FactAlign: Fact-Level Hallucination Detection and Classification Through Knowledge Graph Alignment

Mohamed Rashad, Ahmed Ismail Zahran, Abanoub Amgad Amin, Amr Yassin Abdelaal, Mohamed AlTantawy

Agolo, New York, NY
{mohamed.rashad,ahmed.zahran,abanoub.amgad,
amr.yassin,mohamed}@agolo.com

Abstract

Generative Large Language Models (LLMs) have garnered significant attention for their ability to generate human-like text across diverse domains. However, a major obstacle preventing their widespread adoption in production environments is their propensity for 'hallucinations' – the generation of non-factual statements that can erode confidence in their output. Existing hallucination detection approaches either require access to the categorical distribution of the output or rely on external databases to retrieve evidence about generated output. An alternative strategy employs sampling-based techniques, which generate responses multiple times to identify hallucinations. This paper proposes a novel black-box approach to automatically detect and classify hallucinations at a fact level by transforming the problem into a knowledge graph alignment task. This approach, unique in its applications, also allows us to classify detected hallucinations as either intrinsic or extrinsic. Our methodology was evaluated on the WikiBio GPT-3 hallucination dataset for hallucination detection and the XSum hallucination annotations dataset for hallucination classification. Our method achieved a 0.889 F1 for the hallucination detection and 0.825 F1 for the hallucination type classification, without any further training, fine-tuning, or producing multiple samples of the LLM response.

1 Introduction

Large Language Models (LLMs) have showcased impressive performance in significant tasks such as natural language understanding (Du et al., 2022), language generation (Axelsson and Skantze, 2023), and complex reasoning (Hao et al., 2023). Despite their widespread applications, LLMs are prone to hallucinate (Ji et al., 2023), which makes them difficult to rely on.

Existing literature focuses on robust hallucination detection mechanisms to ensure the reliability and accountability of NLP systems (Corlett et al., 2019). However, recent approaches require access to either the token-level probability distribution (Manakul et al., 2023) or external databases (Bayat et al., 2023) that are rarely available. Another approach relies on sampling that requires multiple LLM calls (Manakul et al., 2023).

Due to these limitations, we introduce a novel approach that transforms hallucination detection into a knowledge graph alignment task.

Our approach is established on the notion that faithful generation should be semantically aligned with the source text. The degree of alignment was modeled as a metric to score the faithfulness of the generated text. Extending beyond mere detection, our approach is capable of classifying detected hallucinations into intrinsic and extrinsic categories. According to (Maynez et al., 2020), intrinsic hallucinations are defined as manipulation of the information present in the input document, while extrinsic hallucinations involve adding information not directly inferable from the input document. By distinguishing between these categories, our method enhances the interpretability of detected hallucinations, providing valuable insights into the underlying causes.

2 Related Work

Current hallucination detection approaches can be classified according to the type of input required from the generative model as grey-box or black-box. Grey-box approaches, such as average and maximum token-level log probabilities (Manakul et al., 2023) are not restricted in their access to the generated text. However, token-level probabilities are not always accessible (e.g.: ChatGPT). Black-box approaches handle this limitation by only requiring the generated text. These approaches include proxy LLM-based approaches, external databases-dependent approaches, and sampling-based approaches.

Proceedings of the 4th Workshop on Trustworthy NLP (TrustNLP 2024), pages 79–84
June 21, 2024

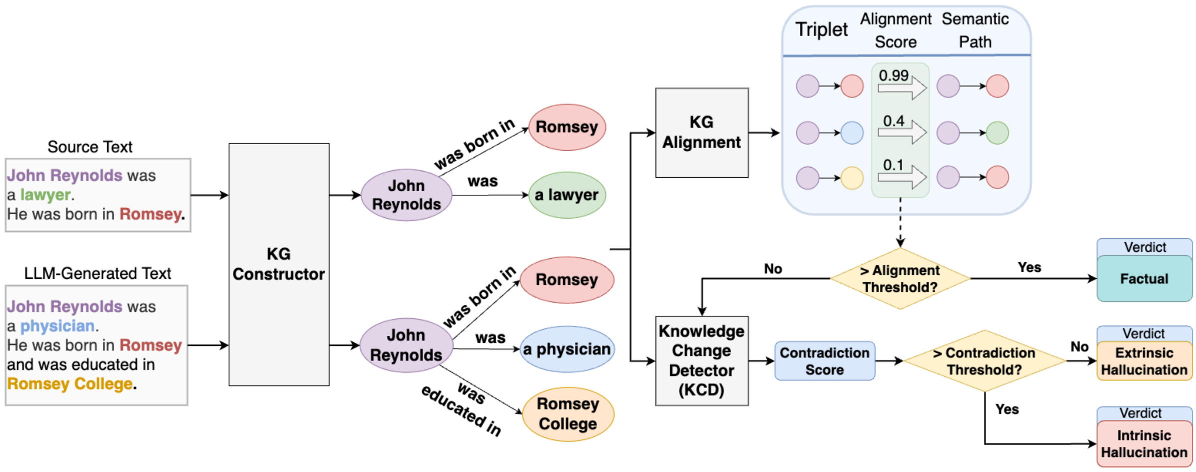

Figure 1: Hallucination detection and classification pipeline

Proxy LLM-based approaches, such as BARTScore (Yuan et al., 2021) use a proxy LLM to obtain token-level probabilities given the input text. The main limitation of these models is that the produced scores cannot be used to classify individual sentences.

Factual data-dependent approaches compare the generated text to factual data. For example, AlignScore (Zha et al., 2023) uses 4.7M training examples from several datasets to train a model on predicting an alignment score between factual and generated data. Other approaches like (Thorne et al., 2018) utilize external sources, which is useful when there is no or limited source text.

Sampling-based approaches stochastically sample multiple outputs and detect hallucinations based on their consistency with the original output. For example, SelfCheckGPT (Manakul et al., 2023) samples outputs and judges their consistency with the original output using either BERTScore (Zhang et al., 2019), multiple-choice question answering, textual entailment, or prompting an LLM. In HaLo (Elaraby et al., 2023), a pairwise entailment is computed between pairs of sentences from the original response and other sample responses using SUMMAC (Laban et al., 2022).

3 Hallucination Detection and Classification Approach

Our approach detects and classifies hallucinations at a fact level using knowledge graph alignment. As shown in Figure 1, the KG Constructor takes source and generated text as inputs and generates the corresponding KGs. The constructed KGs

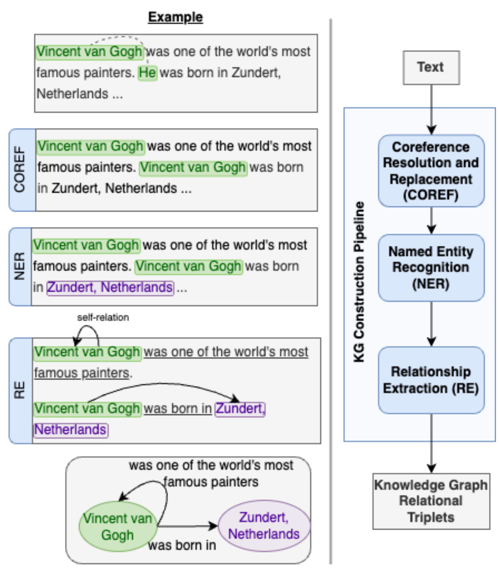

Figure 2: Knowledge graph construction

are passed to the Alignment module to produce the *alignment score* for each generated triplet which is used to determine whether the generated triplet is hallucinated or factual. The KG triplets from the source text and the hallucinated KG triplets from the generated text are passed to the Knowledge Change Detector (KCD), which produces a contradiction score for each of the hallucinated triplets, which in turn is used to classify whether the hallucination in this triplet is intrinsic or extrinsic.

Knowledge Graph Construction We used a simple approach to automatically construct a Knowledge Graph from the text (see Figure 2). First, we resolved each coreference to its reference using coreference resolution model[1]. The text is then passed to NER[2] to extract the named entities[3]. Finally, relation extraction[4] is performed on the text. The produced relational triplets are filtered to remove triplets where the subject or the object is not in the named entities produced by the NER model.

3.1 Hallucination Detection as KG Alignment

A simple approach for solving the KG alignment is to treat it as an assignment problem (Mao et al., 2021). Given the set of all source entities E_s and the set of all generated entities E_g, the input consists of four matrices: $A_s \in \mathbb{R}^{|E_s|\times|E_s|}$ and $A_g \in \mathbb{R}^{|E_g|\times|E_g|}$, which are the adjacency matrices of KG_s and KG_g, respectively, and $H_s \in \mathbb{R}^{|E_s|\times d_e}$ and $H_g \in \mathbb{R}^{|E_g|\times d_e}$ which are the entity representation matrices for KG_s and KG_g, where d_e is the dimension of the entity representation vector space. A permutation matrix P is used to represent the entity correspondences between KG_s and KG_g, such that $P_{i,j} = 1$ indicates that $e_i \in KG_s$ and $e_j \in KG_g$ are an equivalent entity pair. Then, under the one-to-one constraint, the assignment problem can be solved using the following objective function

$$\arg\min_{P\in\mathbb{P}_{|E|}} \sum_{l=1}^{L} ||PA_s^l H_s - A_g^l H_g||_F^2 \quad (1)$$

where l represents the depth of the adjacency matrix, $||.||_F$ represents the Frobenius norm and $\mathbb{P}_N$ represents the set of all N-dimensional permutation matrices.

The above equation can be solved using algorithms like the Hungarian algorithm (Kuhn, 1955) and the Sinkhorn operation (Cuturi, 2013).

We choose to perform alignment on the level of triplets instead of entities. For each triplet, a triplet representation is calculated by concatenating the elements of the triplet as a piece of text and passing it to a transformer-based model[5]. This results in representation matrices $F_s \in \mathbb{R}^{|T_s|\times d_t}$ and $F_g \in \mathbb{R}^{|T_g|\times d_t}$, where T_s is the sets of triplets from KG_s, T_g is the set of triplets from KG_g, and d_t is the dimension of the triplet representation vector space. We simplify Equation 1 by relaxing the one-to-one constraint, such that one triplet from the KG_s can support multiple triplets from the KG_g.

[1]The FastCoref Python package was used (Otmazgin et al., 2022)

[2]Multi-lingual NER BERT was used to obtain named entities (Devlin et al., 2018)

[3]We consider the following entity types: Person, Organization, Location, Date.

[4]Relation Extraction from CoreNLP (Manning et al., 2014) was used to obtain relational triplets.

[5]DistilRoberta pre-trained model from the SentenceTransformers (Reimers and Gurevych, 2019) Python framework was used as our transformer-based model.

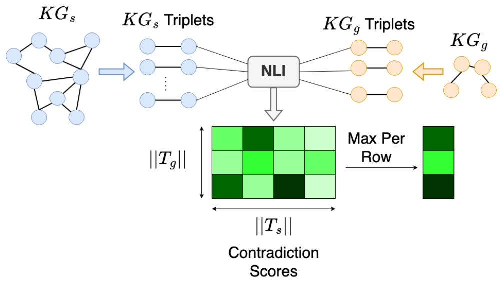

Figure 3: Knowledge Change Detector (KCD) takes the sets of triplets T_g of knowledge graph KG_g and T_s of KG_s. For each triplet $t_j \in T_g$, an NLI model is used to compute the contradiction scores between t_j and t_i $\forall t_i \in KG_s$ and find the maximum contradiction score.

The best match for each generated triplet $t_j \in T_g$ from all source triplets $t_i \in T_s$ is calculated using the following formula

$$\arg\min_{t_i\in T_s} ||v_i^T F_s - v_j^T F_g||_2 \quad (2)$$

where v_i and v_j are the one-hot vectors corresponding to t_i and t_j, respectively.

The corresponding alignment score s_a is computed as

$$s_a = 1 - \min_{t_i\in T_s} ||v_i^T F_s - v_j^T F_g||_2 \quad (3)$$

where $0 \leq s_a \leq 1$. If s_a is higher than a specific threshold (described in Section 4), the triplet is considered to be factual, and is considered to be hallucinated otherwise as shown in Figure 1.

3.2 Hallucination Classification

We extend our approach beyond hallucination detection to classification using a Knowledge Change Detector (KCD) module (see Figure 3) that computes a *contradiction score* (ranging from 0 to 1) between hallucinated and source triplets using an NLI model [6]. This score quantifies knowledge alteration introduced by LLMs. If this score is higher than a specific threshold (described in section 4), the generated knowledge is considered to be manipulated (intrinsic hallucination). Otherwise, it is

[6]DeBERTa-v3-base-mnli-fever-anli was used for NLI (Laurer et al., 2022)

considered to be unsupported by the original text (extrinsic hallucination).

4 Experimental Setup

Datasets To evaluate our hallucination detection approach, we used the WikiBio GPT-3 hallucination dataset (Manakul et al., 2023) which contains 238 Wikipedia-like passages generated using GPT-3 (text-davinci-003). The passages are divided into sentences, each annotated as containing accurate information, minor inaccuracies, or major inaccuracies. We grouped major and minor inaccurate labels into a hallucinated class, labeled as 1, while the accurate class was labeled as 0. 10% of the data was reserved for hyperparameter optimization and the results were reported on the rest of the dataset. For the hallucination classification task, we used the XSum hallucination annotated dataset (Maynez et al., 2020), containing 500 randomly sampled articles from the XSum dataset (Narayan et al., 2018) and the corresponding summaries from multiple generative models. Hallucinated spans were annotated as containing intrinsic or extrinsic hallucination.

Hyperparameter Optimization Bayesian optimization [7] was performed for 30 iterations to decide the alignment and contradiction score thresholds (set to 0.863 and 0.984, respectively).

Baselines We evaluate our method against two baselines: SelfCheck with NLI (Manakul et al., 2023) and AlignScore-Large (Zha et al., 2023). For both methods, the threshold is set to the value that maximizes the F1 score (0.54 for SelfCheck and 0.7 for AlignScore).

5 Results

The proposed method was evaluated on the tasks of hallucination detection using precision, recall, and F1-score. The evaluation was performed on the level of sentences to be compared to sentence-level hallucination detection baselines. Given a generated sentence $s_i \in S$, where S is the set of all generated sentences in the test set, we computed the set of triplets $t_j \in T_g$, where T_g is the set of triplets constructed from the generated sentence s_i. A sentence was classified as hallucinated if it included at least one hallucinated triplet.

As shown in table 1, our hallucination detection method achieves a recall of 0.992 on the task of sentence-level hallucination detection on WikiBio, which is higher than that achieved by the reported baselines without any fine-tuning, training, or using of additional generated samples. While our method obtained less precision compared to the baselines, the overall F1-score of FactAlign is still higher. The results show the effectiveness of fact-level hallucination detection used in our method.

Table 2 reports the fact-level results for intrinsic vs. extrinsic hallucination classification, where each triplet constitutes a generated fact. For the sets of annotated hallucination spans P and the set of extracted triplets T_g in a test example, a triplet $t_j \in T_G$ was annotated as hallucinated if its text overlapped with a hallucinated span $p_i \in P$. As shown in the table, FactAlign achieves reasonable fact-level hallucination classification metrics.

Table 1: Sentence-level hallucination detection results on the WikiBio GPT-3 hallucination dataset

	Precision	Recall	F1
SelfCheck	**0.843**	0.917	0.879
AlignScore	0.809	0.981	0.886
FactAlign	0.805	**0.992**	**0.889**

Table 2: Fact-level hallucination classification results on the XSum hallucination annotations dataset

Precision	Recall	F1
0.833	0.817	0.825

6 Conclusion

In this paper, we introduced a black-box hallucination detection technique based on constructing knowledge graphs from the source and generated text, aligning these knowledge graphs, and comparing the aligned triplets. Our method achieved an F1-score of 0.889 on hallucination detection on the WikiBio dataset and 0.825 on hallucination-type classification on the XSum hallucination annotations dataset. These results show the effectiveness of the knowledge graph alignment approach in the discovery and classification of individual hallucinated triplets. Basing our approach on the level of triplets makes the hallucination detection output explainable and highlights the correct triplets that can later be used to correct hallucinations.

[7] Scikit-optimize (Head et al., 2020) was used for Bayesian optimization.

Limitations

Although our method can obtain high scores on the task of hallucination detection and classifying hallucinations, the method contains some limitations. This section highlights the limitations and possible future research directions.

Knowledge Graph Construction Our approach limits the entities in the constructed triplets to named entities, which means that this knowledge graph construction method may miss important triplets where the entities are not named entities. In future studies, we plan to explore further relation extraction techniques to build more reliable knowledge graphs and explore their effect on hallucination detection.

Large-Scale Hallucination Detection Detecting Hallucination as a KG alignment task on scale presents a formidable challenge, considering that each generated triplet necessitates alignment with the entire source knowledge graph. In future studies, retrieval augmented generation (RAG) (Lewis et al., 2020) is investigated as a way to retrieve relevant triplets. This will allow selective retrieval of the relevant sub-graph that demands alignment, thereby circumventing the need to align with the entirety of the expansive KG.

References

Agnes Axelsson and Gabriel Skantze. 2023. Using large language models for zero-shot natural language generation from knowledge graphs. *arXiv preprint arXiv:2307.07312*.

Farima Fatahi Bayat, Kun Qian, Benjamin Han, Yisi Sang, Anton Belyi, Samira Khorshidi, Fei Wu, Ihab F Ilyas, and Yunyao Li. 2023. Fleek: Factual error detection and correction with evidence retrieved from external knowledge. *arXiv preprint arXiv:2310.17119*.

Philip R Corlett, Guillermo Horga, Paul C Fletcher, Ben Alderson-Day, Katharina Schmack, and Albert R Powers. 2019. Hallucinations and strong priors. *Trends in cognitive sciences*, 23(2):114–127.

Marco Cuturi. 2013. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Mengnan Du, Fengxiang He, Na Zou, Dacheng Tao, and Xia Hu. 2022. Shortcut learning of large language models in natural language understanding: A survey. *arXiv preprint arXiv:2208.11857*.

Mohamed Elaraby, Mengyin Lu, Jacob Dunn, Xueying Zhang, Yu Wang, and Shizhu Liu. 2023. Halo: Estimation and reduction of hallucinations in open-source weak large language models. *arXiv preprint arXiv:2308.11764*.

Shibo Hao, Yi Gu, Haodi Ma, Joshua Jiahua Hong, Zhen Wang, Daisy Zhe Wang, and Zhiting Hu. 2023. Reasoning with language model is planning with world model. *arXiv preprint arXiv:2305.14992*.

Tim Head, Manoj Kumar, Holger Nahrstaedt, Gilles Louppe, and Iaroslav Shcherbatyi. 2020. scikit-optimize/scikit-optimize.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38.

Harold W Kuhn. 1955. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97.

Philippe Laban, Tobias Schnabel, Paul N Bennett, and Marti A Hearst. 2022. Summac: Re-visiting nli-based models for inconsistency detection in summarization. *Transactions of the Association for Computational Linguistics*, 10:163–177.

Moritz Laurer, Wouter van Atteveldt, Andreu Casas, and Kasper Welbers. 2022. Less annotating, more classifying: Addressing the data scarcity issue of supervised machine learning with deep transfer learning and bert-nli. *Political Analysis*, pages 1–33.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

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*.

Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. 2014. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pages 55–60.

Xin Mao, Wenting Wang, Yuanbin Wu, and Man Lan. 2021. From alignment to assignment: Frustratingly simple unsupervised entity alignment. *arXiv preprint arXiv:2109.02363*.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. 2020. On faithfulness and factuality in abstractive summarization. *arXiv preprint arXiv:2005.00661*.

Shashi Narayan, Shay B Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. *arXiv preprint arXiv:1808.08745*.

Shon Otmazgin, Arie Cattan, and Yoav Goldberg. 2022. F-coref: Fast, accurate and easy to use coreference resolution. In *AACL*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

James Thorne, Andreas Vlachos, Oana Cocarascu, Christos Christodoulopoulos, and Arpit Mittal. 2018. The fact extraction and verification (fever) shared task. *arXiv preprint arXiv:1811.10971*.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. Bartscore: Evaluating generated text as text generation. *Advances in Neural Information Processing Systems*, 34:27263–27277.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. Alignscore: Evaluating factual consistency with a unified alignment function. *arXiv preprint arXiv:2305.16739*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.