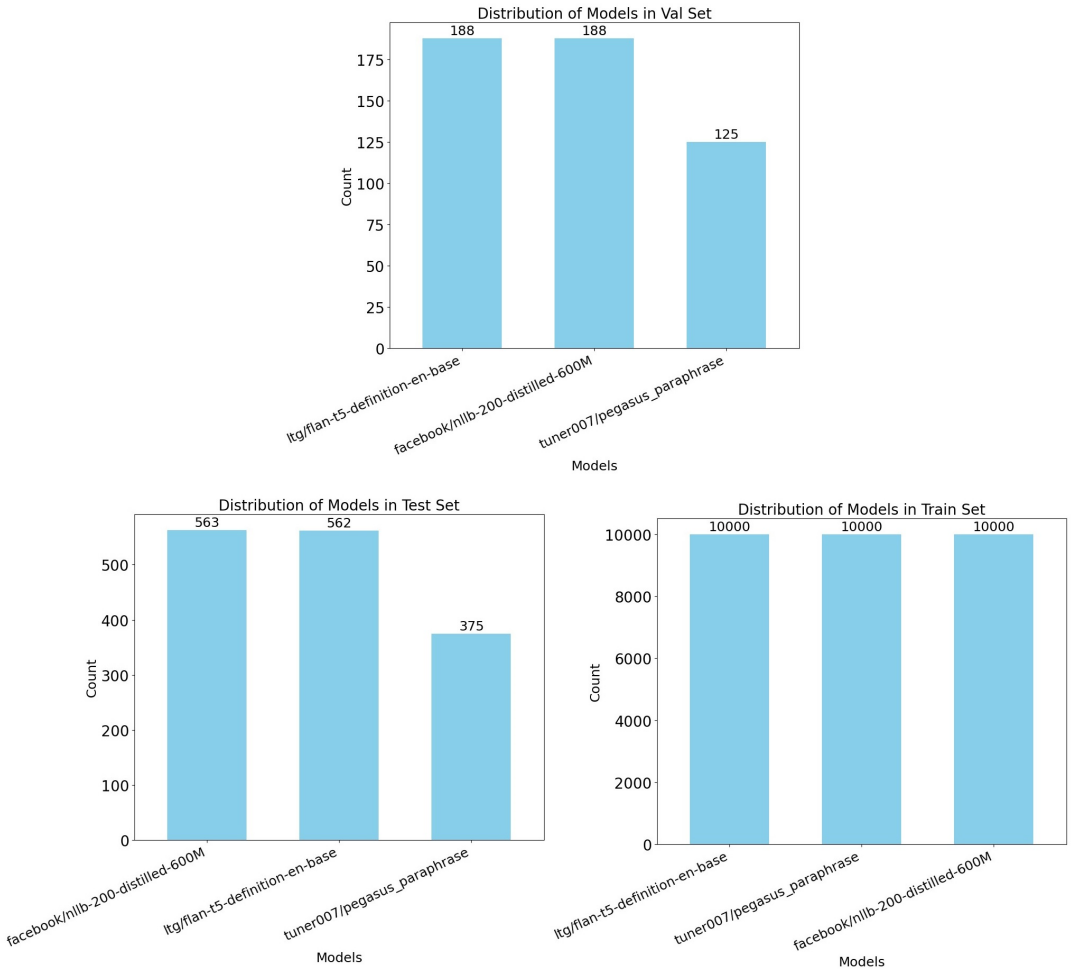

Figure 1: Distribution of Models used in Datasets.