Uncertainty Quantification in Retrieval Augmented Question Answering

Anonymous ACL submission

Abstract

Retrieval augmented Question Answering (QA) helps QA models overcome knowledge gaps by incorporating retrieved evidence, typically a set of passages, alongside the question at test time. Previous studies show that this approach improves QA performance and reduces hallucinations, without, however, assessing whether the retrieved passages are indeed useful at answering correctly. In this work, we propose to quantify the uncertainty of a QA model via estimating the utility of the passages it is provided with. We train a lightweight neural model to predict passage utility for a target QA model and show that while simple information theoretic metrics can predict answer correctness up to a certain extent, our approach efficiently approximates or outperforms more expensive sampling-based methods.[1]

1 Introduction

Retrieval augmented Question Answering (QA) allows QA models to overcome knowledge gaps at test time through access to evidence in the form of retrieved passages (Lewis et al., 2020; Guu et al., 2020; Izacard et al., 2024). Recent work leverages external retrievers (Chen et al., 2017; Izacard and Grave, 2021b) and the language understanding and generation capabilities of Large Language Models (LLMs; Brown et al. 2020; Ouyang et al. 2024) to predict answers based on questions *and* retrieved passages which are provided as input context. In Figure 1, we show an example of a question (*Who sings Does He Love Me with Reba?*), retrieved passages, and predicted answers.

Retrieval augmented QA architectures have proven beneficial in increasing LLM performance on tail knowledge (Izacard et al., 2024; Mallen et al., 2023), reducing hallucinations in the generated answers, and even improving model calibration (Jiang et al., 2021). However, there are various ways in which retrieval augmented QA can go wrong at inference time. The set of retrieved passages is far from perfect (Sciavolino et al., 2021; Yoran et al., 2024; Kasai et al., 2024) containing irrelevant or misleading evidence, the model might be under-trained to read certain passages and reason over these and the question (Izacard et al., 2024; Liu et al., 2024b), and the question can be ambiguous or unanswerable (Kasai et al., 2024). In such cases of uncertainty, QA models should ideally be able to deal with it (e.g., communicating it or abstaining from answering) rather than risking an incorrect response.

A good deal of previous work has focused on quantifying *answer uncertainty* in the context of *closed-book* QA tasks, where the answer is predicted based on a question and the model's encoded knowledge. Sampling-based methods rely on output discrepancies among multiple predictors of the same input (Gal and Ghahramani, 2016; Lakshminarayanan et al., 2017). They measure diversity on a set of answers (Kuhn et al., 2023; Chen and Mueller, 2024) sampled via temperature scaling (Guo et al., 2017), with larger variance indicating higher uncertainty. LLM-based methods rely on the QA model's own judgment of uncertainty (Kadavath et al., 2022; Lin et al., 2022; Tian et al., 2023). Through prompting, the model is encouraged to express its uncertainty (e.g., 0.5 or '*almost certain*'), either alongside the predicted answer (Lin et al., 2022; Tian et al., 2023) or after generating it (Kadavath et al., 2022; Tian et al., 2023).

In this paper, we focus on *retrieval augmented* QA and hypothesize that a passage is *useful*, if a model can correctly answer questions based on it. If passages are informative and prime the QA model towards appropriate knowledge, we expect it to produce a correct answer. On the contrary, if passages are misleading and the question falls outside the QA model's knowledge, it is likely to produce an incorrect answer — either factually in-

[1] Code and data are available at XXXX.

Figure 1: Example question from Natural Questions dataset (Kwiatkowski et al., 2019) with three top-retrieved passages using Contriever-MSMARCO (Izacard et al., 2022). On top of each passage, we show the answer generated by GEMMA2-9B when prompted with that passage and the question. The QA model answers correctly (green) only when prompted with the first passage. Numbers at the bottom right of each passage are utility scores predicted by our model (higher values indicate more useful passages).

accurate or entirely fabricated. We quantify the *utility* of a retrieved passage with a small neural model trained on utility judgements predicted by the target QA model. We borrow ideas from direct uncertainty quantification approaches (Van Amersfoort et al., 2020; Lahlou et al., 2023) but do not decompose uncertainty or outline shifts in the input distribution. We make utility predictions for each retrieved passage which we then use to estimate the uncertainty of the QA model.

We evaluate our approach on short-form question answering tasks (see Figure 1 for an example). Results on six datasets show that our uncertainty estimator outperforms existing sampling-based methods (especially in complex reasoning questions and adversarial QA settings with rare entities or unanswerable questions) while being more test-time efficient. Sampling-based solutions are expensive for in-production QA systems, in terms of latency (see comparison in Appendix A) and cost (e.g., QA engines built on top of proprietary language models would need to process relatively long prompts). Our contributions can be summarized as follows:

- We propose to quantify QA model uncertainty via estimating the utility of the passages it is provided with.
- We (contrastively) train a small neural model on utility scores obtained through combining accuracy (is the generated answer correct?) and entailment (is the answer supported by the passage?) metrics.
- Our approach is lightweight, improves upon more expensive sampling-based methods, and is not tied to the retriever (and passages) used to prompt the QA model.
- We show that utility scores predicted by our model can further improve QA accuracy by re-ranking passages obtained via an external retriever (Liu et al., 2024b).

2 Related Work

Uncertainty Quantification for Question Answering Several methods have been proposed to predict answer uncertainty in QA; however, none of them has analysed uncertainty in retrieval augmented QA models. Many existing approaches rely on the assumption that output variation is an expression of model uncertainty (Kuhn et al., 2023; Farquhar et al., 2024; Chen and Mueller, 2024). For example, Kuhn et al. (2023) first cluster answers with similar meaning (in a sample) via natural language inference before computing entropy while Chen and Mueller (2024) focus on *black-box* models; they also compute similarities in the set of answers but associate them with a model self-judgement of confidence. These approaches are expensive to run at inference time for a production QA system, they require several inference steps in addition to performing similarity computations which can become more complex with longer answers (Zhang et al., 2024). Hou et al. (2024) focus on quantifying aleatoric uncertainty (i.e., uncertainty in the data) caused by ambiguous questions, an approach which could be combined with ours.

Judging the Utility of Retrieved Passages Previous work has analysed the quality of retrieved passages (Yu et al., 2023; Asai et al., 2024; Wang et al., 2024; Xu et al., 2024; Yoran et al., 2024) as they can be irrelevant or misleading. Asai et al. (2024) make use of an external critic model to judge whether a question requires retrieval (or not) and whether the retrieved passages are relevant to formulate the answer. While they analyse passage *relevance*, this decision is taken by an external extreme-scale critic (e.g., GPT-4) and used to fine-tune their QA model. In contrast, we elicit *utility* judgements from the target QA model and use these to train a secondary small-scale model to predict passage utility (i.e., our approach does not require LLM fine-tuning). Other work creates auxiliary tasks around retrieved passages enforcing the QA model to reason on them; e.g., by taking notes about each passage (Yu et al., 2023) or generating passage summaries (Xu et al., 2024). These methods also use extreme-scale LLMs to generate training data for *fine-tuning* a retrieval augmented QA model. Park et al. (2024) select in-context examples with conflicting content (e.g., different dates for a given event) in order to improve LLM reasoning on input passages. These approaches aim at improving QA performance while our primary goal is modelling QA uncertainty.

Improving Retrieval via QA Performance Previous work has focused on jointly training the retriever and QA modules end-to-end (Lee et al., 2019; Lewis et al., 2020; Izacard and Grave, 2021a). This joint training scheme is very expensive for current (extreme-scale) LLMs. Our approach can be seen as an intermediate module between the QA model and the external retriever and could be used to provide feedback (i.e., utility scores) for fine-tuning the retriever, however, we leave this to future work. Salemi and Zamani (2024) evaluate the quality of retrieval on QA tasks and show that external judgements (e.g., query-document relevance labels) of passage utility correlate poorly with QA performance.

Using a Separate Model to Predict Confidence Some approaches train a specific model to predict answer confidence scores (Dong et al., 2018; Kamath et al., 2020; Zhang et al., 2021; Mielke et al., 2022) by incorporating various features from the input and model output. Our approach predicts answer uncertainty directly from individual passage utilities but could be combined with other features (e.g., output sequence probability). Some work (Kamath et al., 2020; Zhang et al., 2021) predicts answer correctness in the context of Reading Comprehension (the task of generating an answer based on a single supportive passage). However, as there is no retrieval involved, the input passage is by default useful, and the main goal is to detect answer uncertainty due to the QA model being under-trained. In our setting, the number and utility of passages varies leading to additional sources of uncertainty (e.g., misleading information).

Our passage utility predictor is related to methods aiming to estimate error *directly* (Lahlou et al., 2023), e.g., by training a secondary model to estimate target model loss; instead, our predictor is trained with sequence-level metrics, i.e., accuracy and entailment, which measure error *indirectly*.

3 Modelling Answer Uncertainty

We formally define retrieval augmented QA as follows. Given question x and a set of retrieved passages $R = \{p_1, p_2, \cdots, p_{|R|}\}$ obtained with retriever $\mathcal{R}$, an LLM-based QA model $\mathcal{M}$ is prompted to generate answer y to question x token-by-token according to its predictive distribution:

$$P(y|x, R; \mathcal{M}) = \prod_{t=1}^{|y|} P(y_t|y_{1..t-1}, x, R; \mathcal{M}). \quad (1)$$

We wish to estimate $\mathcal{M}$'s uncertainty (i.e., chance of error) of generating y given x and R.

When a retrieved passage is useful to answer a given question (such as the first passage in Figure 1 for the question *Who sings Does He Love Me with Reba?*), the QA model is likely to be confident when generating the answer (*Linda Davis*). When the passage is not useful (such as the third passage in Figure 1), the QA model is likely to be uncertain and provide an incorrect answer (*Trisha Yearwood*). Our hypothesis is that the utility of each passage p in R is indicative of the QA model's uncertainty in generating y, when prompted with R. If there are passages in R with high utility (e.g., in Figure 1, the first passage is useful to answer the question), it is likely that the QA model will be confident when generating answer y. In contrast, if all passages in R have low utility, it is likely that the QA model will be uncertain when generating the answer.

The core of our approach is estimating the utility $v_{\mathcal{M}}$ of individual passages for a target QA model $\mathcal{M}$. Specifically, we develop an estimator $\{x, p\} \mapsto v_{\mathcal{M}}(\{x, p\})$ for each passage $p \in R$ (Section 3.1). We then leverage the predicted passage utility $v_{\mathcal{M}}$ to estimate $\mathcal{M}$'s uncertainty when generating answer y to question x based on evidence R, $\{x, R\} \mapsto \mathbf{u}_{\mathcal{M}}(\{x, R\})$ (Section 3.2).

3.1 Passage Utility Prediction

Intuitively, a retrieved passage p is useful for a QA model $\mathcal{M}$ if $\mathcal{M}$ can correctly answer question x when prompted with p. However, the model's dependence on p may vary. In some cases, $\mathcal{M}$ may generate the correct answer even if p does not explicitly contain it, instead it positively primes the model to draw upon its memorised knowledge. In Figure 1, the first passage has high utility because the QA model generates a correct answer when prompted with it, and explicitly states that "Linda Davis sings alongside Reba McEntire". In contrast, the second and third passages, while related to the question's topic, are not useful. The QA model struggles to answer correctly when prompted with them, suggesting uncertainty. Since these passages do not provide helpful information and lead to incorrect answers, their utility is low.

We estimate the utility of passage p in answering question x for QA model $\mathcal{M}$ by combining two key measures: accuracy and entailment. Accuracy, denoted as $a(y)$, indicates whether the generated answer y is correct, while entailment, denoted as $e(y)$, measures the degree to which p supports y. Accuracy is determined by an evaluator A, and entailment is assessed using a Natural Language Inference (NLI) classifier model E. We define the combined passage utility as $\upsilon_{\mathcal{M}} = (a(y)+e(y))/2$ which ranges between 0 and 1, given that a takes values in $\{0, 1\}$ and e falls within the $[0, 1]$ interval.

We train a lightweight neural model on dataset $D_{\mathcal{M}} = \{(x, p, \upsilon_{\mathcal{M}})\}$ to predict passage utility scores, $\{x, p\} \mapsto \upsilon_{\mathcal{M}}(\{x, p\})$, We construct D by pairing input questions x and passages p with utility scores $\upsilon_{\mathcal{M}}$ which we obtain after running $\mathcal{M}$ on examples (x, p) and computing observed answer accuracy and entailment scores from the QA model $\mathcal{M}$. We retrieve $|R| > 1$ passages per question to ensure a diverse range of usefulness and create training instances $\{(x, p_i, \upsilon_i) \,|\, p_i \in R\}$ under model $\mathcal{M}$. We leverage this data to train the passage utility predictor with a contrastive learning scheme. Specifically, if two passages p_i and p_j belong to R and p_i is more useful than p_j for answering question x, the predicted utility score υ_i should be higher than υ_j by margin m, ensuring that p_i is ranked above p_j. We train the utility predictor with the following ranking objective:

$$\mathcal{L}_{rank} = \sum_{((x,p_i),(x,p_j)) \in R \times R, i \neq j} \max(0, -z(\upsilon_i - \upsilon_j) + m)), \quad (2)$$

where z controls the gold order between p_i and p_j (e.g., if $z = 1$, p_i has higher utility, and conversely $z = -1$ indicates the opposite ordering) and m is a hyper-parameter.

We train the passage utility predictor with a Siamese neural network consisting of a BERT-based encoder (Devlin et al., 2019) followed by a pooling and two MLP layers stacked on top of BERT outputs (Fang et al., 2024). The output layer computes the utility score as $\upsilon_i = W_o h^L + b_o$ where h^L is the vector representation for (x, p_i) from the last hidden layer (the L-th layer) of the network. At inference time, we compute a single utility score for each passage. We provide implementation and training details in Section 4.

To strengthen the role of accuracy prediction as a training signal and regularise the range of utility values learned by the ranking scheme, we combine the ranking objective in Equation (2) with the following Binary Cross Entropy (BCE; Sculley 2010) objective:

$$\mathcal{L}_{BCE} = \sum_{(x,p) \in \{(x,p_i),(x,p_j)\}} [a \times \log(p(x,p)) + (1-a) \times \log(1 - p(x,p))], \quad (3)$$

where $p(x, p) = \text{sigmoid}(\upsilon)$ and a is the target accuracy label under model $\mathcal{M}$ taking values in the set $\{0, 1\}$. We train the utility predictor with the following combined objective:

$$\mathcal{L} = \mathcal{L}_{rank} + \lambda \mathcal{L}_{BCE}, \quad (4)$$

where λ is a hyper-parameter.

Both ranking and BCE objectives are compatible with gold annotations that could be provided in active learning or interactive settings (Simpson et al., 2020; Fang et al., 2024). For example, moderators of the QA system would provide judgments on the accuracy of the answers it predicts (e.g., *correct/incorrect*) and whether these are supported by the retrieved passages (e.g., *best* or *worse*).

3.2 Answer Uncertainty Estimation

For our QA task, we want to define an estimator $\{x, R\} \mapsto \mathbf{u}_{\mathcal{M}}(\{x, R\})$ which quantifies the uncertainty of model $\mathcal{M}$ when generating answer y for question x based on a prompt with passages R. We propose estimating $\mathbf{u}_{\mathcal{M}}$ directly from the utility scores of individual passages in R. The key intuition is that the higher the utility of one (or more) passages, the less uncertain $\mathcal{M}$ will be when generating answer y. Specifically, we estimate answer uncertainty $\mathbf{u}_{\mathcal{M}}$ by taking the maximum utility score among the passages in R:

$$\mathbf{u}_{\mathcal{M}}(\{x, R\}) = \max(\upsilon_{\mathcal{M}}(\{x, p\}) \,|\, p \in R). \quad (5)$$

4 Experimental Setup

Datasets We evaluate our approach to predicting answer uncertainty on short-form question answering tasks. Specifically, we present experiments on the following six datasets: Natural Questions (Kwiatkowski et al., 2019), TriviaQA (Joshi et al., 2017), WebQuestions (Berant et al., 2013), SQuAD (Rajpurkar et al., 2016), and PopQA (Mallen et al., 2023). We also evaluate on RefuNQ (Liu et al., 2024a), a dataset with unanswerable questions about non-existing entities. In Appendix B.1, we describe each dataset, provide example questions, and make available details about the splits used in our experiments.

QA Models We consider backbone instruction fine-tuned LLMs from different families of similar size. These are Llama-3.1-8B-Instruct (AI@Meta, 2024), Mistral-7B-Instruct-v0.3 (Jiang et al., 2023), and Gemma2-9B-it (Riviere et al., 2024). We also assess answer uncertainty quantification performance on QA models of the same family but with different sizes. To this end, we additionally evaluate on Gemma2-2B-it and Gemma2-27B-it. For all QA models, we use a simple prompt including the retrieved passages and the question in the context; the prompt is shown in Table 6 in the Appendix. The QA models' response is the most likely answer generated with greedy sampling at temperature equal to 0. Following previous work on retrieval augmented QA, we use Contriever-MSMARCO (Izacard et al., 2022) as our external retriever (Asai et al., 2024) and the target QA models are prompted with $|R| = 5$ passages (Yu et al., 2023; Asai et al., 2024; Xu et al., 2024). In Appendix B.2, we provide more details about inference and passage retrieval.

Accuracy Evaluation A precise metric for measuring accuracy is key when evaluating the quality of uncertainty estimation. Token overlap metrics are imprecise and can over- or under-estimate accuracy (e.g., *5* will not match *five*). Thus, we rely on an LLM-based accuracy evaluator to create training data for the Passage Utility predictor (i.e., A in Section 3.1) and to evaluate retrieval augmented QA performance. We use Qwen2-72B-Instruct (Yang et al., 2024) and the prompt proposed in Sun et al. (2024) to obtain accuracy judgments. More details about the LLM evaluator can be found in Appendix B.2.

Evaluation of Uncertainty Estimation To assess the quality of answer uncertainty prediction, we follow Farquhar et al. (2024) and report the Area Under the Receiver Operator Curve (**AUROC**) on detecting incorrect answers (i.e., answers with high uncertainty). In Appendix C.3, we also report the area under the 'rejection accuracy' curve (**AURAC**) which captures the accuracy a model would have if it refused to answer questions with highest uncertainty. Rejection accuracy is essentially the model's accuracy on the remaining questions. We report accuracy at different answer rejection thresholds, i.e., when models answer 80% and 90% of the least uncertain questions, as well as when always answering. We provide implementation details in Appendix B.2.

Training the Passage Utility Predictor We train a Passage Utility predictor per QA model and QA task. For each task, we create set $D_{\mathcal{M}} = \{(x, p, \upsilon_{\mathcal{M}})\}$ to train and evaluate a Passage Utility predictor for QA model $\mathcal{M}$. We use the train (and development) questions available for the QA task and consider the top five retrieved passages for each question (i.e., $|R| = 5$). Note that this is a hyper-parameter and other values would be also possible. Larger sizes of $|R|$ would yield more training data, since the Utility predictor takes individual passages (together with the question) as input, First, the target QA model $\mathcal{M}$ is prompted with passage $p \in R$ and question x to generate answer y. Then, we annotate passages p with a utility score computed with the accuracy evaluator A and the entailment judge E on the generated answer y (Section 3.1). We use an ALBERT-xlarge (Lan et al., 2020) model optimized on MNLI (Williams et al., 2018) and the VitaminC dataset (Schuster et al., 2021). We report further details about the Passage Utility predictor training in Appendix B.2.

Comparison Approaches and Baselines We compare against several strong uncertainty estimation methods (Fadeeva et al., 2023) which we briefly describe below and also report additional comparisons in Appendix C.3.

Information-Theoretic Measures We compare against uncertainty estimation methods that are based on the predictive probabilities of the target

QA model. For a generated answer y with probability $p(y|x, R; \mathcal{M}) = \prod_{t=1}^{|y|} p(y_t|y_{1..t-1}, x, R; \mathcal{M})$, the Perplexity (PPL) of model $\mathcal{M}$ is computed as:

$$\text{PPL}(x, R, \mathcal{M}) = \exp\{-\frac{1}{|y|}\sum_{t=1}^{|y|} \log p(y_t|y_{1..t-1}, x, R; \mathcal{M})\}, \quad (6)$$

Perplexity essentially calculates *token-level* entropy as it is based on the average negative log-likelihood of the generated tokens.

Regular entropy, on the other hand, is computed over *sequences*, quantifying the entropy of the answers. It is defined as $\mathbb{E}[-\log P(Y|x, R; \mathcal{M})]$ where the expected value, $\mathbb{E}$, is computed on sequences y sampled from the conditional distribution $P(Y|x, R; \mathcal{M})$, where random variable Y denotes the answer sequences, and x and R are the input question and retrieved passages, respectively. In practice, regular entropy is approximated via Monte-Carlo integration, i.e., sampling N random answers from $P(Y|x, R; \mathcal{M})$:

$$\text{RE}(x, R, \mathcal{M}) = -\frac{1}{N}\sum_{n=1}^{N} \log \tilde{P}(y^n \,|\, x, R; \mathcal{M}), \quad (7)$$

where $\tilde{P}(y^n \,|\, x, R; \mathcal{M})$ is the length normalised version of $P(y^n|x, R; \mathcal{M})$.

Kuhn et al. (2023) propose Semantic Entropy, a variant of regular entropy that disregards uncertainty related to the surface form of the generated answers. The method works by sampling several possible answers to each question and grouping the set of N samples into M clusters (with $M \leq N$) that have similar meanings (which are determined on the basis of whether answers in the same cluster entail each other bidirectionally). The average answer probability within each cluster is:

$$\text{SE}(x, R, \mathcal{M}) = -\sum_{m=1}^{M} \hat{P}_m(x, \mathcal{M}) \log \hat{P}_m(x, \mathcal{M}), \quad (8)$$

where $\hat{P}_m(x, \mathcal{M}) = \frac{\sum_{y \in C_m} \tilde{P}(y|x,R;\mathcal{M})}{\sum_{m=1}^{M}\sum_{y \in C_m} \tilde{P}(y|x,R;\mathcal{M})}$.

LLM-based Measures We compare with p(true) which uses the same LLM-based target QA model to assess whether the answers it produces are correct (Kadavath et al., 2022). We follow the p(true) variant used in previous work (Kuhn et al., 2023). The QA model is prompted with the question and a set of candidate answers (consisting of the most likely answer and a sample of N answers) and is instructed to respond whether the most likely answer is true or false (i.e., correct/incorrect). The score produced by this approach is the probability of model $\mathcal{M}$ generating the token True. This method needs several in-context examples to work well; following Kuhn et al. (2023), we use 20 in-context examples. Note that since our backbone LLMs are recent models with a large context (unlike Kuhn et al. 2023), all 20 examples are fed in the context making p(true) an expensive but very strong approach. In the context of retrieval augmented QA, we include in the prompt the set of retrieved passages for the question to evaluate. We illustrate the prompt used by p(true) in Appendix B.3.

For approaches that require sampling, we follow previous work (Farquhar et al., 2024) and take $N = 10$ samples, which we generate with multinomial sampling. We set the sampling temperature to 1, with nucleus sampling ($P = 0.9$; Holtzman et al. 2020) and top$-K$ sampling ($K = 50$; Fan et al. 2018), and use a different random seed to draw each sample. We provide further details about inference in Appendix B.2 and report inference costs for each approach in Appendix A.

5 Results and Analysis

Light-weight answer uncertainty prediction works across model families and question answering tasks. Uncertainty estimation AUROC results for three QA models (GEMMA2-9B, LLAMA-3.1-8B, and MISTRAL-7B-V0.3) are shown in Table 1 (results on the development set are included in Appendix C.3). In general, answer perplexity (PPL) performs rather poorly, especially for GEMMA2-9B and MISTRAL-7B-V0.3. Regular Entropy improves upon PPL by ignoring surface form choices and focusing on meaning, Semantic Entropy further improves AUROC scores. p(true) performs well at detecting answer uncertainty matching or surpassing Semantic Entropy. It is important to note that this method relies on sampled answers and a long prompt with 20 in-context examples. Our Passage Utility approach performs on par or outperforms all other methods with a *single inference step* on each input passage.

Passage Utility performs particularly well on challenging question answering tasks represented

	NQ	TQA	WebQ	SQuAD	PopQA	RefuNQ
			GEMMA2-9B			
PPL	0.64	0.68	0.52	0.53	0.59	0.51
p(true)	**0.73**	0.75	0.67	0.63	0.81	0.62
Regular Entropy	0.66	0.69	0.54	0.56	0.61	0.51
Semantic Entropy	0.70	0.73	0.57	0.64	0.73	0.59
Passage Utility	0.72	**0.82**	**0.70**	**0.79**	**0.84**	**0.81**
			LLAMA-3.1-8B			
PPL	0.75	0.80	0.68	0.74	0.83	0.60
p(true)	**0.79**	**0.88**	0.72	0.77	0.85	0.67
Regular Entropy	0.76	0.81	0.69	0.78	0.83	0.65
Semantic Entropy	0.71	0.78	0.69	0.78	0.79	0.58
Passage Utility	0.78	0.78	**0.76**	**0.82**	**0.86**	**0.82**
			MISTRAL-7B-V0.3			
PPL	0.63	0.71	0.57	0.65	0.64	0.62
p(true)	0.73	**0.82**	0.68	0.74	0.75	0.68
Regular Entropy	0.64	0.75	0.62	0.65	0.66	0.60
Semantic Entropy	0.66	0.77	0.66	0.74	0.74	0.61
Passage Utility	**0.74**	**0.82**	**0.70**	**0.81**	**0.86**	**0.80**

Table 1: AUROC values for QA models GEMMA2-9B, LLAMA-3.1-8B, and MISTRAL-7B-V0.3 on Natural Questions (NQ), TriviaQA (TQA), WebQuestions (WebQ), SQuAD, PopQA, and RefuNQ test sets. Best values are highlighted in **bold**.

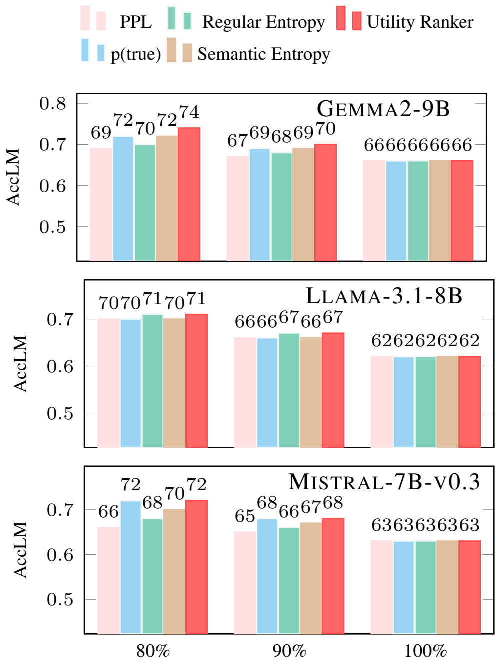

Figure 2: Average QA model accuracy (AccLM) across test sets: models refuse to answer according to different uncertainty estimation methods. 80/90%: the model answers 80/90% of the questions with low uncertainty; 100%: the model answers all questions.

by datasets like WebQ, SQuAD, and RefuNQ. In these cases, our light-weight uncertainty estimation model works better than p(true) which requires the same QA model (i.e., the same backbone LLM) to judge the correctness of its own generated answers. We speculate that for questions with high uncertainty, i.e., where the model does not have the knowledge to answer, it confidently generates a response and also fails at assessing it (e.g., questions about non-existing concepts in RefuNQ). We attribute the Passage Utility's success to the fact that it has been specifically trained to detect situations where the target QA model is prone to answer incorrectly (i.e., when provided with retrieved passages of lower relevance).

Answering selectively based on Passage Utility improves QA accuracy. Answer uncertainty can be used to decide whether to answer or refrain from doing so. Figure 2 shows QA model accuracy across datasets (average AccLM) at different thresholds of answer rejection. We report accuracy when models choose to answer only 80% and 90% of the cases where they are most certain, and when they always answer. Across different LLM backbones, Passage Utility performs on par with or better than more expensive uncertainty estimation approaches.

Passage Utility performance remains consistent across model sizes. Figure 3 (left), shows average AUROC scores on answer uncertainty estimation for Passage Utility and comparison approaches with different GEMMA2 sizes: 2B, 9B, and 27B. Our Passage Utility approach performs best across model sizes. All approaches obtain better AUROC scores for the smaller GEMMA2 model (2B) which makes most errors; we observe a noticeable decrease in performance for most information-theoretic models when using the biggest GEMMA model (27B). We attribute this to the fact that the 27B model more confidently makes less errors. p(true) on the other hand benefits from the largest model's context understanding and memorised knowledge.

Figure 3 (right) shows average accuracy when the target QA models choose to answer 80% of the cases they are most confident about. For comparison, we also show QA accuracy when always answering, i.e., black bold dots. When looking at selective performance according to Passage Utility, the small GEMMA2-2B model surpasses the bigger GEMMA2-27B one when always answering (0.68 vs 0.65), whereas the biggest GEMMA2-27B model improves by +9 points (0.74 vs 0.65).

Passage Utility scores provide good passage re-ranking. We hypothesize that by re-ordering the

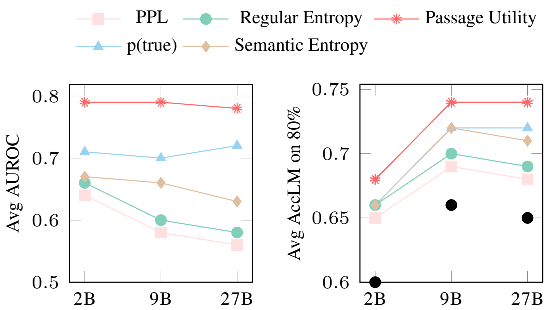

Figure 3: (left) Average AUROC for answer uncertainty estimation methods with varying GEMMA2 sizes: 2B, 9B, and 27B. (right) Average AccLM on the 80% most confident answers; black dots indicate average AccLM when always answering. Results computed on test sets.

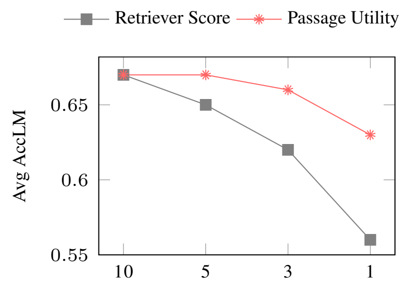

Figure 4: Average retrieval augmented QA accuracy (AccLM) for GEMMA2-9B across five QA test sets (NQ, TQA, WebQ, SQuAD, and PopQA). Points on the x-axis correspond to different context sizes $|R|$, when taking the top-k passages from different rankings produced by the retriever and Passage Utility estimator; k varies over $\{10, 5, 3, 1\}$.

set of retrieved passages according to their Passage Utility score and filtering the top-k ones, it will be possible to improve the accuracy and efficiency of the target QA model (Salemi and Zamani, 2024; Liu et al., 2024b). To test this, we measure QA accuracy for GEMMA2-9B on sets of input passages varying in order and size. Specifically, we measure performance with a set of $|R| = 10$ passages in their original ranking provided by an external retriever and the re-ranking imposed by the Passage Utility score. We then compute accuracy for the top-k passages with k in the range of $\{10, 5, 3, 1\}$. Figure 4 shows average accuracy (AccLM) values across five datasets (NQ, TQA, WebQ, SQuAD, and PopQA) on different input sets created based on the original retriever (gray) and the Passage Utility scores (red). At $k = 10$ both rankings lead to the same average QA accuracy (67%). However, when reducing the size of the context to the top $k = 5, 3, 1$ passages re-ranked by the Passage Utility score, the QA model achieves higher accuracy, indicating that Passage Utility indeed captures which passages are useful for the target QA model.

Training with pairwise judgements on Passage Utility helps improve predictions. Table 2 shows AUROC results on answer uncertainty prediction with Passage Utility estimators trained with different variants of the objective in Equation (4). The first row shows the full objective (see training details in Section B.2), the second row shows a variant where the ranking objective uses only entailment annotations (e), and in the third row the objective is solely based on accuracy prediction ($\mathcal{L}_{BCE}$). As can be seen, there is a drop in performance when the pairwise ranking loss is not used (i.e., last line of Table 2); this component of the objective provides a smoother signal on passage utility which is empirically beneficial. However, when the pairwise ranking loss is only based on entailment (an external critic), performance drops by several points which highlights the importance of training with utility judgements provided by the target QA model.

	G9B	L8B	M7B
$\mathcal{L}_{rank}, (e+a)/2 + \lambda\mathcal{L}_{BCE}$	0.79	0.81	0.79
$\mathcal{L}_{rank}, (e) + \lambda\mathcal{L}_{BCE}$	0.71	0.73	0.71
$\mathcal{L}_{BCE}$	0.77	0.78	0.80

Table 2: Answer uncertainty estimation with Passage Utility predicted by models trained with different variants of the training objective in Equation 4. We report AUROC for GEMMA2-9B (G9B), LLAMA3.1-8B (L8B), and MISTRAL-7B-V0.3 (M7B) averaged over development sets.

6 Conclusions

In this work we focus on retrieval augmented QA and present an approach to answer uncertainty prediction that relies on single passage utilities. We train a small neural model on passage utility judgements collected from the target QA model. We show that our uncertainty estimator is competitive or better than existing strong error prediction methods while being light-weight. Our experiments also show that our approach is particularly good in cases of extreme answer uncertainty such as questions about non-existing entities, where bigger QA models are prone to confidently formulate an incorrect answer. As future work, we would like to explore how to extend our approach to long-form generation tasks, such as query focused-summarisation.

7 Limitations

Instruction-tuned models are known to refuse to answer questions, i.e., they produce answers such as "*This information is not available in the text*". Refusing to answer is an adequate response when none of the input passages contains appropriate information. However, in many cases QA models refuse when in fact they should provide an answer (Adlakha et al., 2024; Liu et al., 2024a). Following previous work (Farquhar et al., 2024), we did not explicitly instruct the QA models to abstain and consider all cases where the answer does not match the goldstandard as incorrect. Refusing to answer is in our setting an indication of uncertainty (i.e., the QA model cannot provide a correct answer) which we aim to predict.

In this work, we focus on answer uncertainty estimation for short-form information-seeking QA tasks (Rodriguez and Boyd-Graber, 2021) where the answer can be often found in one Wikipedia passage. Going forward, it would make sense to extend our approach to multi-hop and related questions involving more complex reasoning (Yang et al., 2018; Pal et al., 2022). Although we expect Passage Utility to be effective in estimating the usefulness of individual passages, it is also possible that a more complex Passage Utility aggregation function is required (Dong et al., 2018).

References

Vaibhav Adlakha, Parishad BehnamGhader, Xing Han Lu, Nicholas Meade, and Siva Reddy. 2024. Evaluating correctness and faithfulness of instruction-following models for question answering. *Transactions of the Association for Computational Linguistics*, 12:681–699.

AI@Meta. 2024. Llama 3 model card.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2024. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *The Twelfth International Conference on Learning Representations*.

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on Freebase from question-answer pairs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1533–1544, Seattle, Washington, USA. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading Wikipedia to answer open-domain questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1870–1879, Vancouver, Canada. Association for Computational Linguistics.

Jiuhai Chen and Jonas Mueller. 2024. Quantifying uncertainty in answers from any language model and enhancing their trustworthiness. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5186–5200, Bangkok, Thailand. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Li Dong, Chris Quirk, and Mirella Lapata. 2018. Confidence modeling for neural semantic parsing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 743–753, Melbourne, Australia. Association for Computational Linguistics.

Ekaterina Fadeeva, Roman Vashurin, Akim Tsvigun, Artem Vazhentsev, Sergey Petrakov, Kirill Fedyanin, Daniil Vasilev, Elizaveta Goncharova, Alexander Panchenko, Maxim Panov, Timothy Baldwin, and Artem Shelmanov. 2023. LM-polygraph: Uncertainty estimation for language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 446–461, Singapore. Association for Computational Linguistics.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 889–898, Melbourne, Australia. Association for Computational Linguistics.

Haishuo Fang, Jeet Gor, and Edwin Simpson. 2024. Efficiently acquiring human feedback with Bayesian deep learning. In *Proceedings of the 1st Workshop on Uncertainty-Aware NLP (UncertaiNLP 2024)*, pages 70–80, St Julians, Malta. Association for Computational Linguistics.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. 2024. Detecting hallucinations in large language models using semantic entropy. *Nature*.

Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1050–1059, New York, New York, USA. PMLR.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1321–1330. PMLR.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020. Retrieval augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 3929–3938. PMLR.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The curious case of neural text degeneration. In *International Conference on Learning Representations*.

Bairu Hou, Yujian Liu, Kaizhi Qian, Jacob Andreas, Shiyu Chang, and Yang Zhang. 2024. Decomposing uncertainty for large language models through input clarification ensembling. In *Forty-first International Conference on Machine Learning*.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2022. Unsupervised dense information retrieval with contrastive learning. *Transactions on Machine Learning Research*.

Gautier Izacard and Edouard Grave. 2021a. Distilling knowledge from reader to retriever for question answering. In *International Conference on Learning Representations*.

Gautier Izacard and Edouard Grave. 2021b. Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 874–880, Online. Association for Computational Linguistics.

Gautier Izacard, Patrick Lewis, Maria Lomeli, Lucas Hosseini, Fabio Petroni, Timo Schick, Jane Dwivedi-Yu, Armand Joulin, Sebastian Riedel, and Edouard Grave. 2024. Atlas: few-shot learning with retrieval augmented language models. *J. Mach. Learn. Res.*, 24(1).

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. 2021. How Can We Know When Language Models Know? On the Calibration of Language Models for Question Answering. *Transactions of the Association for Computational Linguistics*, 9:962–977.

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.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, Deep Ganguli, Danny Hernandez, Josh Jacobson, Jackson Kernion, Shauna Kravec, Liane Lovitt, Kamal Ndousse, Catherine Olsson, Sam Ringer, Dario Amodei, Tom Brown, Jack Clark, Nicholas Joseph, Ben Mann, Sam McCandlish, Chris Olah, and Jared Kaplan. 2022. Language models (mostly) know what they know. *Preprint*, arXiv:2207.05221.

Amita Kamath, Robin Jia, and Percy Liang. 2020. Selective question answering under domain shift. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5684–5696, Online. Association for Computational Linguistics.

Jungo Kasai, Keisuke Sakaguchi, Yoichi Takahashi, Ronan Le Bras, Akari Asai, Xinyan Velocity Yu, Dragomir Radev, Noah A. Smith, Yejin Choi, and Kentaro Inui. 2024. Realtime qa: what's the answer right now? In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *The Eleventh International Conference on Learning Representations*.

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.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, SOSP '23, page 611–626, New York, NY, USA. Association for Computing Machinery.

Salem Lahlou, Moksh Jain, Hadi Nekoei, Victor I Butoi, Paul Bertin, Jarrid Rector-Brooks, Maksym Korablyov, and Yoshua Bengio. 2023. DEUP: Direct epistemic uncertainty prediction. *Transactions on Machine Learning Research*. Expert Certification.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. 2017. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. Albert: A lite bert for self-supervised learning of language representations. In *International Conference on Learning Representations*.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent retrieval for weakly supervised open domain question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6086–6096, Florence, Italy. Association for Computational Linguistics.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. 2024. Awq: Activation-aware weight quantization for on-device llm compression and acceleration. In *Proceedings of Machine Learning and Systems*, volume 6, pages 87–100.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. Teaching models to express their uncertainty in words. *Transactions on Machine Learning Research*.

Genglin Liu, Xingyao Wang, Lifan Yuan, Yangyi Chen, and Hao Peng. 2024a. Examining llms' uncertainty expression towards questions outside parametric knowledge. *Preprint*, arXiv:2311.09731.

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

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9802–9822, Toronto, Canada. Association for Computational Linguistics.

Sabrina J. Mielke, Arthur Szlam, Emily Dinan, and Y-Lan Boureau. 2022. Reducing conversational agents' overconfidence through linguistic calibration. *Transactions of the Association for Computational Linguistics*, 10:857–872.

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. 2024. Training language models to follow instructions with human feedback. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. 2022. Medmcqa: A large-scale multi-subject multi-choice dataset for medical domain question answering. In *Proceedings of the Conference on Health, Inference, and Learning*, volume 174 of *Proceedings of Machine Learning Research*, pages 248–260. PMLR.

SeongIl Park, Seungwoo Choi, Nahyun Kim, and Jay-Yoon Lee. 2024. Enhancing robustness of retrieval-augmented language models with in-context learning. In *Proceedings of the 3rd Workshop on Knowledge Augmented Methods for NLP*, pages 93–102, Bangkok, Thailand. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Gemma Team Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, L'eonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ram'e, Johan Ferret, Peter Liu, Pouya Dehghani Tafti, Abe Friesen, Michelle Casbon, Sabela Ramos, Ravin Kumar, Charline Le Lan, Sammy Jerome, Anton Tsitsulin, Nino Vieillard, Piotr Stańczyk, Sertan Girgin, Nikola Momchev, Matt Hoffman, Shantanu Thakoor, Jean-Bastien Grill, Behnam Neyshabur, Alanna Walton, Aliaksei Severyn, Alicia Parrish, Aliya Ahmad, Allen Hutchison, Alvin Abdagic, Amanda Carl, Amy Shen, Andy

Brock, Andy Coenen, Anthony Laforge, Antonia Paterson, Ben Bastian, Bilal Piot, Boxi Wu, Brandon Royal, Charlie Chen, Chintu Kumar, Chris Perry, Christoper A. Welty, Christopher A. Choquette-Choo, Danila Sinopalnikov, David Weinberger, Dimple Vijaykumar, Dominika Rogozi'nska, D. Herbison, Elisa Bandy, Emma Wang, Eric Noland, Erica Moreira, Evan Senter, Evgenii Eltyshev, Francesco Visin, Gabriel Rasskin, Gary Wei, Glenn Cameron, Gus Martins, Hadi Hashemi, Hanna Klimczak-Pluci'nska, Harleen Batra, Harsh Dhand, Ivan Nardini, Jacinda Mein, Jack Zhou, James Svensson, Jeff Stanway, Jetha Chan, Jin Zhou, Joana Carrasqueira, Joana Iljazi, Jocelyn Becker, Joe Fernandez, Joost R. van Amersfoort, Josh Gordon, Josh Lipschultz, Joshua Newlan, Junsong Ji, Kareem Mohamed, Kartikeya Badola, Kat Black, Katie Millican, Keelin McDonell, Kelvin Nguyen, Kiranbir Sodhia, Kish Greene, Lars Lowe Sjoesund, Lauren Usui, L. Sifre, L. Heuermann, Leticia Lago, Lilly McNealus, Livio Baldini Soares, Logan Kilpatrick, Lucas Dixon, Luciano Martins, Machel Reid, Manvinder Singh, Mark Iverson, Martin Gorner, Mat Velloso, Mateo Wirth, Matt Davidow, Matt Miller, Matthew Rahtz, Matthew Watson, Meg Risdal, Mehran Kazemi, Michael Moynihan, Ming Zhang, Minsuk Kahng, Minwoo Park, Mofi Rahman, Mohit Khatwani, Natalie Dao, Nenshad Bardoliwalla, Nesh Devanathan, Neta Dumai, Nilay Chauhan, Oscar Wahltinez, Pankil Botarda, Parker Barnes, Paul Barham, Paul Michel, Pengchong Jin, Petko Georgiev, Phil Culliton, Pradeep Kuppala, Ramona Comanescu, Ramona Merhej, Reena Jana, Reza Rokni, Rishabh Agarwal, Ryan Mullins, Samaneh Saadat, S. Mc Carthy, Sarah Perrin, S'ebastien Arnold, Sebastian Krause, Shengyang Dai, Shruti Garg, Shruti Sheth, Sue Ronstrom, Susan Chan, Timothy Jordan, Ting Yu, Tom Eccles, Tom Hennigan, Tomás Kociský, Tulsee Doshi, Vihan Jain, Vikas Yadav, Vilobh Meshram, Vishal Dharmadhikari, Warren Barkley, Wei Wei, Wenming Ye, Woohyun Han, Woosuk Kwon, Xiang Xu, Zhe Shen, Zhitao Gong, Zichuan Wei, Victor Cotruta, Phoebe Kirk, Anand Rao, Minh Giang, Ludovic Peran, Tris Brian Warkentin, Eli Collins, Joelle Barral, Zoubin Ghahramani, Raia Hadsell, D. Sculley, Jeanine Banks, Anca Dragan, Slav Petrov, Oriol Vinyals, Jeffrey Dean, Demis Hassabis, Koray Kavukcuoglu, Cl'ement Farabet, Elena Buchatskaya, Sebastian Borgeaud, Noah Fiedel, Armand Joulin, Kathleen Kenealy, Robert Dadashi, and Alek Andreev. 2024. Gemma 2: Improving open language models at a practical size. *ArXiv*, abs/2408.00118.

Pedro Rodriguez and Jordan Boyd-Graber. 2021. Evaluation paradigms in question answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9630–9642, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Alireza Salemi and Hamed Zamani. 2024. Evaluating retrieval quality in retrieval-augmented generation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '24, page 2395–2400, New York, NY, USA. Association for Computing Machinery.

Tal Schuster, Adam Fisch, and Regina Barzilay. 2021. Get your vitamin C! robust fact verification with contrastive evidence. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 624–643, Online. Association for Computational Linguistics.

Christopher Sciavolino, Zexuan Zhong, Jinhyuk Lee, and Danqi Chen. 2021. Simple entity-centric questions challenge dense retrievers. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6138–6148, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

D. Sculley. 2010. Combined regression and ranking. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '10, page 979–988, New York, NY, USA. Association for Computing Machinery.

Edwin Simpson, Yang Gao, and Iryna Gurevych. 2020. Interactive Text Ranking with Bayesian Optimization: A Case Study on Community QA and Summarization. *Transactions of the Association for Computational Linguistics*, 8:759–775.

Kai Sun, Yifan Xu, Hanwen Zha, Yue Liu, and Xin Luna Dong. 2024. Head-to-tail: How knowledgeable are large language models (LLMs)? A.K.A. will LLMs replace knowledge graphs? In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 311–325, Mexico City, Mexico. Association for Computational Linguistics.

Junya Takayama and Yuki Arase. 2019. Relevant and informative response generation using pointwise mutual information. In *Proceedings of the First Workshop on NLP for Conversational AI*, pages 133–138, Florence, Italy. Association for Computational Linguistics.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher Manning. 2023. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5433–5442, Singapore. Association for Computational Linguistics.

Joost Van Amersfoort, Lewis Smith, Yee Whye Teh, and Yarin Gal. 2020. Uncertainty estimation using a single deep deterministic neural network. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 9690–9700. PMLR.

Yuhao Wang, Ruiyang Ren, Junyi Li, Wayne Xin Zhao, Jing Liu, and Ji-Rong Wen. 2024. Rear: A relevance-aware retrieval-augmented framework for open-domain question answering. *arXiv preprint arXiv:2402.17497*.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Fangyuan Xu, Weijia Shi, and Eunsol Choi. 2024. RECOMP: Improving retrieval-augmented LMs with context compression and selective augmentation. In *The Twelfth International Conference on Learning Representations*.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zhihao Fan. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, Brussels, Belgium. Association for Computational Linguistics.

Ori Yoran, Tomer Wolfson, Ori Ram, and Jonathan Berant. 2024. Making retrieval-augmented language models robust to irrelevant context. In *The Twelfth International Conference on Learning Representations*.

Wenhao Yu, Hongming Zhang, Xiaoman Pan, Kaixin Ma, Hongwei Wang, and Dong Yu. 2023. Chain-of-note: Enhancing robustness in retrieval-augmented language models. *Preprint*, arXiv:2311.09210.

Huajian Zhang, Yumo Xu, and Laura Perez-Beltrachini. 2024. Fine-grained natural language inference based faithfulness evaluation for diverse summarisation tasks. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1701–1722, St. Julian's, Malta. Association for Computational Linguistics.

Shujian Zhang, Chengyue Gong, and Eunsol Choi. 2021. Knowing more about questions can help: Improving calibration in question answering. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1958–1970, Online. Association for Computational Linguistics.

Methods	Inference Calls at Test Time
PPL	1 G
p(true)	$(N+1)$ G + 1 E
Regular Entropy	$(N+1)$ G
Semantic Entropy	$(N+1)$ G + $\binom{N}{2}$ E
Passage Utility	$\lvert R \rvert$ Bert-F

Table 3: Number and type of inference calls required to estimate answer uncertainty for question x and set of retrieved passages R. G means inference is performed with a retrieval augmented QA model, i.e., a LLM forward pass with the prompt including the set of $|R|$ retrieved passages and question x to generate a candidate answer y. E is inference with an evaluation model, e.g., a forward pass to ask an LLM for correctness in p(true) or a forward pass with an entailment model in Semantic Entropy. Bert-F is an inference call to predict passage utility for passages p in R and question x.

A Test Time Cost of Uncertainty Estimation Methods

Table 3 shows the cost of estimating uncertainty for question x, measured by the number of inference calls required. Simple information theoretic methods (e.g., PPL) require a single call to the target QA model with the retrieval augmented QA prompt (i.e., $|R|$ retrieved passages and question x). However, approaches that estimate uncertainty based on diversity (e.g., Regular Entropy, Semantic Entropy, and p(true)) require generating N answers, i.e., N inference calls with the retrieval augmented QA prompt. In addition, Semantic Entropy requires the computation of answer clusters (i.e., grouping answers with the same meaning), so additional calls to an entailment model are required to compare the set of sampled answers. p(true) requires one additional LLM call to elicit a True/False answer but with a very long prompt including in-context examples and the assessment question with the $|R|$ retrieved passages, sampled and most likely answers, and question x (see Table 7). In contrast, our approach requires $|R|$ utility predictions with a BERT-sized model.

B Experimental Details

B.1 Datasets and Splits

In our experiments, we use six QA tasks which we describe below. Table 4 shows dataset statistics and example question-answers pairs.

Natural Questions (NQ; Kwiatkowski et al. 2019) is a QA dataset compiled from real user questions submitted to the Google search engine. As part of the dataset curation process, annotators judge the quality of questions and associate them with a short answer that can be extracted from a related Wikipedia page.

TriviaQA (TQA; Joshi et al. 2017) is a question answering dataset designed for training and evaluating machine learning models on open-domain question answering tasks. The dataset was created by gathering questions from trivia websites, along with their corresponding answers, to provide a broad range of factual questions.

WebQuestions (WebQ; Berant et al. 2013) was mined off questions generated with the Google Suggest API. The answers to the questions are defined as Freebase entities (i.e., their string label) and were elicited by Amazon Mechanical Turk (AMT) annotators.

SQuAD (Rajpurkar et al., 2016) contains questions formulated by AMT annotators based on a given Wikipedia paragraph, with the answer being a short span in that paragraph. Annotators were encouraged to use paraphrasing when writing the question. The answer types not only cover named entities but also other categories such as noun- and verb-phrases.

PopQA (Mallen et al., 2023) is an open-domain QA dataset, focusing on popular culture topics, such as movies, TV shows, music, and sports. It contains question-answer pairs derived from (subject, relation, object) triples in Wikidata . Triples were translated into natural language and the object entity was taken to be the gold answer. The collection process focused on gathering questions about subject entities of varying popularity.

RefuNQ (Liu et al., 2024a) is derived from NQ and consists of answerable and unanswerable questions. Unanswerable questions are created by replacing entities in the original NQ question by non-existing concepts.

We follow previous work (Lee et al., 2019) and use only the question and gold answers, i.e., the open versions of NQ, TQA, and SQuAD. We use the unfiltered TQA dataset. We follow the train/dev/test splits as used in previous work (Lee et al., 2019) and randomly split PopQA. RefuNQ only provides a test set so our experiments on this dataset are zero-shot from a Passage Utility predictor trained on SQuAD. We follow Farquhar et al. (2024) and use 400 test examples randomly sampled from the original larger test datasets for evaluation of uncertainty quantification.

B.2 Implementation Details

QA Models For all question answering tasks, we use the off-the-shelf Contriever-MSMARCO (Izacard et al., 2022) tool to retrieve sets of passages R for question x from Wikipedia and the official Wikipedia embeddings based (2018 snapshot) as our document knowledge-base. For PopQA, we follow the work by Mallen et al. (2023) who also use the full 2018 English Wikipedia dump.

The QA prompt used for all models (embedded in the corresponding chat templates) is shown in Table 6. For inference, we set the maximum number of generated tokens to 50 for both the greedy (most likely answer) as well as temperature scaled (sampled candidates) decoding. We use vLLM for inference (Kwon et al., 2023). For all models, inference was run on a single A100-80GB GPU.

Passage Utility Predictor We train a different predictor for each target QA model and QA task. Given the large number of predictors required in our experiments, we initially tested the hyperparameters used in Fang et al. (2024) on the NQ dataset and choose a set thereof for all predictor instances. We train each predictor for 3 epochs, with a batch size of 32 examples, learning rate equal to 2^{e-5}, and weight decay 0.001. For each predictor we performed search on values for λ, i.e., the contribution of the $\mathcal{L}_{BCE}$ loss (Equation 4), and different criteria for model selection, i.e., the best at pairwise ranking or at both pairwise ranking and accuracy prediction (combined).

Table 5 shows the configuration for each predictor. Table cells show selection criteria (R for ranking and C for combined) and the value for λ. A trend that seems to emerge for LLAMA-3.1-8B and MISTRAL-7B-V0.3 is that best predictors tend to rely more on the target QA model accuracy, potentially indicating that their answers in some cases

Dataset	Train	Dev	Test	Example Question	Example Answer
NQ	79,168	8,757	3,610	Who plays Letty in Bring it on all or nothing?	Francia Raisa
TQA	78,785	8,837	11,313	Who was the first artistic director of the National Theatre in London?	Lord Laurence Olivier
WebQ	2,474	361	2,032	What party was Andrew Jackson?	Democratic-Republican Party
SQuAD	78,713	8,886	10,570	What is the Grotto at Notre Dame?	A Marian place of prayer and reflection
PopQA	10,000	1,267	3,000	Who was the director of Champion?	Rabi Kinagi
RefuNQ	—	—	2,173	Who does the voice over in the Requirtion?	—

Table 4: Dataset statistics, number of instances per Train/Development(Dev)/Test sets, and example question-answer pairs (all taken from the Dev set except for RefuNQ).

Models	NQ	TQA	WebQ	SQuAD	PopQA
GEMMA2.9B	R, 0.25	R, 0.25	C, 1	C, 1	R, 0.25
LLAMA-3.1-8B	C, 0.25	C, 1	R, 0.25	C, 1	C, 1
MISTRAL-7B-V0.3	R, 0.25	C, 1	R, 0.25	C, 1	C, 1
GEMMA2.2B	R, 0.25	R, 0.25	R, 0.25	R, 0.25	R, 0.25
GEMMA2.27B	R, 0.25	C, 1	C, 1	R, 0.25	R, 0.25

Table 5: This table shows the λ value and selection criteria (R for pairwise ranking or C for combined pairwise ranking and accuracy prediction) for each Passage Utility predictor in our experiments.

depend less on context. However, for the smaller model GEMMA2-2B all predictors achieve a good ranking, which supports the hypothesis that small models rely more on the provided content to formulate their answers. At inference time we predict a single Passage Utility score given by the selected best checkpoint. For all predictor instances (except for all WebQ and PopQA predictors and the predictor for LLAMA-3.1-8B and NQ), we use half of the available training data to speed up experiments. Training and inference was run on a single A100-40GB GPU, training takes less than 12 hours depending on the dataset.

Comparison Approaches In the additional results section of the appendix (Section C.3), we report the following additional answer uncertainty estimation methods. Maximum Sequence Probability (MSP) is based on the probability of the most likely answer and is computed as

$$\mathrm{MSP}(x, R, \mathcal{M}) = 1 - P(y|x, R; \mathcal{M}). \quad (9)$$

Note that, in contrast to $PPL(x, R, \mathcal{M})$ reported in the main section of the paper, this metric is biased by answer length, i.e., identifying an answer to have low probability (low confidence) because of its length. Despite the fact that QA models are instructed to produce short answers, they do not always follow instructions. For this reason, we consider perplexity a more accurate metric. Indeed, the length of the answer could be a feature indicating that the model is uncertain about the answer.

We estimate answer uncertainty from the Average Answer Length (Avg.Ans.Len) as the average number of words in the sampled answers. We also report Cluster Assignment (CA) which is a variant of SE without answer probabilities where the probability of each generated meaning (i.e., a cluster) is approximated from the number of answers in the cluster. We found that in general CA estimations are very close to Semantic Entropy ones.

Another uncertainty estimation approach is the negative mean Point-wise Mutual Information (PMI; Takayama and Arase 2019) over tokens; i.e., it compares the probability of answer sequence y given a prompt with question x and passages R w.r.t the probability given by $\mathcal{M}$ to y without any context. Intuitively, the higher the point-wise mutual information, the more certain the QA model is on generating y (i.e., the answer is related to or depends on x and R). PMI is computed as

$$\mathrm{PMI}(x, R, \mathcal{M}) = -\frac{1}{|y|}\sum_{t=1}^{|y|}\log\frac{p(y_t|y_{1..t-1}, x, R; \mathcal{M})}{p(y_t|y_{1..t-1}; \mathcal{M})}. \quad (10)$$

We use the implementation provided by Farquhar et al. (2024) to compute Regular Entropy, Semantic Entropy, Cluster Assignment, and p(true).

Metrics We use the implementation provided by Farquhar et al. (2024) for the AUROC, Accuracy at X% of rejection, and AURAC metrics.

We use Qwen2-72B-Instruct (Yang et al., 2024) to obtain accuracy judgments (i.e., A judge, Section 4); specifically, we use the Activation-aware Weight Quantization (Lin et al., 2024) version Qwen2-72B-Instruct-AWQ. We prompt the accuracy evaluator with the prompt proposed by Sun et al. (2024), as we found to perform well. The accuracy evaluation (AccLM) prompt is shown in

Retrieval augmented QA prompt

Knowledge:
[1] passage
[2] passage
...
[$|R|$] passage

Answer the following question with a very short phrase.

Question: question

Table 6: Prompt designed as user turn for QA models.

p(true) prompt

Question: question
Brainstormed Answers: most likely answer
sampled answer 1
...
sampled answer N
Possible answer: most likely answer
Is the possible answer:
A) True
B) False
The possible answer is: correct choice

...

Knowledge:
[1] passage
[2] passage
...
[$|R|$] passage

Question: question
Brainstormed Answers: most likely answer
sampled answer 1
...
sampled answer N
Possible answer: most likely answer
Is the possible answer:
A) True
B) False
The possible answer is:

Table 7: Prompt used for the p(true) comparison approach. The items in blue are filled in with in-context examples from the training set and the current example being evaluated. N represents the number of sampled answers. The 'sequence of in-context examples' prefix is a sequence of examples taken from the training split with the same question format but with the answer to *The possible answer is:* resolved.

Table 8. In a sample of 840 generated answers human and LLM-based judgment of correctness agreed 98% of the time (Sun et al., 2024). As a reference point, to relate to accuracy as computed in previous work, we report retrieval augmented QA accuracy (**Acc**) defined as whether the gold answer is contained in the generated answer (Mallen et al., 2023; Asai et al., 2024).

B.3 Prompts

The prompt we use for our QA models is shown in Table 6. Table 7 illustrates the prompt used for our the p(true) baseline. Table 8 shows the prompt used for the LLM-based accuracy (AccLM) metric.

C Additional Results

C.1 Generalisation of Uncertainty Estimation

In this series of experiments, we assess the generalisation ability of our Passage Utility estimator. To this end, following previous work on question answering and out-of-distribution (o.o.d) scenarios (Kamath et al., 2020; Zhang et al., 2021); we train a Passage Utility predictor on the SQuAD dataset and then use it to predict zero-shot (i.e., without further fine-tuning) passage utilities on all other datasets' test cases. As p(true) relies on 20 in context training examples, we also evaluate its ability to generalise to out of distribution test cases.

Table 9 shows AUROC for answer uncertainty prediction on o.o.d scenarios. We also report PPL as a baseline and the AUROC values for the in distribution scenario as upper-bound. Although Passage Utility performance decreases in o.o.d settings, it remains competitive in three out of five datasets. On NQ and WebQ the performance is slightly above the PPL baseline. Note that we focus on zero-shot accuracy to assess bare transfer performance; however, it would make sense to adapt the model with few examples from the o.o.d data (Kamath et al., 2020; Zhang et al., 2021). Interestingly, p(true)'s performance also drops in all o.o.d test sets showing that relying on a fixed number of training examples is neither robust nor has a principled and scalable adaptation method (e.g., fine-tuning).

C.2 Reference Retrieval Augmented QA Accuracy

Table 10 shows retrieval augmented QA performance (Acc and AccLM) for the five QA models on the development and test sets of the six QA tasks.

C.3 Detailed Uncertainty Estimation Results

Table 11 shows performance of uncertainty quantification approaches on the development sets. We report AUROC and AURAC.

D Examples of False Positives and Negatives

Tables 12–15 illustrate the working of Passage Utility for answer uncertainty estimation. As we report AUROC scores, we do not set any correct/incorrect decision threshold; for the purpose of this discussion, we assume a decision point at 0.5 and analyse clear success and failure cases. For each example, we show the question, gold, and generated answers in the top block. Then, we show three retrieved passages with their estimated Passage Utility and a final block with the ten sampled answers, their grouping into clusters, and the Cluster Assignment entropy.

Table 12 shows an example for a SQuAD question and the LLAMA-3.1-8B QA model. In this case, the QA model correctly answers and the Passage Utility estimate is high (i.e., indicating correct answer). Table 13 illustrates a case where LLAMA-3.1-8B's answer is incorrect and all Passage Utilities are very low (i.e., indicating incorrect answer). The example from NQ in Table 14 shows a case where all Passage Utilities are low but the QA model (GEMMA2-9B) answers correctly. The first passage is not useful, the second does not explicitly mention the answer but still primes the QA model to answer correctly, while the third passage mentions the answer.

In Table 15, Passage Utility scores are high estimating a correct answer for the TQA test question; however, GEMMA2-9B answers with the incorrect magazine name. Note that none of the passages corresponds to the National Geographic magazine but have high token overlap with the question (in particular the first and second passages).

Accuracy evaluation (AccLM) prompt

You need to check whether the prediction of a question-answering system to a question is correct. You should make the judgment based on a list of ground truth answers provided to you. Your response should be "correct" if the prediction is correct or "incorrect" if the prediction is wrong.

Question: Who authored The Taming of the Shrew (published in 2002)?
Ground truth: ["William Shakespeare", "Roma Gill"]
Prediction: W Shakespeare
Correctness: correct

Question: Who authored The Taming of the Shrew (published in 2002)?
Ground truth: ["William Shakespeare", "Roma Gill"]
Prediction: Roma Gill and W Shakespeare
Correctness: correct

Question: Who authored The Taming of the Shrew (published in 2002)?
Ground truth: ["William Shakespeare", "Roma Gill"]"
Prediction: Roma Shakespeare
Correctness: incorrect

Question: What country is Maharashtra Metro Rail Corporation Limited located in?
Ground truth: ["India"]
Prediction: Maharashtra
Correctness: incorrect

Question: What's the job of Song Kang-ho in Parasite (2019)?
Ground truth: ["actor"]
Prediction: He plays the role of Kim Ki-taek, the patriarch of the Kim family.
Correctness: correct

Question: Which era did Michael Oakeshott belong to?
Ground truth: ["20th-century philosophy"]
Prediction: 20th century."
Correctness: correct

Question: Edward Tise (known for Full Metal Jacket (1987)) is in what department?
Ground truth: ["sound department"]
Prediction: 2nd Infantry Division, United States Army
Correctness: incorrect

Question: What wine region is Finger Lakes AVA a part of?
Ground truth: ["New York wine"]
Prediction: Finger Lakes AVA
Correctness: incorrect

Question: question
Ground truth: gold answer
Prediction: generated answer
Correctness:

Table 8: Prompt used for LLM-based accuracy evaluation (AccLM).

	NQ	TQA	WebQ	PopQA	RefuNQ
PPL	0.64	0.68	0.52	0.59	0.51
p(true) (i.i.d)	0.73	0.75	0.67	0.81	—
Passage Utility (i.i.d)	0.72	0.82	0.70	0.84	—
p(true) (o.o.d)	0.67	0.63	0.63	0.72	0.62
Passage Utility (o.o.d)	0.66	0.82	0.57	0.73	0.81

Table 9: Out-of-domain performance of Passage Utility predictor (with GEMMA2-9B). Uncertainty predictors are trained on SQuAD and evaluated zero-shot on NQ, TQA, WebQ, PopQA, and RefuNQ test sets.

	NQ		TQA		WebQ		SQuAD		PopQA		RefuNQ	
	Acc	AccLM	Acc	AccLM	Acc	AccLM	Acc	AccLM	Acc	AccLM	Acc	AccLM
Development												
GEMMA2.9B	0.48	0.66	0.74	0.80	0.46	0.66	0.38	0.60	0.51	0.52	—	—
LLAMA-3.1-8B	0.48	0.62	0.71	0.77	0.53	0.64	0.39	0.57	0.51	0.49	—	—
MISTRAL-7B-V0.3	0.48	0.62	0.72	0.76	0.52	0.69	0.37	0.58	0.53	0.51	—	—
GEMMA2.2B	0.43	0.59	0.67	0.73	0.47	0.65	0.34	0.55	0.48	0.49	—	—
GEMMA2.27B	0.48	0.66	0.75	0.81	0.49	0.67	0.38	0.60	0.52	0.52	—	—
Test												
GEMMA2.9B	0.49	0.65	0.74	0.80	0.40	0.66	0.43	0.60	0.50	0.52	0.26	0.40
LLAMA-3.1-8B	0.49	0.61	0.71	0.77	0.44	0.63	0.43	0.58	0.50	0.49	0.27	0.36
MISTRAL-7B-V0.3	0.49	0.62	0.72	0.77	0.47	0.66	0.41	0.58	0.51	0.50	0.26	0.35
GEMMA2.2B	0.44	0.57	0.67	0.72	0.39	0.61	0.39	0.56	0.48	0.49	0.24	0.33
GEMMA2.27B	0.48	0.65	0.76	0.81	0.41	0.66	0.42	0.61	0.51	0.53	0.26	0.39

Table 10: Performance of target QA models (with $|R| = 5$) on the development and test sets. We report token- and model-based accuracy (Acc and AccLM). AccLM is computed by Qwen2-72B-Instruct.

	NQ	TQA	WebQ AUROC	SQuAD	PopQA	NQ	TQA	WebQ AURAC	SQuAD	PopQA
GEMMA2-9B										
PPL	0.61	0.52	0.58	0.66	0.56	0.67	0.78	0.67	0.65	0.52
MSP	0.64	0.60	0.64	0.71	0.61	0.69	0.80	0.69	0.67	0.56
PMI	0.53	0.46	0.52	0.50	0.48	0.64	0.75	0.64	0.57	0.50
p(true)	0.70	0.71	0.66	0.73	0.83	0.72	0.84	0.70	0.69	0.71
Regular Entropy	0.64	0.54	0.60	0.70	0.58	0.69	0.78	0.68	0.67	0.54
Cluster Assignment	0.68	0.65	0.65	0.70	0.68	0.71	0.82	0.70	0.67	0.60
Semantic Entropy	0.67	0.69	0.64	0.72	0.69	0.71	0.84	0.69	0.68	0.61
Avg.Ans.Len	0.61	0.64	0.65	0.63	0.68	0.68	0.83	0.71	0.65	0.61
Passage Utility	**0.71**	**0.83**	**0.71**	**0.83**	**0.86**	**0.74**	**0.88**	**0.75**	**0.76**	**0.72**
LLAMA-3.1-8B										
PPL	0.75	0.78	0.68	0.75	0.81	0.76	0.85	0.71	0.71	0.68
MSP	0.77	0.80	0.71	0.76	**0.85**	0.76	0.85	0.72	0.72	0.70
PMI	0.55	0.52	0.48	0.54	0.58	0.64	0.73	0.60	0.61	0.53
p(true)	**0.80**	**0.86**	0.73	**0.82**	**0.85**	**0.78**	**0.87**	0.75	**0.75**	**0.71**
Regular Entropy	0.77	0.80	0.69	0.76	0.83	0.76	0.85	0.71	0.72	0.69
Cluster Assignment	0.75	0.83	0.69	0.75	0.82	0.75	0.85	0.71	0.71	0.67
Semantic Entropy	0.74	0.83	0.69	0.74	0.81	0.75	0.86	0.71	0.71	0.68
Avg.Ans.Len	0.73	0.73	0.69	0.69	0.84	0.73	0.82	0.71	0.67	0.69
Passage Utility	0.79	0.84	**0.76**	**0.82**	**0.85**	0.77	**0.87**	**0.76**	0.74	0.70
MISTRAL-7B-V0.3										
PPL	0.66	0.70	0.60	0.63	0.66	0.69	0.84	0.72	0.63	0.63
MSP	0.70	0.75	0.65	0.71	0.77	0.70	0.85	0.73	0.68	0.67
PMI	0.38	0.33	0.42	0.42	0.30	0.53	0.68	0.62	0.52	0.39
p(true)	0.72	**0.82**	0.71	0.75	0.74	0.71	**0.87**	0.76	0.71	0.64
Regular Entropy	0.67	0.71	0.63	0.66	0.68	0.69	0.85	0.73	0.66	0.63
Cluster Assignment	0.72	0.81	0.68	0.73	0.76	0.71	**0.87**	0.75	0.68	0.66
Semantic Entropy	0.72	0.80	0.68	0.73	0.76	0.71	**0.87**	0.76	0.69	0.66
Avg.Ans.Len	0.66	0.75	0.65	0.68	0.81	0.69	0.85	0.73	0.67	0.70
Passage Utility	**0.77**	0.81	**0.74**	**0.83**	**0.84**	**0.74**	**0.87**	**0.79**	**0.74**	**0.71**

Table 11: Answer uncertainty estimation for QA models GEMMA2-9B, LLAMA-3.1-8B, and MISTRAL-7B-V0.3 on NQ, TQA, WebQ, SQuAD, and PopQA development sets. We report AUROC and AURAC.

Question: Due to increased unemployment, who mainly opposed the Shah's regime?
Generated Answer: Millions of youth and poor migrants.
Gold Answer: Millions of youth who had migrated to the cities.

Youth who migrated to cities for construction jobs.

Iran. unemployment, especially among millions of youth who had migrated to the cities of Iran looking for construction jobs during the boom years of the early 1970s. By the late 1970s, many of these people opposed the Shah's regime and began to organize and join the protests against it. The 1979 Revolution, later known as the "Islamic Revolution", began in January 1978 with the first major demonstrations against the Shah. [cont.] 3.93

Unemployed and the poor.

Ruhollah Khomeini. unemployment, ideological disagreement over the economy, and "international pressure and isolation" such as US sanctions following the hostage crisis. Due to the Iran-Iraq War, poverty is said to have risen by nearly 45% during the first 6 years of Khomeini's rule. Emigration from Iran also developed, reportedly for the first time in the country's history. Since the revolution and war with Iraq, an estimated "two to four million entrepreneurs, professionals, technicians, and skilled craftspeople (and their capital)" have emigrated to other countries. [cont.] -1.37

The National Front.

Mohammad Reza Pahlavi. professors issued a public statement criticising the 1953 coup, all were dismissed from their jobs, but in the first of his many acts of "magnanimity" towards the National Front, Mohammad Reza intervened to have them reinstated. Mohammad Reza tried very hard to co-opt the supporters of the National Front by adopting some of their rhetoric and addressing their concerns, for example declaring in several speeches his concerns about the Third World economic conditions and poverty which prevailed in Iran, a matter that had not much interested him before. [cont.] -3.48

['Migrants to the cities.'], ['Millions of youth who had migrated to cities.', 'Millions of youth who migrated to cities.'], ['Cultural and religious conservatives, and the urban poor who had migrated to cities for jobs.'], ['Youth who had migrated to the cities.'], ['Millions of young migrants who had moved to cities in the early 1970s.'], ['Millions of youth and poor migrants to cities.'], ['Cultural and religious conservatives, mostly urban migrants.', 'Cultural and religious conservatives, particularly the migrants to cities.', 'Cultural and religious conservatives, especially those recently migrated to the cities.'] 1.83

Table 12: True negative example (from SQuAD development set): Passage Utility predicts the right answer as well as the QA model (LLAMA-3.1-8B).

Question: Which company was targeted by the NAACP for not having fair practices?
Generated Answer: Target Corporation.
Gold Answer: Lockheed Aircraft Corporation.

Target Corporation.

Target Corporation. of Colored People has repeatedly given Target failing grades on its annual Economic Reciprocity Initiative report card, a measure of the company's "commitment to the African-American citizenry". In 2003 and 2005, the NAACP has rated Target an "F" on this report; in 2004, Target was rated a "D-". In 2006, when Target was asked why it didn't participate in the survey again, a representative explained, "Target views diversity as being inclusive of all people from all different backgrounds, not just one group." In February 2006, the National Federation of the Blind (NFB) filed a class action [cont.] -3.59

None, the NAACP was involved in the Duke lacrosse case.

Reactions to the Duke lacrosse case. formed an opinion on the case. North Carolina NAACP Legal Redress Chair, Al McSurely, explained that "The NAACP stands for fair play for all parties, zealous investigation and deep concern for the survivors of racist/sexist attacks." At the same time, some have criticized the NAACP for making statements that portrayed the players as racist despite evidence to the contrary, using the case to promote the group's cause, and implying guilt. McSurely stated that "[w]ithin five minutes, the men threatened the women with racial and misogynist verbal assaults, [cont.] -3.67

Philadelphia Transit Company (PTC).

Philadelphia transit strike of 1944. (PRTEU), Frank Carney, proved to be equally reticent and claimed that he was not authorized by the union members to consider a request to allow promotions of black employees. The black PTC employees enlisted the help of the NAACP and started lobbying the federal authorities, particularly the Fair Employment Practices Commission (FEPC), to intervene. The Fair Employment Practices Commission, created by an executive order of the President in 1941, was charged with ensuring non-discrimination employment practices by government contractors. [cont.] -3.49

['Target.'], ['Target Corporation.', 'Target Corporation.', 'Target Corporation.', 'Target Corporation.', 'Target Corporation.', 'Target Corporation.', 'Target Corporation.', 'Target Corporation.'], ['Target Corporation and the National Federation of the Blind filed a lawsuit against Michigan.'] 0.33

Table 13: True positive example from (SQuAD development set): Passage Utility predicts the answer provided by the QA model (LLAMA-3.1-8B) which is incorrect.

Question: Close Encounters of the Third Kind Oscar nominations?
Generated Answer: Eight Oscar nominations.
Gold Answer: Eight Oscars.

Sound effects editing.

Close Encounters of the Third Kind. although the Academy honored the film's sound effects editing with a Special Achievement Award (Frank Warner). At the 32nd British Academy Film Awards, "Close Encounters" won Best Production Design, and was nominated for Best Film, Direction, Screenplay, Actor in a Supporting Role (François Truffaut), Music, Cinematography, Editing, and Sound. "Close Encounters" lost the Hugo Award for Best Dramatic Presentation to "Star Wars" [cont.] -1.86

Eight Oscar nominations.

Close Encounters of the Third Kind. Close Encounters of the Third Kind Close Encounters of the Third Kind is a 1977 American science fiction film written and directed by Steven Spielberg, and starring Richard Dreyfuss, Melinda Dillon, Teri Garr, Bob Balaban, Cary Guffey, and François Truffaut. It tells the story of Roy Neary, an everyday blue-collar worker in Indiana, whose life changes after an encounter with an unidentified flying object (UFO). [...] In late 1973, he developed a deal with Columbia Pictures for a science fiction film. Though Spielberg received sole credit for the script. [cont.] -1.58

Eight Oscar nominations.

Close Encounters of the Third Kind. in 2017, in tribute to its 40th anniversary, the movie was given a 4K restoration of the original camera negative. Following its theatrical re-release of the director's cut, the movie was released in 4K and Blu-ray with all three versions given the same 4K treatment. The film was nominated for 8 Oscars at the 50th Academy Awards, including Best Director, Supporting Actress (Melinda Dillon), Visual Effects, Art Direction (Joe Alves, Daniel A. Lomino, Phil Abramson), Original Music Score, Film Editing, and Sound (Robert Knudson, Robert Glass, Don MacDougall and Gene Cantamessa). The film's only win was for Vilmos Zsigmond's cinematography [cont.] -1.48

['Eight nominations.', 'Eight Oscar nominations.', 'Eight Oscar nominations.'], ['Eight nominations, one win.', 'Eight nominations, one win.'], ['Eight, including Best Director.'], ['Eight nominations at the 50th Academy Awards.', '8 nominations at the 50th Academy Awards.'], ['8 nomin ations, 1 win for cinematography.', 'Eight Oscars, won for cinematography.'] 1.56

Table 14: False positive example from (NQ development set): Passage Utility predicts a wrong answer whereas the prediction of the QA model (GEMMA2-9B) is correct.

Question: What magazine, with its iconic yellow border, was first published on Sept 22, 1888?
Generated Answer: The Yellow Book.
Gold Answer: National Geographic.

The New Yorker.

Time (magazine). plays, musicals, television programs, and literary bestsellers similar to "The New Yorker"'s "Current Events" section. "Time" is also known for its signature red border, first introduced in 1927. The border has only been changed five times since 1927: The issue released shortly after the September 11 attacks on the United States featured a black border to symbolize mourning. However, this edition was a special "extra" edition published quickly for the breaking news of the event; the next regularly scheduled issue contained the red border. Additionally, the April 28, 2008, Earth Day issue, dedicated to environmental issues, contained a green border. [cont.] 1.92

The Yellow Book.

The Yellow Book. The Yellow Book The Yellow Book was a British quarterly literary periodical that was published in London from 1894 to 1897. It was published at The Bodley Head Publishing House by Elkin Mathews and John Lane, and later by John Lane alone, and edited by the American Henry Harland. The periodical was priced at 5 shillings and lent its name to the "Yellow Nineties", referring to the decade of its operation. It was a leading journal of the British 1890s; [cont.] 0.35

The Crisis.

The Colored American Magazine. The Colored American Magazine The Colored American Magazine was the first American monthly publication that covered African-American culture. The magazine ran from May 1900 to November 1909. It was initially published out of Boston by the Colored Co-Operative Publishing Company, and from 1904, forward, by Moore Publishing and Printing Company of New York. Pauline Hopkins, its most prolific writer from the beginning, sat on the board as a shareholder, was editor from 1902 to 1904, though her name was not on the masthead until 1903. [cont.] -1.27

['The Yellow Book', 'The Yellow Book', 'The Yellow Book', 'The Yellow Book', 'The Yellow Book', 'Time magazine.', 'The Yellow Book', 'The Yellow Book', 'The Yellow Book'], ['Time magazine'] 0.50

Table 15: False negative (from TQA development set): Passage Utility predicts a correct answer, and the answer by the QA model (GEMMA2-9B is wrong.