TRAQ: Trustworthy Retrieval Augmented Question Answering via Conformal Prediction

Anonymous ACL submission

Abstract

When applied to open-domain question answering, large language models (LLMs) frequently generate incorrect responses based on made up facts, which are called *hallucinations*. Retrieval augmented generation (RAG) is a promising strategy to avoid hallucinations, but it does not provide guarantees on its correctness. To address this challenge, we propose the Trustworthy Retrieval Augmented Question Answering, or *TRAQ*, which provides the first end-to-end statistical correctness guarantee for RAG. TRAQ uses conformal prediction, a statistical technique for constructing prediction sets that are guaranteed to contain the semantically correct response with high probability. Additionally, TRAQ leverages Bayesian optimization to minimize the size of the constructed sets. In an extensive experimental evaluation, we demonstrate that TRAQ provides the desired correctness guarantee while reducing prediction set size by 18.4% on average compared to an ablation.

1 Introduction

Large Language Models (LLMs) have achieved State-Of-The-Art (SOTA) results on many question answering (QA) tasks (OpenAI, 2023; Touvron et al., 2023a,b). However, in open-domain QA tasks where no candidate answers are provided, LLMs have also been shown to confidently generate incorrect responses, called *hallucinations* (Ouyang et al., 2022; Kuhn et al., 2023). Hallucinations have already led to real-world consequences when end users rely on the accuracy of the generated text. As a consequence, there is an urgent need for techniques to reduce hallucinations.

We propose a novel framework, *Trustworthy Retrieval Augmented Question Answering (TRAQ)*, summarized in Figure 1, that combines Retrieval Augmented Generation (RAG) (Guu et al., 2020; Lewis et al., 2021) with conformal prediction (Vovk et al., 2005; Shafer and Vovk, 2007; Park et al., 2020; Angelopoulos and Bates, 2022) to provide theoretical guarantees on question answering performance.

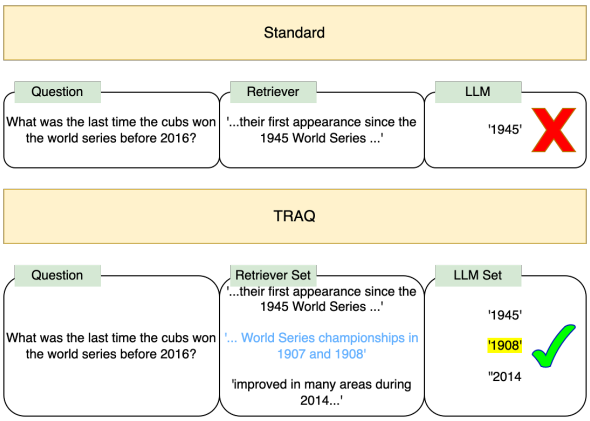


Figure 1: Using the standard retrieval augmented generation (RAG), the retrieved passage may not be relevant to answering the question. In contrast, TRAQ uses conformal prediction to guarantee both that the gold standard passage is in the retrieved set with high probability, and that a semantically correct answer is in the generated set of answers with high probability. By composing these two prediction sets, TRAQ guarantees that a semantically correct answer is in its set of answers with high probability. Furthermore, it uses Bayesian optimization to minimize the size of the prediction set.

RAG reduces hallucinations by retrieving passages from a knowledge base such as Wikipedia, and then using an LLM to answer the question. If the retrieved passages are relevant to the question, the LLM can use this information to generate accurate answers. However, RAG can fail for two reasons: either the retrieved passage is not relevant to the question, or the LLM generates the incorrect answer despite being given a relevant passage.

To avoid these issues, TRAQ uses conformal prediction, an uncertainty quantification technique that modifies the underlying model to predict sets of outputs rather than a single output. These *prediction sets* are guaranteed to contain the true output at

a user-specified rate, e.g., at least 90% of the time. In particular, TRAQ applies conformal prediction separately to the retrieval model (to obtain sets of retrieved passages guaranteed to contain the gold standard passage with high probability), and the generator (to obtain sets of answers that contain the true answer with high probability, assuming the gold standard passage is given). Then, TRAQ outputs all possible answers for all possible retrieved passages. By a union bound, both these prediction sets are valid with high probability, establishing that the prediction set output by TRAQ contains the ground truth answer with high probability.

A major challenge to this basic pipeline is that there may be many different ways of expressing the correct answer in natural language. For example, for the responses *deep learning is a subset of machine learning* and *machine learning is a superset of deep learning* are different ways of expressing the same meaning (Kuhn et al., 2023; Lin and Demner-Fushman, 2007). This diversity of possible responses also makes prediction probabilities less reliable, since if an answer can be expressed in many different but equivalent ways, then the probabilities may be divided across these different responses, making them all smaller even if the model is confident it knows the correct answer.

TRAQ addresses this challenge by modifying the notion of coverage to focus on semantic notions of uncertainty. In particular, TRAQ aggregates semantically equivalent answers across a large number of samples from the LLM, and uses the number of clusters of non-equivalent answers as a measure of uncertainty. This measure is used as a nonconformity measure to construct prediction sets. Finally, the prediction sets are over clusters of equivalent answers rather than individual answers. This strategy also enables TRAQ to work for blackbox APIs such as *GPT-3.5-Turbo*, where predicted probabilities for individual tokens are unavailable.

A second challenge is that the prediction sets can become very large since we are aggregating uncertainty across multiple components. This complexity introduces hyperparameters into TRAQ; while TRAQ guarantees correctness regardless of the choice of these hyperparameters, they can affect the performance of TRAQ in terms of the average prediction set size. To address this challenge, TRAQ uses Bayesian optimization to minimize the set of the overall prediction sets it generates.

We evaluate TRAQ in conjunction with several generative LLMs, including both GPT-3.5-Turbo-0613 and Llama-2-7B, on four datasets, including a biomedical question answering dataset. Our experiments demonstrate that TRAQ empirically satisfies the coverage guarantee (i.e., the prediction set it outputs contains a semantically correct answer with the desired probability), while reducing the average prediction set size compared to an ablation by 18.4%. Thus, TRAQ is an effective strategy for avoiding hallucinations in applications of LLMs to open domain question answering.

Contributions. We offer the first conformal prediction guarantees for retrieval augmented generation (RAG) targeted question answering. Our framework, TRAQ, introduces a novel nonconformity measure that measures uncertainty for each semantically distinct meaning and obtains a coverage guarantee at the semantic level. Furthermore, TRAQ leverages Bayesian optimization to minimize the average size of the generated prediction sets. Finally, our experiments demonstrate that TRAQ is effective at avoiding hallucinations in open domain question answering.

2 Related Work

Retrieval for Open-Domain QA. A two-stage approach is often used for open-domain question answering (QA): first, a *retriever* is used to obtain informative passages; and second, a *generator* produces answers based on the retrieved passages. A popular choice for retrieval is the Dense Passage Retriever (DPR) (Karpukhin et al., 2020b), which measures similarity by taking the inner product of the BERT (Devlin et al., 2019) embeddings of the question and the passage. Other works (Lin and Lin, 2022; Salemi et al., 2023; Lin et al., 2022; Zhang et al., 2021) have improved the performance of DPR and extended it to more diverse settings. Retrieval Augmented Generation (RAG) (Lewis et al., 2021) proposes to jointly finetune the retriever and the generator for QA tasks.

Conformal Prediction. Conformal prediction (Vovk et al., 2005; Papadopoulos, 2008) is a general distribution-free approach to quantifying uncertainty for machine learning (ML) models. It is based on a *nonconformity measure* (e.g., probabilities predicted by an ML model) $s : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ and a held-out calibration set $B = \{(x_i, y_i)\}_{i=1}^{N}$ sampled i.i.d. from the data distribution $\mathcal{D}$, as well as a user-specified error level α. The prediction set

for a testing data point X_{test} is then constructed as

$$C(X_{\text{test}}) = \{y \in \mathcal{Y} \mid s(X_{\text{test}}, y) \leq \tau\}, \quad (1)$$

where τ is the $\frac{\lceil (1-\alpha)(N+1) \rceil}{N}$-th smallest score in $\{s(x_i, y_i)\}_{i=1}^{N}$. Conformal prediction guarantees that the true labels are contained in the constructed prediction sets with probability at least $1 - \alpha$:

Theorem 1. *Conformal Prediction Guarantee (Angelopoulos and Bates, 2022, Theorem 1). Suppose $\{(x_i, y_i\}_{i=1}^{N}$ and (X_{test}, Y_{test}) are i.i.d. from $\mathcal{D}$, and $C(X_{test})$ is constructed by (1); then, we have*

$$\Pr_{(X_{test}, Y_{test}) \sim \mathcal{D}}(Y_{test} \in C(X_{test})) \geq 1 - \alpha. \quad (2)$$

We call this guarantee a *coverage guarantee*. An extension of conformal prediction is *Probably Approximately Correct prediction sets* (Park et al., 2019) (PAC prediction set) or *training-conditional conformal prediction* (Vovk, 2012). Compared with vanilla conformal prediction, where the coverage guarantee holds on average, PAC prediction sets guarantee that coverage is satisfied with high probability given the current calibration set:

Theorem 2. *PAC Guarantee (Park et al., 2019, Theorem 1). Suppose $\{(x_i, y_i\}_{i=1}^{N}$ and (X_{test}, Y_{test}) are sampled i.i.d. from $\mathcal{D}$, and $C(X_{test})$ is constructed via (4) in the Appendix; then, we have*

$$\Pr_{B \sim \mathcal{D}^n}[\Pr_{(X,Y) \sim \mathcal{D}}(Y_{test} \in C(X_{test})) \geq 1 - \alpha] \geq 1 - \delta.$$

Further details on conformal prediction and PAC prediction sets are in Appendices A.1 & A.2, respectively; a brief comparison between the two is given in Appendix A.3. Both vanilla conformal prediction and PAC prediction sets have been applied to deep learning (Park et al., 2019; Angelopoulos et al., 2020; Bates et al., 2021).

Uncertainty Quantification for LLMs. Uncertainty quantification for Large Language Models (LLMs) has been gaining attention due to their hallucinations. A recent study (Kuhn et al., 2023) combined confidence calibration with natural language inference to measure the LLMs' certainty in responding to an input question. However, this work does not guarantee the accuracy of the responses. Other studies have applied conformal prediction to LLM predictions, mainly focusing on the multiple choice question answering problem and using vanilla conformal prediction to ensure correctness (Kumar et al., 2023; Ren et al., 2023). However, these methods necessitate a finite set of labels, such as {*True*, *False*} or {*A*, *B*, *C*}, and cannot be used for open-domain question answering. A related work concurrent to ours is Quach et al. (2023), which applies conformal prediction to open-domain QA. However, they only consider the generator, whereas our approach provides conformal guarantees for RAG. Furthermore, their approach requires the generation probability from the LLM, which is not available in many blackbox APIs.

3 The TRAQ Framework

TRAQ is composed of two steps. The first is the *Prediction Set Construction* step, where a question q is used to create a *retrieval set* $C_{\text{Ret}}(q)$ for the retriever, and a *LLM set* $C_{\text{LLM}}(q, p)$ for each pair (question q, passage p). These sets are aggregated into an *Aggregation Set* $C_{\text{Agg}}(q)$. The second step is the *Performance Improvement* step, where promising error budgets α_{Ret} and α_{LLM} are sampled from a Bayesian model. Using these budgets, prediction sets are constructed on the optimization set and evaluated for their performance. This process is repeated N times, and the final output is the error budgets α_{Ret} and α_{LLM} with the highest performance. The chosen hyperparameters are used to construct prediction sets as in the first step using a separate held-out calibration set. The overall TRAQ framework is summarized in Figure 2.

3.1 Assumptions

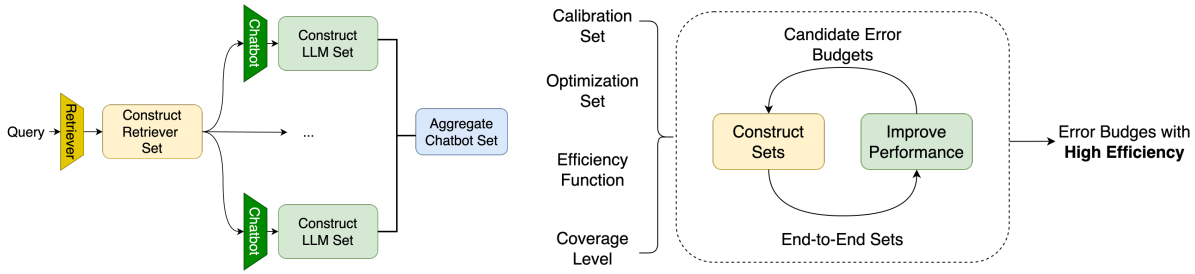
To construct provable prediction sets, we first make three necessary assumptions:

Assumption I.I.D. *For both the retrieval and LLM tasks, the examples are drawn independently and identically from the data distribution $\mathcal{D}$.*

Assumption *Retriever Correctness*. *Given a question q, the underlying retriever is able to retrieve the most relevant passage p^* within the top-K retrieved passages.*

Assumption *LLM Correctness*. *Given a question q and its most relevant passage p^*, the LLM is able to generate a semantically correct response within the top-M samples.*

Assumption I.I.D is a standard assumption from the conformal prediction literature, and is needed to apply conformal prediction algorithms (it can be slightly relaxed to exchangeable distributions, but we make the i.i.d. assumption for simplicity).



(a) Prediction Set Construction

(b) Performance Improvement

Figure 2: Given a question, TRAQ first constructs the retriever prediction; then, for every *(question, contained passage)* pair, TRAQ constructs a LLM prediction on the LLM generated responses. Finally, LLM prediction sets are aggregated as the final output. In Figure 2b, TRAQ takes in candidate error budgets from the Bayesian optimization; it then constructs aggregated prediction sets on the optimization set. Next, the average semantic counts in constructed sets are computed to update the Gaussian process model in Bayesian optimization.

Assumptions *Retriever Correctness* and *LLM Correctness* are needed to ensure that most relevant passages and semantically correct answers can be contained in the prediction sets if the prediction sets are sufficiently large. In principle, we can use very large values of K and M to satisfy this assumption, though there are computational and cost limitations in practice. We discuss ways to remove these assumptions in Section 7.

3.2 Prediction Set Construction

Retriever Set: To construct the retriever sets C_{Ret}, we use the negative inner product between the question q and the annotated most relevant passage p^*, denoted as R_{q,p^*}, as the nonconformity measures (NCMs). Given N such NCMs $\{s_1, \ldots, s_N\}$ in the calibration set and the error budget α_{Ret} for the retriever set , we construct the retriever set by

$$C_{\text{Ret}}(q) = \{p \mid -R_{q,p} \leq \tau_{\text{Ret}}\}, \tag{3}$$

where

$$\tau_{\text{Ret}} = \text{Quantile}\left(\{s_k\}_{k=1}^{K}; \frac{\lceil (K+1)(1-\alpha_{\text{Ret}}) \rceil}{K}\right).$$

Given this construction and Assumptions I.I.D and Assumption *Retriever Correctness*, the retriever sets are guaranteed to contain the most relevant passage with probability at least $1 - \alpha_{\text{Ret}}$:

Lemma 2.1. *Suppose the questions q and their corresponding most relevant passage p^* are sampled from distribution $\mathcal{D}_{passage}$. Given the error budget α_{Ret}, the retriever prediction sets satisfy*

$$\Pr_{(q,p^*)\sim\mathcal{D}_{Passage}}(p^* \in C_{Ret}(q)) \geq 1 - \alpha_{Ret}.$$

This result follows straightforwardly from Theorem 1 and Assumptions I.I.D & *Retriever Correctness*. We give a proof in Appendix B.

LLM Set: We utilize Monte Carlo sampling to approximate confidences for different semantic meanings; then, we use the approximated confidences as the NCMs to construct LLM sets. Specifically, for each *(question, passage)* pair, we ask the LLM to generate M responses ($M = 30$ in our experiments). Given these responses, we cluster them by their semantic meanings using Rouge-score (Lin, 2004) or using BERT embeddings (Kuhn et al., 2023). After clustering, for each cluster i, let N_i be the number of responses in the cluster; we approximate the confidence of i-th cluster by N_i/M. We then put the confidences of semantically correct answers into the calibration set $\{s_1, \ldots, s_N\}$. Finally, given the error budget for the LLM α_{LLM}, we can utilize a similar process as in (3) to construct LLM sets. The constructed sets satisfy the following:

Lemma 2.2. *Suppose the questions q, their corresponding most relevant passage p^*, and semantically correct responses r^* are sampled from distribution $\mathcal{D}_{Response}$. Given error budget α_{LLM}, if Assumptions I.I.D & LLM Correctness hold, the LLM sets satisfy*

$$\Pr_{(q,p^*,r^*)\sim\mathcal{D}_{Response}}(r^* \in C_{LLM}(q, p^*)) \geq 1 - \alpha_{LLM}.$$

The proof of Lemma 2.2 is similar to that of Lemma 2.1; we give it in Appendix B.

Aggregated Set: To obtain an overall correctness guarantee, we construct LLM sets for each passage

contained in the retriever set. Then, we aggregate these individual LLM sets by removing duplicated and re-clustering semantic meanings. The resulting Aggregated set C_{Agg} satisfies the following:

Theorem 3. *Suppose the questions q and semantically correct responses r^* are sampled from distribution $\mathcal{D}$, and a user-specified error level α is given. By aggregating retriever sets with error budget α_{Ret} with LLM sets with error budget α_{LLM}, with $\alpha = \alpha_{Ret} + \alpha_{LLM}$, the aggregated sets satisfy*

$$\Pr_{(q,r^*)\sim\mathcal{D}}(r^* \in C_{Agg}(q)) \geq 1 - \alpha.$$

We give a proof in Appendix B. Note that this aggregation process is actually a global hypothesis testing method called the Bonferroni Correction. Lemmas 2.1 & 2.2 and Theorem 3 can be straightforwardly extended to the probably approximately correct (PAC) guarantee by constructing PAC prediction sets; see Appendix B.1 for details.

3.3 Performance Improvement

By Theorem 3, we can guarantee that semantically correct responses are included in the aggregated set with a probability of at least $1 - \alpha$, assuming $\alpha = \alpha_{\text{Ret}} + \alpha_{\text{LLM}}$. This theorem is valid for any combination of the two error budgets. However, the predictive performance of the aggregation sets is influenced by the specific choice of the error budgets. This issue has been discussed in the Bonferroni correction and the global testing literatures (Neuwald and Green, 1994; Wilson, 2019; Poole et al., 2015).

Therefore, we optimize the error budgets using Bayesian optimization, a sampling-based global optimization technique suitable for nonconvex, non-closed form problems; see Appendix A.4 for details. In TRAQ, Bayesian optimization first models the underlying performance landscape using a Gaussian process; then, it samples potential error budgets (i.e., α_{Ret} and α_{LLM}) based on the Gaussian process. After assessing the performance of the sampled error budgets on a held-out optimization set, the Gaussian process is modified to more accurately reflect the performance landscape. This process is repeated for N times. Pseudo-code for this procedure is shown in Algorithm 1.

Algorithm 1 Prediction Set Optimization

input: Calibration set B_{Cal}, optimization set B_{Opt}, performance metric f, error level α
Initialize Gaussian process G
for $t \in \{1, ..., T\}$ **do**
  Sample α_{Ret} and α_{LLM} basing on G
  Compute τ_{Ret} and τ_{LLM} using sampled budgets and calibration set B_{Cal}
  Construct aggregation prediction set C_{Agg} on the optimization set B_{Opt}
  Evaluate the performance of the sets using f
  Update G using the evaluation results
end for
return: the best error budgets α_{Ret} and α_{LLM}

4 Experiments

Experiment Setup. We evaluated TRAQ on four datasets, including three standard QA datasets (Natural Question (Kwiatkowski et al., 2019), TriviaQA (Joshi et al., 2017), SQuAD-1 (Rajpurkar et al., 2016)), and a biomedical QA dataset (BioASQ (Tsatsaronis et al., 2012)). We gathered 1,000 samples that met the criteria of Assumptions *Retriever Correctness* & *LLM Correctness*. We divided each dataset into calibration, optimization, and testing sets, with 300, 300, and 400 data points, respectively.

We employed two fine-tuned DPR models, one (Karpukhin et al., 2020a) trained on Natural Question, TriviaQA, and SQuAD-1 datasets, and the other fine-tuned on BioASQ (see Appendix D.2 for training details). Additionally, we used two generative large language models (LLMs): *GPT-3.5-Turbo-0613* (*GPT-3.5*), whose internal embedding and prediction probabilities are not accessible, and *Llama-2-7B* (*Llama-2*). We separately fine-tuned Llama-2 on Natural Question, TriviaQA, and SQuAD-1, with hyperparameters given in Appendix D.1.

For each question, we retrieved the top-20 passages; for each *(question, passage)* pair, we sampled 30 responses, with a temperature of 1.0.

We evaluated using coverage levels 50%, 60%, 70%, 80%, and 90%. For the PAC guarantee, we use confidence level 90%. We used five random seeds for each experiment. To investigate the influence of prompt design, we designed two prompts, a zero-shot and a few-shot prompt; the few-shot prompt included two demonstrations. The prompt templates are provided in Appendix D.3. Unless otherwise specified, the zero-shot prompt was used for both GPT-3.5 and Llama-2.

We evaluated the performance of our approach using two metrics. The first metric is the *coverage rate*, which is the rate at which correct responses

are contained in the constructed sets. We consider responses to be *correct* if their *Rouge-1* (Lin, 2004) scores with the annotated answers were higher than 0.3. The coverage rate is expected to be no less than the desired level on average across different random seeds. The second metric is the prediction set size. Specifically, we consider two size measures: (i) the average number of semantic clusters, and (ii) the average number of unique answers. Lower values indicate better performance. In general in conformal prediction, the goal is to obtain the smallest prediction set size subject to satisfying the desired coverage rate.

We compared our approaches, *TRAQ* and *TRAQ-P* (the PAC version), to several baselines, including *Vanilla*, *Bonf*, and *Bonf-P*. Vanilla is a baseline that does not construct prediction sets and only uses the top retrieved passage and generated answers. Bonf and Bonf-P are ablations that omit Bayesian optimization. In all plots, we also show the *Reference* line indicating the desired coverage level.

Our experiments aim to answer the following:
(Q_1) *Do the coverage guarantees hold for the retriever and the generator?*
(Q_2) *Does the overall coverage guarantee hold?*
(Q_3) *Does Bayesian optimization help?*
(Q_4) *Does TRAQ work for different semantic clustering and performance metrics?*
(Q_5) *How does the prompt affect the results?*

Q1: Do the coverage guarantees hold for the retriever and generator? To validate the retriever and generator coverage guarantees, we consider the coverage rates of retriever and LLM sets (named *Ret* and *LLM*), and with the PAC guarantee (named *Ret-P* and *LLM-P*). We report results on BioASQ using GPT-3.5 in Figure 3; results for other datasets and different LLMs are reported in Figure 10, and are qualitatively similar. As shown in Figure 3, the empirical coverage levels of the retrieval and QA prediction sets are close to the desired coverage levels. Thus, the coverage guarantees hold for individual components, as desired.

We also report empirical coverage rates with 20 random seeds in Figure 11. Compared to results with 5 random seeds, empirical coverages with more random seeds become closer to the desired level. Furthermore, when using the PAC prediction sets, the empirical coverage levels were almost always above the expected coverage levels across all random seeds, as desired.

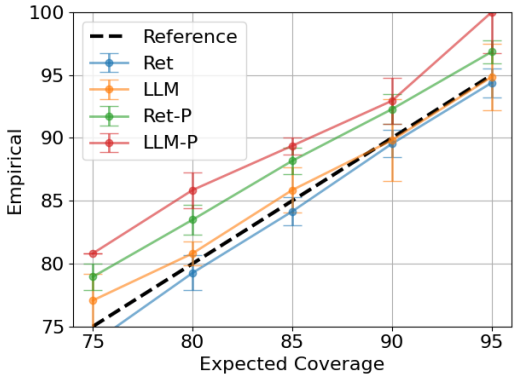


Figure 3: Retriever and generator coverage rates on the BioASQ dataset.

Q2: Does the end-to-end coverage guarantee hold? To verify the end-to-end guarantees from TRAQ, we report two rates. The first is the rate that the correct responses are covered considering only the annotated most relevant passages:

$$\Pr(p^* \in C_{\text{Ret}}(q)) \times \Pr(r^* \in C_{\text{LLM}}(q, p^*)).$$

These results are shown in Figure 4. They show that the rates on average satisfy the desired coverage levels when using conformal prediction. Also, the rates are mostly above the desired coverage levels when using PAC prediction sets. Second, we report the rate that the correct responses are covered in the aggregated prediction set:

$$\Pr(r^* \in C_{\text{Agg}}(q)).$$

The results are shown in Figure 5. Different from Figure 4, empirical levels of both conformal prediction and PAC prediction sets were above the expected coverage levels most of the time. This is because the generator can output the correct response even if it is not given a relevant passage.

Q3: How does the Bayesian optimization help? To demonstrate the advantages of incorporating Bayesian optimization, we evaluate the average prediction set size (in terms of number of semantic clusters) across different approaches. We show results for different coverage levels and random seeds on BioASQ dataset in Figure 6. We reported the semantic counts on other datasets in Table 1 (for GPT-3.5) and Table 2 (for Llama-2). Both TRAQ and TRAQ-P, which use Bayesian optimization, are able to construct significantly smaller prediction sets, reducing them on average by 18.4% (15.3% in Table 1 and 21.5% in Table 2). These results

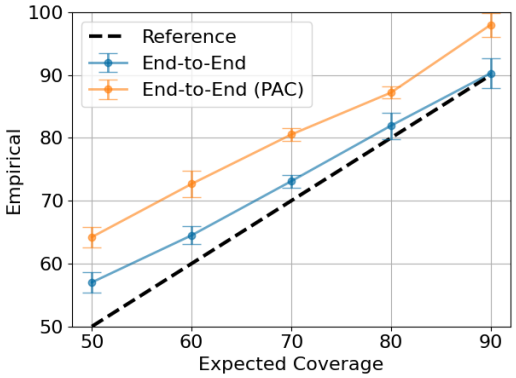


Figure 4: End-to-end Guarantee Considering Only the Most Relevant Passage on BioASQ Dataset

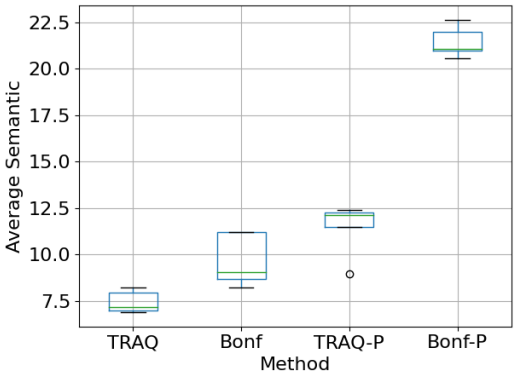


Figure 6: Prediction set sizes according to the average number of semantic clusters

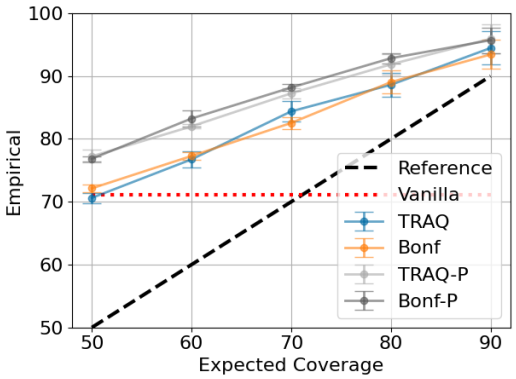


Figure 5: Overall coverage guarantee considering all passages on the BioASQ dataset

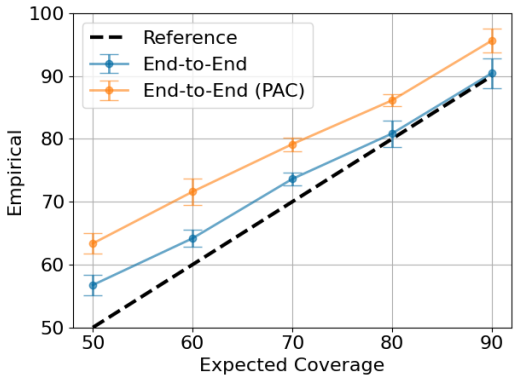


Figure 7: Coverage rate using BERT embeddings on SQuAD-1 dataset

demonstrate that Bayesian optimization can effectively improve performance. Importantly, even though the prediction sets are smaller, the desired overall coverage guarantees still hold.

Q4: Does TRAQ work for different semantic clustering and performance metrics? We evaluated whether TRAQ remains effective with a different semantic clustering method and performance metrics. We use the semantic clustering method proposed by Kuhn et al. (2023), which is based on BERT (Devlin et al., 2019), and specified the performance metric as the average number of unique answers in the aggregated prediction sets. We evaluated this setup on the SQuAD-1 dataset using GPT-3.5. The results, shown in Figure 7 & 8, demonstrate that TRAQ remains successful. Furthermore, Figure 7 demonstrates that the overall coverage guarantee still holds, and Figure 8 demonstrates that TRAQ and TRAQ-P still reduce prediction set size compared to their ablations Bonf and Bonf-P, respectively.

Q5: How does the prompt affect the results? We investigated how prompt engineering affects the performance of TRAQ by using a few-shot prompt with two demonstrations. The prompt template is provided in Appendix D.3. We evaluated TRAQ on SQuAD-1 using GPT-3.5. The end-to-end coverage rates and prediction set sizes using different methods are shown in Figure 16. TRAQ with a few shot prompt achieved the desired coverage rate on average, and reduced prediction set size compared to its ablation. In Figure 9, we also compared the zero-shot and few-shot prompts in terms of performance. Interestingly, zero-shot prompting consistently yielded better efficiencies. This could be because zero-shot prompting generated more diverse answers and had lower confidence in wrong answers. An example of the comparison between responses using different prompts is given in the Appendix D.3.

Qualitative Analysis. By incorporating multiple passages, TRAQ guarantees that it considers a rel-

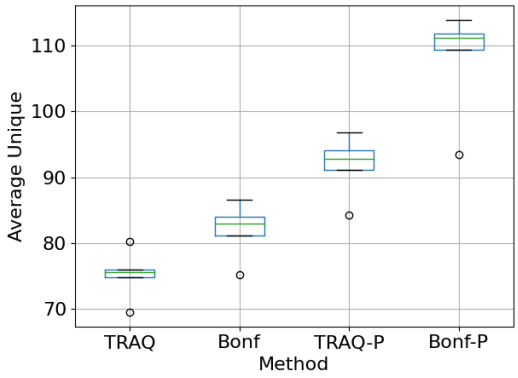


Figure 8: Prediction set size according to the average number of unique responses

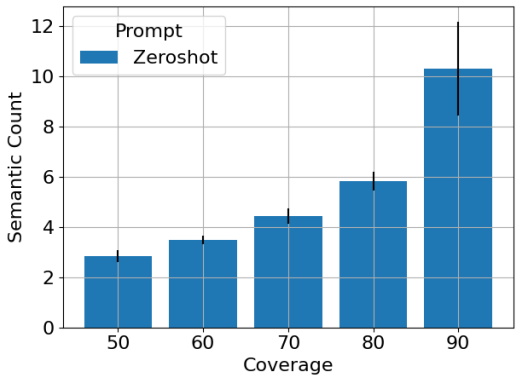


Figure 9: Comparison between zero-shot and few-shot prompts on prediction set size.

evant passage with high probability. For example, we consider the following question: *Who played in the movie a star is born with Judy Garland?*, where *James Mason* is a correct answer. For this example, we obtained the following outputs:

```
Question: who played in the movie a star is born
     with judy garland

Gold Answer: {'James Mason', 'Charles Bickford',
     'Jack Carson'}

vanilla: {'Gary Busey', 'Judy Garland', 'Barbra
    Streisand'}

TRAQ: {'Judy Garland', 'James Mason', 'Lady Gaga
    ', 'Sid Luft', 'Danny Kaye'}

Bonf {'Gary Busey', 'Judy Garland', 'James Mason
    ', 'Lady Gaga', 'Bradley Cooper', 'Sidney
    Luft', 'Danny Kaye'}
```

We show additional examples in Appendix C.6.

5 Conclusion

We proposed an algorithm, called *Trustworthy Retrieval Augmented Question Answering (TRAQ)*, that applies conformal prediction to construct prediction sets for Retrieval Augmented Generation (RAG). TRAQ first constructs prediction sets for the retriever and generator, and then aggregates these sets. TRAQ guarantees that for each question, a semantically correct answer is included in the prediction set it outputs with high probability. To the best of our knowledge, this guarantee is the first conformal guarantee for retrieval augmented generation. Additionally, to minimize prediction set size, TRAQ leverages Bayesian optimization to identify optimal hyperparameters. In our comprehensive experiments, we demonstrate that TRAQ provides an overall semantic level coverage guarantee across different tasks, and that Bayesian optimization consistently reduces prediction set size.

6 Broader Impacts

The need for trustworthy AI algorithms has recently become paramount due to the risks for spreading misleading information (Biden, 2023; Commission, 2023). We propose TRAQ, a framework that aims to address the hallucination problem by using conformal prediction to provide probabilistic guarantees for retrieval augmented generation (RAG). In addition, TRAQ leverages novel techniques for improving performance that may be useful more broadly in conformal prediction.

7 Limitations

TRAQ makes three assumptions: that the data is independent and identically distributed (I.I.D), that the retriever has good performance (*Retriever Correctness*), and that the language model can generate a response to the input question (*LLM Correctness*). Our experiments have verified I.I.D, but *Retriever Correctness* and *LLM Correctness* may not hold if the underlying retriever and language model do not have good performance. To relax *Retriever Correctness*, we can select more passages than the top-**20** used in our experiments. To remove *LLM Correctness*, we propose providing a guarantee of including *I do not know* in the aggregation set if the language model cannot answer the input question. We describe how TRAQ can be modified to provide such guarantees in Appendix E.

TRAQ is a post-hoc method, so its prediction sets may be larger than necessary if the underlying

models, such as the retriever and large language model, do not work properly. Additionally, if the semantic clustering techniques (Rouge-score based or BERT-based) are invalid, then some semantically unrelated answers may be aggregated.

References

Anastasios Angelopoulos, Stephen Bates, Jitendra Malik, and Michael I. Jordan. 2020. Uncertainty sets for image classifiers using conformal prediction.

Anastasios N. Angelopoulos and Stephen Bates. 2022. A gentle introduction to conformal prediction and distribution-free uncertainty quantification.

Stephen Bates, Anastasios Angelopoulos, Lihua Lei, Jitendra Malik, and Michael I. Jordan. 2021. Distribution-free, risk-controlling prediction sets.

Joseph R. Jr. Biden. 2023. Presidential actions archives | the white house. https://www.whitehouse.gov/briefing-room/presidential-actions/2023/10/30/. (Accessed on 12/12/2023).

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.".

European Commission. 2023. Commission welcomes political agreement on ai act. https://ec.europa.eu/commission/presscorner/detail/%20en/ip_23_6473. (Accessed on 12/12/2023).

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Peter I. Frazier. 2018. A tutorial on bayesian optimization.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020. Retrieval augmented language model pre-training. In *International conference on machine learning*, pages 3929–3938. PMLR.

Haystack. Quick start | haystack. https://haystack.deepset.ai/overview/quick-start. (Accessed on 11/27/2023).

Tim Head, Manoj Kumar, Holger Nahrstaedt, Gilles Louppe, and Iaroslav Shcherbatyi. 2021. scikit-optimize/scikit-optimize.

Matthew Honnibal and Ines Montani. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada. Association for Computational Linguistics.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020a. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen tau Yih. 2020b. Dense passage retrieval for open-domain question answering.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation.

Bhawesh Kumar, Charlie Lu, Gauri Gupta, Anil Palepu, David Bellamy, Ramesh Raskar, and Andrew Beam. 2023. Conformal prediction with large language models for multi-choice question answering.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

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. 2021. Retrieval-augmented generation for knowledge-intensive nlp tasks.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Jimmy Lin and Dina Demner-Fushman. 2007. Semantic clustering of answers to clinical questions. *AMIA ... Annual Symposium proceedings / AMIA Symposium. AMIA Symposium*, 2007:458–62.

Sheng-Chieh Lin, Minghan Li, Jimmy Lin, and David R. Cheriton. 2022. Aggretriever: A simple approach to aggregate textual representation for robust dense passage retrieval.

Sheng-Chieh Lin and Jimmy Lin. 2022. A dense representation framework for lexical and semantic matching. *ACM Transactions on Information Systems*, 41:1 – 29.

Meta. 2023. Llama access request form - meta ai. https://ai.meta.com/resources/models-and-libraries/llama-downloads/. (Accessed on 12/13/2023).

Andrew F. Neuwald and Philip Green. 1994. Detecting patterns in protein sequences. *Journal of Molecular Biology*, 239(5):698–712.

OpenAI. 2023. Gpt-4 technical report.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback.

Harris Papadopoulos. 2008. *Inductive conformal prediction: Theory and application to neural networks*. INTECH Open Access Publisher Rijeka.

Sangdon Park, Osbert Bastani, Nikolai Matni, and Insup Lee. 2019. Pac confidence sets for deep neural networks via calibrated prediction. *arXiv preprint arXiv:2001.00106*.

Sangdon Park, Osbert Bastani, Nikolai Matni, and Insup Lee. 2020. Pac confidence sets for deep neural networks via calibrated prediction.

Sangdon Park, Edgar Dobriban, Insup Lee, and Osbert Bastani. 2021. Pac prediction sets under covariate shift. *arXiv preprint arXiv:2106.09848*.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

William Poole, David L. Gibbs, Ilya Shmulevich, Brady Bernard, and Theo Knijnenburg. 2015. Combining dependent p-values with an empirical adaptation of brown's method. *bioRxiv*.

Victor Quach, Adam Fisch, Tal Schuster, Adam Yala, Jae Ho Sohn, Tommi S. Jaakkola, and Regina Barzilay. 2023. Conformal language modeling.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text.

Allen Z. Ren, Anushri Dixit, Alexandra Bodrova, Sumeet Singh, Stephen Tu, Noah Brown, Peng Xu, Leila Takayama, Fei Xia, Jake Varley, Zhenjia Xu, Dorsa Sadigh, Andy Zeng, and Anirudha Majumdar. 2023. Robots that ask for help: Uncertainty alignment for large language model planners.

Alireza Salemi, Juan Altmayer Pizzorno, and Hamed Zamani. 2023. A symmetric dual encoding dense retrieval framework for knowledge-intensive visual question answering. *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*.

Glenn Shafer and Vladimir Vovk. 2007. A tutorial on conformal prediction.

Glenn Shafer and Vladimir Vovk. 2008. A tutorial on conformal prediction. *Journal of Machine Learning Research*, 9(12):371–421.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023a. Llama: Open and efficient foundation language models.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023b. Llama 2: Open foundation and fine-tuned chat models.

George Tsatsaronis, Michael Schroeder, Georgios Paliouras, Yannis Almirantis, Ion Androutsopoulos, Eric Gaussier, Patrick Gallinari, Thierry Artieres, Michael R. Alvers, Matthias Zschunke, and Axel-Cyrille Ngonga Ngomo. 2012. BioASQ: A challenge on large-scale biomedical semantic indexing and Question Answering. In *Proceedings of AAAI Information Retrieval and Knowledge Discovery in Biomedical Text*.

Vladimir Vovk. 2012. Conditional validity of inductive conformal predictors.

Vladimir Vovk, Alex Gammerman, and Glenn Shafer. 2005. *Algorithmic learning in a random world*. Springer Science & Business Media.

Daniel Wilson. 2019. The harmonic mean p -value for combining dependent tests. *Proceedings of the National Academy of Sciences*, 116:201814092.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Huggingface's transformers: State-of-the-art natural language processing.

Hang Zhang, Yeyun Gong, Yelong Shen, Jiancheng Lv, Nan Duan, and Weizhu Chen. 2021. Adversarial retriever-ranker for dense text retrieval. *ArXiv*, abs/2110.03611.

A Conformal Prediction and PAC Guarantees

A.1 Conformal Prediction and Hypothesis Testing

Conformal prediction is a distribution-free uncertainty quantification technique that constructs provable prediction sets for black-box models. Specifically, let $\mathcal{X}$ and $\mathcal{Y}$ be the input and label spaces, respectively, and (x, y) be an input-label pair. Conformal prediction assumes given a calibration set $B = \{x_i, y_i\}_{i=1}^N$ with N input-label pairs, along with a *nonconformity measure* $s : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ that measures how different a pair (x, y) is from the examples sampled from the distribution $\mathcal{D}$. Given a new input x_{test}, conformal prediction constructs a prediction set $C(x_{\text{test}}) \subseteq \mathcal{Y}$ using Algorithm (Angelopoulos and Bates, 2022). Intuitively, for every label $y \in \mathcal{Y}$, this algorithm checks whether (x_{test}, y) is similar to examples in the B according to the nonconformity measure $s(x_{\text{test}}, y)$. If $s(x, y)$ is lower enough, then y is included in the prediction set $C(x_{N+1})$; otherwise, y is excluded from $C(x_{N+1})$. To connect these ideas with multiple hypothesis testing, we note that conformal prediction can be framed as an application of the Neyman-Pearson theory for hypothesis testing (Shafer and Vovk, 2008).

Algorithm 2 The Conformal Algorithm

Input: Nonconformity measure s, significance level α, calibration st $B = \{x_n, y_n\}_{n=1}^N$, a new input x_{test}, label space $\mathcal{Y}$
Compute the threshold τ as the $\frac{\lceil (1-\alpha)(N+1) \rceil}{N}$-th smallest score in $\{s(x_i, y_i)\}_{i=1}^N$.
Construct prediction set for x_{test} by

$$C(x_{\text{test}}) = \{y \mid s(x_{\text{test}}, y), y \in \mathcal{Y}\}$$

Return: $C(x_{\text{test}})$.

A.2 PAC Prediction Set

PAC prediction sets (Vovk, 2012; Park et al., 2021) are a variant of conformal prediction approache that satisfy stronger PAC-style guarantees. Let $\mathcal{D}$ be the distribution of samples, $B = \{x_n, y_n\}_{n=1}^N$ be a held-out calibration set of i.i.d. data points from $\mathcal{D}$ of size N. We denote the joint distribution over N samples as $\mathcal{D}^N$. The goal is to find a set of a small size satisfying the PAC property, i.e., given $\alpha, \delta \in (0, 1)$,

$$\Pr_{Z \sim \mathcal{D}^n}[L_{\mathcal{D}}(C) \leq \alpha] \geq 1 - \delta,$$

where the $\Pr_{Z \sim \mathcal{D}^n}$ refers to the chances of calibration succeeding. In this case, we say C is (α, δ)-*probably approximately correct (PAC)*. To construct (α, δ)-PAC sets, the PAC prediction set considers the following one-dimensional parameterization of the prediction sets:

$$C_\tau(x) = \{y \in \mathcal{Y} \mid g(x, y) \geq \tau\},$$

where $\tau \geq 0$ and $g : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}_{\geq 0}$ is any given scoring function (e.g., the label probabilities output by a deep neural network). The threshold τ is computed by solving the following optimization problem:

$$\hat{\tau} = \underset{\tau \geq 0}{\arg\max}\ \tau \ \text{ subj. to } \sum_{(x,y) \in Z} \mathbb{I}[y \notin C_\tau(x)] \leq k^*, \tag{4}$$

where

$$k^* = \underset{k \in \mathbb{N} \cup \{0\}}{\arg\max}\ k \qquad \text{subj. to} \qquad F(k; N, \alpha) \leq \delta,$$

where $F(k; N, \alpha)$ is the cumulative distribution function of the binomial random variable Binomial(N, α) with N trials and success probability α. Maximizing τ corresponds to minimizing the prediction set size. We have the following theorem:

Theorem 4 ((Vovk, 2012; Park et al., 2021)). *$C_{\hat{\tau}}$ is (α, δ)-correct for $\hat{\tau}$ as in (4).*

A.3 Conformal Prediction and PAC Prediction Set Comparison

Conformal Prediction Guarantee Formally, we can write conformal prediction guarantee as

$$Pr_{(X,Y)\sim\mathcal{D}}(Y \in C(X)) \geq 1-\epsilon.$$

In other words, prediction sets constructed by conformal prediction guarantee that over the whole distribution $\mathcal{D}$, the probability that the true label is contained in the set is at least $1-\epsilon$. Note that this coverage probability is marginalized over all possible calibration sets. On the other hand, for a specific calibration set B, this guarantee might not hold. For instance, the guarantee will not hold if samples in B are concentrated in a small region of the joint distribution and therefore are not representative of the joint distribution $\mathcal{D}$.

PAC Prediction Set Guarantee Formally, we can write PAC prediction set guarantee as

$$\Pr_{B\sim\mathcal{D}^N}(Pr_{(X,Y)\sim\mathcal{D}} \geq 1-\epsilon) \geq 1-\delta.$$

Compared with the conformal prediction guarantee, the difference is the outer probability, which is on the given calibration set B. Intuitively, the guarantee of the PAC prediction set says that conditioning on the given calibration set B, we can say with high confidence (at least $1-\delta$) that the true label is contained in the constructed set $C(X)$ with high probability $(1-\epsilon)$. As a result, PAC prediction set guarantee is stronger than conformal prediction guarantee as PAC prediction set guarantee is over an individual calibration set, while the conformal prediction guarantee is marginalized over all possible calibration sets.

A.4 Bayesian Optimization

Bayesian optimization (BO) is a technique for finding the global optimum of a potentially non-convex, non-linear, or non-closed form objective function f with decision variables $\{b^1, \ldots, b^M\}$. It builds a probabilistic model of the objective function, then selects parameters that could maximize it. The model is then refined using the chosen parameters. This process is repeated until an iteration budget T is reached, as shown in Algorithm 3 (Frazier, 2018). Our implementation of the Bayesian optimization is based on *scikit-optimization* (Head et al., 2021).

Algorithm 3 Bayesian Optimization

Place a Gaussian process prior on f.
Observe f at t_0 points according to an initial space-filling experimental design. Set $t = t_0$.
while $t \leq T$ **do**
  Update the posterior probability distribution on g using all available data.
  Let b_t be a minimizer of the acquisition function over b, where the acquisition function is computed using the current posterior distribution.
  Observe $f(b_t)$.
  Increment t.
end while
Return a solution b: either the point evaluated with the smallest $f(b)$ or the point with the smallest posterior.

B Proofs

Proof of Lemma 2.1. First, basing on Assumption I.I.D, samples collected for the construction of the retrieval prediction set construction share is i.i.d. with unobserved samples, satisfying the i.i.d. (exchangeability) assumption required by conformal prediction (PAC prediction set).

Second, basing on Assumption *Retriever Correctness*, for every input question q, since its relevant passage can be retrieved, the prediction set can contain the relevant passage if the threshold τ_{Ret} is appropriately set. (Otherwise, the prediction set cannot contain the relevant passage even if all retrieved passages are included.)

Third, since we construct the retriever set following conformal prediction with the error level being α_{Ret}, the resulting retriever prediction sets satisfy:

$$\Pr_{(q,p^*)\sim\mathcal{D}_{\text{Passage}}}(p^* \in C_{\text{Ret}}(q)) \geq 1 - \alpha_{\text{Ret}}.$$

□

Proof of Lemma 2.2. First, basing on Assumption I.I.D, samples collected for the construction of the LLM prediction set construction share is i.i.d. with unobserved samples, satisfying the i.i.d. (exchangeability) assumption required by conformal prediction (PAC prediction set).

Second, basing on Assumption *LLM Correctness*, for every input question and its most relevant passage q^*, since its semantically correct responses can be retrieved, the prediction set can contain the correct responses if the threshold τ_{LLM} is appropriately set. (Otherwise, the prediction set cannot contain correct responses even if all responses are included.)

Third, since we construct the LLM prediction set following conformal prediction with the error level being α_{LLM}, the resulting retriever prediction sets satisfy:

$$\Pr_{(q,p^*,r^*)\sim\mathcal{D}_{\text{Response}}}(r^* \in C_{\text{Ret}}(q,p^*) \geq 1 - \alpha_{\text{LLM}}..$$

□

Proof of Theorem 3. Given Lemmas 2.1 & 2.2 and $\alpha_{\text{Ret}} + \alpha_{\text{LLM}} = \alpha$, we can prove the end-to-end guarantee in the following way:

$$\begin{aligned}
\Pr_{(q,r^*)\sim\mathcal{D}}(r^* \in C_{\text{Agg}}(q)) &= \Pr_{(q,p^*)\sim\mathcal{D}_{\text{Passage}}}(p^* \in C_{\text{Ret}}(q)) \times \Pr_{(q,p^*,r^*)\sim\mathcal{D}_{\text{Response}}}(r^* \in C_{\text{Ret}}(q,p^*) \\
&\geq (1-\alpha_{\text{Ret}})(1-\alpha_{\text{LLM}}) \\
&= 1 - \alpha_{\text{Ret}} - \alpha_{\text{LLM}} + \alpha_{\text{Ret}} \times \alpha_{\text{LLM}} \\
&\geq 1 - \alpha_{\text{Ret}} - \alpha_{\text{LLM}} \\
&= 1 - \alpha.
\end{aligned}$$

□

B.1 PAC Prediction Set Construction

To construct prediction sets with probably approximately correct (PAC) guarantees, we use the same nonconformity measures states in 3.2 for retrieval and LLM tasks, respectively. Also, we will assign the error budgets α_{Ret} and α_{LLM} with $\alpha_{\text{Ret}} + \alpha_{\text{LLM}} = \alpha$. Additionally, we need to specify confidence levels for PAC prediction set. In our work, we specify $1 - \frac{\delta}{2}$ to the retriever and LLM PAC prediction set. Then, we have the following Corollaries:

Lemma 4.1. *Suppose the questions and their corresponding most relevant passage p^*'s are subject to the distribution $\mathcal{D}_{passage}$. Given the error budget α_{Ret} and confidence level $1 - \frac{\delta}{2}$,, the constructed retriever prediction sets satisfy the following inequality:*

$$\Pr_{B\sim\mathcal{D}_{Passage}}[\Pr_{(q,p^*)\sim\mathcal{D}_{Passage}}(p^* \in C_{Ret}(q)) \geq 1 - \alpha_{Ret}] \geq 1 - \frac{\delta}{2}. \tag{5}$$

Lemma 4.2. *Suppose the questions, their corresponding most relevant passage p^*'s, and semantically correct responses r^* are subject to the distribution $\mathcal{D}_{Response}$. Given the error budget α_{LLM} and confidence level $1 - \frac{\delta}{2}$, if Assumption I.I.D and Assumption LLM Correctness hold, the LLM sets using PAC prediction set satisfy the following inequality:*

$$\Pr_{B\sim\mathcal{D}_{Response}{}^N}[\Pr_{(q,p^*,r^*)\sim\mathcal{D}_{Response}}(r^* \in C_{LLM}(q,p^*)) \geq 1 - \alpha_{LLM}] \geq 1 - \frac{\delta}{2}. \tag{6}$$

Theorem 5. *Suppose the questions, and semantically correct responses r^* are subject to the distribution $\mathcal{D}$; a user-specified error level α is given. By aggregating retriever sets with error budget α_{Ret} with LLM sets with error budget α_{LLM} and confidence levels $1-\delta/2$, with $\alpha = \alpha_{Ret} + \alpha_{LLM}$, the aggregation sets satisfy the following inequality:*

$$\Pr_{B\sim\mathcal{D}}[\Pr_{(q,r^*)\sim\mathcal{D}}(r^* \in C_{Agg}(q)) \geq 1-\alpha] \geq 1-\delta.$$

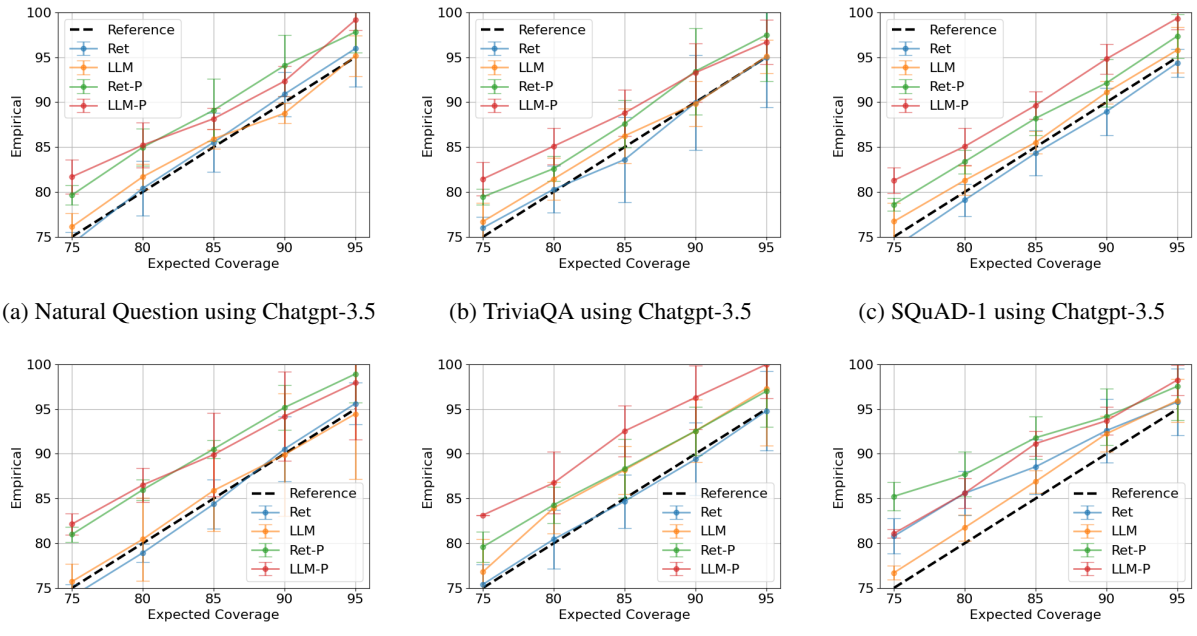
Proof of Theorem 5. Given Lemmas 4.1 & 4.2 and $\alpha_{\text{Ret}} + \alpha_{\text{LLM}} = \alpha$, we can prove the end-to-end guarantee in the following way:

$$\begin{aligned}
\Pr_{(q,r^*)\sim\mathcal{D}}(r^* \in C_{\text{Agg}}(q)) &= \Pr_{(q,p^*)\sim\mathcal{D}_{\text{Passage}}}(p^* \in C_{\text{Ret}}(q)) \times \Pr_{(q,p^*,r^*)\sim\mathcal{D}_{\text{Response}}}(r^* \in C_{\text{Ret}}(q,p^*) \\
&\geq (1-\alpha_{\text{Ret}})(1-\alpha_{\text{LLM}}) \\
&= 1-\alpha_{\text{Ret}}-\alpha_{\text{LLM}}+\alpha_{\text{Ret}}\times\alpha_{\text{LLM}} \\
&\geq 1-\alpha_{\text{Ret}}-\alpha_{\text{LLM}} \\
&= 1-\alpha.
\end{aligned}$$

Similarly, the confidence bound holds $(1-\delta)$ by taking a union bound over the outer probabilities of Equation (5) and (6). □

C Additional Results

C.1 Individual Coverage

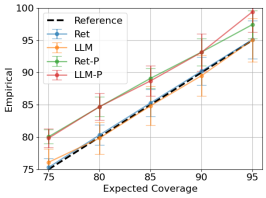


(a) Natural Question using Chatgpt-3.5 (b) TriviaQA using Chatgpt-3.5 (c) SQuAD-1 using Chatgpt-3.5

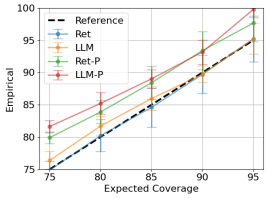
(d) Natural Quesiton using Llama-2 (e) TriviaQA using Llama-2 (f) SQuAD-1 using Llama-2

Figure 10: Individual Coverages on All Datasets and Both LLMs.

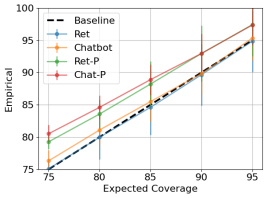
C.2 Individual Coverage with More Random Seeds



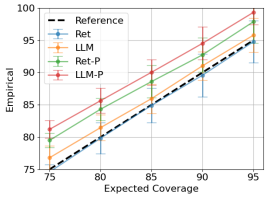
(a) BioASQ using Chatgpt-3.5



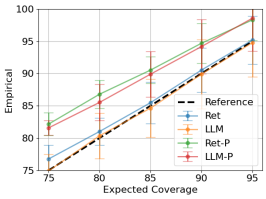
(b) Natural Question using Chatgpt-3.5



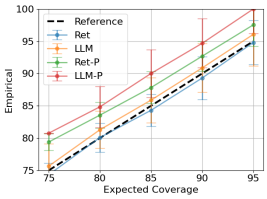
(c) TriviaQA using Chatgpt-3.5



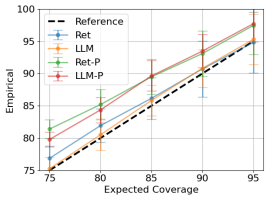
(d) SQuAD-1 using Chatgpt-3.5



(e) Natural Quesiton using Llama-2



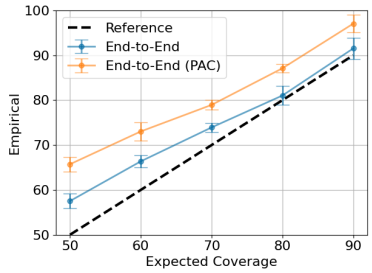
(f) TriviaQA using Llama-2



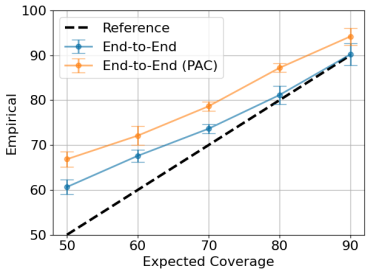
(g) SQuAD-1 using Llama-2

Figure 11: Individual Coverages on All Datasets and Both LLMs with More Random Seeds.

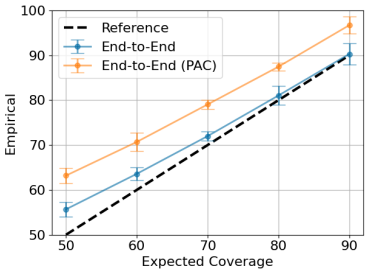
C.3 End-to-end Coverages



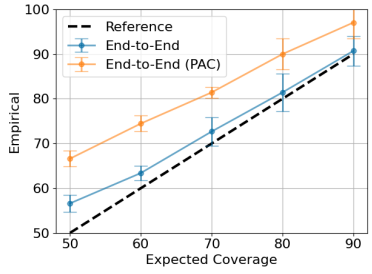
(a) Natural Question using Chatgpt-3.5



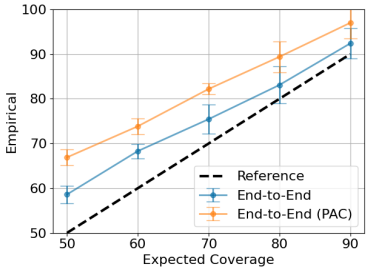
(b) TriviaQA using Chatgpt-3.5



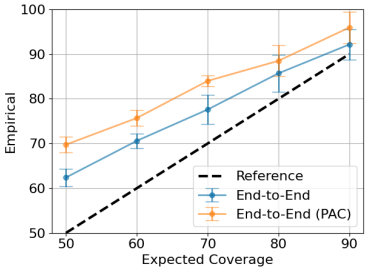
(c) SQuAD-1 using Chatgpt-3.5



(d) Natural Quesiton using Llama-2



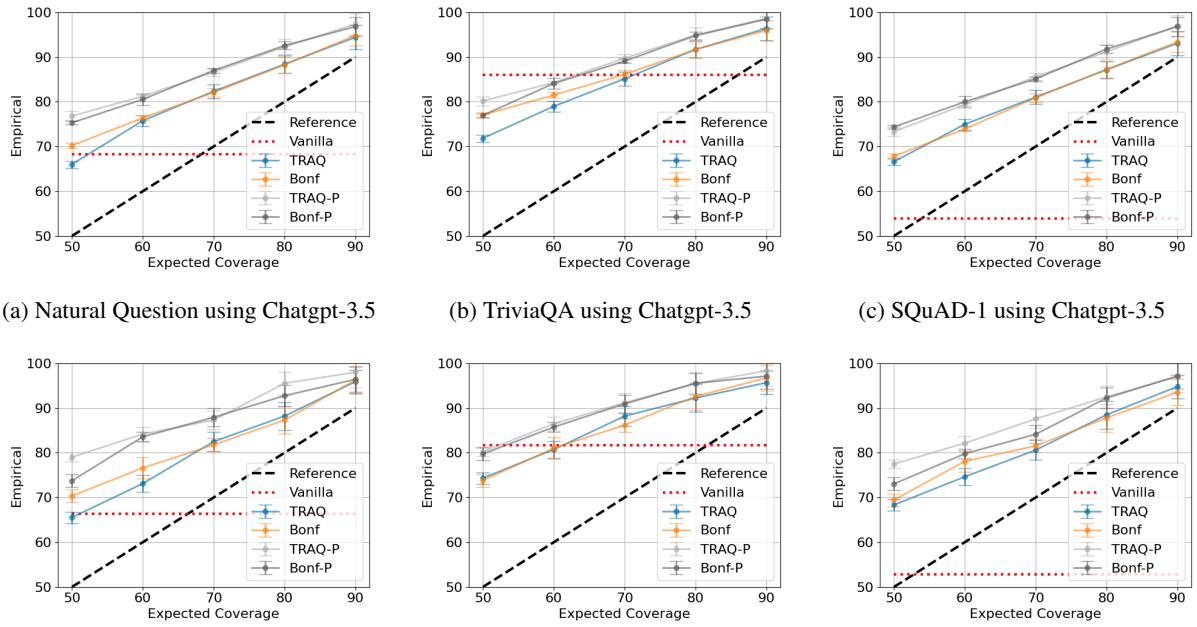
(e) TriviaQA using Llama-2



(f) SQuAD-1 using Llama-2

Figure 12: End-to-end Coverage on All Datasets and Both LLMs.

C.4 End-to-end Coverages



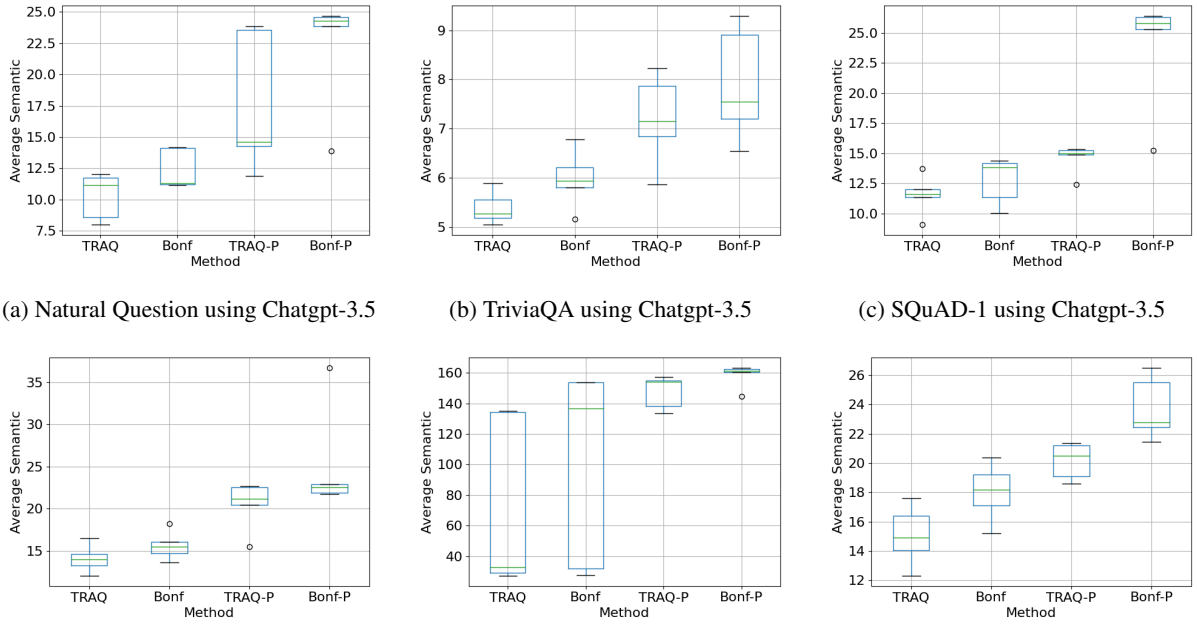
(a) Natural Question using Chatgpt-3.5 (b) TriviaQA using Chatgpt-3.5 (c) SQuAD-1 using Chatgpt-3.5

(d) Natural Quesiton using Llama-2 (e) TriviaQA using Llama-2 (f) SQuAD-1 using Llama-2

Figure 13: End-to-end Coverage on All Datasets and Both LLMs.

C.5 Performance

Most of the results are similar to those in Figure 6. The results on TriviaQA using Llama-2 have relatively large prediction set size. This could be explained by the fact that true scores on this task have a large variance. Therefore, the identified threshold τ_{LLM} was relative low (as in Figure 15a compared to other tasks (as in Figure 15b).



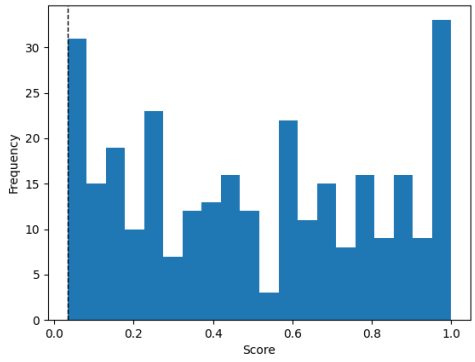
(a) Natural Question using Chatgpt-3.5 (b) TriviaQA using Chatgpt-3.5 (c) SQuAD-1 using Chatgpt-3.5

(d) Natural Quesiton using Llama-2 (e) TriviaQA using Llama-2 (f) SQuAD-1 using Llama-2

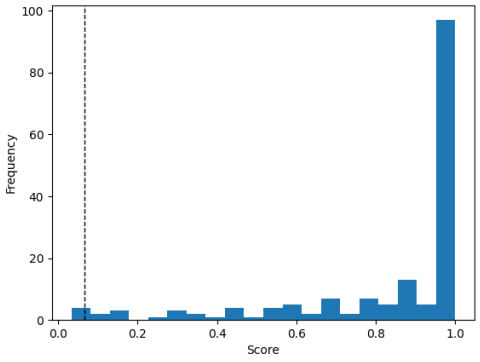
Figure 14: Efficiency on All Datasets and Both LLMs.

task	Cov	TRAQ	Bonf	TRAQ-P	Bonf-P
NQ	50	$2.84_{0.25}$	$2.98_{0.19}$	$3.47_{0.17}$	$3.61_{0.14}$
	60	$3.49_{0.17}$	$3.66_{0.13}$	$4.25_{0.19}$	$4.42_{0.31}$
	70	$4.45_{0.31}$	$4.62_{0.25}$	$5.47_{0.34}$	$5.77_{0.48}$
	80	$5.83_{0.38}$	$6.32_{0.79}$	$7.27_{0.69}$	$9.42_{1.80}$
	90	$10.26_{1.68}$	$12.56_{1.48}$	$17.07_{5.23}$	$22.45_{4.22}$
SQuAD1	50	$3.40_{0.11}$	$3.46_{0.12}$	$3.90_{0.07}$	$4.05_{0.05}$
	60	$3.94_{0.08}$	$4.07_{0.05}$	$4.42_{0.10}$	$5.07_{0.29}$
	70	$4.60_{0.09}$	$5.20_{0.21}$	$5.38_{0.25}$	$6.98_{0.64}$
	80	$6.12_{0.44}$	$7.84_{0.75}$	$7.96_{0.79}$	$11.12_{1.71}$
	90	$11.88_{1.68}$	$12.93_{1.79}$	$14.64_{1.11}$	$24.18_{4.39}$
Trivia	50	$1.90_{0.28}$	$2.03_{0.20}$	$2.25_{0.28}$	$2.36_{0.23}$
	60	$2.34_{0.24}$	$2.40_{0.22}$	$2.67_{0.25}$	$2.74_{0.25}$
	70	$2.80_{0.29}$	$2.92_{0.30}$	$3.38_{0.21}$	$3.47_{0.31}$
	80	$3.58_{0.30}$	$3.81_{0.26}$	$4.55_{0.31}$	$4.89_{0.40}$
	90	$5.60_{0.61}$	$6.16_{0.70}$	$7.38_{0.95}$	$8.15_{1.21}$
Average		$4.87_{0.46}$	$5.40_{0.50}$	$6.27_{0.73}$	$7.91_{1.10}$

Table 1: Average Semantic Counts using Chatgpt-3.5



(a) True Scores on TriviaQA using Llama-2



(b) True Scores on Natural Quesiton using Llama-2

Figure 15: True Scores Collected on TriviaQA and Natural Question using Llama-2

C.6 Additional Qualitative Results

C.6.1 All Covered

As shown in the example below, when the first retrieved passage is sufficiently informative, the LLM can probably generate correct responses for the question. In this case, TRAQ and Bonf can also include semantically correct responses in the aggregated sets. Again, TRAQ included as semantic meanings than Bonf did.

```
Query: who plays zack and cody in the suite life

Golden answer: ['Dylan and Cole Sprouse']

vanilla: {'Dylan and Cole Sprouse', 'Dylan and Cole Sprouse.'}

TRAQ: {'Dylan and Cole Sprouse', 'Dylan Sprouse', 'Phill Lewis'}

Bonf: {'Dylan and Cole Sprouse', 'Cole Sprouse', 'Dylan Sprouse'}
```

task	Cov	TRAQ	Bonf	TRAQ-P	Bonf-P
NQ	50	$4.81_{0.77}$	$5.00_{0.78}$	$6.14_{0.99}$	$6.57_{0.93}$
	60	$5.83_{0.85}$	$6.13_{1.02}$	$7.99_{0.91}$	$8.45_{0.95}$
	70	$7.69_{0.99}$	$7.86_{0.99}$	$10.28_{0.84}$	$10.67_{1.25}$
	80	$10.08_{0.73}$	$10.72_{1.23}$	$13.26_{1.61}$	$14.72_{1.23}$
	90	$14.10_{1.63}$	$15.63_{1.72}$	$20.5_{2.93}$	$25.16_{6.46}$
SQuAD1	50	$4.05_{0.26}$	$5.12_{0.47}$	$4.98_{0.55}$	$6.33_{0.49}$
	60	$5.37_{0.51}$	$6.52_{0.60}$	$6.60_{0.57}$	$7.74_{0.68}$
	70	$7.13_{0.42}$	$8.05_{0.66}$	$8.54_{0.55}$	$10.15_{0.61}$
	80	$9.34_{0.70}$	$11.3_{0.94}$	$11.54_{1.11}$	$14.45_{1.92}$
	90	$15.05_{2.05}$	$18.0_{1.98}$	$20.15_{1.25}$	$23.74_{2.15}$
Trivia	50	$3.99_{0.61}$	$4.51_{1.16}$	$5.27_{0.58}$	$6.47_{1.20}$
	60	$5.59_{0.97}$	$6.84_{1.22}$	$7.30_{1.25}$	$9.25_{2.21}$
	70	$8.48_{1.54}$	$10.22_{1.94}$	$11.30_{1.39}$	$18.48_{6.17}$
	80	$14.09_{2.14}$	$19.22_{6.13}$	$22.75_{6.84}$	$71.32_{60.67}$
	90	$71.67_{57.5}$	$100.68_{65.19}$	$147.61_{10.86}$	$158.35_{7.81}$
Average		$12.48_{4.78}$	$15.72_{5.74}$	$20.28_{2.15}$	$26.12_{6.31}$

Table 2: Average Semantic Counts using Llama-2

C.7 Miscovered

```
Query: who sang i love rock and roll original

GOLDEN ANSWER: ['Alan Merrill']

vanilla: {'Joan Jett'}

TRAQ: {'Joan Jett', 'Elvis Presley', 'Lou Reed', 'Joan Jett \& the Blackhearts', 'Alan Merrill', '
    Chuck Berry', 'Donna Summer', 'Kevin Johnson', 'Joan Jett and The Arrows'}

Bonf: {'Joan Jett', 'Elvis Presley', 'The Velvet Underground', 'Lou Reed', 'Joan Jett & the
    Blackhearts', 'Alan Merrill', 'Chuck Berry', 'Donna Summer', 'Bobby Vee', 'Buddy Holly', 'Kevin
    Johnson', 'Mac Davis', 'The original version of "I Love Rock and Roll" was sung by The Arrows.',
     'The Runaways', 'The answer to the question is not provided in the given context.', 'The
    Runaways sang the original version of "I Love Rock and Roll".', 'Joan Jett and The Arrows'}
```

D Implementation Details

D.1 Llama-2 Finetune Hyperparameters

We used 4-bit QLoRA (Dettmers et al., 2023) to fine-tune Llama-2 (Touvron et al., 2023b) models on Natural Question, TriviaQA, and SQuAD-1 datasets separately. Hyperparameters used for QLoRA are listed in Table 3; for fine-tuning are listed in Table 4.

Name	Value	Name	Value
r	64	alpha	16
dropout	0.1	precision	4bit

Table 3: QLoRA Hyperparameters

Name	Value	Name	Value
batch_size	16	learning rate	2e-4
weight_decay	0.001	lr scheduler	constant
warmup ratio	0.03	epoch	3

Table 4: Fine-tuning Hyperparameters

D.2 Finetune Dense Passage Retriever (DPR) on the Biomedical Dataset (BioASQ)

We collected our dataset for DPR fine-tuning by using the collection of all passages mentioned in BioASQ as our knowledge corpus, resulting in **56,795** passages. Following the method in (Karpukhin et al., 2020a), we created negative contexts for each sample in BioASQ by first retrieving the **top-20** passages; and labeling contexts that did not contain the golden answers as the **negative passages**. We then divided the original BioASQ dataset into training, validation, and testing sets, with 3,775, 471, and 469 data points, respectively.

We fine-tuned the DPR model (Karpukhin et al., 2020a) using the *Haystack* framework (Haystack), adjusting key hyperparameters to **epochs=5** and **batch size=16**. Other hyperparameters were left at their default values. To evaluate the performance of the fine-tuned DPR, we used *hit rate*, which is the rate of relevant passages included in the top-k retrieved passages. With k set to **20**, the fine-tuned DPR achieved hit rates of **77.2%** on the training set, **72.8%** on the validation set, and **75.7%** on the testing set.

D.3 Different Prompts

Zero-shot Prompt

Answer the following question based on the given context; Answer the question shortly.

Question: {question}
Context: {context}
Answer:

Few-shot Prompt

Answer the following question based on the given context; Answer the question shortly.

Question: {question 1}
Context: {context 1}
Answer: {answer 1}

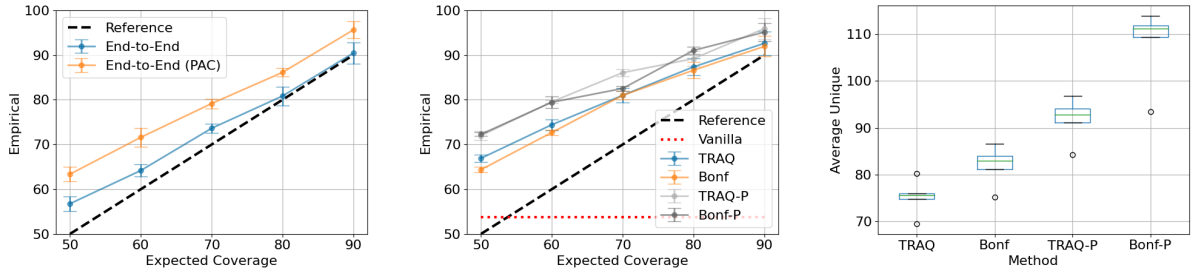
Question: {question 2}
Context: {context 2}
Answer: {answer 2}

Question: {question}
Context: {context}
Answer:

```
['The Great Lakes do not meet the ocean.',
'The Great Lakes meet the ocean at the Saint Lawrence River.',
'The Great Lakes meet the ocean through the Saint Lawrence River.',
'The Great Lakes do not meet the ocean.',
'The Great Lakes do not directly meet the ocean.',]

['There is no specific answer given in the provided context about where the Great Lakes meet the
    ocean.',
```

```
'Atlantic Ocean',
'Saint Lawrence River',
'The Great Lakes do not meet the ocean.',
'The Great Lakes do not meet the ocean. They are primarily connected to the Atlantic Ocean through
    the Saint Lawrence River.',
'The Great Lakes do not meet the ocean. They connect to the Atlantic Ocean through the Saint Lawrence
     River.',
'The Great Lakes meet the ocean through the Saint Lawrence River.',
'They do not meet the ocean.']
```



(a) End-to-end Coverage (b) End-to-end Coverage 2 (c) Performance

Figure 16: Results using Few-shot Prompting on Natural Question using Chatgpt-3.5.

D.4 Main Packages

Package	Version	Package	Version
transformer (Wolf et al., 2020)	4.32.1	nltk (Bird et al., 2009)	3.8.1
spacy (Honnibal and Montani, 2017)	3.6.1	torch (Paszke et al., 2019)	2.0.1
rouge-score (Lin, 2004)	0.1.2	scikit-optimize (Head et al., 2021)	0.9.0

D.5 Artifact License and Terms

Our implementation is based on *haystack*, *transformers* and *DPR* (Karpukhin et al., 2020a). The first two are licensed under **Apache License 2.0**, the third is licensed under **Attribution-NonCommercial 4.0 International**. We used four datasets, namely BioASQ, Natural Quesiton, TriviaQA, and SQuAD-1. BioASQ is licensed under the **CC BY 2.5 license**, Natural Question is under **CC BY-SA 3.0 license**, TriviaQA is under the **Apache License 2.0**, and SQuAD-1 is under the **CC BY-SA 4.0 license**. We used two LLMs, namely *Chatgpt-3.5* and *Llama-2*. Chatgpt-3.5 usage is subject to OpenAI's *Sharing & Publication Policy* and *Usage Policies*. Llama-2 is under Llama-2 Community License (Meta, 2023). Our implementation and the data collected are under the **MIT License**.

Our use of the existing artifacts is consistent with their original intended use. Our created artifacts intends to verify our proposed method in our submission, which is consistent with original access conditions.

E Removing Assumption *LLM Correctness*

In certain scenarios, even if the most pertinent passage is identified and given to the language understanding model (LLM), the LLM is still unable to answer the question with accurate answers. This could be due to a variety of reasons, such as the passage not being sufficiently specific or the LLM not being able to extract enough information from the passage. If the LLM is unable to generate correct responses even when the most pertinent passage is provided, our guarantee regarding the LLM and end-to-end pipeline may not hold. This problem can be alleviated by annotating better passages or creating more powerful LLMs.

To address the issue with the existing datasets and language models, we offer a guarantee of claiming *I do not know* if the language model is unable to generate a correct response to a question and its most

relevant passage. We collected questions and their most relevant passages, and also whether the Chatgpt-3.5 could generate a correct response. We then divided the dataset into training, validation, and testing sets, with 6,899, 1,725, and 1,725 data points respectively. We labeled **True** if the language model could generate a correct response, and **False** otherwise. We then trained a BERT-based text classifier which takes in the questions and their most relevant passages, and predicts whether the Chatgpt-3.5 can generate a correct response. We named the trained classifier *Conf-Classifier*. Surprisingly, Conf-Classifier achieved an accuracy of 95% on the testing set. To provide guarantees, we applied conformal prediction to the outputs of Conf-Classifier. We included *I do not know* in the LLM set if the constructed prediction set contained **False**.

To construct the calibration set, we collected estimated confidences on **not being able to answer the question** on input questions that the LLM failed to generate correct response. We denote these estimated confidences as $\{s_1, \ldots, s_K\}$. Given a user-specified coverage level, we then utilized conformal prediction to identify the $\frac{\lceil (K+1)(1-\alpha) \rceil}{K}$) quantile as the threshold τ_{Ign} to construct the set. Given an input question q, we then include *I do not know* in the aggregation set $C_{Agg}(q)$ if the estimated confidence n_{K+1} is above τ_{Ign} Then, we can guarantee the following:

Lemma 5.1. *Given an input question q that the LLM cannot correct answer and a user-specified error level α, if α_{Ign} is used to decide whether to include I do not know, the aggregation set satisfies the following property:*

$$\Pr_{q \sim \mathcal{D}}[\textit{I do not know} \in C_{Agg}(q)]$$

This results follows straightforwardly from Theorem I.I.D.

We tested our guarantee by using five distinct random seeds and five different coverage levels. The results are shown in Figure 17. As the figure illustrates, our method can include *I do not know* at various required coverage levels. By combining this with our guarantee on the LLM, we can guarantee all questions.

Theorem 6. *Given a user-specified error level α, if aggregation are constructed with error level α, the resulting prediction sets contain true answers (i.e., semantically correct responses if the input question is answerable; or I do not know if the input question is unanswerable) with probability at least $1 - \alpha$, i.e.,*

$$\Pr_{q \sim \mathcal{D}}[\textit{True answer} \in C_{Agg}(q)] \geq 1 - \alpha.$$

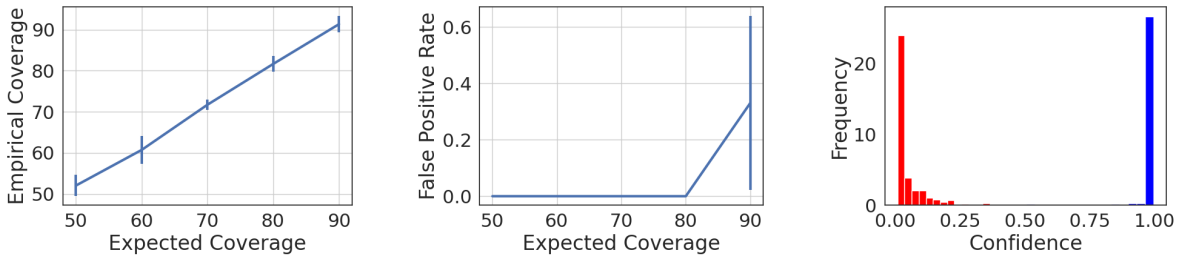
Proof. Suppose we construct the aggrgation set and ignorance set both with coverage level $1 - alpha$, then we have the following inequalities:

$$\begin{aligned}
&\Pr_{q \sim \mathcal{D}}[\text{True answer in the resulting set}] \\
&= \Pr_{q \sim \mathcal{D}}[\text{Correct response} \in C_{\text{Agg}}(q)] \times \Pr[\text{q is answerable}] \\
&+ \Pr_{q \sim \mathcal{D}}[\text{I do not know} \in C_{\text{Agg}}(q)] \times \Pr[\text{q is unanswerable}] \\
&\leq (1-\alpha) \times \Pr[\text{q is answerable}] + (1-\alpha) \times \Pr[\text{q is unanswerable}] \\
&= 1 - \alpha.
\end{aligned}$$

□

F AI Assistant Usage

We used *Copilot* to assist our coding.



(a) Coverage Rate on *I do not know*.

(b) False Positive Rates (claiming *I do not know but actually being able to answer*.

(c) The distribution of confidence on claiming *I do not know* using the training classifier.

Figure 17: Results on Identifying Whether A Given Prompt Is Answerable or Not.