Towards Trustworthy Knowledge Graph Reasoning: An Uncertainty Aware Perspective

Bo Ni[1], **Yu Wang**[2], **Lu Cheng**[3], **Erik Blasch**[4], **Tyler Derr**[1]
[1]Vanderbilt University
[2]University of Oregon
[3]University of Illinois Chicao
[4]Air Force Research Lab
{bo.ni, tyler.derr}@vanderbilt.edu, yuwang@uoregon.edu, lucheng@uic.edu, erik.blasch.1@us.af.mil

Abstract

Recently, Knowledge Graphs (KGs) have been successfully coupled with Large Language Models (LLMs) to mitigate their hallucinations and enhance their reasoning capability, e.g., KG-based retrieval-augmented framework. However, current KG-LLM frameworks lack rigorous uncertainty estimation, limiting their reliable deployment in high-stakes applications Directly incorporating uncertainty quantification into KG-LLM frameworks presents challenges due to their complex architectures and the intricate interactions between the knowledge graph and language model components. To address this crucial gap, we propose a new trustworthy KG-LLM framework, UAG (**U**ncertainty **A**ware Knowledge-**G**raph Reasoning), which incorporates uncertainty quantification into the KG-LLM framework. We design an uncertainty-aware multi-step reasoning framework that leverages conformal prediction to provide a theoretical guarantee on the prediction set. To manage the error rate of the multi-step process, we additionally introduce an error rate control module to adjust the error rate within the individual components. Extensive experiments show that UAG can achieve any pre-defined coverage rate while reducing the prediction set/interval size by 40% on average over the baselines.

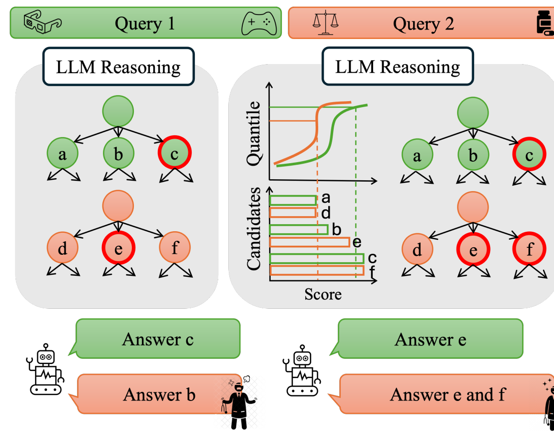


Figure 1: An illustration of uncertainty quantification in the context of knowledge graph question answering. By quantifying a confidence boundary (dashed lines), UQ methods can capture more correct answers (red circles) when facing uncertainties under different scenarios.

Introduction

Large Language Models (LLMs) have recently achieved impressive performance in question-answering tasks due to their unprecedented capability of understanding complex linguistic patterns and generating coherent responses (OpenAI et al. 2024; Huang and Chang 2022). However, LLM's frequent hallucination (Huang et al. 2023b; Mündler et al. 2024) and lack of reasoning capability (Lin, Hilton, and Evans 2021; Fu et al. 2023) still limit their practical usage when facing domain-specific or complex questions.

To address the challenges, recently knowledge graphs (KGs) have been coupled with LLMs to provide factual data and contextual grounding, enhancing the reliability and accuracy of their responses. Knowledge graphs, as a form of structural knowledge representation consisting of factual triplets, provide two unique advantages to overcome the aforementioned challenges of LLMs. First, the additional information retrieved from KGs is directly extracted from reliable external sources, ensuring the currency and faithfulness of the additional context upon which LLMs generate answers. Second, KGs are typically represented as triplets where entities are connected by heterogeneous relations, providing explicit logic to enable reasoning and fulfilling complex tasks. To effectively leverage the above two advantages of KGs to enhance LLMs, two representative methods have been developed: LLM-based Structural Query and KG-based Retrieval-augmented Generation (KG-RAG). Structural Query relies on LLMs to generate structural queries such as SPARQL (Jiang et al. 2023) or traverse the KG via beam search (Sun et al. 2023; Wang et al. 2024). KG-RAG works by deriving factually correct and context-relevant content from KG to augment the generation of LLMs (Luo et al. 2024; Li et al. 2023; Xie et al. 2022; Baek, Aji, and Saffari 2023; Yang et al. 2023; Wang et al. 2023a).

Despite KG-RAG and Structural Queries' effectiveness in various real-world applications, there is no systematic investigation into the uncertainty quantification (UQ) aspect of KG-LLM systems, and therefore, their deployment remains limited in high-stakes scenarios (e.g., medical analysis, financial decision-making, etc.) where the cost-of-errors is significant. As illustrated in Figure 1, without proper uncertainty quantification, existing approaches fail to generate answers that accurately reflect output confidence. Recently, several methods have been proposed to model uncertainty for LLMs (Ye et al. 2024; Lin, Trivedi, and Sun 2023).

One popular technique for uncertainty quantification is conformal prediction (CP) (Angelopoulos and Bates 2021). Given a user-specified error rate α, CP produces a prediction set that guarantees a coverage rate $1-\alpha$ by calibrating the model prediction on a hold-out calibration set. Despite the model agnostic, distribution free characteristics of CP, directly applying it to LLMs is challenging because of the unbounded output space. To address it, Ye et al. and Kumar et al. used the logits generated by the contemporary causal language models as the prediction probability; Su et al. and Lin, Trivedi, and Sun considered the commercial black-box language models and sampled the responses to approximate the confidence; Most recently, Quach et al. extended the general risk control framework to enable conformal prediction in large language models (Quach et al. 2023).

Despite the numerous methods that have utilized conformal prediction (CP) to model the uncertainty of large language models (LLMs), no previous work has focused on equipping knowledge graph-language models (KG-LLMs) with uncertainty quantification. This task presents several unique challenges. First, the complexity of knowledge graphs (KGs), characterized by multiple hops connecting queries to solutions, makes it difficult to accurately calibrate the graph reasoning process. Second, the high-dimensional output distribution of the underlying language models further complicates the calibration process. In KG-LLM applications, there are often multiple valid solutions to a single query, making the coverage of these solutions critically important. Thus, the uncontrolled high-dimensional output poses a significant challenge because it complicates the process of identifying and validating the correct solutions among many possibilities. This lack of control can lead to the propagation of errors and reduced reliability in the model's predictions. To address these challenges, novel techniques specifically tailored for knowledge graph question answering (KGQA) need to be developed.

Therefore, to fill this gap, in this paper, we introduce UAG (**U**ncertainty **A**ware Knowledge-**G**raph Reasoning), a novel uncertainty-aware knowledge graph reasoning framework that takes advantage of both the structural knowledge representation and the reasoning capability of LLM. Specifically, instead of relying on LLM to generate the answers from the open domain, we retrieve an initial set of uncertainty aware answers through beam searching the knowledge graph. We guide the beam search process with conformal prediction to achieve the theoretically guaranteed coverage rate. In order to facilitate more faithful reasoning in LLM, we also retrieve reasoning paths through a *planning-retrieval* module (Luo et al. 2024). Then, we leverage the LLM to generate the answer with the retrieved reasoning paths. To combine the power of generation and the fidelity of the retrieved candidates, we additionally calibrate the similarity measurements between the retrieved candidates and the generated answers. One challenge of the multi-step framework is the propagation of the error rates. In order to adjust the error rate aggregation across multiple components, we leverage the Learn Then Test (LTT) framework (Angelopoulos et al. 2021) to control the error rate in the individual components.

We perform experiments on two widely used multi-hop knowledge graph QA datasets and demonstrate that UAG is able to satisfy the uncertainty constraint while maintaining a reasonable size of prediction set.

In summary, our contributions are summarized as follows:

- We propose a novel framework, UAG, for uncertainty quantification with LLM-based KGQA, addressing a significant gap in the existing literature.
- We extend the *learn-then-test* paradigm to KGQA by modeling the distribution of the error propagation, improving the reliability of the model outputs.
- Extensive experiments demonstrate the effectiveness of our framework on two multi-hop knowledge graph QA benchmarks, outperforming baseline methods in terms of both coverage and robustness under uncertainty.

The rest of the paper is organized as follows: Section 2 provides the preliminaries; Section 3 introduces our UAG in detail; experiments and ablation studies in Sections 4 and 5; related work in Section 6; lastly, conclusions and future work are presented in Sections 7 and 8.

Preliminiaries

Conformal Prediction

Conformal prediction (CP) is a distribution-free model-agnostic approach for uncertainty quantification. By calibrating the model prediction on a held-out calibration set, CP produces sets of predicted intervals that contain the ground-truth labels with a user-specified error rate α (Angelopoulos and Bates 2021; Shafer and Vovk 2008).

Without loss of generality, considering the i^{th} sample (x_i, y_i) with x_i being the input feature and y_i being the corresponding ground-truth output, we denote the calibration set as $\mathcal{D}^{\text{cal}} = \{(x_i, y_i)\}_{i=1}^{n}$. Conformal Prediction (CP) guarantees the error rate of the prediction set with the following steps.

- **Define non-conformal score:** A heuristic-based uncertainty estimation function $S : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$ is defined to calculate the non-conformal score $s_i = S(x_i, y_i)$, which measures the uncertainty of the prediction y_i given the input x_i with larger score indicating worse agreement between x_i and y_i.
- **Compute the quantile:** CP then calculates the non-conformal score for every input-output pair in the calibration set to obtain $\mathcal{S}^{\text{cal}} = \{s_i\}_{i=1}^{n}$ and further compute the conformal score $q_\alpha^{S,\mathcal{D}_{\text{cal}}}$ as the $\frac{\lceil (n+1)(1-\alpha) \rceil}{n}$ quantile of the calibration scores $\mathcal{S}^{\text{cal}}$:

$$q_\alpha^{S,\mathcal{D}_{\text{cal}}} = \text{Quant}\Big(\{S(x,y) \mid (x,y) \in \mathcal{D}_{\text{cal}}\}, \frac{\lceil (n+1)(1-\alpha) \rceil}{n}\Big) \tag{1}$$

- **Prediction set:** the final prediction set satisfying the α error rate with respect to the calibration set $\mathcal{D}_{\text{cal}}$ would be $C(X_{\text{test}}) = \{y | y \in \mathcal{Y}, S(X_{\text{test}}, y) \leq q_\alpha^{S,\mathcal{D}_{\text{cal}}}\}$.

Learn Then Test (LTT)

Recently, the Learn Then Test (LTT) framework (Angelopoulos et al. 2021) was introduced that extends conformal prediction to manage the expectation of any loss functions. Specifically, it is achieved by approaching the hyperparameter selection as a multiple-hypothesis testing problem. Next, we formally introduce the LTT framework.

Formally, let $L_\lambda : \mathcal{Y} \times \hat{\mathcal{Y}} \rightarrow \mathbb{R}$ be any loss function with a hyperparameter configuration $\lambda \in \Lambda$. Notably, λ could be multi-dimensional. Let $\alpha \in \mathbb{R}$ be the user-defined error rate for L_λ (i.e., $\mathbb{E}(L_\lambda) \leq \alpha$). Using the calibration set $\mathcal{D}^{\text{cal}}$, LTT then computes a set of valid configurations $\Lambda_{\text{valid}} \in \Lambda$ that satisfies

$$\mathbb{P}\left(\sup_{\lambda \in \Lambda_{\text{valid}}} \mathbb{E}[L_\lambda | \mathcal{D}_{\text{cal}}] \leq \alpha\right) \geq 1 - \delta \tag{2}$$

where δ represents the desired confidence level on the selection of configurations. Intuitively, δ is the probability that the valid configurations we identify will truly meet our error guarantee. To calibrate the configurations on $\mathcal{D}_{\text{cal}}$, we define the null hypothesis $H_0^\lambda : \mathbb{E}[L_\lambda] > \alpha$ for each $\lambda \in \Lambda$ and calculate a super-uniform p-value p_λ using concentration inequalities. Then we can leverage any family-wise error rate (FWER) controlling algorithms to identify the non-rejected configurations Λ_{valid}.

Theorem 1 (Learn Then Test (Angelopoulos et al. 2021))**.** Suppose p_λ is super-uniform under $H_0^\lambda \forall \lambda \in \Lambda$. Let $\mathcal{T}$ be any FWER-controlling algorithm at level δ. Then Λ_{valid} satisfies Eq. (2).

Problem Definition

Given a knowledge graph $\mathcal{G} = (\mathcal{E}, \mathcal{R})$, where $\mathcal{E}$ is the set of entities and $\mathcal{R}$ is the set of relations. The edges in the knowledge graph are represented as triplets (i.e., $(e_s, r, e_o) \in \mathcal{G}$) with $e_s \in \mathcal{E}$ being the head entity, $r \in \mathcal{R}$ being the relation, and $e_o \in \mathcal{E}$ being the tail entity. For the rest of this paper, without further specification, we denote the calligraphic font as sets (such as $\mathcal{X}, \mathcal{Y}$) and capital letters as functions (such as S, F, for score functions and loss functions, respectively).

Definition 1 (Uncertainty quantification for multi-hop knowledge graph question answering)**.** Given a question q in natural language with the ground-truth answer set being $\mathcal{Y}_{\text{test}} \subset \mathcal{E}$, assuming the user specified error rates as α and δ, the task here is to derive a algorithm C_λ hyper-parametrized by λ, which takes the input X_{test} and predicts the output set $C_\lambda(X_{\text{test}})$ that satisfies:

$$\mathbb{P}\left(\mathbb{P}\left(\hat{e} \in \mathcal{Y}_{\text{test}}, \forall \hat{e} \in C_\lambda(X_{\text{test}}) \mid \mathcal{D}_{\text{cal}}\right) \geq 1 - \alpha\right) \geq 1 - \delta \tag{3}$$

Remark 1. The error rate δ controls the probability that the inner probabilistic guarantee holds. Specifically, it ensures that with at least $(1 - \delta)$ confidence, the prediction set $C_\lambda(X_{\text{test}})$ will contain the correct answers with a probability of at least $(1 - \alpha)$.

Intuition: Consider α as a measure of how tolerant we are to errors in individual predictions. The parameter δ, on the other hand, controls our overall confidence in this process.

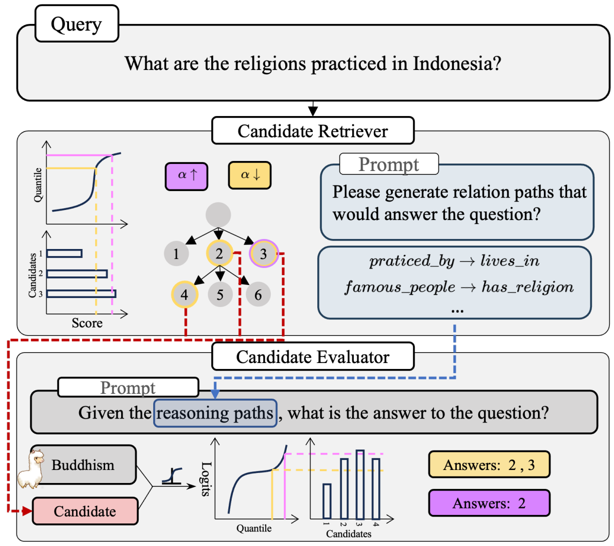


Figure 2: An illustration of our UAG framework.

Method

UAG is a multi-hop knowledge graph reasoning framework that incorporates uncertainty quantification (UQ) into its reasoning process. It consists of three components: UQ-aware candidate retriever, UQ-aware candidate evaluator, and Global Error Rate Controller, as shown in Figure 2. Given a user-specified query, the **UQ-aware Candidate Retriever** selectively retrieves neighboring nodes and formulates reasoning paths to ensure a pre-defined error rate. The **UQ-aware Candidate Evaluator** then reasons over the retrieved candidates and reasoning paths to produce the final answer set. Subsequently, to avoid overly conservative predictions, the **Global Error Rate Controller** adjusts individual error rates by calibrating them using the LTT framework introduced in the previous section.

UQ-aware Candidate Retriever

The first component of our UAG framework is the candidate retriever, which retrieves candidate entities from the knowledge graph based on the question or current paths. Previous research shows that multi-hop graph traversal frameworks improve LLM reasoning and answer faithfulness (Wang et al. 2024; Sun et al. 2023). However, these frameworks typically use heuristics like Top-K candidate selection based on textual similarity, which lack a theoretical basis and cannot ensure a consistent error rate, making them unreliable for high-stakes scenarios. To address this, we incorporate uncertainty quantification into graph traversal, replacing heuristic selection with error-bounded selection. Specifically, we apply conform prediction to two steps: retrieving candidate paths and candidate neighbors.

Retrieving Candidate Path Formally, for each traversed path currently ending at node v, we update the breadth-first-search tree queue by following:

$$\left\{ s \;\middle|\; s \in \mathcal{N}(v),\; S_1\left(Q||(||_{i=0}^{j-1} r_i), r_j\right) < q_{\alpha_1}^{S_1, \mathcal{D}_{\text{cal}}} \right\} \tag{4}$$

where $\mathcal{N}(v)$ is the neighborhood set of v, $Q||(||_{i=0}^{j-1} r_i)$ are the concatenated($||$) relations traversed so far in the path with starting passages being Q, r_j is the current relation between the entities v and s, and S_1 is the uncertainty estimation score for the candidate retriever. In our context, we choose S_1 to be the textual similarity between two textual passages. Note that the graph traversal starts with a provided entity/node $v = e_0$ and the initial relation is chosen by calculating the similarity between the question and the relations of e_0. In other words, for the initial step, we are calculating $S_1(Q, r_j)$ as in Eq. (4).

Retrieving Candidate Neighbor While traversing the graph, we also want to determine if the current node should be added into the candidate set. As a result, we add the following set to the prediction set while visiting node v:

$$\left\{s \;\middle|\; s \in \mathcal{N}(v), S_1\left(Q, ||_{i=0}^{j} r_i\right) < q_{\alpha_2}^{S_1, \mathcal{D}_{\text{cal}}}\right\} \tag{5}$$

Eq. (4) differs from Eq. (5) in their focus during the traversal process. Eq. (4) focuses on identifying the likely answer paths by assessing the similarity between the traversed path and the current relation. In contrast, Eq. (5) evaluates if the entire path up to the current relation sufficiently answers the query by comparing it to the initial query. This distinction allows for dynamic adjustment of the traversal process and candidate set based on the evolving query context and encountered relations.

UQ-aware Candidate Evaluator

The second component of UAG is an uncertainty-aware candidate evaluator based on large language models (LLMs). Directly prompting the language model has demonstrated significant advantages in knowledge graph question answering (KGQA) tasks, particularly due to its ability to generate diverse and contextually relevant answers. However, this open-ended generation approach lacks any theoretical guarantees on the correctness of the answers, which can lead to unreliable outputs. Conversely, the retrieved candidates are constrained by a predefined error rate, ensuring the reliability of the prediction set. However, this constraint often results in overestimating the prediction set size, encompassing a broader range of potential answers than necessary.

Thus, to harness the strengths of both approaches, we introduce a calibration process that optimizes the balance between them. By aligning the similarity between the LLM-generated candidates and the retrieved set, we aim to refine the prediction set size while maintaining a bounded error rate. This ensures that the final output is not only theoretically sound but also practically effective.

To calculate the final answer set, we define the non-conformal score for the evaluator by taking the similarity between the retrieved answers and the generated answers. Formally, given a set of candidates $\mathcal{C}$ and retrieved reasoning paths $\mathcal{P}$, let Φ be the LLM generation function. Then, the final answer set is defined as follows:

$$\left\{a \in \mathcal{C} \mid S_1\big(a, \Phi(\mathcal{P})\big) < q_{\alpha_{\text{llm}}}^{S_1, \mathcal{D}_{\text{cal}}}\right\} \tag{6}$$

Global Error Rate Controller

Since UAG involves multiple components, directly applying the user-defined error rate α to Eqs. (4)-(6) may not achieve the desired error rate. Errors from the knowledge graph traversal can propagate to the inference stage, potentially exceeding the user-specified tolerance unless errors across components are perfectly correlated.

To address this issue, we leverage the LTT framework (Section 2) to find the best error rate for each component by treating their error rates as hyper parameters. Let $\lambda = (\alpha_1, \alpha_2, \alpha_3) \in (0, 1]^3$ be the individual error rates for each component. The finite search space Λ can be represented as the Cartesian product of these sets for α_1, α_2, and α_3:

$$\Lambda = \{(\alpha_1, \alpha_2, \alpha_3) \mid \alpha_1, \alpha_2, \alpha_3 \in \{h, 2h, \ldots, 1\}\}$$

where h is the hyper parameter to control the size of the search space, bounded by 0 and 1. For each $\lambda \in \Lambda$, we compute a valid p-value p_λ for the null hypothesis $H_0^\lambda : \mathbb{E}[L_\lambda] > \alpha$ by computing the concentration inequality on the calibration dataset D_{cal}.

Theorem 2 (Binomial tail bound p-values (Quach et al. 2023))**.** Let Binom(n, α) denote a binomial random variable with sample size n and success probability α. Then $p_\lambda = \mathbb{P}(\text{Binom}(n, \alpha) \le \sum_{D_{cal}} L_\lambda$ is a valid p-value for $H_0^\lambda : \mathbb{E}[L_\lambda] > \alpha$.

In our context, we define the loss function F as the error rate in the system. Let $\mathcal{T}$ be a FWER-controlling algorithm[1] $\mathcal{T} : (\mathcal{P}, \mathbb{R}) \to \mathcal{P}'$ where $\mathcal{P}$ denotes a family of p-values that we want to control based on a given error rate. Then we can identify a set of configurations $\Lambda_{valid} \subseteq \Lambda$ that satisfies Eq. (2) by taking $\mathcal{T}(\{p_\lambda \mid \lambda \in \Lambda\}, \delta)$ as Λ_{valid}.

Theorem 3 (Coverage Guarantee of UAG)**.** If $\lambda \in \Lambda_{valid}$, then Eq. (3) is satisfied.

Proof. From Eq. (2), $\forall \lambda \in \Lambda_{valid}$,

$$\mathbb{P}\left(E[L_\lambda \mid D_{\text{cal}}] \le \alpha\right) \ge 1 - \delta$$

In the context of KGQA,

$$L_\lambda = \mathbb{1}\{\nexists \hat{e} \in C_\lambda(X) : \hat{e} \in \mathcal{E}_{ans}\}$$

Note that,

$$\begin{aligned} E[L_\lambda \mid D_{\text{cal}}] &= E[\mathbb{1}\{\nexists \hat{e} \in C_\lambda(X) \colon \hat{e} \in \mathcal{E}_{ans}\} | D_{cal}] \\ &= \mathbb{P}\left(\nexists \hat{e} \in C_\lambda(X) \colon \hat{e} \in \mathcal{E}_{ans} | D_{cal}\right) \end{aligned}$$

Substitute $E[L_\lambda \mid D_{\text{cal}}]$ in eq. 2, then we have

$$\mathbb{P}\left(\mathbb{P}\left(\nexists \hat{e} \in C_\lambda(X) \colon \hat{e} \in \mathcal{E}_{ans} | D_{cal}\right) \le \alpha\right) \ge 1 - \delta$$

This is equivalent to

$$\mathbb{P}\left(\mathbb{P}\left(\hat{e} \in \mathcal{E}_{\text{ans}}, \forall \hat{e} \in C_\lambda(X_{\text{test}}) \mid D_{\text{cal}}\right) \ge 1 - \alpha\right) \ge 1 - \delta \tag{7}$$

□

In addition, to minimize the returned set size to make the prediction most useful to the users, we use the validation set to select the configuration $\lambda \in \Lambda_{valid}$ with the smallest average set size.

[1] FWER-controlling algorithms take a set of null hypothesis and ensures the type-I error rate is controlled by a pre-defined value δ.

Datasets	#Train	#Test	Extracted KG Subgraphs $\frac{Max}{\#hops}$	$\frac{Avg}{\#Nodes}$	$\frac{Avg}{\#Edges}$
WebQSP	2,826	1,628	2	1,374	2,909
CWQ	27,639	3,531	4	1,256	2,615

Table 1: Statistics of KGQA datasets.

Experiments

In this section, we conduct extensive experiments on two widely used multi-hop KGQA datasets (Luo et al. 2024; Sun et al. 2023; Lan et al. 2022). Our experiments are designed to rigorously evaluate the uncertainty quantification performance of UAG, employing standard UQ metrics to assess its effectiveness against state-of-the-art baselines. We provide detailed analysis to identify key strengths and potential areas of improvement. This includes a case study and ablation studies beyond our primary experiments, such as . Additional experiments include comparing UAG to standard KGQA methods are further detailed in the ablation studies.

Datasets

We evaluate UAG with two widely used benchmark dataset for KGQA: WebQuestionSP (WebQSP) (Yih et al. 2016) and Complex WebQuestions (CWQ) (Talmor and Berant 2018). WebQSP contains up to 2 hop questions, and CWQ contains up to 4 hop questions. Additionally, Freebase (Bollacker et al. 2008) is used as the underlying knowledge graph for both datasets. The dataset statistics are presented in Table **??**. Note that for calibration, we use the training partition.

Evaluation Metrics

Compared to traditional KGQA tasks that typically rely on evaluation metrics such as hits@1 (Luo et al. 2024), our approach focuses on measuring the uncertainty quantification of the methods. To this end, we use empirical coverage rate and average prediction set size as the key metrics to measure the accuracy and efficiency of the models, respectively (Angelopoulos and Bates 2021). We aim to provide insights into how well the various methods balance accuracy with the efficiency of its predictions. The details of these metrics are the following:

- **ECR (Empirical Coverage Rate)**: This measures how well the uncertainty quantification model satisfies the error rate. For knowledge graphs, we consider the accuracy of the prediction set.
- **APSS (Average Prediction Set Size)**: This evaluates the effectiveness of the uncertainty quantification model, with smaller average prediction set sizes indicating greater efficiency in selecting the most likely answers.

Baselines

We compare UAG with the following uncertainty quantification baselines. Because there has been no existing work considering uncertainty quantification in the KG-LLM paradigm, we include the previously proposed existing LLM-based uncertainty quantification methods as baselines.

- **Top-K**: Non-CP method without coverage guarantee. It includes responses with the K highest probabilities for each question in the test set (Huang et al. 2023a).
- **Standard Split Conformal Prediction (SplitCP**). Standard conformal prediction (Shafer and Vovk 2008) where we follow the framework outlined in previous work for its application on language models (Ye et al. 2024).
- **Conformal Language Modeling (CLM)** A logit-based CP method that utilizes the general risk control framework directly on the output of the large language models (Quach et al. 2023).
- **Logit-free Conformal Prediction for LLMs (LoFreeCP)**. A state-of-the-art CP method that designs specific techniques for handling low-frequency events in language models (Su et al. 2024).

Implementation Details

For the implementation, we set δ to be 0.05, and use Llama3-8b as our backbone large language model. For the encoder g, we use the SentenceTransformer model and pre-train it on our training data. Other implementation details are further documented in the appendix and our code is publicly available[2].

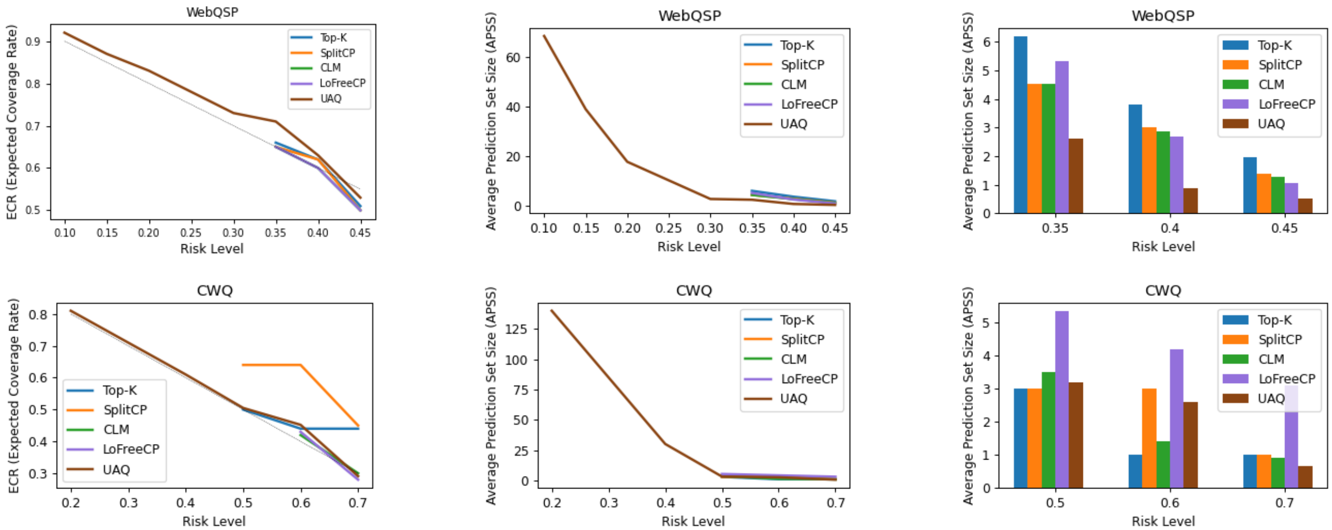
Primary Results

The experimental results are presented in Figure 3. For clarity and conciseness, we provide the most relevant graphs in the main paper and include the detailed tabular results in the supplementary material. In Figure 3(a), we first plot the Empirical Coverage Rate (ECR) against the risk level. The region above the diagonal dashed line, represents the acceptable coverage rate given the specified risk tolerance level. As anticipated, all CP-based methods satisfy the error guarantee within this boundary. However, we observe that the baseline methods fail to achieve error rates below 0.35, as their generation components alone cannot produce results better than this threshold. In contrast, our proposed UAG, which integrates both retrieval and generation, consistently achieves the desired error rates across a wider range of risk levels.

In Figure 3(b) (i.e., the second column), we plot the prediction set size against the risk level. As expected, with an increase in risk level, there is a corresponding decrease in prediction set size, reflecting the trade-off between risk and prediction confidence. This trend is observed as the model becomes less conservative at higher risk levels, thereby reducing the number of elements in the prediction set.

Finally, we compare the zoomed-in view of prediction set sizes across different methods in Figure 3(c). Despite the performance at the 0.6 level in CWQ, our proposed UAG consistently outperforms or performs competitively with the baseline methods, demonstrating its ability to maintain high accuracy while reducing uncertainty in predictions. This advantage can be attributed to the expressive power of the traversed reasoning paths, which effectively guide the retrieval process, as well as the robust combination of retrieval and generation that enhances the precision of the prediction set.

[2]https://github.com/Arstanley/UAG



(a) Risk Level vs. ECR (b) Risk Level vs. APSS (c) APSS

Figure 3: Uncertainty quantification results for UAG and the baselines.

Question:	*What religions are practiced in Indonesia?*
Ground Truth:	*Catholicism*, *Hinduism*, *Protestantism*, *Islam*
Predicted Answers:	
RoG (Luo et al. 2024)	**Islam**
UAG	**Catholicism**, **Hinduism**, **Protestantism**, The Religion of Java, **Islam**, The Language of the Gods

Table 2: Case Study on Sample Question from WebQSP.

Case Study

We demonstrate the real-life implications and usefulness of UAG through a detailed case study. Specifically, we select an instance among the samples where existing KGQA methods produce mispredicted or incomplete answers and analyze how UAG overcomes these challenges.

As shown in Table 2, the ground truth to the question *What are the religions practiced in Indonesia?* includes four distinct answers, showcasing the diversity of religious practices in the country. However, directly prompting the question to the language models results in only a single answer—Islam—even when using the state-of-the-art KGQA model. This outcome poses a significant threat to fairness and may introduce bias, as it fails to capture the full spectrum of religious diversity in Indonesia. UAG addresses this challenge by incorporating risk level control, which allows for a more comprehensive and balanced prediction.

The prediction set is constructed by setting the risk tolerance level to 0.2. While the resulting set includes several additional related answers—The Religion of Java, a reference to a popular book by anthropologist Clifford Geertz focused on the religion of Java (an island of Indonesia), and The Language of the Gods, an alias for Sanskrit, the ancient classical language of Hinduism—it successfully covers all four ground truth answers. This ensures a more accurate and inclusive representation of the answer to the given question, thereby reducing the risk of bias and enhancing the trustworthiness of the predictions.

Ablation Studies

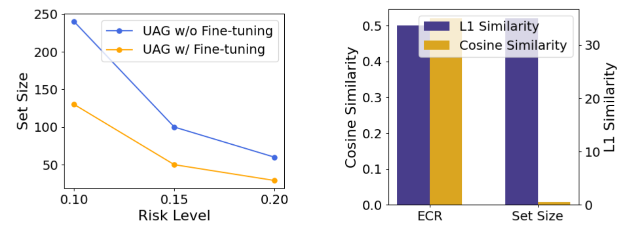
We conduct ablation studies on UAG to comprehensively evaluate its effectiveness and robustness. Specifically, we first examine how UAG compares to state-of-the-art KGQA methods, as robust performance on standard KGQA tasks is critical in daily uses. Additionally, we investigate the effects of pre-training and different similarity measures on the model performance.

Comparing to KGQA Methods Here we compare our method to state of the art KGQA methods (Luo et al. 2024) by selecting the top responses from UAG given a fixed error rate α. We select $\alpha = 0.2$ for the purpose of testing. Our result is shown in Table **??**. Overall, the table highlights that our model attains performance on par with current state-of-the-art for the traditional benchmark KGQA tasks (that utilize Hits@1 for evaluation). This underscores the robust reasoning capabilities of the UAG and the efficiency of the candidate retrieval process.

Effects of LM Pre-Training As shown in Figure 4(a), pre-training the sentence transformers (specifically for graph traversal) significantly reduces the size of the prediction set. This improvement is likely because fine-tuning at this stage enhances the quality of the initial traversal on the knowledge graph, leading to more accurate retrieval of answers. This process is analogous to better indexing in the context of Retrieval-Augmented Generation (RAG). However, when

Category	Method	Hits@1
Fine-tuned	TIARA (Shu et al. 2022)	75.2
	DeCAF (Yu et al. 2023)	82.1
	RoG (Luo et al. 2024)	**85.7**
Prompting	COT (Wei et al. 2022)	39.1
	ToG (Sun et al. 2023)	57.4
	KD-CoT (Wang et al. 2023b)	63.7
	StructGPT (Jiang et al. 2023)	72.6
	UAG-Top-1	66.8
	UAG-Top-3	**73.4**

Table 3: KGQA Results on WebQSP.



(a) Pre-Training (b) Similarity Measures

Figure 4: Ablation Studies where (a) shows the benefits of pre-training the sentence transformer used in graph traversal and (b) shows a comparison when using $L1$ similarity as opposed to cosine similarity within UAG.

we pre-trained the large language model backbone, we did not observe a substantial improvement. This outcome aligns with previous findings suggesting that fine-tuning large language models can increase model uncertainty (Ye et al. 2024).

Alternative Similarity Measures We also experimented with alternative methods for measuring textual similarity, specifically S_1 as defined in Eq. 4. As shown in Figure 4(b), we found that using cosine similarity significantly improves prediction efficiency while maintaining the user-defined error rate compared to L1 similarity. This improvement can be attributed to cosine similarity's focus on the angle between vectors, which makes it less sensitive to variations in vector magnitudes. This is particularly helpful in high-dimensional spaces, as it allows cosine similarity to more effectively capture semantic relationships between text representations.

Ablation Studies Discussion Overall, UAG demonstrates strong performance compared to existing KGQA methods even though it was not specifically designed for the traditional setting. Through the ablation studies, we observed that while UAG benefits significantly from fine-tuning sentence transformers for graph traversal, the pre-training of large language models does not yield substantial gains, which we leave further investigation on finding improvements with fine-tuning LLMs as one future work in this direction. Lastly, through a case study we were able to more deeply understand the benefits of the proposed trustworthy knowledge graph reasoning framework.

Related Work

Knowledge Graph Reasoning. Knowledge graph reasoning enables effective inference by leveraging structured information for tasks like question answering, entity prediction, and link prediction (Chen, Jia, and Xiang 2019; Liang et al. 2022; Chen et al. 2020; He et al. 2021; Xie et al. 2022). Recent methods, such as MINES (Liang et al. 2024), enhance inductive reasoning by utilizing neighbor-enhanced sub-graph structures, improving embedding adaptability. Moreover, to leverage the powerful capabilities of LLMs with structured data, researchers have developed RAG-based approaches that combine symbolic and neural reasoning methods to produce more accurate and contextually relevant answers. For example, Luo et al. fine-tuned LLMs to jointly generate relation plans and answers (Luo et al. 2024), while Sun et al. and Wang et al. proposed LLM-guided graph traversal methods for efficient retrieval (Sun et al. 2023; Wang et al. 2024). This integration of has shown promising results in complex reasoning tasks, offering improved performance across many domains.

LLM Uncertainty Quantification. Uncertainty quantification has emerged as a critical research area for developing trustworthy AI systems, especially in contexts where decision-making involves significant risks. Recently, growing attention has been directed towards modeling uncertainty in large language models (LLMs). Researchers have extended the general risk control framework to text generation and robot planning (Quach et al. 2023; Ren et al. 2023). LofreeCP further extended conformal prediction to black-box language models by approximating the non-conformity score with repeated prompting, thereby improving the reliability of LLM outputs (Su et al. 2024). Additionally, researchers have proposed novel approaches to quantify uncertainty in retrieval-augmented generation by calibrating uncertainty across both the retrieval and generation processes with parameter tuning (Rouzrokh et al. 2024; Li et al. 2024).

Conclusion

In this paper, we tackle the challenge of uncertainty quantification in knowledge graph question answering by integrating conformal prediction with KG-LLM models. Our architecture leverages the Learn Then Test (LTT) framework for multi-step calibration, delivering reliable results with pre-defined error rates and practical prediction set sizes. Extensive experiments demonstrate our method's effectiveness in balancing accuracy and uncertainty, making it suitable for real-world applications. While focused on knowledge graphs, this approach could extend to open-domain QA, though challenges persist due to the lack of structured graph properties. Future work can explore these adaptations to ensure robust uncertainty quantification across tasks.

Acknowledgements

This research is supported by the National Science Foundation (NSF) under grant number IIS2239881.

References

Angelopoulos, A. N.; and Bates, S. 2021. A Gentle Introduction to Conformal Prediction and Distribution-Free Uncertainty Quantification. *arXiv preprint arXiv:2107.07511*.

Angelopoulos, A. N.; Bates, S.; Candès, E. J.; Jordan, M. I.; and Lei, L. 2021. Learn then test: Calibrating predictive algorithms to achieve risk control. *arXiv preprint arXiv:2110.01052*.

Baek, J.; Aji, A. F.; and Saffari, A. 2023. Knowledge-Augmented Language Model Prompting for Zero-Shot Knowledge Graph Question Answering. arXiv:2306.04136.

Bollacker, K. D.; Evans, C.; Paritosh, P. K.; Sturge, T.; and Taylor, J. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *SIGMOD Conference*.

Chen, X.; Jia, S.; and Xiang, Y. 2019. A review: Knowledge reasoning over knowledge graph. *Expert Systems with Applications*, 141: 112948.

Chen, Z.; Wang, Y.; Zhao, B.; Cheng, J.; Zhao, X.; and Duan, Z. 2020. Knowledge Graph Completion: A Review. *IEEE Access*, 8: 192435–192456.

Fu, Y.; Peng, H.; Khot, T.; Sabharwal, A.; Clark, P.; and Choi, Y. 2023. Are Emergent Abilities of Large Language Models a Mirage? *arXiv preprint arXiv:2304.15004*.

He, G.; Lan, Y.; Jiang, J.; Zhao, W. X.; and Wen, J.-R. 2021. Improving Multi-hop Knowledge Base Question Answering by Learning Intermediate Supervision Signals. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*. ACM.

Huang, J.; and Chang, K. C.-C. 2022. Towards Reasoning in Large Language Models: A Survey. *arXiv preprint arXiv:2212.10403v2*.

Huang, J.; Xi, H.; Zhang, L.; Yao, H.; Qiu, Y.; and Wei, H. 2023a. Conformal prediction for deep classifier via label ranking. *arXiv preprint arXiv:2310.06430*.

Huang, L.; Yu, W.; Ma, W.; Zhong, W.; Feng, Z.; Wang, H.; Chen, Q.; Peng, W.; Feng, X.; Qin, B.; and Liu, T. 2023b. A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions. Submitted on 9 Nov 2023, arXiv:2311.05232.

Jiang, J.; Zhou, K.; Dong, Z.; Ye, K.; Zhao, W. X.; and Wen, J.-R. 2023. StructGPT: A General Framework for Large Language Model to Reason over Structured Data. arXiv:2305.09645.

Kumar, B.; Lu, C.; Gupta, G.; Palepu, A.; Bellamy, D.; Raskar, R.; and Beam, A. 2023. Conformal Prediction with Large Language Models for Multi-Choice Question Answering. *arXiv preprint arXiv:2305.18404*. Updated sections on prompt engineering. Expanded sections 4.1 and 4.2 and appendix. Included additional references. Work published at the ICML 2023 (Neural Conversational AI TEACH) workshop.

Lan, Y.; He, G.; Jiang, J.; Jiang, J.; Zhao, W. X.; and Wen, J.-R. 2022. Complex knowledge base question answering: A survey. *IEEE Transactions on Knowledge and Data Engineering*.

Li, S.; Park, S.; Lee, I.; and Bastani, O. 2024. TRAQ: Trustworthy Retrieval Augmented Question Answering via Conformal Prediction. *arxiv*. Submitted on 7 Jul 2023 (v1), last revised 5 Apr 2024 (this version, v2).

Li, X.; Zhao, R.; Chia, Y. K.; Ding, B.; Bing, L.; Joty, S.; and Poria, S. 2023. Chain of Knowledge: A Framework for Grounding Large Language Models with Structured Knowledge Bases. arXiv:2305.13269.

Liang, K.; Meng, L.; Liu, M.; Liu, Y.; Tu, W.; Wang, S.; Zhou, S.; Liu, X.; and Sun, F. 2022. A Survey of Knowledge Graph Reasoning on Graph Types: Static, Dynamic, and Multimodal. *arXiv preprint arXiv:2212.05767*. Version 7, last revised on 22 Jul 2023.

Liang, K.; Meng, L.; Zhou, S.; Tu, W.; Wang, S.; Liu, Y.; Liu, M.; Zhao, L.; Dong, X.; and Liu, X. 2024. MINES: Message Intercommunication for Inductive Relation Reasoning over Neighbor-Enhanced Subgraphs. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(9).

Lin, S.; Hilton, J.; and Evans, O. 2021. TruthfulQA: Measuring How Models Mimic Human Falsehoods. *arXiv preprint arXiv:2109.07958*.

Lin, Z.; Trivedi, S.; and Sun, J. 2023. Generating with Confidence: Uncertainty Quantification for Black-box Large Language Models. arXiv:2305.19187.

Luo, L.; Li, Y.-F.; Haffari, G.; and Pan, S. 2024. Reasoning on Graphs: Faithful and Interpretable Large Language Model Reasoning. In *International Conference on Learning Representations*.

Mündler, N.; He, J.; Jenko, S.; and Vechev, M. 2024. Self-contradictory Hallucinations of Large Language Models: Evaluation, Detection and Mitigation. *arXiv preprint arXiv:2301.00427*. ICLR 2024 poster.

OpenAI; et al. 2024. GPT-4: A Large-Scale Multimodal Model for Text and Image Generation. arXiv:2303.08774.

Quach, V.; Fisch, A.; Schuster, T.; Yala, A.; Sohn, J. H.; Jaakkola, T. S.; and Barzilay, R. 2023. Conformal Language Modeling. *arXiv preprint arXiv:2306.10193*. V1.

Ren, A. Z.; Dixit, A.; Bodrova, A.; Singh, S.; Tu, S.; Brown, N.; Xu, P.; Takayama, L.; Xia, F.; Varley, J.; Xu, Z.; Sadigh, D.; Zeng, A.; and Majumdar, A. 2023. Robots That Ask For Help: Uncertainty Alignment for Large Language Model Planners. *arXiv preprint arXiv:2307.01928*. Conference on Robot Learning (CoRL) 2023, Oral Presentation.

Rouzrokh, P.; Faghani, S.; Gamble, C. U.; Shariatnia, M.; and Erickson, B. J. 2024. CONFLARE: CONFormal LArge language model REtrieval. *arxiv*. Submitted on 4 Apr 2024.

Shafer, G.; and Vovk, V. 2008. A tutorial on conformal prediction. *Journal of Machine Learning Research*, 9: 371–421.

Shu, Y.; Yu, Z.; Li, Y.; Karlsson, B. F.; Ma, T.; Qu, Y.; and Lin, C.-Y. 2022. TIARA: Multi-grained Retrieval for Robust Question Answering over Large Knowledge Bases. arXiv:2210.12925.

Su, J.; Luo, J.; Wang, H.; and Cheng, L. 2024. API Is Enough: Conformal Prediction for Large Language Models Without Logit-Access. *arXiv preprint arXiv:XXXX.XXXXX*.

Zhejiang University - University of Illinois Urbana-Champaign Institute, Zhejiang University and Department of Computer Science, University of Illinois Chicago.

Sun, J.; Xu, C.; Tang, L.; Wang, S.; Lin, C.; Gong, Y.; Shum, H.-Y.; and Guo, J. 2023. Think-on-Graph: Deep and Responsible Reasoning of Large Language Model with Knowledge Graph. arXiv:2307.07697.

Talmor, A.; and Berant, J. 2018. The web as a knowledge-base for answering complex questions. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 641–651.

Wang, J.; Sun, Q.; Chen, N.; Li, X.; and Gao, M. 2023a. Boosting Language Models Reasoning with Chain-of-Knowledge Prompting. arXiv:2306.06427.

Wang, K.; Duan, F.; Wang, S.; Li, P.; Xian, Y.; Yin, C.; Rong, W.; and Xiong, Z. 2023b. Knowledge-Driven CoT: Exploring Faithful Reasoning in LLMs for Knowledge-intensive Question Answering. arXiv:2308.13259.

Wang, Y.; Lipka, N.; Rossi, R. A.; Siu, A.; Zhang, R.; and Derr, T. 2024. Knowledge Graph Prompting for Multi-Document Question Answering. *AAAI Technical Track on Natural Language Processing II*, 38(17).

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Ichter, B.; Xia, F.; Chi, E. H.; Le, Q. V.; and Zhou, D. 2022. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. In *Advances in Neural Information Processing Systems (NeurIPS)*. Published: 31 Oct 2022, Last Modified: 11 Jan 2023.

Xie, T.; Wu, C. H.; Shi, P.; Zhong, R.; Scholak, T.; Yasunaga, M.; Wu, C.-S.; Zhong, M.; Yin, P.; Wang, S. I.; Zhong, V.; Wang, B.; Li, C.; Boyle, C.; Ni, A.; Yao, Z.; Radev, D.; Xiong, C.; Kong, L.; Zhang, R.; Smith, N. A.; Zettlemoyer, L.; and Yu, T. 2022. UnifiedSKG: Unifying and Multi-Tasking Structured Knowledge Grounding with Text-to-Text Language Models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 602–631. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics.

Yang, L.; Chen, H.; Li, Z.; Ding, X.; and Wu, X. 2023. ChatGPT is not Enough: Enhancing Large Language Models with Knowledge Graphs for Fact-aware Language Modeling. arXiv:2306.11489.

Ye, F.; Yang, M.; Pang, J.; Wang, L.; Wong, D. F.; Yilmaz, E.; Shi, S.; and Tu, Z. 2024. Benchmarking LLMs via Uncertainty Quantification.

Yih, W.-t.; Richardson, M.; Meek, C.; Chang, M.-W.; and Suh, J. 2016. The value of semantic parse labeling for knowledge base question answering. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 201–206.

Yu, D.; Zhang, S.; Ng, P.; Zhu, H.; Li, A. H.; Wang, J.; Hu, Y.; Wang, W.; Wang, Z.; and Xiang, B. 2023. DeCAF: Joint Decoding of Answers and Logical Forms for Question Answering over Knowledge Bases. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics.