EG-RAG: From Passages to Graph Triples for Explainable LLM Factuality

Anonymous ACL submission

Abstract

This work introduces a graph-based retrieval-augmented generation (RAG) approach aimed at validating large language models (LLMs) to enhance their factual accuracy and explainability. While most existing RAG architectures prioritize reducing hallucinations, they often neglect interpretable justifications, leaving explainability underexplored. We argue that hallucination metrics alone are insufficient without interpretable justifications. To this end, we propose an approach that utilizes knowledge graph extraction of queries and claims with graph context matching to provide evidence. The approach includes steps for reasoning related subgraphs using text embedding language models and the all-Steiner tree method, validation, and explanation.

Our method explicitly derives LLM validation and interpretability from the semantic relationships that represent atomic facts. As a step toward explainability, we generate explanations for these relationships and propose explanation-based reasoning using prominent atomic facts from the Knowledge graphs.

Our experiments demonstrate that our proposed explanation-based reasoning improves the factual accuracy of several LLMs, up to 40 percent, on the Fever and Pub Health datasets. Our explainability experiments demonstrate that our method improves judgment, and the digested inference context it generates yields LLM inferences that are more accurate than those generated from textual LLM context. In addition, our method transparently exposes matched entities from the knowledge graph of facts and the user query.

1 Introduction

Large language models (LLMs) continue to advance in areas such as code generation, reasoning, and multimodal prediction, but still generate fluent yet potentially inaccurate information. This challenge raises concerns about the trustworthiness and factual consistency of LLM-based systems. State-of-the-art models, such as GPT-5, demonstrate remarkable capabilities in text generation. However, like earlier models, they remain prone to hallucinations, in which they confidently generate factually incorrect statements. Furthermore, it is suggested that Structural Hallucinations could remain a permanent challenge of LLMs as an intrinsic nature of such systems [Banerjee et al., 2024, Kalai et al., 2025]

Factfulness often refers to the factual consistency of a statement with a given background information, rather than objective factuality or universal truth [Dreyer et al., 2023]. In the context of Large Language Model (LLM) inference, factfulness requires avoiding hallucinations and demands both formal correctness and truthful content. Even when an inference follows a valid relational pattern, it may still be factually incorrect if its underlying assumptions do not hold. For example, inferring biological parenthood from a spousal relationship is correct in some cases, such as for Marie and Pierre Curie with respect to Irène Curie, but it is not universally valid. For instance, when a child is from a previous marriage. A model that preserves the syntactic structure of such relations may therefore produce well-formed but factually incorrect conclusions, highlighting the need to evaluate factual plausibility beyond structural validity.

To address factuality challenges, retrieval-augmented generation (RAG) systems [Lewis et al., 2020] retrieve documents relevant to an input query and incorporate the evidence into the LLM context, thereby reducing hallucinations and improving factual alignment. Traditional RAG systems focus solely on textual relevance and are not attentive to the semantic relationships of concepts, which are usually represented by graph structures. Therefore, their effectiveness on complex, interconnected data is limited [Barry et al., 2025].

In addition, RAG systems face the challenge

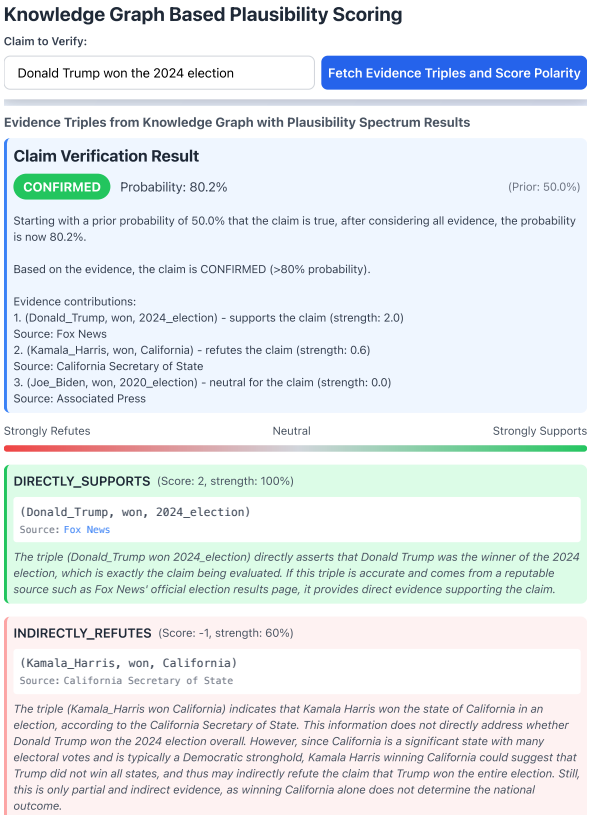

Figure 1: Illustrating retrieved triples with their plausibility score, indicating the contributions to plausibility of a statement.

of corpus split size, in which a text corpus is partitioned into text snippets. In this setting, determining an optimal snippet length is challenging: too short snippets introduce redundancy and data fragmentation, whereas excessively long snippets distract the retrieval method and obscure key information [Liu et al., 2024a].

In the following, we argue that explainable graph-based retrieval is not merely an enhancement but potentially a prerequisite for building trustworthy and explainable LLM systems. In contrast to traditional RAG systems, Graph-based RAGs retrieve information related to queries by leveraging a network of facts, represented as a subgraph of joint triples from a knowledge graph, to provide supporting evidence for the queried facts.

Graph-based RAGs address several limitations of traditional RAG systems; however, they do not resolve data-related challenges that are ubiquitous in RAG pipelines. Graphs, like the textual resources used in conventional RAGs, may contain irrelevant, noisy, overly long, or inconsistent information. Knowledge graph construction from textual documents and graph-based information undertaken by graph-based RAGs does not necessarily resolve these underlying challenges. Manual knowledge graph construction, despite addressing data-cleaning challenges, also entails addressing issues arising from factually incorrect or inconsistent data. In addition, human error and scaling bottlenecks contribute to the RAG and Graph RAG challenges.

Therefore, to address these challenges, RAG systems must provide a means of transparency and explanation that facilitates human and machine collaboration in fact-checking. An example from the FEVER fact-checking challenge [Thorne et al., 2018] is the claim "Hunter Biden had no experience in Ukraine or in the energy sector when he joined the board of Burisma.". In this example, the baseline text indicates: "No, Hunter Biden's previous career history does not include work for energy companies.". When a claim is validated by a human judge using a short paragraph from the "Answer" dataset, the human judge can conclude that the Fact is valid. However,when we use LLM as a judge within our experimental setting, the result would indicate that the provided answer is insufficient to accept or reject it. Our explanation approach breaks the claim into two aspects of experience in Ukraine and experience in the Gas industry, and shows that the provided answer only covers one aspect of the claim. A human judge or an LLM-based judge, when assisted by an explanatory approach, can make better judgments based on evidence analysis. This example shows that attempts to target hallucinations pose trustworthiness challenges that increase the AI system's risks. To minimize these risks, we define three criteria for transparency and explainability as requirements for Graph RAG systems and show how we meet these requirements.

- **Req-1** Such a system should disclose the source of data, information about intermediate steps, and the final selected supporting context. Exposing intermediate steps results in a more transparent approach to how a result inference was generated. In addition, providing metadata about information resources in a RAG system enables users to obtain an overview and subjectively accept or reject an LLM inference based on source reliability and intermediate steps.

- **Req-2** To promote explainability, users should be shown exactly which parts of the evidence

data support or reject a fact, thereby mitigating the concern that AI serves as its own judge [Cheng et al., 2025].

- **Req-3** From the extracted related parts of the evidence that support the RAG-based judgment, the piece of information that most influences the judgment of a claim must be highlighted, and, if possible, its connection must be explained. This allows a user to judge whether the inference drawn from a set of related evidence is reasonable.

This study proposes EG-RAG, an Explainability-based Graph RAG method that meets these requirements for explainability. With this method, we aim to enhance and assess the factfulness of LLMs grounded in a Graph, with a focus on explainability. In the following, we describe EG-RAG's contributions with respect to the three requirements of explainability.

Our method is available in two settings: (1) LLM as a judge and (2) LLM for user input validations (fact-checking). Figure 1 illustrates an example of our method for LLM fact-checking, showing how retrieved facts' polarity with respect to a query is explained. We address Req 1-3 with our approach by presenting the retrieved most prominently related triples, how and how much each of these triples relates to the claim, the information on the intermediate steps, and links to their resources. In the fact-checking setting, this approach allows users to subjectively accept or reject an LLM inference with an extra insight. In the LLM-as-judge setting, our approach enables an LLM to produce better judgments using more comprehensive information than plain text or subgraphs offered in traditional RAG or Graph-RAGs.

2 Related Work

Prior works on decompositional fact-checking and graph-based RAG methods that feature explainability align with the direction of our study. We review related work on aspects of both topics in the following.

Decompositional Fact Verification: FactScore [Min et al., 2023] is a fact-checking validation approach that targets factual precision in long-form LLM Text Generation. Similar to our study, it decomposes facts into their atomic form; however, it does so in sentence form rather than as triples in Knowledge graphs. FactScore is complementary to our study as it decomposes claims, but we decompose both claims and resource fact graphs. Another decompositional approach is [Kamoi et al., 2023], which decomposes claims into subclaims and assigns entailment labels based on Wikipedia evidence sentences that support a subclaim. Wadden et al. [2020] have trained a BERT-based model on domain knowledge to retrieve and reason about the factfulness of scientific statements, where they verify parts of a claim using background facts that relate to them. The provided method, however, predicts a claim's correctness in a black-box manner, thereby reducing the transparency of the method.

Graph-RAGs and Explanability: Few Graph-based approaches provide minimal factual transparency by providing the retrieved subgraph supporting the inference. Medical GRAG [Wu et al., 2024] provides a resource and an explanation of the medical terms in their response. On the other hand, KG^{2}RAG [Zhu et al., 2025a] models fact-level relationships among retrieved chunks by utilizing a resource graph. Path-based explanations from paths between entities and from subgraphs that justify an answer, as in [Das et al., 2018], could be used to explain the extent to which a graph supports the fact. In this direction, KGRAG-Ex [Balanos et al., 2025] provides graph-based explainability on how a component of a graph influences the RAG result by removing related context from graphs and evaluating their effects. In contrast to these studies, we not only provide transparency of the RAG system through generated explanations but also use them to improve the method's accuracy.

While existing graph-RAG approaches to mitigate LLM hallucinations share common concepts, they often fail to fully leverage the advantages of knowledge graphs for explainability. For instance, FLEEK [Bayat et al., 2023] indicates to users whether a resource knowledge graph supports parts of an inferred statement; however, it does not identify the specific supporting facts or their source resources. Furthermore, it lacks a mechanism to quantify the extent to which the statement is grounded in the underlying knowledge graph. Our work addresses this gap. The following section details the proposed method.

3 EG-RAG: Explainable Graph RAG

We present EG-RAG, an explainable graph RAG approach for fact and LLM validation. Figure 2 illustrates the two complementary validation workflows of EG-RAG that share a common evidence

construction pipeline based on structured knowledge graphs. The workflows differ in the object being validated and in the role assigned to the language model.

Workflow 1: Fact Checking In the first workflow, the goal is to validate the factual correctness of the user-provided claim. The user prompt, together with the refined KG-based evidence context, is provided to a language model acting as a validator. The model performs evidence-grounded reasoning over atomic factual claims and outputs a decision (confirming *True* or refuting *False*) indicating whether the evidence supports the claim. Importantly, the language model does not generate new content in this setting, but is restricted to factual verification.

Workflow 2: LLM Validation In the second workflow, the object of validation is the output generated by a language model. A generator LLM first produces an answer to the user prompt. Subsequently, a judge LLM is provided with both the generated output and the refined evidence context constructed from the KG. The judge then evaluates the factual correctness of the generated response and outputs a binary judgment. This workflow decouples generation from verification, enabling post hoc, evidence-based validation of language model outputs.

In both workflows, we extract knowledge graphs (KGs) from textual resources (evidence) and user input user claim, reason the related triples within the extracted evidence KG. Subsequently, by extending the triples with general related knowledge, we apply the Steiner graph minimization method to remove irrelevant triples. We then deploy an LLM to further minimize the established context graph by identifying semantically prominent related triples. We then deploy LLMs to explain the relationships between triples in the refined related subgraph and the claim. We use these explanations in two ways: in Workflow 1, we directly feed them into the fact-checking LLM; in Workflow 2, we feed them into the LLM-as-Judge. In Workflow 1, we also propose a Bayesian inference approach as a white-box alternative to the LLM for fact-checking or query plausibility estimation. In the following, we explain the main components of our approach common to both workflows.

3.1 Evidence Context Construction

This part presents the right-hand side of the workflow in Figure 2. Given a user prompt, the method first extracts relevant entities and uses them to retrieve a reasoning-related subgraph from an external KG. In parallel, it extracts relations from an unstructured text corpus and represents them as a graph, enabling the augmentation of the KG with evidence.

Graph construction relies on data in graph format. Because factfulness datasets are often in raw text format, our retrieval approach treats knowledge graph construction as a preprocessing step. This step is usually referred to as the Open Information Extraction (OIE) task [Liu et al., 2024b]. Neural OIE methods, particularly when trained on test-set data, are more effective than unsupervised OIE methods. However, they are less practical due to their training requirements. We favor using LLMs with few-shot samples, which are more effective than supervised OpenIE methods and are typically trained on a more diverse training set. In experiments of [Ling et al., 2023], Llama gains the best precision on OIE2016, and GPT-3.5-TURBO gains the best recall result on ReOIE among several Rule-based and generation-based methods. We apply few-shot prompting with curated examples that distinguish relations between one subject and two objects, given a paragraph.

We match the input prompt and the triples in the resource graph by linking the entities extracted from the prompt to find relevant relations in the resource knowledge graph. To improve the matching rate and capture implicit and multi-hop relations, we identify equivalent entities with different names and contextualize the query entities with neighbours up to two hops away. In addition, we enrich the prompt entities with triples from the Wikidata corpus that describe them. We minimize the matched context graph using the all-Steiner-trees method [Sadeghi and Fröhlich, 2013], which enumerates all variants of Steiner trees on a set of terminal nodes, yielding a cluster of candidate minimal evidence subgraphs. This graph is subsequently refined into a compact evidence context, which is provided to downstream language models in both workflows. On top of the extracted context knowledge graph, we apply prompting to extract the few most prominent triples.

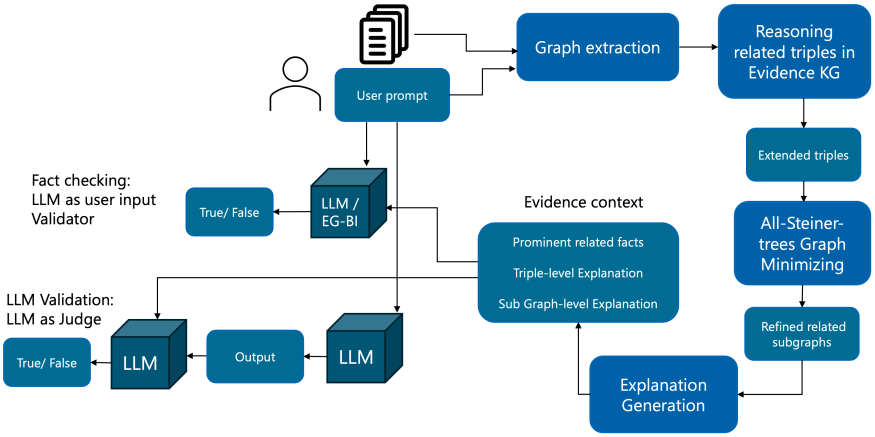

Figure 2: Two workflows for fact-checking and LLM factfulness validation.

3.2 Explanatory based Validation

This section and the following sections describe the left-hand side of the workflow diagram. By validation based on explanation, we indicate assessing factual consistency by referencing a corresponding triple in a knowledge graph and detecting contradictions or confirmations with those triples and their explanations.

Given the few prominent triples provided as evidence context, the approach further applies prompting to explain each triple's meaning and to determine whether it refutes or confirms the claim. An explanation is also generated for the subgraph of triples to justify whether, based on the provided triples and the explanation, the claim should be accepted or refuted. In both fact-checking and LLM validation workflows, the LLM makes a final judgment based on this information.

3.3 Explanability-based Plausibility Estimation

The preceding sections have described the method illustrated in Figure 2. This approach, in addition to meeting Req-1 and Req-2, covers Req-3 by generating explanations that explain how the prominent triples align with or refute claims. To provide a transparent way to deploy such explanations and demonstrate their effectiveness, we further introduce a white-box method that directly infers claim judgments from explanations. We refer to it as EG-BI, which indicates a Bayesian Inference approach that, given a set of explanations inferred from a knowledge graph, deducts the claim's plausibility. We evaluate this method only within the fact-checking workflow in this work. However, as Figure 2 illustrates, it can be applied in both workflows and we define the two key concepts below.

Key Concept 1: Spectrum of Polarity While in linguistics the polarity of statements is typically treated as binary (affirmative vs. negative), we extend the notion of polarity between facts to involve convergence, as a continuous spectrum that captures how directly or indirectly they confirm or reject each other. In addition, we define a magnitude value that indicates the strength of an affirmative or dismissive relation between two statements.

Definitions. Let H be a hypothesis ("the claim is true"). For each evidence item $i = 1, \ldots, n$, let $E_i \in [-2, 2]$ denote its *polarity score*. Where positive = confirming, negative = refuting, supporting directly +2 and supporting indirectly +1, to refute indirectly -1 and to refute directly as -2. And let $C_i \in [0, 1]$ denote the *magnitude* of that polarity estimate. The strength level ranges from 0 to 1, indicating the degree to which it supports or refutes the claim, with one denoting strong support and zero denoting weak support. We use the odds transform $O(p) = \frac{p}{1-p}$ to represent *plausibility*: the plausibility of H is $O(H) = \frac{P(H)}{1-P(H)}$. The *likelihood ratio* (LR) supplied by evidence item i is the multiplicative update on plausibility.

The above concept of fact plausibility differs from semantic alignment, in which facts primarily confirm one another through entailment or semantic correlation. Instead, it generalizes to capture indirect relations. Figure 1 illustrates a fact: Kamala Harris won the state of California in 2024. This fact does not directly support or refute the claim; it carries indirect negative polarity but does not entirely reject it. We query an LLM, using evidence resources, for the strength of its support

for the claim. Using these two values, we calculate a plausibility score indicating that it is weak evidence against Donald Trump winning the 2024 presidential election; however, it cannot refute this outright, as it reflects only a single state outcome.

From Polarity to Triple Plausibility: To quantify the uncertainty of how well a triple confirms or refutes a fact, we define the plausibility of a triple as the likelihood ratio (natural log) of polarity and magnitude. We map polarity and magnitude to a log-likelihood ratio (LR) via

$$\ln LR_i \;=\; \alpha\, E_i\, C_i,$$

so that

$$LR_i \;=\; \exp\bigl(\alpha\, E_i\, C_i\bigr).$$

Here E_i determines the *convergence polarity* (how directly or indirectly confirming vs. refuting), C_i scales the *strength* of the contribution, and $\alpha > 0$ is an optional calibration constant learned on validation data ($\alpha = 1$ if no calibration is performed).

Consequently, if $E_i > 0$, then $LR_i > 1$, meaning the evidence favors H. If $E_i < 0$, then $LR_i < 1$, meaning the evidence favors $\neg H$. The magnitude C_i scales the strength of the evidence: Higher magnitude increases the magnitude of the log-likelihood ratio, while a lower magnitude dampens it. This formulation ensures that polarity determines the direction of support, while magnitude adjusts the weight of the evidence.

Key Concept 2: Unified Factual Plausibility We define a statement's (claim) plausibility as a function of the confirmation plausibility of the resource's facts in refuting or confirming it, thereby generalizing plausibility from individual graph triples to a complete supporting subgraph.

Prior studies often treat retrieved facts as either missing knowledge or confirming information, without requiring processing, and typically provide them directly to the LLM as context. ArgRAG [Zhu et al., 2025b], in contrast, considers the inconsistency of retrieved knowledge and formulates their inconsistency as an argument resolution. However, it regards the polarities of all the evidence included in the argument as either positive or negative.

Bayesian updating with independent evidence: Assuming conditional independence of the evidence items given H and $\neg H$, Bayes' rule in odds form yields

$$O(H \mid E_{1:n}) \;=\; O(H) \prod_{i=1}^{n} LR_i,$$

$$P(H \mid E_{1:n}) \;=\; \frac{O(H \mid E_{1:n})}{1 + O(H \mid E_{1:n})},$$

where $LR_i = \frac{P(\text{obs}_i \mid H)}{P(\text{obs}_i \mid \neg H)}$ denotes the likelihood ratio contributed by the i-th evidence item.

4 Evaluation

We present four experiments on the proposed RAG system, comprising one evaluation of retrieval efficiency and three experiments on the effectiveness of explainability. In the following, we first describe the experimental setup; next, we detail each experiment; and finally, we discuss the evaluation results.

4.1 Experimental Setup

Datasets: AVeriTeC [Schlichtkrull et al., 2023] is a dataset from the series of FEVER benchmarks [Thorne et al., 2018] with 4568 real-world claims covering fact-checks by 50 different organizations. We refer to this dataset as FEVER in our experimental results. Each claim is annotated with question-answer pairs supported by evidence available online, along with textual justifications explaining how the evidence combines to produce a verdict. The Claims in AVeriTeC are classified into four labels: "Supported", "Refuted", "Not Enough Evidence", and "Conflicting Evidence/Cherry-picking". However, instead of training and development subsets from AVeriTeC that have direct answers to claims and are labeled, we take on the more realistic and challenging document subset of Evidence Collection. This dataset subset, in contrast, is unlabeled, and the provided context for rating claims can be insufficient to make a judgment.

Per Claim, this set includes three types of documents: one "answer", one "gold evidence", and several "background evidence". We utilize these in separate benchmarks as resource documents. The "answer" is a sentence or short paragraph that attempts to provide a direct answer to claims but may not cover the entire claim. The "gold" document is a text document related to the claim but does not necessarily directly answer it and may contain conflicting content. The category of "background evidence" may comprise several documents per claim, and each background document may be directly or indirectly related to the claims. In addition, they may include irrelevant data. For our evaluations, we consider up to 5 documents from

the "background evidence" set per claim. As part of establishing our benchmark dataset, we extract 190 claim topics and supporting document data across the three categories and manually label each claim.

We do not include the KG construction phase in the evaluations, and to ensure a reproducible experimental setting, we use the same KG across all experiments. To foster reproducibility and support future studies, we will release the knowledge graphs generated in our experiment upon acceptance. We use the manual claim ratings as a reference set, generated by a human who uses the "answer" paragraph, our extracted KG from it, and our approach's LLM-generated explanations of the claims. In our experiments, we use three labels: Confirmed, Rejected, or Insufficient evidence.

PubHealth [Kotonya and Toni, 2020] is a dataset for explainable fact-checking of public health claims. It contains 11,832 claims covering a wide range of topics, including biomedical science and government healthcare policy. This dataset includes one resource document per claim, which may consist of several paragraphs confirming, rejecting, or providing mixed information, as well as text that is inconclusive for a claim. We selected 150 claims from PubHealth and generated graphs for the claims and their corresponding resource documents.

We compare different methods across 4 language models Llama-3.3-70B-Instruct, Mixtral-8x7B-Instruct, Qwen2.5-VL-72B-Instruct, and GPT-4.1 The Llama, Mixtral, and Qwen models were obtained from their official HuggingFace page. We used GPT-4.1 for LLM-based graph extraction to construct evidence graphs and Qwen3-Embedding-0.6B to match entities between the query and evidence graphs. For the experiment involving traditional RAG, we used a LangChain[1] RAG comprising FAISS as the vector retriever and the all-MiniLM-L6-v2 embedding retrieved from Hugging Face.

4.2 Experiments

Evaluation of retrieval-based judgment accuracy: This evaluates the right-hand side of the workflow and assesses the accuracy of the retrieved data by estimating the percentage of judgments based on the retrieved data that are correct. We conduct this experiment on the FEVER "background evidence" subset, which comprises five documents per claim. Because the manually generated labels are based on the "answer" subset, we exclude claims for which the human rating or a tested method deems the supporting context insufficient for judgment, and we test the methods only on claims labeled as Confirmed or Rejected. Similarly, to compare methods fairly in estimating accuracy, we limit the claims validated by the vanilla LLMs to those for which the compared method could find sufficient evidence to make a judgment. Because this experiment involves retrieval among a corpus of more than one document, we include evaluation of a traditional RAG as a baseline.

Method	FEVER		PubHealth
	F-Ans	F-Gold	
LLaMA	0.730	0.576	0.540
LLaMA + TG	0.658	0.423	0.189
LLaMA + EG-BI	0.935	0.583	–
GPT	0.455	0.780	0.270
GPT + TG	0.650	0.571	0.378
GPT + EG-BI	0.760	0.888	–
GPT + EG-RAG	0.764	0.954	0.378
Mistral	–	–	0.486
Mistral + TG	–	–	0.297
Mistral + EG-RAG	–	–	0.432
Qwen	–	–	0.270
Qwen + TG	–	–	0.378
Qwen + EG-RAG	–	–	0.378

Table 1: Fact checking accuracy comparison across LLaMA3.3-70B-Instruct, Mixtral-8x7B-Instruct, Qwen2.5-VL-72B-Instruct, and GPT-4.1 methods with and without Text Grounding (TG) and with EG-RAG and Explaining based Graph Bayesian Inference (EG-BI) on FEVER_Answer and FEVER_Gold datasets.

Evaluation of effect of triple-based representations on LLM comprehension: To measure this, we evaluate how the triples, together with their explanations, perform relative to the raw text when fed into LLMs, indicating whether they are more understandable to the LLMs.

Evaluation of explanation-based reasoning: We measure the percentage of judgments of the probabilistic Validation method (EG-BI) in comparison to the LLM reasoning using explanations.

Evaluation of system correction measure: We measure the percentage of mistakes made by an LLM that an LLM-as-judge can correct when supported by our method. To calculate this measure, we count the instances in which the LLM makes a judgment error and where our method produces a correct inference for that query.

[1] https://www.langchain.com/

Method	FEVER-BG
LLama	0.741
LLama + EG-RAG	0.852
LLama	0.833
LLama + EG-BI	0.833
LLama	0.560
LLama + RAG	0.109
GPT	0.808
GPT + EG-RAG	0.909
GPT	0.750
GPT + EG-BI	0.793

Table 2: Accuracy comparison of claim judgment inference across different methods on evaluation that includes retrieval of the most related facts among 5 related background evidence documents.

Method	FEV_Ans	FEV_Gold	FEV_BG
LLaMA + EG-RAG	0.427	0.285	0.192
LLaMA + EG-BI	0.192	0.053	0.083
GPT+ EG-RAG	0.431	0.136	0.091

Table 3: Fact checking correction rate results using EG-RAG and EG-BI in comparison to vanilla LLMs on FEVER_Answer, FEVER_Gold, and FEVER_Background datasets.

4.3 Results

As part of the first experiment, Table 1 provides fact-checking accuracy comparison across multiple LLMs with and without Text Grounding (TG) and our methods on the "answer" and "gold" subset datasets of FEVER and the PubHealth dataset. This table shows striking improvements in both EG-BI and EG-RAG across all LLMs on our generated dataset, with gains ranging from 0.1 percent to 40 percent relative to vanilla LLMs. The accuracy improvement of EG-RAG is more substantial on the "gold" and EG-BI on the "answer", indicating EG-RAG's competence as we scale the amount of provided resources. Notably, Mistral and LLaMa perform better on PubHealth in the vanilla setting. This could potentially be attributed to data leakage. Reduction in performance with our method could be due to lower quality context being provided in comparision to data leakage.

Also in Table 1, as part of the second experiment, a comparison of LLMs grounded in text with our explanation-based methods can be inferred. The results indicate that explanation-based methods achieve better or on-par performance across all benchmarks, demonstrating the effectiveness of making resource documents understandable.

For the third experiment, Table 2 shows the accuracy comparison of claim judgment inference across different methods on evaluation that includes the retrieval of the most related facts among five background evidence documents of the more challenging FEVER background subset dataset. We observe that the traditional RAG achieves lower accuracy than the vanilla language model, indicating the complexity of the FEVER background document datasets. Both our EG-RAG and EG-BI methods improve LLM factuality, whereas the GPT model benefits most from explanation generation, with a 10 percent increase in accuracy. The results also show that the black-box EG-RAG outperforms EG-BI, which nevertheless improves accuracy relative to the vanilla LLM while inferring judgments transparently.

As part of the fourth experiment, Table 3 demonstrates the fact-checking correction rate results using EG-RAG and EG-BI in comparison to vanilla LLMs on the "answer", "gold", and "background" subsets of the FEVER dataset. We observe that the overall occurrence of corrections in all the benchmarks. A comparison of the results of the methods in this Table and Table 1 that while the LLM corrections are greater on the "answer" dataset, particularly for EG-RAG, on this dataset the correct Judgement of the LLMs and our methods have the least intersection.

5 Conclusion

Explainability is a critical component of our method, as it enables human understanding and verifiability of the system's outputs. A system designed to improve the reliability of LLMs becomes more trustworthy if its own operation is interpretable. Our goal is not only to detect hallucinations that challenge factfulness, but also to provide interpretable justifications grounded in atomic facts derived from knowledge graphs (KGs). We reach this goal by fulfilling explainability Req 1-3, by supporting LLMs in claim validation from evidence, broken down to a triple level, and providing a means to quantify how each component of evidence is related to the claim and how it supports or refutes the claim. In addition, we propose a method for deploying our explanation to quantify the uncertainty of the judgment of a claim.

Although we execute the modules sequentially as a workflow, we outline that it can be adopted into an agentic design, where an orchestrator layer invokes modules differently based on input data coverage and validation signals. We leave such dynamic control for future work.

6 Limitations

As shown in Table 1 for the PubHealth dataset, our approach might underperform in cases where data leakage could lead to vanilla models having better performance than our approach. If a leakage of the test dataset into trainig of LLMs be true, it suggests a core methodological risks on evaluation results that include this open-source dataset.

In addition, Errors or noise in the generated graphs can cascade through the whole approach and significantly impacts the performance. Although we address this by allowing explainability for users to make decisions, in situations where several claims are provided, users might end up with explanation overload. The tradeoff between the completeness of explanations and cognitive overload should be considered. Another limitation is that the generation of the explanations adds costs to the inference in comparison to the vanilla LLMs.

References

Georgios Balanos, Evangelos Chasanis, Konstantinos Skianis, and Evaggelia Pitoura. 2025. Kgrag-ex: Explainable retrieval-augmented generation with knowledge graph-based perturbations. *CoRR*, abs/2507.08443.

Sourav Banerjee, Ayushi Agarwal, and Saloni Singla. 2024. Llms will always hallucinate, and we need to live with this. *CoRR*, abs/2409.05746.

Mariam Barry, Gaëtan Caillaut, Pierre Halftermeyer, Raheel Qader, Mehdi Mouayad, Fabrice Le Deit, Dimitri Cariolaro, and Joseph Gesnouin. 2025. Graphrag: Leveraging graph-based efficiency to minimize hallucinations in llm-driven RAG for finance data. In *Proceedings of the 31st International Conference on Computational Linguistics, COLING 2025 - Workshops, Abu Dhabi, UAE, January 19-24, 2025*, pages 54–65. Association for Computational Linguistics.

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. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023 - System Demonstrations, Singapore, December 6-10, 2023*, pages 124–130. Association for Computational Linguistics.

Zerui Cheng, Stella Wohnig, Ruchika Gupta, Samiul Alam, Tassallah Abdullahi, João Alves Ribeiro, Christian Nielsen-Garcia, Saif Mir, Siran Li, Jason Orender, and 1 others. 2025. Position: Benchmarking is broken-don't let ai be its own judge. In *The Thirty-Ninth Annual Conference on Neural Information Processing Systems Position Paper Track*. NeurIPS 2025.

Rajarshi Das, Shehzaad Dhuliawala, Manzil Zaheer, Luke Vilnis, Ishan Durugkar, Akshay Krishnamurthy, Alex Smola, and Andrew McCallum. 2018. Go for a walk and arrive at the answer: Reasoning over paths in knowledge bases using reinforcement learning. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Markus Dreyer, Mengwen Liu, Feng Nan, Sandeep Atluri, and Sujith Ravi. 2023. Evaluating the tradeoff between abstractiveness and factuality in abstractive summarization. In *Findings of the Association for Computational Linguistics: EACL 2023, Dubrovnik, Croatia, May 2-6, 2023*, pages 2044–2060. Association for Computational Linguistics.

Adam Tauman Kalai, Ofir Nachum, Santosh S Vempala, and Edwin Zhang. 2025. Why language models hallucinate. *arXiv preprint arXiv:2509.04664*.

Ryo Kamoi, Tanya Goyal, Juan Diego Rodriguez, and Greg Durrett. 2023. Wice: Real-world entailment for claims in wikipedia. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 7561–7583. Association for Computational Linguistics.

Neema Kotonya and Francesca Toni. 2020. Explainable automated fact-checking for public health claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 7740–7754. Association for Computational Linguistics.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Chen Ling, Xujiang Zhao, Xuchao Zhang, Yanchi Liu, Wei Cheng, Haoyu Wang, Zhengzhang Chen, Takao Osaki, Katsushi Matsuda, Haifeng Chen, and Liang Zhao. 2023. Improving open information extraction with large language models: A study on demonstration uncertainty. *CoRR*, abs/2309.03433.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024a. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Pai Liu, Wenyang Gao, Wenjie Dong, Lin Ai, Ziwei Gong, Songfang Huang, Zongsheng Li, Ehsan Hoque,

Julia Hirschberg, and Yue Zhang. 2024b. A survey on open information extraction from rule-based model to large language model. In *Findings of the Association for Computational Linguistics: EMNLP 2024, Miami, Florida, USA, November 12-16, 2024*, pages 9586–9608. Association for Computational Linguistics.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12076–12100.

Afshin Sadeghi and Holger Fröhlich. 2013. Steiner tree methods for optimal sub-network identification: an empirical study. *BMC bioinformatics*, 14(1):144.

Michael Sejr Schlichtkrull, Zhijiang Guo, and Andreas Vlachos. 2023. Averitec: A dataset for real-world claim verification with evidence from the web. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and VERification. In *NAACL-HLT*.

David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. 2020. Fact or fiction: Verifying scientific claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 7534–7550. Association for Computational Linguistics.

Junde Wu, Jiayuan Zhu, and Yunli Qi. 2024. Medical graph RAG: towards safe medical large language model via graph retrieval-augmented generation. *CoRR*, abs/2408.04187.

Xiangrong Zhu, Yuexiang Xie, Yi Liu, Yaliang Li, and Wei Hu. 2025a. Knowledge graph-guided retrieval augmented generation. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2025 - Volume 1: Long Papers, Albuquerque, New Mexico, USA, April 29 - May 4, 2025*, pages 8912–8924. Association for Computational Linguistics.

Yuqicheng Zhu, Nico Potyka, Daniel Hernández, Yuan He, Zifeng Ding, Bo Xiong, Dongzhuoran Zhou, Evgeny Kharlamov, and Steffen Staab. 2025b. Argrag: Explainable retrieval augmented generation using quantitative bipolar argumentation. In *19th International Conference on Neurosymbolic Learning and Reasoning (NeSy)*, volume vol.284. Proceedings of Machine Learning Research.