Hallucination Detection In Large Language Models (LLMs)

Karthikeyan S (3122225001056), Krithika C (3122225001066), and Lavanya Vasudevan (3122225001067)

Department of Computer Science and Engineering Sri Sivasubramaniya Nadar College of Engineering, Kalavakkam, Chennai

Abstract. The surge in applications of Large Language Models (LLMs) [1] has prompted concerns about the generation of misleading or fabricated information, known as hallucinations. Therefore, detecting hallucinations has become critical to maintaining trust in LLM-generated content. Supervised learning [2] offers a robust approach to hallucination detection by training classifiers on labeled datasets to distinguish between factual and hallucinated outputs. This approach enhances the reliability and interpretability of AI systems in domains like healthcare and computer vision. Experimental results demonstrate improved accuracy and robustness, making it a promising solution for real-world applications requiring high precision.

Keywords: Large Language Models (LLMs) · Artificial Intelligence (AI) · Generative Pre-trained Transformer (GPT) · Bidirectional Encoder Representations from Transformers (BERT) · Term Frequency-Inverse Document Frequency (TF-IDF) · Adaptive Synthetic Sampling (ADASYN)

1 Introduction

In today's rapidly evolving landscape of machine learning, large language models (LLMs) have emerged as transformative forces shaping various applications such as question answering, summarization, translation, and content generation. However, a major challenge persists: hallucinations—fabricated or factually incorrect outputs that undermine trust in LLM-generated content. These hallucinations often arise when the desired information is absent from the model's training data or when reasoning within retrieved knowledge fails. The impact of hallucinations can be especially detrimental in critical domains like healthcare, law, and scientific research, where accuracy and reliability are paramount. To address this issue, supervised learning offers a robust approach to hallucination detection. By leveraging labeled datasets, classifiers can be trained to distinguish between factual and hallucinated outputs, enabling the identification of inaccuracies in LLM-generated responses. This approach enhances the reliability and transparency of AI systems while building user trust. Furthermore, it is a vital step toward ensuring that LLMs deliver precise and dependable information. In addition, techniques like Retrieval-Augmented Generation (RAG) [3] attempt to minimize hallucinations by retrieving relevant information from external knowledge bases. While effective, RAG systems are not immune to hallucinations due to errors in retrieval or reasoning. As a result, detecting hallucinations in LLM outputs remains a crucial area of focus. This project explores the use of supervised learning to develop an effective hallucination detection framework, providing a pathway to creating robust and trustworthy AI systems for real-world applications.

1.1 Problem Statement

This report focuses on detecting hallucinations in Large Language Models (LLMs) using supervised learning, ensuring accurate and trustworthy outputs for real-world applications. Hallucination detection in Large Language Models (LLMs) [4] has profound applications in critical domains where accuracy, reliability, and trust are of utmost importance. In healthcare, it ensures that AI systems provide accurate medical diagnoses, treatment recommendations, and documentation, preventing life-threatening errors. In the legal domain, it safeguards the integrity of AI-generated contracts, legal opinions, and compliance documents, avoiding costly misinterpretations or inaccuracies. In finance, hallucination detection helps ensure the reliability of market predictions, risk assessments, and automated reports, reducing the chances of decisions based on fabricated insights. Similarly, in education, it ensures factual accuracy in AI-generated learning materials, enhancing trust in AI tutors. Beyond these, applications in autonomous systems, such as driverless vehicles and robotic assistants, rely on hallucination detection to prevent misjudgments that could result in harm. As AI continues to evolve, hallucination detection will play a pivotal role in ensuring safe and responsible deployment in real-world, high-stakes scenarios.

2 Literature Survey

This section explores the existing literature surrounding the critical issue of hallucinations in Large Language Models (LLMs) and the advancements made to address this phenomenon.

2.1 Large Language Models (LLMs)

Large Language Models (LLMs) are advanced AI systems that use transformer-based architectures, such as GPT [5] and BERT [6], to process and generate human-like text. Trained on vast datasets, these models encode contextual information through self-attention mechanisms, enabling them to perform tasks like question answering, translation, and summarization with high accuracy. Their pre-trained knowledge allows for minimal fine-tuning on specific applications, making them versatile across domains. For instance, GPT-3 [7] demonstrated exceptional zero-shot and few-shot learning capabilities, setting a benchmark for natural language understanding and generation.Their ability to understand and generate contextually relevant text has revolutionized various industries, including healthcare, education, and customer service, paving the way for innovative applications and advancements in artificial intelligence.

2.2 Hallucinations in Large Language Models (LLMs)

One of the major downsides of Large Language Models (LLMs) is their tendency to generate hallucinations—outputs that are factually incorrect, fabricated, or irrelevant to the given input. While these issues may seem trivial in low-stakes scenarios, they pose significant risks in critical domains. For instance, in healthcare, hallucinated outputs can result in incorrect medical advice, misdiagnosis, or dangerous treatment suggestions, jeopardizing patient safety. In legal applications, hallucinations could lead to the generation of misleading case summaries or incorrect interpretations of legal statutes, potentially impacting judicial decisions. Similarly, in finance, the generation of inaccurate reports or fabricated data can result in financial losses and damage to institutional credibility. These risks are compounded by the confidence with which LLMs present their outputs, making it difficult for users to discern the validity of the information. As such, the presence of hallucinations underscores the urgent need for robust detection mechanisms, enhanced verification processes, and stringent evaluation frameworks to ensure reliability in high-stakes domains.

2.3 Retrieval-Augmented Generation (RAG)

Retrieval-Augmented Generation (RAG) is a promising framework designed to improve the factual accuracy of Large Language Models (LLMs) by combining them with external knowledge retrieval systems. In RAG, relevant information is fetched from a knowledge base or database using techniques like sparse or dense retrieval and is then fed into the LLM alongside the input query to generate more grounded outputs. While RAG addresses some issues of hallucinations by anchoring responses to external data, it does not entirely eliminate the problem. The effectiveness of RAG depends on the quality of the retrieved data. If the retrieval system provides incomplete, irrelevant, or outdated information, hallucinations can still occur as the LLM extrapolates beyond the given input. Even with accurate retrievals, hallucinations may arise due to misinterpretation or reasoning errors, particularly for tasks requiring complex synthesis. Thus, while RAG helps ground responses, it cannot fully eliminate hallucinations, emphasizing the need for complementary detection of hallucinations.

2.4 Supervised Learning for Hallucination Detection

Supervised learning is a key approach for detecting hallucinations in the outputs of Large Language Models (LLMs), relying on labeled datasets to train classifiers that differentiate between factual and hallucinated content. By analyzing annotated examples, supervised models can identify patterns indicative of hallucinations, ensuring higher accuracy in distinguishing reliable outputs. Techniques such as semantic similarity analysis and contextual understanding have been integrated to enhance detection capabilities, particularly in tasks like summarization, translation, and question answering. This approach has been instrumental in improving the reliability of AI systems across critical domains, ensuring their outputs are consistent and trustworthy.

3 Proposed System

This section outlines the workflow and components of the proposed system for detecting hallucinations in responses generated by Large Language Models (LLMs).

3.1 System Architecture Diagram

1. **Loading the Dataset**
   - The hallucination dataset is loaded into the environment, consisting of labeled data with columns: `Id`, `Prompt`, `Answer`, and `Target`.
2. **Exploratory Data Analysis (EDA)**
   - Dataset Information: Inspect the dataset structure, columns, and features.
   - Descriptive Statistics: Generate statistical summaries to identify trends, distributions, and anomalies.
   - Word Cloud: Create visual word clouds to identify the most frequently occurring words in hallucinated (“1”) and factual (“0”) responses.
   - Text Length Analysis: Analyze the length of responses (word or character count), visualizing trends between the two classes (“1” and “0”).
   - Class Imbalance: Examine the imbalance, where non-hallucinated responses (“0”) dominate the dataset.
3. **Preprocessing**
   - Lowercasing: Convert all text to lowercase.
   - Removing Special Characters: Remove punctuation, numbers, and special symbols, keeping only alphabetic characters and spaces.
   - Tokenization: Split the text into tokens.
   - Stopword Removal: Remove common words (e.g., “the,” “is”) .
   - Lemmatization: Convert words to their base form (e.g., “running” → “run”).
   - Joining Tokens: Recombine processed tokens into strings for vectorization.
4. **Vectorization**
   - **TF-IDF:** Term Frequency-Inverse Document Frequency (TF-IDF) is a statistical technique that converts text into numerical features by reflecting the importance of a word in a document relative to the entire corpus.
   - **BERT Vectorization:** BERT embeddings use a pre-trained transformer-based model to generate contextualized vector representations of text, capturing semantic and syntactic nuances.

5. **Machine Learning Models**
   - Train and evaluate supervised learning models on both TF-IDF and BERT features:
     - Logistic Regression
     - Naive Bayes
     - Random Forest
     - K Nearest Neighbours Classifier (KNN)
6. **Evaluation Metrics**
   - Evaluate model performance using:
     - Accuracy
     - Precision, Recall, F1-Score
     - Confusion Matrix
7. **Visualization of Results**
   - Plot ROC curves and confusion matrices to interpret model performance and dataset trends.

Figure [1] shows the architecture diagram of the system.

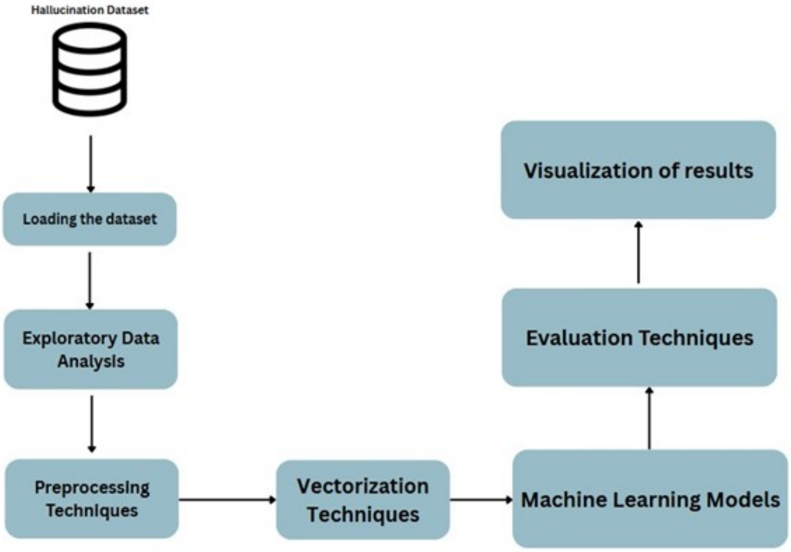

Fig. 1. System Architecture Diagram.

4 Implementation

4.1 Development Environment

The development environment for this machine learning task is set up using Python as the primary programming language due to its versatility and extensive library support for data science and machine learning. The implementation is carried out on Google Colab, a cloud-based platform that provides free access to GPUs and TPUs, enabling efficient computation and experimentation. Key libraries include pandas and numpy for data handling, matplotlib, seaborn, and wordcloud for visualization, nltk [8] for text preprocessing, torch and transformers for BERT embeddings, and scikit-learn for machine learning models and evaluation.This combination of tools and resources ensures a seamless workflow for data preprocessing, model development, and performance evaluation.

4.2 Dataset Description

The dataset utilized for hallucination detection, downloaded from a Kaggle repository, consists of 16,687 entries and is organized into four main columns: Id, Prompt, Answer, and Target. The Id column serves as a unique identifier for each record, ensuring traceability and consistency. The Prompt column contains the input queries or instructions provided to the language model, while the Answer column stores the corresponding responses generated by the model. The Target column acts as the ground truth label, indicating whether the generated response is factual (0) or hallucinated (1). This dataset is divided into two distinct classes: “1” (hallucinated responses) and “0” (accurate responses). Out of the total records, approximately 15,000 are labeled as "0" and only around 1,000 are labeled as "1" demonstrating a significant class imbalance [Fig 2].

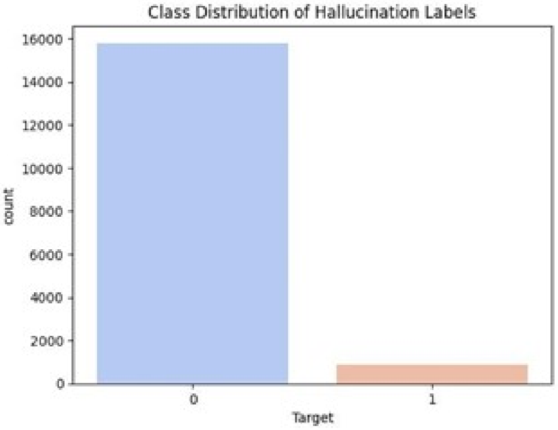

Fig. 2. Class Distribution of Hallucination Labels.

4.3 Balancing the Dataset for Fair Representation

To address the issue of class imbalance in the hallucination dataset, SMOTE (Synthetic Minority Oversampling Technique) is applied to oversample the minority class. Initially, the textual data from the Prompt and Answer columns is combined into a single feature (X), while the Target column serves as the label (y). SMOTE is usedgenerate synthetic samples for the minority class, ensuring a more balanced class distribution. After resampling, the new distribution of labels is analyzed and a bar graph is created to visualize the improved balance between classes. However, for high-dimensional embeddings, such as BERT embeddings, SMOTE is less effective because of its higher dimensionality, which makes generating meaningful synthetic samples challenging. Therefore, ADASYN (Adaptive Synthetic Sampling) is utilized with BERT embeddings to duplicate existing minority-class samples, achieving balance without introducing synthetic noise or complexity.

Figure [3] shows the class distrbution after balancing the dataset.

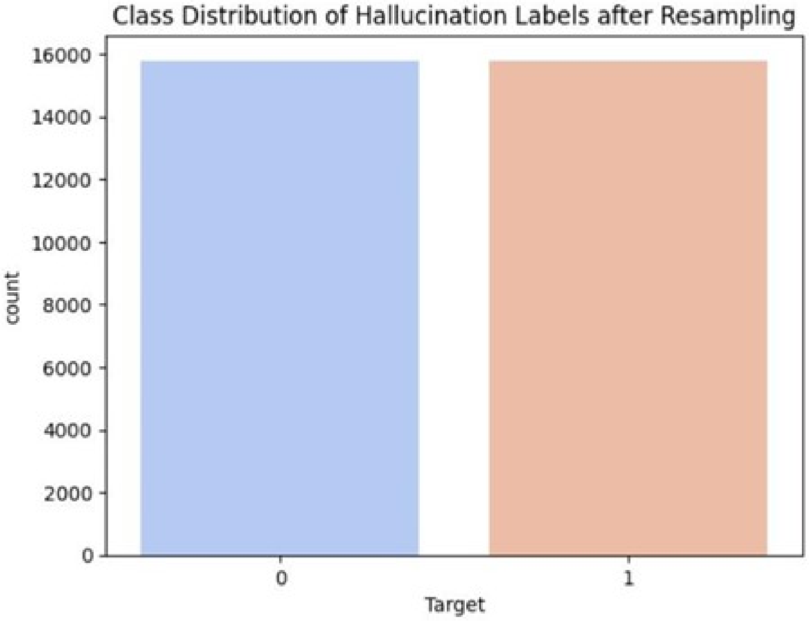

Fig. 3. Class Distribution of Hallucination Labels after balancing.

4.4 Exploratory Data Analysis (EDA)

To understand the characteristics of the textual responses in the dataset, exploratory analysis was conducted on two aspects: word frequency and response length.

Word Clouds: Word clouds were generated to visualize the most frequently occurring words in hallucinated ("No") and factual ("Yes") responses. As shown in Figure 4, there are clear differences in word usage patterns between the two classes.

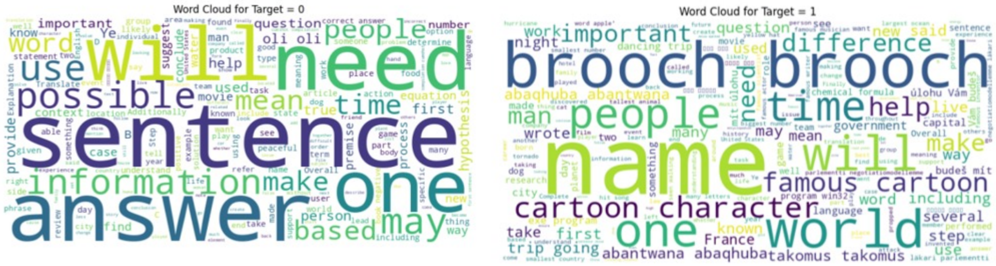

Fig. 4. Word Cloud for Factual and Hallucinated Responses.

Response Length Distribution: The number of words in each response was calculated to analyze verbosity. Figure 5 shows the distribution of the response lengths.

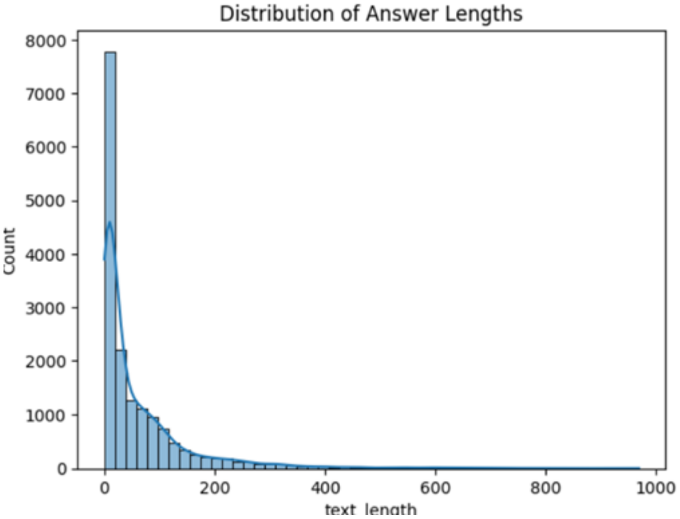

Fig. 5. Distribution of Response Lengths for Hallucinated and Factual Responses.

4.5 Preprocessing

Before feeding the textual data into machine learning models, a series of preprocessing steps are applied to ensure consistency and improve the quality of features extracted from the text. First, all text was converted to lowercase to maintain uniformity and avoid case sensitivity issues during analysis. Special characters, punctuation marks, and numbers were then removed, leaving behind only alphabetic characters and spaces to focus on meaningful textual content. The cleaned text was subsequently tokenized, breaking it down into individual words or tokens for easier manipulation. Common stopwords such as "the," "is," and "and" were filtered out predefined stopword list to highlight more significant words. Lemmatization was then employed to reduce words to their base or dictionary form, for example, converting "running" to "run," which helps minimize redundancy and reduce dimensionality. Finally, the processed tokens were rejoined into coherent strings, preparing them for the vectorization techniques that followed.

4.6 Vectorization Techniques

To convert textual data into a format suitable for machine learning models, vectorization techniques are employed. This section explores two prominent methods used in this study: TF-IDF vectorization and BERT embeddings.

TF-IDF Vectorization (Term Frequency-Inverse Document Frequency) is a statistical method used to represent text data by assigning a weight to each term based on its importance in a document relative to a corpus. Term Frequency (TF) measures how frequently a word appears in a specific document, indicating its relevance within that context. However, some words might appear frequently across many documents and may not carry significant meaning—this is where Inverse Document Frequency (IDF) plays a role. IDF reduces the weight of such commonly occurring terms, highlighting words that are more unique to each document. The TF-IDF score, calculated as the product of TF and IDF, strikes a balance between local and global term importance. This results in a sparse matrix where rows represent documents and columns correspond to terms, making it a suitable input format for traditional machine learning algorithms.

BERT Embeddings, on the other hand, leverage a deep learning-based approach using the Bidirectional Encoder Representations from Transformers (BERT) model. BERT is a transformer-based language model pre-trained on large corpora to understand the context of words based on surrounding text. In this approach, the tokenizer first processes the input text into tokens compatible with BERT's vocabulary. The processed text is then passed through the model to obtain contextualized embeddings from the last hidden layer. These embeddings capture rich semantic and syntactic information, allowing models to understand the nuances of language more effectively. The resulting dense vector representations are applied to each text sample and used as input features for downstream classification tasks.

4.7 Machine Learning Models

This section outlines the machine learning models employed for the classification of hallucinated and factual responses. A variety of supervised learning algorithms were explored to determine which performs best under different vectorization techniques. Each model was trained using both TF-IDF and BERT embeddings, allowing for a comparative analysis of their effectiveness in identifying hallucinations in text data.

Naive Bayes: Naive Bayes is a probabilistic classifier based on Bayes' Theorem with the assumption of independence between features. It is particularly effective for text classification tasks due to its simplicity and speed. Despite its naive assumption, it often performs surprisingly well in high-dimensional spaces.

Logistic Regression: Logistic Regression is a linear model used for binary classification problems. It estimates the probability of a class label using the logistic (sigmoid) function. This model is interpretable and works well when the classes are linearly separable.

Random Forest: Random Forest is an ensemble learning method that builds multiple decision trees and merges their results. It improves classification performance by reducing overfitting compared to individual decision trees. Random Forests are robust, handle nonlinear data well, and are effective for imbalanced datasets.

K-Nearest Neighbors (KNN): KNN is a non-parametric, instance-based learning algorithm that classifies data points based on the majority label of their nearest neighbors. It relies on distance metrics like Euclidean distance to measure similarity between data points. KNN is simple, effective for small datasets, and performs well when the decision boundary is not linear.

4.8 Evaluation Metrics

To assess the performance of the classification models, a variety of evaluation metrics were utilized. These metrics provide a comprehensive understanding of how well each model distinguishes between hallucinated and factual responses.

Confusion Matrix: The confusion matrix is a tabular representation of the actual versus predicted classifications. It consists of True Positives (TP), False Positives (FP), True Negatives (TN) and False Negatives (FN). It provides information on the types of errors made by the model.

Accuracy: Accuracy measures the proportion of correctly predicted instances from the total predictions. It provides a general idea of how well the model is performing. It is most reliable when the dataset is balanced.

Precision: Precision shows the proportion of positive predictions that were actually correct. It is useful when the cost of false positives is high. A higher precision indicates fewer irrelevant results.

Recall: Recall measures the proportion of actual positives that were correctly identified. It is important when the cost of missing positive instances is high. A higher recall indicates fewer missed relevant results.

F1-Score: The F1-Score balances both precision and recall in one metric. It is especially useful when dealing with imbalanced datasets. A higher F1-Score means better overall model performance across classes.

5 Results and Discussions

This section presents the performance outcomes of the implemented machine learning models on the hallucination detection dataset. The results are analyzed using various evaluation metrics to compare model effectiveness and identify strengths and weaknesses. Visualizations such as confusion matrices and classification performance plots further aid in understanding the models' behavior. Key observations and insights derived from the experiments are discussed in detail.

5.1 Model Performance Evaluation

The performance of different machine learning models was assessed using both training and testing datasets. Each model was evaluated based on various metrics including accuracy, precision, recall, and F1-score. The experiments were conducted using two popular vectorization techniques: TF-IDF and BERT embeddings. Tables 1 and 2 summarize the training and testing results respectively.

Table 1. Training Performance of Models with Different Vectorization Techniques

Model	Vectorization Technique	Accuracy	Precision	Recall	F1-Score
Logistic Regression	TF-IDF	0.77	0.83	0.69	0.75
Logistic Regression	BERT embeddings	0.76	0.75	0.77	0.76
Random Forest	TF-IDF	0.79	0.89	0.67	0.76
Random Forest	BERT embeddings	0.82	0.82	0.81	0.82
KNN	BERT	0.86	0.76	0.79	0.76

Table 2. Testing Performance of Models with Different Vectorization Techniques

Model	Vectorization Technique	Accuracy	Precision	Recall	F1-Score
Logistic Regression	TF-IDF	0.77	0.95	0.95	0.95
Logistic Regression	BERT embeddings	0.75	0.83	0.83	0.83
Random Forest	TF-IDF	0.80	0.99	0.99	0.99
Random Forest	BERT embeddings	0.81	0.99	0.99	0.99
KNN	BERT	0.9210	0.9308	0.9224	0.9207
Naive Bayes	TF-IDF	0.94	0.95	0.95	0.95
Naïve Bayes	BERT embeddings	0.69	0.69	0.69	0.69

Logistic Regression performs decently with both TF-IDF (Accuracy: 0.77) and BERT embeddings (Accuracy: 0.75). The slightly lower performance with BERT embeddings suggests that Logistic Regression is better suited for simpler, sparse representations like TF-IDF, as it relies on linear relationships and performs best when features are independent and less context-driven. **Naive Bayes**, on the other hand, excels with TF-IDF (Accuracy: 0.94) due to its assumption of feature independence, which aligns well with the bag-of-words nature of TF-IDF. However, it struggles significantly with BERT embeddings (Accuracy: 0.69) as these dense, contextual vectors violate the core assumption of feature independence.

Random Forest performs consistently well with both TF-IDF (Accuracy: 0.80) and BERT embeddings (Accuracy: 0.81), showcasing its versatility. Its ensemble, tree-based structure allows it to model non-linear relationships and handle both sparse and dense feature spaces effectively. This highlights that models like Logistic Regression and Random Forest are better suited for handling complex embeddings like BERT, as they can learn interdependencies between features. In contrast, **Naive Bayes** lacks the capacity to utilize the rich contextual information embedded in BERT. This analysis underscores the importance of choosing models aligned with the complexity of the task and the nature of feature representations—especially for context-heavy tasks like hallucination detection.

6 Impact of the project on human, societal, ethical and sustainable development

6.1 Impact on Humans

This project improves the reliability of AI-driven systems, ensuring that users receive factual and trustworthy information. This reduces the risk of misinformation, aiding individuals in making better-informed decisions in areas like education, healthcare, and everyday problem-solving.

6.2 Impact on Society

By minimizing hallucinations in language models, the project contributes to fostering societal trust in AI technologies. Reliable AI systems can support critical societal functions, such as governance, public policy, and disaster response, enabling more effective solutions for collective challenges.

6.3 Ethical Impact

The project promotes ethical AI practices by addressing the challenge of hallucinations, which can lead to harmful consequences when unchecked. By ensuring accurate outputs, it aligns with principles of transparency, accountability, and fairness, essential for the responsible use of AI.

6.4 Impact on Sustainable Development

Enhancing the accuracy of AI systems contributes to sustainable development by fostering innovation and enabling data-driven solutions to global challenges. Reliable AI applications can be leveraged in achieving goals like quality education, good health, and reduced inequalities, driving sustainable progress across sectors.

7 Conclusion and Future Work

7.1 Conclusion

This study provides a comprehensive framework for detecting hallucination in textual responses, addressing a critical challenge in ensuring the reliability of Large Language Models (LLMs). By employing preprocessing, resampling techniques, and rigorous evaluation metrics, the project emphasizes the importance of building balanced and robust models capable of distinguishing between factual and hallucinated content. The integration of vectorization techniques and systematic exploration of the dataset has illuminated the complexities involved in textual analysis, particularly in identifying patterns that differentiate hallucinated responses. This work has broader implications in ethical and societal contexts, as hallucination detection is vital for fostering trust in AI systems—especially in sensitive domains such as healthcare, law, and education. By reducing inaccuracies and enhancing accountability, this project contributes to developing more reliable and sustainable AI-driven solutions.

7.2 Future Work

Future advancements could focus on adapting hallucination detection models for domain-specific applications, particularly in high-stakes areas like healthcare and the legal field. In healthcare, ensuring the correctness of AI-generated content is critical to prevent the spread of incorrect medical information, which could have serious consequences. Fine-tuning the model using medical datasets could enhance its ability to identify hallucinations in clinical texts, thereby safeguarding patient safety. Likewise, in legal contexts, the model could be optimized to detect fabricated case references, misrepresented statutes, or misleading arguments, improving the reliability of AI-generated legal documents. Such targeted specialization would promote the ethical and responsible use of AI, aligning its capabilities with societal expectations and professional standards.

References

1. Chang, Y., Wang, X., Wang, J., Wu, Y., Yang, L., Zhu, K., Chen, H., Yi, X., Wang, C., Wang, Y. and Ye, W.: A survey on evaluation of large language models. *ACM Transactions on Intelligent Systems and Technology*, **15**(3), 1–45 (2024)
2. Cunningham, P., Cord, M. and Delany, S.J.: Supervised learning. In *Machine learning techniques for multimedia: case studies on organization and retrieval*, 21–49. Springer Berlin Heidelberg (2008)
3. Gupta, S., Ranjan, R. and Singh, S.N.: A Comprehensive Survey of Retrieval-Augmented Generation (RAG): Evolution, Current Landscape and Future Directions. *arXiv preprint arXiv:2410.12837* (2024)
4. Huang, L., Yu, W., Ma, W., Zhong, W., Feng, Z., Wang, H., Chen, Q., Peng, W., Feng, X., Qin, B. and Liu, T.: A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems* (2024)

5. Yenduri, G., Ramalingam, M., Selvi, G.C., Supriya, Y., Srivastava, G., Maddikunta, P.K.R., Raj, G.D., Jhaveri, R.H., Prabadevi, B., Wang, W. and Vasilakos, A.V.: GPT (Generative Pre-trained Transformer) – a comprehensive review on enabling technologies, potential applications, emerging challenges, and future directions. *IEEE Access* (2024)
6. Ravichandiran, S.: Getting Started with Google BERT: Build and train state-of-the-art natural language processing models using BERT. *Packt Publishing Ltd.* (2021)
7. Kublik, S. and Saboo, S.: GPT-3. *O'Reilly Media, Inc.* (2022)
8. Hardeniya, N., Perkins, J., Chopra, D., Joshi, N. and Mathur, I.: Natural Language Processing: Python and NLTK. *Packt Publishing Ltd.* (2016)