Management Summary

Project Title:
Hallucination Detection

Group 2

Authors:

Hanna Müller (12407821)
Rita Selimi (12332281)
Theresa Brucker (01609010)
Rea Kasumi (12332282)

194.093 Natural Language Processing and Information Extraction

Date:
January 25, 2025

1 Background

Our project focused on developing a model to detect hallucinations in large language models (LLMs) across different tasks. In this context, hallucinations refer to fluent outputs that contain incorrect or unsupported information. The interpretation of hallucinations varies depending on the task the LLM is performing. We concentrated on three major tasks:

1. **Machine Translation (MT)**: The LLM translates a sentence from one language to another. A hallucination occurs when the translation fails to capture one or more meanings of the input sentence.
2. **Definition Modelling (DM)**: The LLM defines a word within a specific context. A hallucination is present when the output provides an incorrect definition due to missing or inaccurate information.
3. **Paraphrase Generation (PG)**: The LLM transforms a sentence into a paraphrase while preserving its semantic meaning. Hallucinations occur when the transformation adds, removes, or alters critical information, thereby failing to retain the original meaning.

To establish a unified approach, we focused on detecting **fluent overgeneration**, which involves comparing the model's output to a reference sentence. If the output contains information not supported by the reference, it is classified as a hallucination. For MT, the reference is a "gold-standard" translation; for DM, it is a "gold-standard" definition; and for PG, the reference is the input sentence itself, which serves as the baseline for transformation.

2 Methodology

Our approach can be split up in multiple steps which are discussed in the following section.

2.1 Data Inspection & Preprocessing

Our dataset consisted of training, validation and testing data. Each consisted of outputs for the three tasks with a task label including a reference to compare the output with. In addition, the data was already labeled as Hallucination or Non-Hallucination. For the test and validation data, this label was done by a majority voting of 5 non-experts that had the criterium *"Does the following AI output only contain information supported by the Reference?"* to classify the output as Non-Hallucination. The labels of the training data were generated by a *gpt-3.5-turbo* model.

When looking at the test data and its label, we already noticed some problems regarding the labels. For example the model output *"Perfect in every way."* with the reference *"(archaic) Wholly perfect."* was labeled as Hallucination although there was no wrong or added information. This was not the only example were we did not agree with the labels which can also affect the later accuracy of our model and has to be kept in mind.

The preprocessing focused on the output and reference sentence which we brought to the same format and transformed the words into so called lemmas which only keeps the word in its base form to make it better comparable.

2.2 Baselines

We created two models as baseline, introducing them as a minimal standard of performance to achieve. We will later compare our more advanced models to these baselines.

Our first model is a simple Machine Learning Model called **Naive Bayes** which learns from the labeled training data. For this, the data was first transformed into vectors with *TF-IDF vectorization*, converting the words into numbers based on how frequently they appear, with which the model can then work. This model showed a non-satisfactory result by only correctly classifying 59% of the test examples correctly. This can be explained by the oversimplification of the model which assumes that the words are independent as well as the lack of semantic understanding which is crucial when it comes to detecting hallucinations.

Our second baseline was a more advanced deep learning model **Vectara** which is already trained on the task of hallucination detection. You can input a reference and an output and the Vectara model outputs how much the output is supported by the reference on a scale of 0 to 1. To create a threshold when it is a Hallucination we used the validation data. With our test data we then achieved an accuracy of 70% with the model performing a bit better in precisely detecting Hallucinations.

2.3 Advanced Models

2.3.1 Feature-based Classifier

To improve the performance of a machine learning (ML) model, we tried to extract features from the data that helps the model in classifying something as a Hallucination. For this we designed 4 features. The first one is a word overlap score which caluclates the percentage of words of the reference that are contained in the output. To focus more on a sentence level, the second feature did not only look at words but at word sequences, also called n-grams. To also capture specific characteristics of the sentences, we additionally used a named entitity overlap, which computes how many words are labeled the same in reference and output. The last feature focused more on the semantics and compared the similarity between the semantic embeddings.

These features were then used as input for a Random Forest Model, which was the ML approach that worked the best. The model was trained on the training data where we balanced out the number of Hallucinations and Non-Hallucinations and finally tested on the testing data. We observed that 75% of the testing data was classified correctly with an advantage for the classification of Hallucinations. When looking at the problems of the classifier, we noticed that they were mainly due to semantic differences between reference and output despite high similarity regarding word use. Therfore, including more than one feature covering the meaning of the sentence could help in improving the results.

2.3.2 BERTScore

In the initial approach, we used BERTScore to measure semantic similarity between hypotheses and references for hallucination detection. BERTScore leverages pre-trained embeddings (e.g., RoBERTa-large) which can be used to detect similarity between output and reference. Hypotheses were classified as "Hallucination" or "Not Hallucination" based on a threshold set on the performance for the validation data. However, since the pre-trained model was not fine-tuned for hallucination detection, it struggled with task-specific nuances.

To improve performance, we fine-tuned a RoBERTa-based model using labeled data for hallucination detection. This allowed the model to learn patterns specific to our task, improving its ability to detect subtle inconsistencies in the data. Fine-tuning helped the model adapt to the datasets, resulting in more accurate predictions compared to relying solely on pre-trained embeddings.

The model's performance varies across tasks but predicts only 66% of the training data correctly. For **PG**, it struggles with hallucination detection but performs well on non-hallucinations of which it detects 90%. For **MT**, there is also a bias toward non-hallucinations. For **DM**, the model performs the worst with only classifying 62% of the instances correctly. Overall, the model struggles to balance the predictions for Hallucinations and Non-Hallucinations.

3 Results

The feature-based classifier demonstrated the best performance, achieving an accuracy of 75%. Even majority voting with various rating scenarios failed to improve performance, indicating that the feature-based classifier significantly outperforms BERTScore and Vectara.

4 Conclusions and Recommendations

On our analysis of various models revealed that those with an understanding of semantic meanings perform more effectively in hallucination detection, as hallucinations are deeply connected with the underlying meaning of the produced output. Given that our best performing model is feature-based, a potential improvement could involve the usage of even more features designed to capture and describe the semantic relations within sentences more efficiently.

The advantage of having a machine-learning approach with self-designed features is also that the model is more explainable than using a deep learning model which is also often described as a black box. Additionally when having extracted the features, the training process is less energy consuming and therefore making it an efficient strategy when it comes to classifying hallucinations. Because of the already good results the classifier achieved with only four features, this model shows potential for future improvements and optimizations in the topic of hallucination detection.