



# A Survey on Uncertainty Quantification of Large Language Models: Taxonomy, Open Research Challenges, and Future Directions

OLA SHORINWA, Princeton University, Princeton, United States

ZHITING MEI, Princeton University, Princeton, United States

JUSTIN LIDARD, Princeton University, Princeton, United States

ALLEN Z. REN, Princeton University, Princeton, United States

ANIRUDHA MAJUMDAR, Princeton University, Princeton, United States

The remarkable performance of large language models (LLMs) in content generation, coding, and common-sense reasoning has spurred widespread integration into many facets of society. However, integration of LLMs raises valid questions on their reliability and trustworthiness, given their propensity to generate hallucinations: plausible, factually-incorrect responses, which are expressed with striking confidence. Previous work has shown that hallucinations and other non-factual responses generated by LLMs can be detected by examining the uncertainty of the LLM in its response to the pertinent prompt, driving significant research efforts devoted to quantifying the uncertainty of LLMs. This survey seeks to provide an extensive review of existing uncertainty quantification methods for LLMs, identifying their salient features, along with their strengths and weaknesses. We present existing methods within a relevant taxonomy, unifying ostensibly disparate methods to aid understanding of the state of the art. Furthermore, we highlight applications of uncertainty quantification methods for LLMs, spanning chatbot and textual applications to embodied artificial intelligence applications in robotics. We conclude with open research challenges in uncertainty quantification of LLMs, seeking to motivate future research.

CCS Concepts: • **Computing methodologies** → **Artificial intelligence**; **Natural language processing**; **Natural language generation**;

Additional Key Words and Phrases: Uncertainty Quantification; Large Language Models (LLMs); Confidence Estimation.

## 1 Introduction

Large language models have demonstrated remarkable language generation capabilities, surpassing average human performance on many benchmarks including math, reasoning, and coding [1, 7, 23, 34, 47, 206]. For example, recent (multi-modal) large language models were shown to achieve impressive scores, e.g., in the 90% percentile, on simulated Law School Admission Test (LSAT) exams, the American Mathematics Competition (AMC) contests, the Multistate Bar Exam, and the Graduate Record Exam (GRE) General Test, outperforming a majority of test takers [1, 7, 98]. Likewise, LLMs have advanced the state of the art in machine translation, text summarization, and question-and-answer tasks. However, LLMs also tend to produce plausible, factually-incorrect responses to their input prompts, termed *hallucinations* [113]. In some scenarios, the hallucinated response is overtly incorrect; however, in many cases, the factuality of the LLM response is harder to discern, posing significant risk as a user might falsely assume factuality of the response, which can result in devastating

Authors' Contact Information: Ola Shorinwa, Princeton University, Princeton, New Jersey, United States; e-mail: shoa@princeton.edu; Zhiting Mei, Princeton University, Princeton, New Jersey, United States; e-mail: maymei@princeton.edu; Justin Lidard, Princeton University, Princeton, New Jersey, United States; e-mail: jlidard@princeton.edu; Allen Z. Ren, Princeton University, Princeton, New Jersey, United States; e-mail: allen.ren@princeton.edu; Anirudha Majumdar, Princeton University, Princeton, New Jersey, United States; e-mail: ani.majumdar@princeton.edu.



This work is licensed under a Creative Commons Attribution International 4.0 License.

© 2025 Copyright held by the owner/author(s).

ACM 1557-7341/2025/6-ART

<https://doi.org/10.1145/3744238>

consequences, especially when safety is of paramount importance. As a result, hallucinations pose a notable danger to the safe, widespread adoption of LLMs.

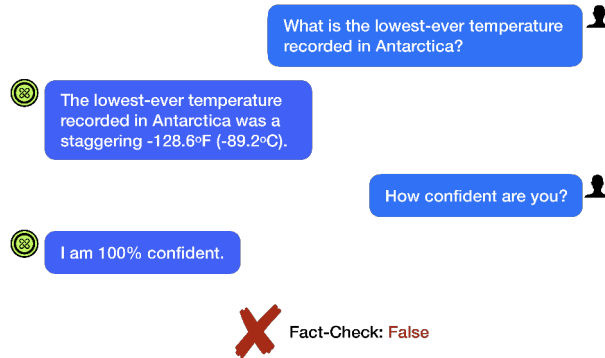


Fig. 1. A user asks an LLM the question: *What is the lowest-ever temperature recorded in Antarctica?*; in response, the LLM answers definitively. Afterwards, the user asks the LLM how confident the LLM is. Although the LLM states that it is “100% confident,” the LLM’s response fails to pass a fact-check test. Confidence scores provided by LLMs are generally miscalibrated. UQ methods seek to provide calibrated estimates of the confidence of LLMs in their interaction with users.

To ensure the trustworthiness of LLMs, substantial research has been devoted to examining the mechanisms behind hallucinations in LLMs [11, 31, 86, 113, 227], detecting its occurrence, identifying potential causes, and proposing mitigating actions. However, even in the absence of hallucinations, LLMs are susceptible to doubt when given prompts at the boundary of their knowledge base. In these situations, prior work has shown that LLMs fail to accurately convey their uncertainty to a user, either implicitly or explicitly, unlike typical humans [6, 134]. In fact, LLMs tend to be overconfident even when they should be uncertain about the factuality of their response [66, 225]. We provide an example in Figure 1, where an LLM is asked: “What is the lowest-ever temperature recorded in Antarctica?”, to which the LLM responds definitively. Even when prompted for its confidence in its answer, the LLM claims that it is “100% confident.” However, the LLM’s answer fails to pass a fact-check test. Knowing how much to trust an LLM-generated response is critical for users [101], helping inform the development of contingency strategies commensurate with the degree of uncertainty of the LLM in its response. For example, in applications such as robotics, an LLM-equipped robot could seek human guidance [178] or necessitate further review in the judicial practice [42]. Uncertainty quantification (UQ) methods for LLMs seek to address this challenge by providing users with an estimate of an LLM’s confidence in its response to a given prompt. Indeed, uncertainty quantification can be important in factuality analysis [84].

The rapid adoption of LLMs in many applications has contributed to the fast-paced development of UQ methods for LLMs to promote their safe integration into a wide range of applications. However, the huge volume of UQ methods for LLMs has made it particularly challenging to ascertain the research scope and guarantees provided by existing UQ methods, complicating the identification of useful UQ methods for practitioners seeking to leverage them in their application areas, as well as the identification of impactful future directions for research. We claim that this challenge arises from the lack of a taxonomy that unifies related existing methods and presents an organized view of existing work in this research area.

Through this survey, we seek not only to enumerate existing work in UQ for LLMs, but also to provide a useful taxonomy of UQ methods for LLMs to aid understanding the state of the art in this research area. We reiterate that the introduction of an effective taxonomy for these methods can facilitate their adoption in wide-ranging applications, such as in factuality analysis, hallucination detection, and robotics. We categorize existing

uncertainty quantification methods for LLMs into four main classes: (1) token-level uncertainty quantification methods; (2) self-verbalized uncertainty quantification methods; (3) semantic-similarity uncertainty quantification methods; and (4) mechanistic interpretability methods. These categories encompass uncertainty quantification of multi-claim, multi-sentence LLM responses. We elaborate on each category in this survey, identifying the key features shared by methods within each category. Moreover, we identify open research challenges and provide directions for future research, hoping to inspire future effort in advancing the state of the art.

### Comparison to other Surveys

A number of surveys on hallucinations in LLMs exists, e.g., [14, 82, 128, 173, 205]. These surveys discuss hallucinations in detail, introducing the notion of hallucinations [173], identifying its types and potential causes [82], and presenting mitigation techniques [205]. However, these papers provide little to no discussion on uncertainty quantification methods for LLMs, as this research area lies outside the scope of these surveys. In contrast, only two surveys on uncertainty quantification methods for LLMs exist, to the best of our knowledge. The first survey [61] categorizes confidence estimation and calibration methods into two broad classes: methods for generation tasks and methods for classification tasks, defined by the application domain. The survey in [61] focuses more heavily on calibration methods, with a less extensive discussion on confidence estimation methods. In contrast, our paper provides an extensive survey of uncertainty quantification methods with a brief discussion on calibration of uncertainty estimates. For example, whereas [61] lacks a detailed discussion on the emerging field of mechanistic interpretability, our survey presents this field in detail, along with potential applications to uncertainty quantification. Moreover, our survey discusses a broad range of applications of uncertainty quantification methods for LLMs, e.g., embodied applications such as in robotics, beyond those discussed in [61]. A concurrent survey [81] on uncertainty quantification of LLMs categorizes existing uncertainty quantification methods within more traditional classes, which do not consider the unique architecture and characteristics of LLMs. In contrast, our survey categorizes existing work within the lens of LLMs, considering the underlying transformer architecture of LLMs and the autoregressive token-based procedure utilized in language generation.

### Organization

In Section 2, we begin with a review of essential concepts that are necessary for understanding the salient components of uncertainty quantification of LLMs. We discuss the general notion of uncertainty and introduce the main categories of uncertainty quantification methods within the broader field of deep learning. Subsequently, we identify the relevant metrics utilized by a majority of uncertainty quantification methods for LLMs. In Sections 3, 4, 5, and 6, we discuss the four main categories of uncertainty quantification methods for LLMs, highlighting the key ideas leveraged by the methods in each category. In Section 7, we provide a brief discussion of calibration techniques for uncertainty quantification, with applications to uncertainty quantification of LLMs. In Section 8, we summarize the existing datasets and benchmarks for uncertainty quantification of LLMs and present applications of uncertainty quantification methods for LLMs in Section 9. We highlight open challenges in Section 10 and suggest directions for future research. Lastly, we provide concluding remarks in Section 11. Figure 2 summarizes the organization of this survey, highlighting the key details presented therein.

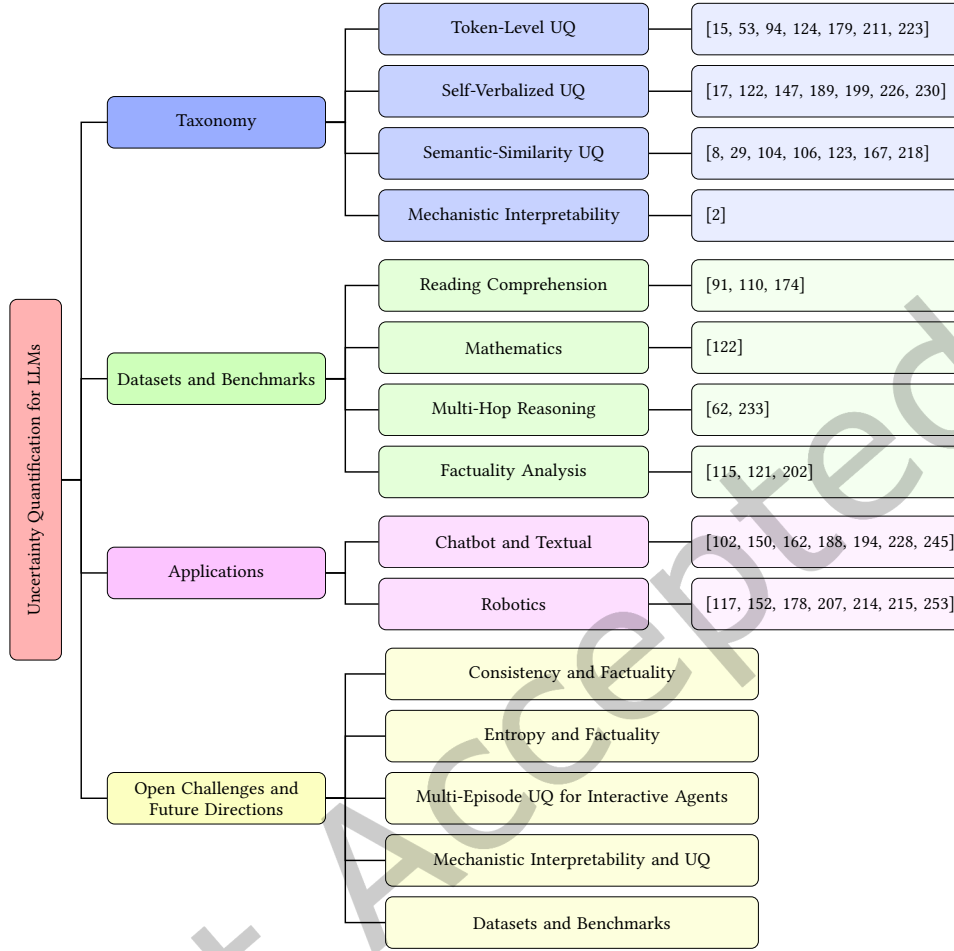


Fig. 2. The overview of this survey, including a taxonomy of uncertainty quantification methods for LLMs, relevant datasets and benchmarks, applications, and open challenges and directions for future research.

## 2 Background

We review fundamental concepts that are crucial to understanding uncertainty quantification of LLMs. We assume basic familiarity with deep learning and build upon this foundation to introduce more specific concepts, describing the notion of uncertainty, the inner workings of LLMs, and the development of metrics and probes to illuminate the uncertainty of LLMs in their response to a user's prompt.

### 2.1 Uncertainty

Uncertainty is a widely-known, yet vaguely-defined concept. For example, people generally associate uncertainty with doubt or a lack of understanding, knowledge, or control, but cannot generally provide a precise definition, especially a mathematical one. This general ambiguity applies to the field of LLMs [99]. For example, a subset of the LLM research field considers the uncertainty of a model to be distinct from its level of confidence in a response generated by the model [123], stating that confidence scores are associated with a prompt (input) and a

prediction by the model, whereas uncertainty is independent of the model's prediction. However, a large subset of the field considers uncertainty and the lack of confidence to be mostly-related, generally-interchangeable concepts. In this section, for simplicity, we consider uncertainty and confidence to be mostly interchangeable.

When prompted, LLMs tend to hallucinate when uncertainty about the correct answer exists, e.g., when a lack of understanding or a lack of knowledge exists (see Figures 3 and 4). In Figures 3 and 4, we ask GPT-4o mini to name the best cooking book written by a (likely) fictional person Jamie Feldman. GPT-4o mini provides a confident response: "The Ultimate Guide to Cooking for One." However, based on an internet search, this cookbook does not exist (although many similar ones do). Moreover, when prompted about its confidence, GPT-4o mini apologizes before providing yet another confident, but factually-incorrect response: "The Jewish Cookbook." This book is authored by *Leah Koenig*, not Jamie Feldman. Uncertainty quantification (UQ) methods aim to provide a more rigorous estimate of the model's confidence in its response, e.g., from the entropy of the distribution from which the tokens are sampled. Before discussing UQ techniques for LLMs, we identify the types of uncertainty and the methods suitable for characterizing uncertainty in deep-learned models, more broadly.

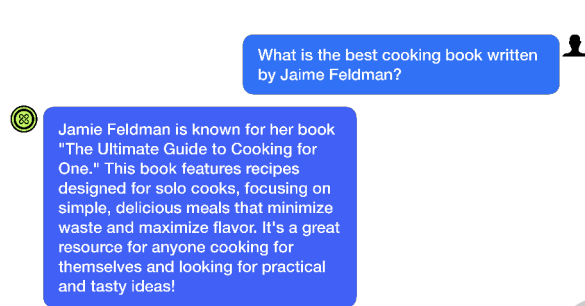


Fig. 3. Hallucination in LLMs: When asked for information about a possibly fictional person, LLMs tend to fabricate a response that sounds coherent but is entirely false.

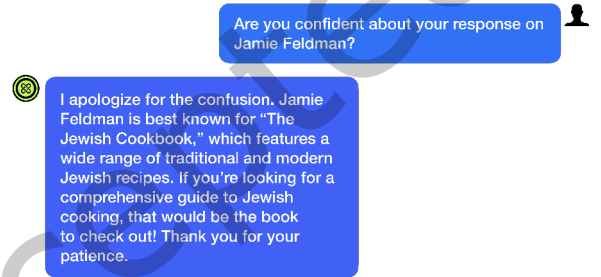


Fig. 4. Hallucination in LLMs: When asked about its confidence, the LLM apologizes before hallucinating another response. The Jewish Cookbook is authored by *Leah Koenig*, not Jaime Feldman.

## 2.2 Types of Uncertainty

Uncertainty can be broadly categorized into two classes, namely: *aleatoric* uncertainty and *epistemic* uncertainty. When considered collectively, the resulting uncertainty is referred to as *predictive* uncertainty, without a distinction between the two components.

**2.2.1 Aleatoric Uncertainty.** Aleatoric uncertainty encompasses the lack of definiteness of the outcome of an event due to the inherent randomness in the process which determines the outcome of the event. For example, a model cannot predict with certainty the outcome of an unbiased coin toss due to the random effects in the coin toss, regardless of the complexity of the model or the size of the training dataset used in training the model. This irreducible uncertainty is referred to as aleatoric uncertainty. For example, in the case of LLMs, aleatoric uncertainty can arise when there is inherent randomness in the ground-truth response, e.g., when prompted with "What will the temperature be tomorrow?", the uncertainty associated with the LLM's output can be characterized as aleatoric uncertainty, which is entirely due to the random effects associated with daily weather conditions. In essence, daily weather conditions cannot be predicted with absolute certainty, irrespective of the amount of training data available.

**2.2.2 Epistemic Uncertainty.** In contrast to aleatoric uncertainty, epistemic uncertainty characterizes the doubt associated with a certain outcome (prediction) due to a lack of knowledge or "ignorance" by a model, often

due to limited training data. For example, when prompted to provide the digit in the 7th decimal place in the square-root of 2, GPT-4o mini responds with the answer 6. However, this answer is wrong: the digit in the 7th decimal place is 5. The uncertainty in the LLM's output can be characterized as epistemic uncertainty, which can be eliminated by training the LLM on more data specific to this prompt. In other words, epistemic uncertainty describes reducible uncertainty, i.e., epistemic uncertainty should reduce when there is more knowledge about the state on which the decision is being made, e.g., via choosing the right model to use for learning, using more training data, or by incorporating any prior knowledge. The uncertainty associated with the response in Figure 3 is entirely epistemic and stems from missing training data. If we train the LLM on more data, including the fact that Jamie Feldman did not write a cookbook, we can eliminate the uncertainty associated with the model's response. Before concluding, we note that prior work has explored decomposing predictive uncertainty into epistemic and aleatoric components [78].

### 2.3 Uncertainty Quantification in Deep Learning

Broadly, uncertainty quantification for deep learning lies along a spectrum between two extremes: *training-based* and *training-free* methods, illustrated in Figure 5. Whereas training-based methods assume partial or complete visibility and access to the internal structure of the neural network, modifying it to probe its uncertainty, training-free methods use auxiliary models or additional data to quantify the uncertainty of the model post-hoc.

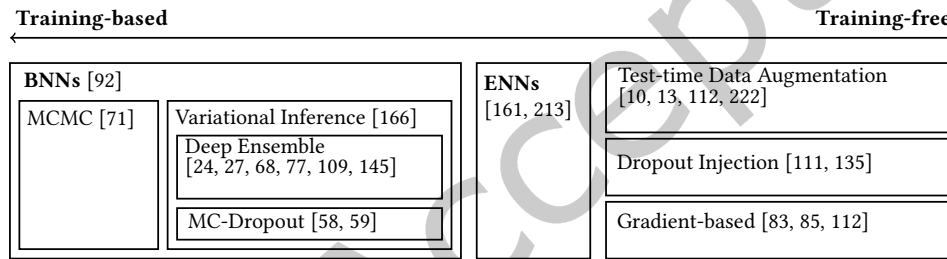


Fig. 5. Uncertainty quantification methods in deep learning span the spectrum from training-based methods to training-free methods.

**2.3.1 Training-Based Methods.** Training-based uncertainty quantification methods span Bayesian Neural Networks, Monte Carlo Dropout methods, and Deep Ensembles, which we review in the subsequent discussion. Instead of training a set of parameters to predict a single outcome, a Bayesian neural network (BNN) [92] learns a distribution over the model's weights  $\theta$ . Specifically, a BNN learns a distribution over the parameters,  $p(\theta|D)$ , with dataset  $D$ , with its prediction consisting of two parts: a maximum a posteriori estimation component  $\hat{y}$ , and the uncertainty associated with it, defined by the covariance of the prediction  $\Sigma_{\hat{y}|x,D}$ .

Despite being statistically principled, the prohibitive computational costs associated with BNNs prevent them from being directly employed. In order to train BNNs, a variety of methods have been proposed, among which the most popular ones are Markov Chain Monte-Carlo (MCMC) [71] and variational inference [166]. The former samples from the exact posterior distribution, while the latter learns to approximate the posterior with a variational distribution,  $q_{\phi}$ . Due to the relaxed requirement of access to large amounts of samples, the variational inference method has been more widely used, with Monte-Carlo dropout [58, 59] and Deep ensemble [109] being representative methods. More recently, epistemic neural networks (ENNs) [161, 213] have been introduced to reduce the computational challenges associated with BNNs. To make ensemble methods more efficient, e.g., in out-of-distribution detection [212], pruning methods [27, 68, 145], which reduce redundancy among ensemble members, and distillation methods [24, 77], which reduce the number of networks to one, teaching it to represent the knowledge of a group of networks, have been introduced. While these methods are easy to

implement and require much less computation compared to regular BNNs or MCMC, they do suffer from being an approximation of the true posterior distribution. In fact, the model’s uncertainty predictions could be worse when data augmentation, ensembling, and post-processing calibration are used together [171].

**2.3.2 Training-Free Methods.** Training-free methods for estimating uncertainty have become popular due to their ease of implementation. Since neither the network architecture nor the training process need to be revised, training-free methods work well with large-scale foundation models that are costly to train or fine-tune. In [10, 13, 112, 222], the authors perform data augmentation at test time to generate a predictive distribution, quantifying the model’s uncertainty. Similarly, dropout injection [111, 135] extends MC-dropout to the training-free domain by only performing dropout at inference time to estimate epistemic uncertainty. In [146], the authors estimate uncertainty for regression using similar perturbation techniques. Lastly, gradient-based uncertainty quantification methods [112] generate gradients at test-time, which provide an signal for epistemic uncertainty and for OOD detection in [83, 85], by constructing confounding labels.

## 2.4 Uncertainty Quantification for LLMs

The introduction of the transformer [210] for sequence-to-sequence machine translation tasks spurred the development of large language models. However, as noted in the preceding discussion, LLMs have gained some notoriety for their tendency to hallucinate when uncertain about a response to a specified prompt. Here, we review the general architecture of LLMs and provide some motivation for the development of LLM-specific metrics for quantifying uncertainty.

**2.4.1 LLM Architecture.** LLMs use the transformer architecture to provide free-form responses to input prompts specified in natural language. The transformer architecture consists of an encoder, which processes the input to the model, and a decoder, which generates the model’s outputs auto-regressively, where the previous outputs of the model are passed into the model to generate the future outputs. Given an input prompt, the words (elements) of the prompt are tokenized (i.e., the sentences/phrases in natural-language are decomposed into simple units referred to as tokens) and transformed to input embeddings using a learned model. The encoder takes in the input embeddings augmented with positional encodings to incorporate positional context and generates a sequence of latent embeddings, which serves as an input to the decoder, using a stack of  $N$  multi-head attention sub-blocks and fully-connected feedforward networks. The decoder takes in the embeddings associated with the previous outputs of the decoder, preceded by a *start* token, and computes an output embedding using a similar stack of multi-head attention heads and feedforward networks as the encoder. The resulting output embeddings are passed into a linear layer prior to a softmax output layer, which converts the decoder embeddings to a probability distribution over the tokens in the dictionary of the model. In subsequent discussion, we denote the probability of the  $j$ ’th token in the  $i$ ’th sentence of an LLM’s output as  $p_{ij}$ . The output token is selected from this probability distribution: e.g., by greedily taking the token associated with the maximum probability mass. The resulting output is passed into the decoder for auto-regressive generation of text.

While early LLM models utilized encoder-only or encoder-decoder transformer architectures, state-of-the-art LLMs now generally utilize a decoder-only architecture. For example, the GPT family of models, such as GPT-4 [1], and the Llama family of models, such as Llama 3 [47], are decoder-only transformers. In Figure 6, we show a decoder-only transformer model. These state-of-the-art models leverage advances in transformers to improve computational efficiency, given the huge size of these models: Llama 3 has 8B parameters for the small variant and 70B parameters for the large variant, while GPT-4 is rumored to have over one trillion parameters. Llama 3 uses rotary positional embeddings (RoPE) [190] instead of absolute positional embeddings, which have been shown to be more effective than alternative embedding schemes. For a more detailed review of LLMs, we refer

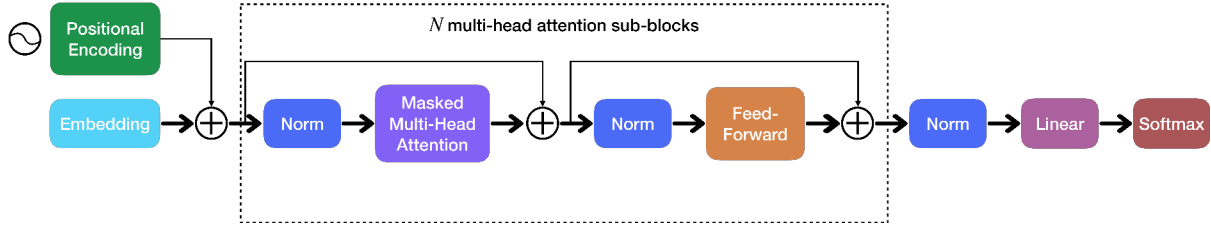


Fig. 6. Many state-of-the-art LLMs are decoder-only transformers, with  $N$  multi-head attention sub-blocks, for auto-regressive output generation.

readers to [149]. Before presenting the metrics utilized by UQ methods for LLMs, we discuss natural-language inference, which is an important component of many UQ methods for LLMs.

**2.4.2 Natural-Language Inference.** Natural-language inference (NLI) refers to the task of characterizing the relationship between two text fragments, where one text fragment represents a premise (i.e., a statement that is believed to be true) while the other fragment represents a hypothesis (i.e., a statement whose veracity we seek to evaluate based on the premise) [40, 57, 221]. Given a premise and a hypothesis, we can classify the relation between the text pair as: an *entailment*, if one can infer that the hypothesis is most likely true given the premise; a *contradiction*, if one can infer that the hypothesis is most likely false given the premise; or a *neutral* label, if one cannot infer the truthfulness of the hypothesis from the premise [36, 137, 151]. In Figure 7, we provide some examples of text pairs that exhibit entailment, contradiction, or neutrality. In the first example, the premise indicates that the student *presented* a research paper at a conference (i.e., the student *did not* skip the conference), hence, the contradiction. In the second example, the premise indicates that the orchestra enjoyed the concert, but does *not* state whether the orchestra performed at the concert (or just observed the event), hence the neutral label. In the third example, we can infer that the hypothesis is true, since the premise indicates that the team was on vacation, hence, *not* in the office.

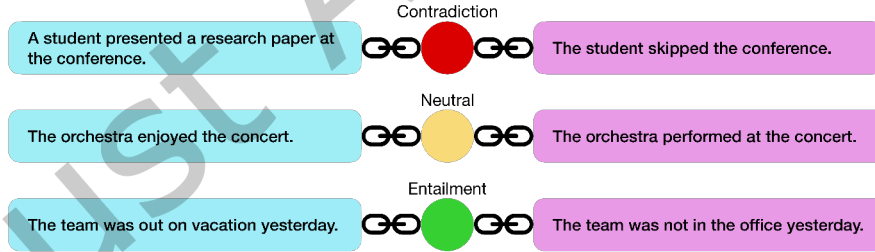


Fig. 7. Natural-language inference models characterize the relationship between a pair of texts, namely: a premise and a hypothesis. The possible relations include: (1) an entailment where the hypothesis can be inferred from the premise; (2) a contradiction, where the hypothesis is more likely false given the premise; and (3) a neutral relation, where the veracity of the hypothesis cannot be determined from the premise.

NLI methods play an important role in uncertainty quantification of LLMs. Many UQ methods for LLMs rely on characterization of the semantic relationship between multiple realizations of the LLM's responses to a given input prompt to determine the confidence of the model. Many of these methods rely on learned models for natural-language inference, such as BERT [45], which utilizes a transformer-based architecture to learn useful



language representations that are crucial in natural-language tasks such as question answering and natural-language inference. Unlike many standard language models, e.g., Generative Pre-trained Transformer (GPT) [170], which impose a unidirectionality constraint where every token can only attend to previous tokens, BERT employs a bidirectional approach where each token can attend to any token regardless of its relative position, using a masked language model, potentially enabling the model to capture broader context, especially in sentence-level tasks. In [132], the authors demonstrate that the performance of BERT is limited by inadequate pre-training and propose an improved model, named RoBERTa [132], which retains the same architecture as BERT but is trained for longer with larger mini-batches of data with longer sequences. DeBERTa [74] further improves upon the performance of RoBERTa by introducing a disentangled attention mechanism and an enhanced mask decoder.

**2.4.3 Metrics for Uncertainty Quantification for LLMs.** Uncertainty quantification in the LLM community has largely eschewed traditional UQ methods for learned models due to the notable computation cost of running inference on LLMs [16], although, a few UQ methods for LLMs utilize deep ensembles, e.g., [9, 16, 216, 243], generally based on low-rank adaptation (LoRA) [79]. Consequently, many UQ methods in this space have introduced less computationally intensive approximate quantification methods that directly harness the unique architecture of LLM models to assess the uncertainty of these models. In some cases, these methods retain the high-level idea of ensemble methods, quantifying the uncertainty of the model on a given prompt using the outputs of a set of individual models or a collection of outputs from the same model, with a temperature parameter less than one to promote greater stochasticity in the tokens generated by the model. UQ methods for LLMs can be broadly categorized into white-box models and black-box models [129, 209], illustrated in Figure 8 and Figure 9, respectively.

**White-Box UQ Methods.** White-box UQ models assume that the underlying architecture of the model is partially or completely visible and accessible—hence the term *white-box*—taking advantage of access to the intermediate outputs of the underlying models, such as the probability distribution over the generated tokens or outputs at the inner layers of the model, to assess the uncertainty of the model [12, 53, 106]. We provide some metrics utilized by white-box UQ methods for LLMs, where  $p_{ij}$  denotes the conditional probability of token  $j$  (conditioned on all preceding tokens) in sentence  $i$ :

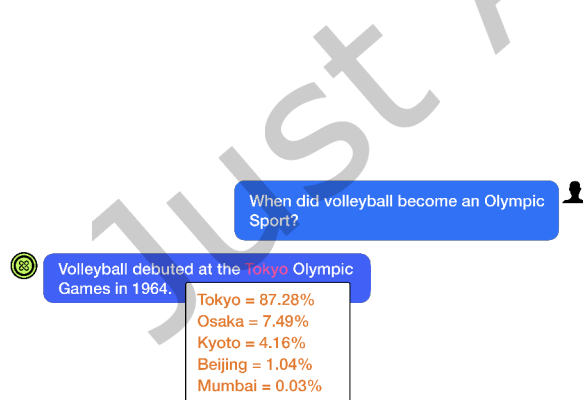


Fig. 8. White-box uncertainty quantification methods utilize an LLM’s internal information, e.g., the model’s probabilities for the token associated with each output.

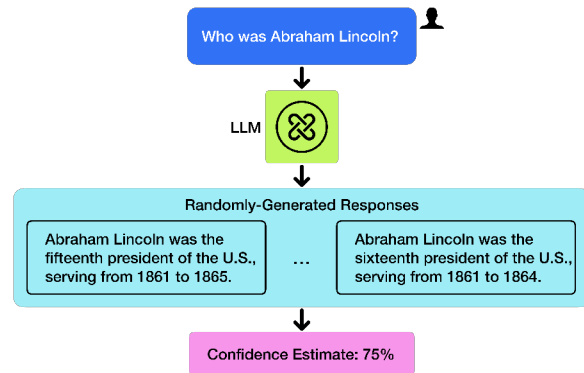


Fig. 9. Black-box uncertainty quantification methods do not access the internal states or probabilities computed by the model, quantifying the model’s uncertainty entirely from its natural-language response.

- (1) Average Token Log-Probability. The average of the negative log-probability of the tokens, which captures the average confidence of the model [141], is given by:  $\text{Average}(p) = -\frac{1}{L_i} \sum_j \log(p_{ij})$ , where sentence  $i$  consists of  $L_i$  tokens. Note that the value of this metric increases as the conditional probability distribution of each token decreases, signifying an decrease in the model's confidence. The average token probability is related to the product of the token probabilities.
- (2) Perplexity. The perplexity of a model's prediction represents the exponential of the average of the negative log-probability of the tokens which comprise the sentence (response) generated by the LLM [53]. Perplexity is given by:  $\text{Perplexity}(p) = \exp\left(-\frac{1}{L_i} \sum_j \log(p_{ij})\right)$ .
- (3) Maximum Token Log-Probability. The maximum token log-probability captures the token with the lowest conditional probability, which is given by:  $\text{Maximum}(p) = \max_j -\log(p_{ij})$ .
- (4) Response Improbability. This metric entails computing the probability of a given sentence given the conditional distribution for each token [53], where the probability distribution is conditioned on preceding tokens, and subtracting the resulting value from one. The uncertainty metric is defined as:  $\text{Improb.} = 1 - \prod_j p_{ij}$ .
- (5) Entropy. The maximum entropy of the probability distribution associated with each token can be utilized as a metric for UQ, given by:  $\text{Entropy} = \max_j \mathcal{H}(p_j)$ , where  $\mathcal{H}$  represents the entropy of the probability distribution  $p_j$  of token  $j$ . Some existing methods claim that this metric is better than the perplexity [53]. Similarly, the predictive entropy [139] at input  $x$  and output  $y$  is defined as:  $\mathcal{H}(Y | x) = -\int p(y | x) \ln p(y | x) dy$ . In the discrete case, the entropy associated with the output distribution of token  $j$  in sentence  $i$  is defined by:  $\mathcal{H}_{ij} = -\sum_{w \in \mathcal{D}} p_{ij}(w) \log p_{ij}(w)$ , where  $\mathcal{D}$  denotes the dictionary containing all possible words in the model and  $w$  represents a word in  $\mathcal{D}$ .

**Black-Box UQ Methods.** In contrast, *black-box* methods assume that the model's internal outputs cannot be accessed externally [29, 141]. Hence, these methods quantify the uncertainty of the model entirely from the model's response to an input prompt. Prior work has discussed the pros and cons of both categories of UQ methods [123]. Concisely, white-box methods generally require access to the underlying architecture and intermediate outputs of an LLM, which is increasingly difficult to obtain given that many LLMs have become closed-source models, posing a significant limitation. In contrast, black-box models enable UQ of closed-source models such as OpenAI's GPT-4 [1] and Anthropic Claude [7], which do not provide complete access to the model. In general, black-box UQ methods for LLMs require the evaluation of the similarity between multiple responses generated by an LLM or an ensemble of LLMs on the same or similar prompts to quantify the uncertainty of the LLM on a given input prompt. Other black-box UQ methods, such as self-verbalized UQ methods, train the model to directly provide a natural-language estimate of its confidence. Here, we identify some prominent techniques for measuring the similarity between a pair of text fragments:

- (1) NLI Scores. As described in Section 2.4.2, NLI models, such as RoBERTa [132] and DeBERTa [74], classify the relationship between a pair of text fragment as either an entailment, a contradiction, or a neutral relation. Many black-box methods utilize the probabilities (or logits) predicted by the NLI model for these three classes as a measure of the similarity between the two text fragments, which is ultimately used to quantify the uncertainty of the LLM. For example, given the probability  $p_{\text{entail}}$  predicted by an NLI model that a text fragment  $t_1$  entails another text fragments  $t_2$ , we can define the similarity between both text fragments as:  $\text{sim}(t_1, t_2) = p_{\text{entail}}$ . Conversely, given the probability of contradiction  $p_{\text{contradict}}$ , we can define the similarity between  $t_1$  and  $t_2$  as:  $\text{sim}(t_1, t_2) = 1 - p_{\text{contradict}}$ .
- (2) Jaccard Index. The Jaccard index, also referred to as Intersection-over-Union measures the similarity between two sets by computing the ratio of the intersection of both sets and the union of both sets. Hence, the Jaccard index  $J$  between two sets  $\mathcal{T}_1$  and  $\mathcal{T}_2$ , where each set consists of the words that make up its associated text fragment, is given by:  $J(\mathcal{T}_1, \mathcal{T}_2) = \frac{|\mathcal{T}_1 \cap \mathcal{T}_2|}{|\mathcal{T}_1 \cup \mathcal{T}_2|}$ . Although the Jaccard index always lies between 0

and 1, making it a suitable metric [37, 164, 169], the Jaccard index does not consider the context of the text fragments, which is important in evaluating the similarity between both text fragments.

- (3) **Sentence-Embedding-Based Similarity.** The similarity between two text fragments can also be determined by computing the cosine-similarity between the sentence embeddings associated with each text fragment. Sentence-embedding models transform natural-language inputs (or tokens) into a vector space, enabling direct computation of the similarity between two sentences (phrases). For example, Sentence-BERT (SBERT) [175] builds upon the pretrained BERT architecture to train a model that computes semantically-relevant sentence embeddings. Other similar models include LaBSE [54] and SONAR [49]. Since the sentence embeddings capture the context of the text fragment, this approach is less susceptible to the challenges faced by the Jaccard index, such as those that arise with negated words.
- (4) **BERTScore.** The BERTScore [244] measures the similarity between two sentences by computing the cosine-similarity between the contextual embedding of each token (word) in the reference sentence  $t_r$  and the contextual embedding of the associated token in the candidate sentence  $t_c$ . The token embeddings are generated from NLI models to capture the context of the sentence. As a result, a given word might have different embeddings, depending on the context of the sentence in which it is used, addressing the challenges faced by the Jaccard similarity metric and word-embedding-based metrics. The BERTScore is composed of a precision  $P_{BS}$ , recall  $R_{BS}$ , and F1  $F_{BS}$  score, given by:

$$P_{BS} = \frac{1}{|t_c|} \sum_{\hat{w}_j \in t_c} \max_{w_i \in t_r} w_i^\top \hat{w}_j, \quad R_{BS} = \frac{1}{|t_r|} \sum_{w_i \in t_r} \max_{\hat{w}_j \in t_c} w_i^\top \hat{w}_j, \quad F_{BS} = 2 \frac{P_{BS} \cdot R_{BS}}{P_{BS} + R_{BS}}, \quad (1)$$

where each token in the candidate sentence is matched to its most similar token in the reference sentence. The BERTScore is obtained by computing the cosine-similarity between matched pairs. Since each token embedding is normalized, the cosine-similarity between a pair of embeddings simplifies to the inner-product. The recall score is related to the ROUGE metric [120] used in evaluating text summaries and more broadly to the BARTScore [237]. However, the ROUGE metric utilizes human-provided summaries as the reference.

In the following sections, we describe the main categories of UQ methods for LLMs in detail, namely: (1) Token-Level UQ Methods; (2) Self-Verbalized UQ Methods; (3) Semantic-Similarity UQ Methods; and (4) Mechanistic Interpretability, outlined in Figure 2. Although mechanistic interpretability has not been widely applied to uncertainty quantification, we believe that insights from mechanistic interpretability can be more extensively applied to the uncertainty quantification of LLMs; hence, we include these methods within our taxonomy.

### 3 Token-Level UQ

We recall that the outputs of an LLM are generated by sampling from a probability distribution over the tokens that make up the outputs, conditioned on the preceding tokens in the outputs (see Section 2.4). Token-level UQ methods leverage the probability distribution over each token to estimate the probability of generating a given response from an LLM. Although a high predicted probability in a particular generation may not be indicative of its correctness over another, direct quantification of the model's generating distribution may lead to better understanding of the stochasticity of generations. Token-level UQ methods utilize the white-box UQ metrics discussed in Section 2.4.3 to estimate the randomness in the probability distribution associated with an LLM's response. For example, some token-level UQ methods compute the entropy of the underlying probability distribution over the tokens [124, 223] or semantic clusters [106] (referred to as semantic entropy) to estimate the LLM's confidence. Likewise, a variant of SelfCheckGPT [141] trains an  $n$ -gram model using multiple samples of the response of an LLM to a given query including its main response. Subsequently, SelfCheckGPT estimates the LLM's uncertainty by computing the average of the log-probabilities of the tokens generated by the  $n$ -gram

model, given the original response of the LLM. Moreover, SelfCheckGPT proposes using the maximum of the negative log-probability to estimate the LLM’s uncertainty.

Token-based UQ methods generally perform poorly with long-form responses, because the product of the token probabilities decrease with longer responses, even when the responses are semantically equivalent to a shorter response. To address this limitation, token-based UQ methods employ a length-normalized scoring function [139, 201], to reduce the dependence of the UQ metrics on the length of the sequence, given by:

$\text{Product}(p) = \prod_{j=1}^{L_i} p_{ij}^{\frac{1}{L_i}}$ , where  $L_i$  denotes the length of sentence  $i$ , and  $p_{ij}$  is the conditional probability of token  $j$ , given all preceding tokens, in sentence  $i$ . The work in [15] introduces Meaning-Aware Response Scoring (MARS) as an alternative to length-normalized scoring. MARS utilizes an importance function to assign weights to each token based on its contribution to the meaning of the sentence. The contribution of each token to the meaning of the sentence is determined using BEM [25], a question-answer evaluation model. Taking a different approach, Claim-Conditioned Probability (CCP) [53] decomposes the output of an LLM into a set of claims and computes the token-level uncertainty of each claim from its constituent tokens. CCP utilizes the OpenAI Chat API [1, 23] to identify the main claims in a given response. By examining the component claims, CCP provides finer-grained uncertainty quantification compared to other UQ methods for LLMs.

As described, token-level UQ methods estimate the uncertainty of an LLM based on the conditional distribution associated with each token. Although this approach is effective in general, the conditional distribution of the tokens can be misleading in certain scenarios, especially when an initial token is incorrect but all the succeeding tokens are highly probable given the initial token. Trainable attention-based dependency (TAD) [211] trains a regression model on the conditional dependence between the tokens and applies the predicted factors to improve the estimated uncertainty of the LLM. Lastly, we present token-level UQ methods that use specific prompting strategies to estimate the model’s confidence. The work in [94] shows that token-based UQ methods can be particularly effective in estimating the confidence of LLMs when the model is prompted to select an option when given a multiple-choice question. Specifically, the authors show that the model’s probability distribution over the options in the prompt is well-calibrated, when presented with multiple-choice problems or problems with a True/False answer. Further, the authors fine-tune an LLM with a value head to predict the probability that the model knows the answer to a given question for each token. The probability associated with the LLM’s final token is defined as the confidence of the LLM in its response for the given prompt. The results demonstrate that the LLM predictions of these probability values are well-calibrated, with an improvement in the calibration performance with larger models. Other follow-on work leveraging multiple-choice problems to estimate the uncertainty of LLMs includes [179].

#### 4 Self-Verbalized UQ

Self-verbalized uncertainty quantification methods seek to harness the impressive learning and reasoning capabilities of LLMs to enable an LLM to express its confidence in a given response through natural-language. Self-verbalized uncertainty estimates (e.g., expressed as probabilities) are more easily interpretable to humans, especially when the estimates are provided using widely-used epistemic uncertainty markers [197, 235], e.g., words like *I am not sure...* or *This response might be...* Figure 10 illustrates the use of epistemic markers by an LLM to convey its uncertainty, when asked of the team that won the 2022 NBA Finals. The response of the LLM is actually incorrect; however, by expressing its uncertainty, a user may be more inclined to verify the factuality of the LLM’s response. Prior work has shown that LLMs typically fail to accurately express their confidence in a given response, often using decisive words that suggest confidence, while at the same time being unsure of the accuracy of their response. Empirical studies [105] have shown that poor calibration of LLM’s self-verbalized confidence estimates is more pronounced in low-data language settings, e.g., Hindi and Amharic.

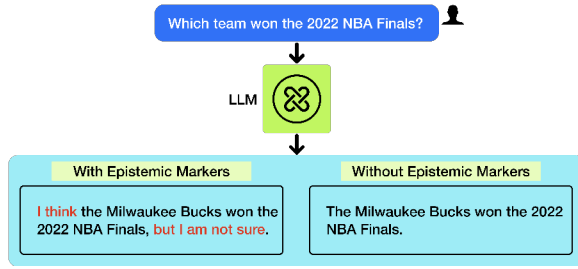


Fig. 10. The LLM provides an incorrect response, but communicates its uncertainty using epistemic markers, e.g., “I think.”

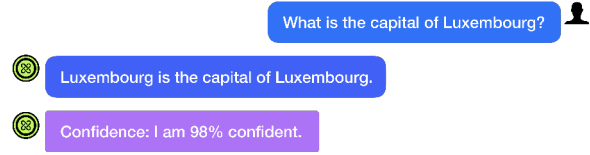


Fig. 11. LLMs can be trained or fine-tuned to provide numeric estimates of their confidence in the factuality of their response.

To address this challenge, prior work in [147] trains a learned model (calibrator) that predicts the probability that an LLM’s response to a given prompt is correct, given the input prompt, its response, and the LLM’s representations of the prompt and its response. In addition, the output of the calibrator and the LLM’s original response are subsequently used in fine-tuning a generative model [186] to produce a linguistically calibrated response, aligning the verbal expression of the LLM’s confidence with its probability of factual correctness. However, the resulting verbalized uncertainty lacks a numerical value, making it difficult for users to assess the relative confidence of the LLM. Follow-on work in [122] introduces the notion of *verbalized probability*, providing a definite numerical value of the model’s confidence, e.g., in Figure 11, or a scaled characterization of the model’s confidence in words, e.g., *low*, *medium*, or *high* confidence. The authors of [122] fine-tune GPT-3 on their proposed CalibratedMath benchmark dataset using supervised learning, demonstrating that the verbalized probability generalizes well; however, best performance is achieved in in-distribution scenarios.

More recent work has investigated other training approaches for fine-tuning LLMs to accurately express their confidence verbally. LACIE [189] introduces a two-agent speaker-listener architecture to generate training data for fine-tuning an LLM, where the reward signal is a function of the ground-truth answer and the listener’s perceived confidence of the speaker’s response. In essence, LACIE aims to fine-tune an LLM to produce a response composed of epistemic markers that are aligned with the model’s confidence in the correctness of its response. Likewise, the work in [230] proposes a knowledge-transfer training architecture where the knowledge from a bigger LLM (the teacher), e.g., GPT-4 [1], is distilled into a smaller LLM (the student), e.g., Vicuna-7B [34], using chain-of-thought reasoning. The student LLM is fine-tuned to provide its confidence (expressed as a value between 0 and 100) along with its response to an input prompt. A line of existing work [199, 226] utilizes reinforcement learning to fine-tune an LLM to improve the alignment of the confidence estimates expressed by the LLM with its factual accuracy. While SaySelf [226] relies on self-reflective rationales to improve the calibration of the verbalized confidence, the work in [199] uses reinforcement learning from human feedback (RLHF) to define a reward function consisting of a quality component in addition to an alignment component. Similarly, the work in [17] fine-tunes Llama 2 [206] using supervised learning and reinforcement learning, to produce calibrated verbalized confidence estimates that enable a user to make informed decisions on related questions. Lastly, other recent work, e.g., [55, 231], seeks to fine-tune LLMs to abstain from providing an answer to a question when faced with doubt [204], which is illustrated in Figure 12.

Despite these efforts, in many cases, LLMs still fail to accurately express their confidence verbally [66, 225], typically exhibiting overconfidence, with confidence values primarily between 80% and 100%, often in multiples of 5, similar to the way humans interact. This weakness decreases with the size of an LLM. Nonetheless, large-scale LLMs still display overconfidence, albeit at a smaller rate. However, effective prompting strategies to reduce

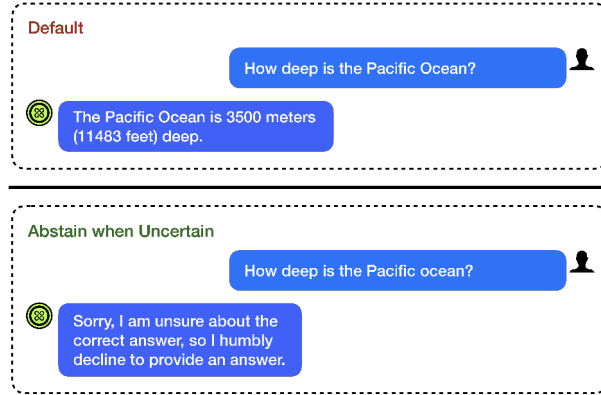


Fig. 12. Some self-verbalized UQ methods fine-tune an LLM to refrain from answering when it is uncertain about the answer.

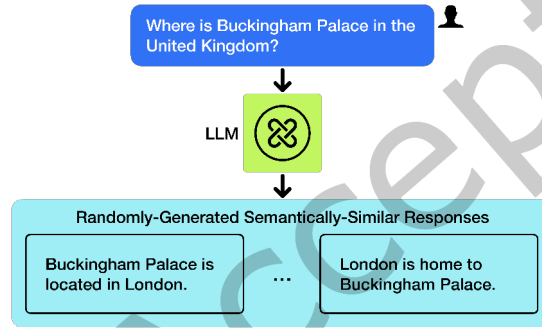


Fig. 13. When prompted to answer a question, e.g., "Where is Buckingham Palace in the United Kingdom?", an LLM might generate many variations of the same sentence. Although the form of each response may differ at the token-level, the semantic meaning of the sentences remains consistent. Semantic-similarity UQ techniques exploit semantic clustering to derive UQ methods that are robust to these variations in the form of the responses.

the calibration error of these models exist. Although verbalized confidence estimates are better calibrated than raw, conditional token probabilities [203], existing empirical studies [155] suggest that token-based UQ methods generally yield better-calibrated uncertainty estimates compared to their self-verbalized UQ counterparts.

## 5 Semantic-Similarity UQ

Semantic-similarity uncertainty quantification methods examine the similarity between multiple responses of an LLM to the same query [29, 106, 123] by focusing on the *meaning* (i.e., the semantic content of a generated sentence) rather than the *form* (i.e., the string of tokens that the model predicts) [106] of the responses. For example, consider the prompt (question) given to an LLM: *Where is Buckingham Palace in the United Kingdom?* Standard sampling from an LLM can produce many variations of the same answer when prompted with this question, as illustrated in Figure 13. However, while an LLM may be uncertain about which sequence the user may prefer, most variations do not alter the meaning of the sentence. This difference in the ordering of the tokens in each response may lead to different token probabilities, which in turn may negatively impact the accuracy of other uncertainty quantification methods, such as token-level UQ methods.

Since semantic similarity is a relative metric, its outputs are in general model-dependent, posing a central challenge. A recent line of work uses NLI models, such as RoBERTa [132] and DeBERTa [74] (discussed in Section 2.4.2), to compute entailment probabilities. The work in [4] proposes upweighting tokens that have large gradients with respect to the NLI model to maximize the probability of contradiction to generate semantically-varied responses. In addition, the method in [198] proposes using a chain-of-thought agreement (CoTA) metric that uses entailment probabilities to evaluate the agreement between CoT generations, concluding that CoTA semantic uncertainty leads to more robust model faithfulness estimates than either self-verbalized or token-level uncertainty estimates. The authors of [29] propose using a combined measure of confidence that incorporates entailment probabilities along with a verbalized confidence score, and selects the generation with the highest confidence. The UQ method in [18] proposes generating multiple explanations for each plausible response and then summing the entailment probabilities. Another work [104] introduces semantic entropy probes, wherein semantic clusters are grown iteratively using entailment probabilities. Each new generation is either added to an existing cluster if entailment holds, or added to a new cluster. Then, a linear classifier is trained to predict high-entropy prompts. Furthermore, the method in [144] uses a database of user-verified false statements to build a semi-automated fact-checking system that uses entailment probabilities with database queries as a metric for confidence in a statement's falseness.

In addition to using NLI models to evaluate factual similarities between responses, some methods use language embeddings [163] to cluster responses based on their semantic similarity and reason about uncertainty over the clusters, e.g., semantic density in [167]. First, several reference responses are generated by sampling the model. Then, the overall uncertainty per response is computed using the entailment scores, taking values in the set  $\{0, 0.5, 1\}$ . The semantic density is then used to accept or reject a target response based on the similarity to the target responses. The supervised approach in [73] utilizes the K-means algorithm to first cluster synonyms, which are attended by the LLM during training. The work in [80] introduces a method to achieve semantically-aligned item identification embeddings based on item descriptions, which aid in aligning LLM-based recommender systems with semantically-similar generations when item descriptions are sparse. Further, the method in [218] prompts an LLM to generate concepts (effectively semantic clusters) and uses an NLI-based concept scorer along with the entropy over the concepts to quantify the overall uncertainty of the LLM. ClusterLLM [246] uses a frozen instruction-trained LLM to guide clustering based on triplet queries (e.g., does A match B better than C?), achieving more semantically-aligned embeddings.

However, assigning responses to a single cluster precludes assignment to another, when in reality a response may belong to more than one class, limiting the effectiveness of clustering-based semantic-similarity UQ methods. To address this challenge, another line of work extends clustering-based methods to graphs, which may express the complex relationship between responses more explicitly. The work in [8] proposes Contrastive Semantic Similarity, which uses responses as vertices and CLIP cosine similarities as edges. The overall uncertainty is computed from the eigenvalues of the graph Laplacian, and the eigenvectors can be used to assign clusters more effectively. Similarly, the approach in [39] uses edges weights determined directly from NLI models and extends the graph-Laplacian-based uncertainty metric to include additional semantic uncertainty, such as Jaccard similarity. The authors of [89] introduce a claim-and-response structure wherein edges are added between a claim and response if the response entails the claim. The centrality metrics are used to estimate per-claim uncertainty and integrate low-uncertainty claims into further generations. In addition, Kernel Language Entropy [157] clusters responses to construct a kernelized graph Laplacian, which is used to estimate fine-grained differences between responses in a cluster.

A few works that learn to estimate semantic meanings without NLI models using supervised approaches have also been proposed. In [129], the authors use an auxiliary tool LLM to compute a similarity score between the target LLM's generation and the tool LLM's generation and learns an uncertainty estimation function to estimate the similarity score. In [93], the authors propose a cascading chain of increasingly complex LLM judges to evaluate

the predecessor’s preference between two generations. A calibration dataset is used to learn a threshold that determines each judge’s minimum confidence level. The confidence thresholds are tuned in order to guarantee that the appropriate judge is selected to generate a satisfactory response.

## 6 Mechanistic Interpretability

Mechanistic interpretability (MI) aims to understand the inner workings of LLMs to pinpoint the potential sources of uncertainty, by uncovering causal relationships [20]. Several survey papers have provided a taxonomy of mechanistic interpretability in the field of transformer-based language models [172], focused on AI safety [20] or interpretability of language models in general [247].

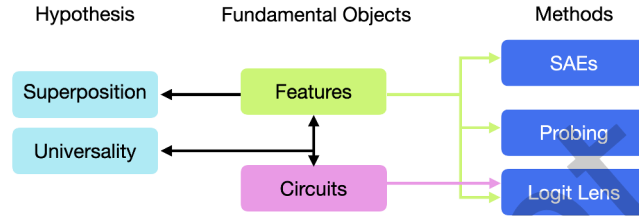


Fig. 14. Taxonomy of Mechanistic Interpretability [172].

We start by discussing a few key concepts of mechanistic interpretability, summarized in Figure 14. Features are the unit for encoding knowledge in a neural network. For example, a neuron or set of neurons consistently activating for Golden Gate Bridge can be interpreted as the “Golden Gate Bridge” feature [200]. *Superposition* [51] is often a key hypothesis in mechanistic interpretability [20], due to the fact that the same neuron seems to activate in multiple, distinct contexts, a phenomenon known as polysemanticity [38]. The superposition hypothesis claims that the set of  $N$  neurons encode  $M > N$  features, by allocating each feature to a linear combination of neurons, which are in almost orthogonal directions, leading to an overcomplete set of basis. On the other hand, the work in [52] suggests that there exists circular features corresponding to days of the week and months of the year, breaking the assumption that high-level features are linearly represented in the activation space. Circuits, another fundamental concept, refers to sub-graphs of the network that consist of features and weights connecting them. Recent research have aimed to perform comprehensive circuit analysis on LLMs in order to construct a full mapping from specific circuits to functionalities of the language model [48, 119]. The hypothesis of universality, related to both features and circuits, claims that similar features and circuits exist across different LLMs.

Methods in MI can be broadly classified into the following categories: logit lens, probing, and sparse auto-encoders methods, each discussed briefly. Logit lens methods project the activations from various layers of the LLM back into the vocabulary space, allowing for interpreting intermediate predictions and information encoded in activations [63, 118]. Probing methods aim to find which intermediate activations encode specific information (e.g., syntactic, semantic, or factual knowledge), by training a linear classifier as a *probe* to predict the existence of a certain feature [19, 69]. Despite being simple and successful, probing methods only reveal correlations instead of causal relations, limiting their use in MI. Sparse auto-encoders (SAEs) represent a popular architecture applied in MI to directly identify meaningful feature activations within LLMs and the causal relations between them. SAEs map the feature vectors onto a much higher dimensional space with strong sparsity, in order to disentangle the features that were in superposition. In these methods, an encoder-decoder pair  $(z, \hat{x})$  is trained to map  $\hat{x}(z(x))$  back to the model’s activation  $x$ , given by:  $z = \sigma(W_{\text{enc}}x + b_{\text{enc}})$ ,  $\hat{x} = W_{\text{dec}}z + b_{\text{dec}}$ . The



specific implementation of the activation function can vary, with a common choice of the activation function given by the ReLU [38, 48]. In [60],  $\sigma = \text{TopK}$  is used to keep only the  $k$ -largest latents, simplifying tuning and outperforming ReLU. In [119],  $\sigma = \text{JumpReLU}$  is chosen due to its slightly better performance and the ability to allow for a variable number of active latents at different tokens. In [48], the authors train the architecture differently with transcoders, where the faithfulness term in the loss function measures the error between the output and the original MLP sub-layer output, instead of the original input. In [238], the authors hypothesize that contextualized word embeddings are linear superpositions of transformer factors. For example, the word “apple” can be decomposed into:  $\text{apple} = 0.09\text{dessert} + 0.11\text{organism} + 0.16\text{fruit} + 0.22\text{mobile\&IT} + 0.42\text{other}$ . The authors aim to learn a comprehensive dictionary of word factors. In doing so, they distinguish between low, mid, and high-level factors by looking at the change in the importance score across layers. Low-level factors correspond to word-level polysemy disambiguation; mid-level factors are sentence-level pattern formation; and high-level factors correspond to long-range dependency, which have to be manually distinguishable from mid-level factors, although it could be done with black-box interpretation algorithms as well. In [196], the authors quantize features into sparse “codebook” features, providing the capability to control the network behavior.

Prior work has employed techniques from mechanistic interpretability to track the progress of models during training [154], to explain the outputs of models [182], and to improve the accuracy of LLMs [26]. The work in [26] demonstrates that the accuracy of the latent knowledge of LLMs is less sensitive to the input prompts, with its accuracy remaining relatively constant even when the LLM is prompted to generate incorrect responses. Likewise, ReDeEP [193] examines the latent knowledge of an LLM to decouple the effects of external knowledge from knowledge bases and the internal knowledge in the model on hallucinations in retrieval-augmented generation. Further, prior work has examined hallucinations in LLMs through the lens of mechanistic interpretability [217, 236]. The work in [236] investigates the role of an LLM’s hidden states in contributing to hallucinations, quantifying the contributions of lower-layer and upper-layer MLPs and attention heads to factual errors. In addition, the method in [56] leverages mechanistic interpretability to identify the boundaries of an LLM’s internal knowledge of its own capabilities, which could be used to prevent a model from answering questions on certain subjects (i.e., in safeguarding the model) or to prevent hallucinations when the model does not know about certain subjects. Lastly, the work in [2] trains small classifiers (linear and non-linear MLPs) on the activations of a small LLM to predict the uncertainty level of a larger LLM, demonstrating that the classifiers generalize to unseen distributions. Although there is an inextricable link between understanding the inner workings of LLMs and quantifying their uncertainty when prompted by a user, the connections between mechanistic interpretability and uncertainty quantification have not been extensively explored. For example, certain neural activation patterns in LLMs might be associated with the expression of uncertainty by the model. In addition, when faced with doubt, an LLM might utilize certain features (words/concepts), that could be detected from its neural activations. Identifying the specific intermediate activations and features of an LLM that are relevant for uncertainty quantification remains an open research challenge. We describe this open challenge in Section 10.4.

## 7 Calibration of Uncertainty

In many cases, the confidence estimates computed by the UQ methods presented in the preceding sections are not well-calibrated i.e., aligned with the observed frequencies of the responses (accuracy of the model). However, reliability of the confidence estimates of an LLM’s output remains crucial to the safe deployment of LLMs. As a result, we would like the confidence estimates to be calibrated. Formally, for a perfectly-calibrated confidence estimate  $p$ , we have that,  $\forall p \in [0, 1]$ :

$$\mathbb{P}[Y = \hat{Y} \mid \hat{P} = p] = p, \quad (2)$$

where  $Y$  and  $\hat{Y}$  represent random variables denoting the ground-truth and predicted outputs from the model, respectively, and  $\hat{P}$  represents a random variable denoting the confidence associated with the predicted output  $\hat{Y}$ .

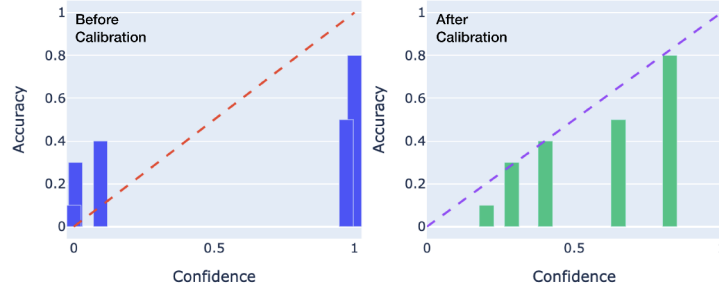


Fig. 15. The confidence estimates provided by many UQ methods are not always calibrated, i.e., the observed frequencies do not match the estimates. Calibration techniques correct these confidence estimates for better alignment with the observed accuracy.

[67]. In Figure 15, we show poorly-calibrated confidence estimates on the left, where the estimated confidence of the model is not well-aligned with the observed accuracy of the model. The dashed-line illustrates perfect alignment between the estimated confidence of the model and its accuracy. In this example, confidence estimates of the model above 0.5 tend to be overconfident, exceeding the accuracy of the model. Conversely, confidence estimates that are less than 0.5 tend to be underconfident. Calibration techniques improve the alignment of the estimated confidence of the model with the observed accuracy, with the estimated confidence more closely following the dashed-line, as shown on the right in Figure 15. We review some metrics for quantifying the calibration of a model’s confidence estimates.

**Expected Calibration Error (ECE).** The Expected Calibration Error (ECE) measures the expected deviation between the left-hand side and right-hand side of (2), with:  $\mathbb{E}_{\hat{P}} [|\mathbb{P}[Y = \hat{Y} \mid \hat{P} = p] - p|]$ , where the expectation is taken over the random variable  $\hat{P}$ . Computing the expectation in the ECE is intractable in general. Hence, the work in [153] introduces an approximation of the ECE, which partitions the confidence estimates into equal-width bins and computes the difference bin-wide, with:  $\text{ECE} = \sum_{m=1}^M \frac{|B_m|}{n} |\text{acc}(B_m) - \text{conf}(B_m)|$ , where the confidence estimates are divided into  $M$  bins with the  $i$ ’th bin denoted by  $B_i$ , and  $\text{acc}$  and  $\text{conf}$  denote the average accuracy and confidence of the samples in a bin.

**Maximum Calibration Error (MCE).** Alternatively, we may seek to quantify the maximum deviation between the left-hand and right-hand sides of (2), representing the worst-case error, which is often useful in safety-critical applications. The Maximum Calibration Error (MCE) is given by:  $\max_{p \in [0,1]} |\mathbb{P}[Y = \hat{Y} \mid \hat{P} = p] - p|$ , which is also challenging to compute exactly, like the ECE. As a result, we can estimate an upper bound, given by:  $\text{MCE} = \max_{m \in \{1, \dots, M\}} |\text{acc}(B_m) - \text{conf}(B_m)|$ , as introduced in [153]. Metrics for quantifying the calibration error of confidence estimates are further discussed in [67, 156, 159].

We can categorize calibration techniques for uncertainty estimation as either *training-based* or *training-free* calibration methods. Training-based calibration methods comprise supervised techniques that modify the network’s weights and various types of self-verbalization, where the model qualifies and refines its outputs based on its own reasoning or feedback about uncertainty. In contrast, training-free calibration methods include statistical techniques that operate on a frozen learned model.

### 7.1 Training-Free Calibration Methods

Training-free calibration methods do not modify the weights of the model to produce calibrated predictions, e.g., Platt scaling [165], isotonic regression [239, 240], and conformal prediction [184]. Here, we discuss conformal

prediction in greater detail. Conformal prediction (CP) is a powerful technique used to quantify the uncertainty of a model's predictions by providing prediction sets that are guaranteed to contain the true outcome with a specified probability. Given a prediction model  $f$  and a calibration dataset  $\mathcal{D}_{\text{cal}} = \{(x, y)_i\}_{i=1}^N$ , conformal prediction aims to compute a set of nonconformity scores  $\mathcal{S} = \{(s)_i\}_{i=1}^N$  over  $\mathcal{D}_{\text{cal}}$ , which reflect how closely each prediction  $f(x_i)$ —such as the confidence estimate provided by the aforementioned UQ methods—aligns with the true label  $y_i$ . Given a coverage level  $\hat{\varepsilon}$  (effectively a budget for incorrect predictions) and  $\mathcal{S}$ , CP aims to construct a prediction set  $C(x_{n+1})$  for a new *test* data point  $x_{n+1}$ :  $C(x_{n+1}) = \{y : f(x_{n+1}) \leq q_{1-\hat{\varepsilon}}(s_1, s_2, \dots, s_n)\}$ , along with the probabilistic guarantee:  $\mathbb{P}(x_{n+1} \in C(x_{n+1}) | \mathcal{D}_{\text{cal}}) \geq 1 - \varepsilon(\delta)$ , where  $q_{1-\hat{\varepsilon}}$  is the  $(1 - \hat{\varepsilon})$ -quantile of the nonconformity scores from the calibration set and  $\delta$  is a tunable failure probability associated with the randomness in sampling  $\mathcal{D}_{\text{cal}}$ . By applying a Hoeffding-style argument [184], one can show that  $\varepsilon$  can be selected, e.g., using the cumulative distribution function of the Beta distribution:  $\varepsilon := \text{Beta}_{N+1-v, v}^{-1}(\delta)$ ,  $v := \lfloor (N+1)\hat{\varepsilon} \rfloor$ , where  $\hat{\varepsilon}$  is the target coverage level.

Provided that the nonconformity scores represent the true conditional probabilities, conformal prediction produces the tightest prediction set that minimizes the number of false positives (i.e., maximizes the discriminative power) among all set-valued predictors such that the user-specified coverage level holds [181, Theorem 1]. As a result, LLMs that are calibrated with conformal prediction will have the smallest prediction sets on average, and therefore the least ambiguity in their responses. A number of papers employ conformal prediction for uncertainty quantification of LLMs, e.g., for semantic uncertainty quantification [219] and calibration [130]. In addition to conformal prediction, information-theoretic approaches have been developed to manage and calibrate uncertainty in sequential decision-making processes [251], e.g., entropy-rate control and multicalibration [44], which involves grouping data points into subgroups and ensuring the model is calibrated with respect to each of these subgroups. A model can also be calibrated to control a heuristic estimate of risk, such as human agreement [93] or Pareto-optimality of the response correctness [249].

## 7.2 Training-Based Calibration Methods

We can group training-based calibration techniques into ensemble-based calibration methods, few-shot calibration methods, and supervised calibration methods.

**7.2.1 Ensemble-Based Calibration.** Ensemble-based calibration (model ensembling) seeks to estimate uncertainty by querying many similar models (for example, the same architecture trained with different random seeds) and comparing their outputs. Prompt ensembles enhance calibration by combining the outputs of multiple prompts [88]. One common and effective ensembling strategy involves utilizing the majority vote. Given  $K$  models predicting a response  $l_i$ , the majority vote is selected as:  $P_{\text{acc}}(\hat{y} = l_i) = \sum_{k=1}^K P_k(\hat{y}_k = l_i) \mathbb{I}(\hat{y}_k = l_i)$ . The ensemble vote is then the response  $l_i$  with the highest aggregate confidence. Another class of ensemble-based methods evaluates *overall* (rather than pre-choice) uncertainty, e.g. binning the model's responses into semantic categories and computing the entropy [15, 208]. An ensemble-like effect can also be realized by varying the in-context examples provided to the LLM [114].

**7.2.2 Few-Shot Calibration.** Few-shot calibration techniques employ several queries to the same model and benefit from sequential reasoning as the model evaluates its intermediate generations. For instance, prompting models to begin their responses with a fact and justification for the fact has been shown to improve calibration versus other types of linear reasoning, such as tree-of-thought [220, 250]. In the domain of code generation, calibration techniques have also been applied to improve the reliability of generated code [187]. Furthermore, inferring human preferences with in-context learning has been explored as a means to calibrate models in alignment with human judgments [133].

**7.2.3 Supervised Calibration.** Supervised calibration approaches, which mainly involve modifying the LLM’s weights via additional losses, auxiliary models, or additional data, are also crucial in enhancing model calibration. In supervised methods, learning to classify generated responses as correct (i.e., via a cross-entropy loss) can result in better calibration than non-learning-based approaches and can help to combat overconfidence [32, 90, 254]. In fact, some existing work argue that fine-tuning is necessary for the calibration of uncertainty estimates of LLMs [97]. Given a language generator  $\hat{f}$ , score model (confidence)  $\hat{P}$ , and a dataset  $\mathcal{D} := \{(x, y)_i\}_{i=1}^N$  of data-label pairs, the token-level cross-entropy loss seeks to measure the uncertainty of the predicted labels  $\hat{f}(x)$ , on average, over the dataset:  $L_{CE} = -\mathbb{E}^{(x,y) \sim \mathcal{D}} [\log \hat{P}(y = \hat{f}(x))]$ , to improve the calibration of the confidence estimates of the model. While LLMs exhibit high-quality text generations ( $\hat{f}$ ), their confidences ( $\hat{P}$ ) may be improved by fine-tuning the model with a cross-entropy loss on the full dataset or a subset. Besides the cross-entropy function, other proper-scoring rules can also be used for achieving calibration [64, 65]. Reinforcement learning (with human feedback in some applications) may be used to fine-tune a model to produce realistic confidence estimates, e.g., [17, 143]. Techniques such as learning to rationalize predictions with generative adversarial networks [183], applying regularization [103], and biasing token logits [131, 252] have also been explored. Finally, sequence-level likelihood calibration has been proposed to improve the quality of LLM generations [251]. Instead of modifying the model’s weights, another class of supervised calibration methods seeks to modify model *hyperparameters* in a post-hoc manner. These include temperature tuning [43] and methods involving entropy and logit differences [QQ] [136].

## 8 Datasets and Benchmarks

Here, we present useful benchmarks in uncertainty quantification for LLMs. The rapid development of highly-capable LLMs has led to the introduction of a slate of benchmarks for measuring advances on the different capabilities of these models. Some examples of these datasets include: GPQA [176], a domain-specific dataset with multiple-choice questions in the physical sciences; MMLU [76], a multi-task dataset for evaluating the breadth of knowledge of LLMs across a wide range of subjects, e.g., the humanities and sciences; HellaSwag [241], a dataset for evaluating LLM’s common-sense reasoning capability in sentence-completion tasks; RACE [108], a dataset for reading-comprehension evaluation; GSM8K [35], a dataset for evaluating the grade-school, math-solving capability of LLMs; and APPS [75], a code-generation benchmark for LLMs. There have been a related line of work in developing datasets with inherent ambiguities [95, 125, 148, 195], e.g., “the cat was lost after leaving the house” meaning either that the cat was unable to find the way, or the cat was unable to be found [148, Fig. 1], as well as datasets modeling clarifying questions in multi-turn conversations [5]. However, experimental results associated with these datasets do not necessarily incorporate uncertainty evaluation beyond answering accuracy.

Although many of the aforementioned benchmarks have not been widely adopted in research on uncertainty quantification, a few benchmarks in natural-language processing have proven highly amenable to research in uncertainty quantification of LLMs, e.g., TriviaQA [91], a dataset which consists of 95K question-answer pairs for evaluating an LLM’s reading-comprehension skill. TriviaQA [91] has been widely utilized in evaluating many methods for uncertainty quantification of LLMs [106, 147, 189]. Likewise, other methods have employed CoQA [174], a dataset containing conversational question-answer pairs, and WikiBio [110], a dataset containing biographies from Wikipedia, in evaluating the performance of UQ methods for LLMs. The CalibratedMath benchmark was introduced in [122] for examining the ability of LLMs to verbally express their confidence in solving arithmetic tasks. Moreover, datasets for evaluating the consistency of LLMs exist, e.g., ParaRel [50], which consists of 328 paraphrases, generated by altering a set of prompts while keeping the semantic meaning of the prompts the same. Furthermore, HotpotQA [233] and StrategyQA [62] represent question-answering benchmarks consisting of question-answer pairs generated from Wikipedia, specifically designed to assess the ability of LLMs to perform multi-hop reasoning. Similarly, TruthfulQA [121] represents a factuality-oriented dataset, designed to

evaluate the ability of LLMs to generate factual responses to questions that some humans might answer wrongly based on misconceptions. Noting the connection between hallucination and uncertainty quantification, uncertain quantification methods can leverage benchmarks for hallucination detection, e.g., HaluEval [115], and datasets for factuality analysis and claim verification, e.g., FEVER [202]. Lastly, we note that there has been some work that aims to standardize the tasks for evaluating the performance of LLMs by explicitly accounting for the uncertainty of LLMs in specific tasks, e.g., based on selective classification and generation [209] or conformal prediction [234].

## 9 Applications

We highlight a few application areas of uncertainty quantification of LLMs, including its applications to chatbots and other textual use-cases and robotics.

### 9.1 Chatbot and Textual Applications

Given that LLMs are prone to hallucinate, existing work examines the integration of uncertainty quantification techniques in LLM-enabled chatbots. For example, recent work leverages uncertainty quantification techniques for LLMs in hallucination detection [104, 204, 228, 245] and content and factuality analysis [162, 194]. Semantic entropy probes (SEPs) [245] utilize linear logistic models to predict semantic entropy from the hidden states of an LLM, demonstrating its effectiveness in detecting hallucinations on a variety of tasks. The approach in [228] introduces an information-theoretic metric for hallucination detection by estimating both the aleatoric and epistemic uncertainty of the LLM, with the premise that large epistemic uncertainty corresponds to hallucinations. Other downstream applications leverage hallucination detection to estimate the confidence of the LLM on the factuality of its response [138] or to actively improve the factuality of LLMs during the token-generation step [28].

In Figure 16, we illustrate an application of uncertainty quantification to detect hallucinations in LLMs. When asked for the smallest country in Asia by land area, the LLM provides a confident response. However, the low token-level confidence estimate reveals the uncertainty of the LLM, indicating a high likelihood of hallucination by the LLM. Drawing upon the association between factuality analysis and uncertainty quantification, the work in [150] employs conformal prediction to actively generate outputs that have a high probability of being facts. Further, the work in [162] trains a logistic regression classifier to detect outright lies in LLMs (i.e., false information provided by the LLM when the factual answer is known as opposed to hallucinations where the LLM does not know the factual answer), by asking the LLM follow-up questions unrelated to the original prompt. Applications in sentiment analysis [140] and content analysis [33, 41, 224] utilize LLMs in characterizing the sentiments or opinions implied in text sources and in deductive coding to aid the identification of relevant themes across highly-varied documents, respectively. However, noting that LLMs are not necessarily consistent in their outputs, the LLMq method [194] examines the LLM's outputs for the presence of epistemic linguistic uncertainty markers and the consistency of the LLM's outputs to identify the thematic codes associated with the text. Further applications arise in text summarization [102], examining the alignment of uncertainty markers in the original source document and the LLM-generated summary.

Uncertainty quantification has also been explored within the context of *jailbreaking* LLMs. For example, the work in [188] examines the connections between predictive entropy and jailbreak prompts, showing that the entropy of the LLM's tokens increases when an LLM is given jailbreak prompts. However, the LLM's uncertainty can be directly manipulated during the jailbreaking attempt [242]. In addition, the evaluation study in [126] highlights that safeguard models for LLMs often show notable miscalibration in jailbreaking attempts. Further, existing work employs uncertainty quantification techniques to improve LLMs via fine-tuning [158, 160, 229, 232]. Other applications have explored uncertainty quantification in multi-step interaction and chain-of-thought

prompting settings [70, 248], where the final output of an LLM depends on intermediate responses. To account for the influence of preceding responses, these methods propagate the LLM’s uncertainty at each interaction phase. Similar uncertainty propagation techniques have been applied to sequential labeling problems [72]. In other applications, uncertainty quantification methods for LLMs have been utilized in retrieval-augmented generation [116, 180], using the framework of conformal prediction to provide provable guarantees. Moreover, some existing work utilizes conformal prediction in response generation from an LLM to identify prediction sets that are likely to contain the ground-truth with some guarantees [107, 168]. Although prior work employing conformal prediction generally assume access to the LLM’s logits, conformal prediction can also be utilized with black-box LLMs, e.g., [191]. Lastly, techniques and results from mechanistic interpretability can be used to predict performance of LLMs at test time. In [182], the authors train a causal explanation model to estimate model performance using sensitivity to input features. In [154], the authors find that sudden emergent qualitative changes in LLMs can be predicted by reverse engineering the model. Further, recent works [255] have shown that scaling up LLMs in terms of model size or dataset does not improve interpretability as previously believed, by surveying human participants.

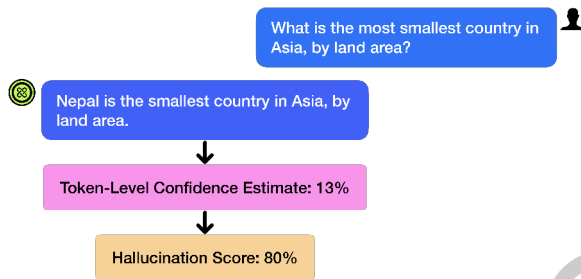


Fig. 16. Uncertainty quantification methods for LLMs have been employed in hallucination detection. LLMs tend to be less confident when hallucinating (measured via token-based metrics), although their responses may sound overly confident. In this example, although the LLM provides a confident response to the prompt, a token-level UQ method indicates that the LLM is uncertain, enabling hallucination detection.

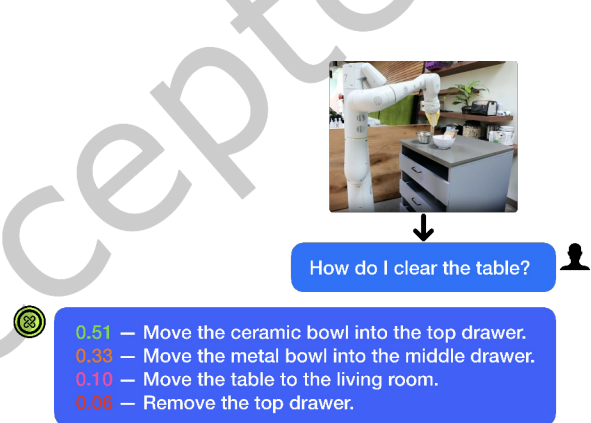


Fig. 17. Robotics applications utilize UQ methods to estimate the LLM’s confidence in the sub-tasks proposed by the LLM, to determine when human assistance is required.

## 9.2 Robotics

Endowing LLMs with an embodiment (physical form) presents unique challenges, as is the case in robotics. Such embodiment essentially empowers LLMs to be agents of physical change, which can lead to potentially disastrous outcomes if the outputs of the LLMs are not reliable or trustworthy. Although LLMs (and vision-language models) have found widespread applications in robotics, e.g., robotic manipulation [3, 21, 22, 100], robotic navigation and exploration [46, 177, 185], and multi-robot collaboration [30, 96, 142], only a few of these applications explicitly consider the uncertainty of the LLMs to ensure safety, although other existing work [214] utilize LLMs to assess the success of a task without explicitly reasoning about the confidence of the LLM.

The work in [207] fine-tunes the Mistral-7B LLM [87] to generate possible next actions for a decision-making agent and trains a neural point-wise dependency estimator to predict the compatibility score between a user-provided prompt and all generated actions. Subsequently, the authors employ conformal prediction to identify

more likely actions for a given prompt, which is presented to the user to select the next action. A collection of LLM-based task planning work for robots examine the confidence an LLM assigns to its generated next-step plans to determine when human assistance or verification is required, illustrated in Figure 17. To determine when an LLM requires clarification from a human, KnowNo [178] utilizes a token-based UQ approach to estimate the uncertainty of the LLM in generating possible next steps for a robot given a task, by examining the token probability assigned to each option in the list of possible next steps. Further, KnowNo employs conformal prediction to generate prediction sets over the possible next steps, with provable theoretical guarantees, prompting the human for help, if the prediction set consists of more than one possible action. HERACLES [215] presents a similar pipeline within a Linear Temporal Logic framework, with multiple high-level sub-goals.

IntroPlan [117] extends KnowNo [178] through introspective planning, where, given a task, the LLM retrieves the most relevant instance from a knowledge base constructed from few-shot, human-provided examples and reasons about the feasibility of the possible next actions. Introspective planning enables IntroPlan to generate prediction sets with tighter confidence bounds, minimizing human intervention. LAP [152] further introduces an action-feasibility metric to improve the alignment of the LLM’s confidence estimate with the probability of success, resulting in fewer clarification queries. S-ATLAS [214] extends KnowNo to LLM-based multi-robot task planning, where a team of robot collaborate to complete a task. In addition, KnowLoop [253] utilizes a multi-modal large language model (MLLM), e.g., LLaVa [127] or ChatGPT-4V, for failure detection in LLM-based task planning. The MLLM evaluates the success of the task, given images of the environment at each stage, providing its feedback along with its estimated confidence, using either a self-verbalized approach or a token-level UQ method. KnowLoop [253] demonstrates that token-level UQ approaches yield better-aligned uncertainty estimates compared to a self-verbalized UQ approach. Lastly, TrustNavGPT [192] employs a similar architecture to evaluate the trustworthiness of human commands to an LLM in LLM-based, audio-guided robot navigation.

## 10 Open Research Challenges

We enumerate a number of open research challenges, hoping to drive future research to address these challenges.

### 10.1 Consistency is not Factuality

Many uncertainty quantification methods for LLMs rely on evaluating the consistency between multiple realizations of the response generated by LLMs. This approach faces fundamental limitations, since consistency is not necessarily aligned with factuality. For example, in Figure 18, when prompted to provide a response to the question: “What happened to Google in June 2007, in a single sentence?” GPT-4 claims that Google announced its mobile operating system Android in June 2007, which is incorrect, given that Android was launched in November 2007. In fact, when creating the set of responses for uncertainty quantification, multiple queries to GPT-4 generate the same incorrect response, which can lead to a miscalibrated confidence estimate. Notably, black-box methods that rely entirely on consistency are most susceptible to this challenge.

Nonetheless, consistency is often a good predictor of factuality, especially when given a sufficiently large number of samples. However, many existing methods do not rigorously examine the number of samples required to define a reliable set of responses when evaluating the consistency of an LLM on a given prompt, which constitutes a critical component for any guarantee on the estimated confidence of the model or factuality of the model’s response. Moreover, this challenge might be mitigated by a principled selection of the temperature parameter in an LLM to increase the randomness of the mode; however, the effectiveness of this strategy is quite limited, as excessive randomness in the LLM’s outputs defeats the purpose of examining the confidence of the model on a given prompt.

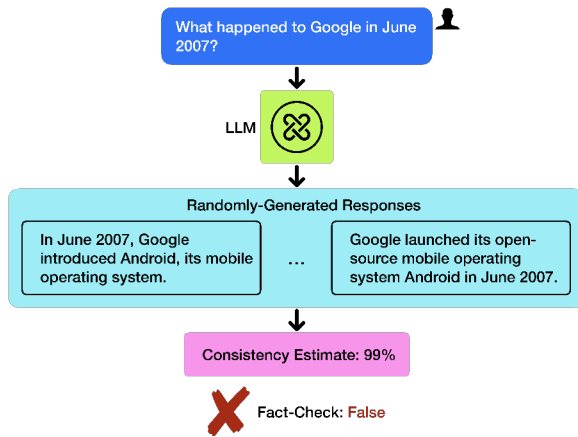


Fig. 18. Consistency is not factuality. Semantic-similarity UQ methods for LLMs might provide misleading confidence estimates, e.g., when multiple random responses from the LLM are consistent but false. In this example, the LLM consistently claims that Google introduced Android in June 2007, which is incorrect, given that Android was introduced in November 2007.

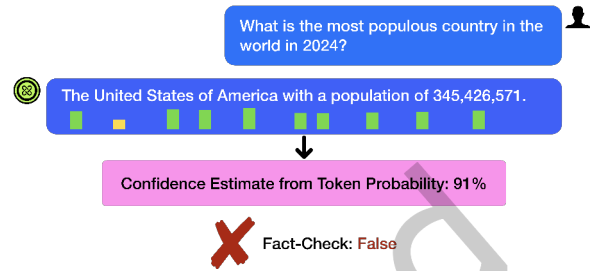


Fig. 19. Using the conditional distribution of tokens for uncertainty quantification (e.g., in token-level UQ methods) can lead to misleading uncertainty estimates. In this example, the uncertainty of the LLM is notably low, since the succeeding tokens are highly likely given the preceding tokens. However, the claim is incorrect. The most populous country in the world in 2024 is India, not the United States of America. The bars denote the probability of each token.

## 10.2 Entropy is not Factuality

Entropy and other token-based UQ metrics of the token probability distribution in an LLM's output are not necessarily aligned with the factuality of the model's output, although entropy and factuality are often aligned. In particular, the distribution over the tokens is a function of the size of the LLM (including its dictionary of tokens) and the diversity and size of the training data, which can influence the alignment of entropy and factuality. Hence, token-based UQ methods might produce highly miscalibrated confidence estimates for a given prompt, when these estimates are computed entirely from the distribution over the tokens. For example, in a worst-case scenario where the training data is corrupted or insufficient, an LLM might assign most of its probability to an incorrect answer (token) which is most closely related to the training data, leading to a miscalibrated estimate of its confidence. Moreover, reinforcement learning with human feedback (RLHF), which is utilized in fine-tuning LLMs, generally reduces the calibration of the LLM's confidence estimates [1]. Further, the conditional distribution of each token might not be indicative of the factuality of an LLM's response at the claim-level (sentence-level), i.e., although each generated token might be highly likely given the preceding token, the overall claim expressed by the LLM might not be correct [211], as illustrated in Figure 19.

Future research should explore aligning the entropy of tokens with the factuality of the claims expressed by LLMs and examine augmentation strategies that consider the training distribution of LLMs to better account for the influence of the training data on the probability distribution associated with the generated tokens to ultimately improve the alignment of entropy and other token-based measures of uncertainty with factuality. Moreover, the probability distributions over the tokens of an LLM can be manipulated in jailbreaking attacks, leading to misleading confidence estimates and, in some cases, non-factual responses [242]. Future research should seek to improve the robustness of token-level uncertainty quantification methods to adversarial attacks. Further, few existing methods explore uncertainty quantification of LLMs in text summarization, which is critical to the preservation of factual records, constituting an important direction for future research.



### 10.3 Applications in Interactive LLM-Enabled Agents

Although some existing applications explore uncertainty quantification in LLM-enabled agents, e.g., see Section 9.2, many of these applications only estimate the LLM’s uncertainty at each episode without considering the history of the agent’s interaction with the LLM. However, many practical applications require multi-episode interactions, where the LLM generates successive responses based on the information from preceding episodes with the agent. For example, in the scenario depicted in Figure 17, the robot may be asked to prepare a meal for a user, which would require multi-episode interactions, where each episode corresponds to a given sub-task, such as dicing some vegetables before sautéing it. Note that utilizing many existing techniques for uncertainty quantification would require the assumption that the LLM’s uncertainty at each episode is independent of its prior interaction history, an assumption that is generally not satisfied in real-world applications. Rigorous uncertainty quantification of the LLM’s outputs requires the consideration of the history of the agent’s interaction with the LLM and its observations (e.g., camera images), in the case of VLMs. This yet-unexplored research area constitutes an exciting direction for future research.

### 10.4 Applications of Mechanistic Interpretability to Uncertainty Quantification

The connections between interpretability of LLMs and uncertainty quantification have been relatively unexplored, despite the intuitive relationship between both concepts. Mechanistic interpretability holds notable potential in exploiting the synergy between both areas to derive solutions to some of the aforementioned research challenges. For example, the work in [2] predicts the token-level confidence of large LLMs using small linear probes (models) trained on the embeddings of frozen pretrained models. This work suggests the existence of a relationship between the internal states of LLMs and their confidence. The authors indicate that their findings suggest that information on the internal state of an LLM could be utilized in distinguishing epistemic uncertainty of the model from aleatoric uncertainty. However, this research area is relatively unexplored, presenting a potentially fruitful direction for future research.

### 10.5 Datasets and Benchmarks

Although a number of datasets and benchmark for uncertainty quantification exists [91, 121, 174, 233], to the best of our knowledge, no dataset exists for uncertainty quantification of LLMs in multi-episode interaction scenarios. Future research should examine the creation of versatile, standardized datasets that aid research on uncertainty quantification of LLMs, taking into consideration the history of interaction between a user and an LLM. Moreover, benchmarks on uncertainty quantification of LLMs can help inform researchers on the relative performance of their proposed methods. Unfortunately, widely-accepted benchmarks for uncertainty quantification of LLMs do not exist, although some work has been devoted to developing such benchmarks. Future work should seek to create suitable benchmarks for this purpose, especially benchmarks that evaluate the calibration, tightness (conservativeness), and interpretability of uncertainty quantification methods. However, benchmarks can also introduce other challenges, by disconnecting research from practical concerns, overly simplifying the assessment of research advances to outperforming existing work on some metric defined in a benchmark. Hence, care must be taken to ensure that benchmarks remain relevant to practical effectiveness.

## 11 Conclusion

In this survey, we provide a comprehensive review of existing uncertainty quantification methods for LLMs, including relevant background information necessary for readers. We categorize UQ methods for LLMs into four broad classes based on the underlying technique employed by these methods, namely: token-based UQ methods, self-verbalized UQ methods, semantic-similarity-based methods, and mechanistic interpretability. Token-based UQ methods rely on access to an LLM’s intermediate outputs or architecture to estimate the confidence an LLM,

whereas in self-verbalized UQ methods, the LLM provides its estimated confidence in natural-language. Many semantic-similarity-based methods are black-box methods which only require access to the model’s natural-language output, relying on consistency metrics to estimate the LLM’s confidence. In contrast, mechanistic interpretability requires access to the LLM’s internal activations to identify latent features that explain its activation patterns. Furthermore, we identify relevant datasets and applications for uncertainty quantification of LLMs and highlight open research challenges to inspire future research.

## Acknowledgments

We would like to acknowledge Apurva S. Badithela and David Snyder for their contributions. This work was partially supported by the NSF CAREER Award [#2044149], the Office of Naval Research [N00014-23-1-2148], and the Sloan Fellowship. Justin Lidard was supported by a National Science Foundation Graduate Research Fellowship.

## References

- [1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774* (2023).
- [2] Gustaf Ahlritz, Tian Qin, Nikhil Vyas, Boaz Barak, and Benjamin L Edelman. 2024. Distinguishing the knowable from the unknowable with language models. *arXiv preprint arXiv:2402.03563* (2024).
- [3] Michael Ahn, Anthony Brohan, Noah Brown, Yevgen Chebotar, Omar Cortes, Byron David, Chelsea Finn, Chuyuan Fu, Keerthana Gopalakrishnan, Karol Hausman, et al. 2022. Do as i can, not as i say: Grounding language in robotic affordances. *arXiv preprint arXiv:2204.01691* (2022).
- [4] Lukas Aichberger, Kajetan Schweighofer, Mykyta Ielanskyi, and Sepp Hochreiter. 2024. Semantically Diverse Language Generation for Uncertainty Estimation in Language Models. *arXiv preprint arXiv:2406.04306* (2024).
- [5] Mohammad Aliannejadi, Julia Kiseleva, Aleksandr Chuklin, Jeffrey Dalton, and Mikhail Burtsev. 2021. Building and evaluating open-domain dialogue corpora with clarifying questions. *arXiv preprint arXiv:2109.05794* (2021).
- [6] Hussam Alkaissi and Samy I McFarlane. 2023. Artificial hallucinations in ChatGPT: implications in scientific writing. *Cureus* 15, 2 (2023).
- [7] AI Anthropic. 2024. The Claude 3 model family: Opus, Sonnet, Haiku. *Claude-3 Model Card* 1 (2024).
- [8] Shuang Ao, Stefan Rueger, and Advait Siddharthan. 2024. CSS: Contrastive Semantic Similarity for Uncertainty Quantification of LLMs. *arXiv preprint arXiv:2406.03158* (2024).
- [9] Gabriel Y Arteaga, Thomas B Schön, and Nicolas Pielawski. 2024. Hallucination Detection in LLMs: Fast and Memory-Efficient Finetuned Models. *arXiv preprint arXiv:2409.02976* (2024).
- [10] Murat Seckin Ayhan and Philipp Berens. 2018. Test-time data augmentation for estimation of heteroscedastic aleatoric uncertainty in deep neural networks. In *Medical Imaging with Deep Learning*.
- [11] Razvan Azamfirei, Sapna R Kudchadkar, and James Fackler. 2023. Large language models and the perils of their hallucinations. *Critical Care* 27, 1 (2023), 120.
- [12] Amos Azaria and Tom Mitchell. 2023. The internal state of an LLM knows when it’s lying. *arXiv preprint arXiv:2304.13734* (2023).
- [13] Yuval Bahat and Gregory Shakhnarovich. 2020. Classification confidence estimation with test-time data-augmentation. *arXiv preprint arXiv:2006.16705* (2020).
- [14] Zechen Bai, Pichao Wang, Tianjun Xiao, Tong He, Zongbo Han, Zheng Zhang, and Mike Zheng Shou. 2024. Hallucination of multimodal large language models: A survey. *arXiv preprint arXiv:2404.18930* (2024).
- [15] Yavuz Faruk Bakman, Duygu Nur Yaldiz, Baturalp Buyukates, Chenyang Tao, Dimitrios Dimitriadis, and Salman Avestimehr. 2024. MARS: Meaning-Aware Response Scoring for Uncertainty Estimation in Generative LLMs. *arXiv preprint arXiv:2402.11756* (2024).
- [16] Oleksandr Balabanov and Hampus Linander. 2024. Uncertainty quantification in fine-tuned LLMs using LoRA ensembles. *arXiv preprint arXiv:2402.12264* (2024).
- [17] Neil Band, Xuechen Li, Tengyu Ma, and Tatsunori Hashimoto. 2024. Linguistic Calibration of Long-Form Generations. In *Forty-first International Conference on Machine Learning*.
- [18] Evan Becker and Stefano Soatto. 2024. Cycles of Thought: Measuring LLM Confidence through Stable Explanations. *arXiv preprint arXiv:2406.03441* (2024).
- [19] Yonatan Belinkov. 2022. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics* 48, 1 (2022), 207–219.
- [20] Leonard Bereska and Efstratios Gavves. 2024. Mechanistic Interpretability for AI Safety—A Review. *arXiv preprint arXiv:2404.14082* (2024).

- [21] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, et al. 2023. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818* (2023).
- [22] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, et al. 2022. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817* (2022).
- [23] Tom B Brown. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165* (2020).
- [24] Cristian Bucilua, Rich Caruana, and Alexandru Niculescu-Mizil. 2006. Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. 535–541.
- [25] Jannis Bulian, Christian Buck, Wojciech Gajewski, Benjamin Boerschinger, and Tal Schuster. 2022. Tomayto, tomahto. beyond token-level answer equivalence for question answering evaluation. *arXiv preprint arXiv:2202.07654* (2022).
- [26] Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. 2022. Discovering latent knowledge in language models without supervision. *arXiv preprint arXiv:2212.03827* (2022).
- [27] George DC Cavalcanti, Luiz S Oliveira, Thiago JM Moura, and Guilherme V Carvalho. 2016. Combining diversity measures for ensemble pruning. *Pattern Recognition Letters* 74 (2016), 38–45.
- [28] Haw-Shiuan Chang, Nanyun Peng, Mohit Bansal, Anil Ramakrishna, and Tagyoung Chung. 2024. REAL Sampling: Boosting Factuality and Diversity of Open-Ended Generation via Asymptotic Entropy. *arXiv preprint arXiv:2406.07735* (2024).
- [29] Jiu-hai Chen and Jonas Mueller. 2023. Quantifying uncertainty in answers from any language model via intrinsic and extrinsic confidence assessment. *arXiv preprint arXiv:2308.16175* (2023).
- [30] Yongchao Chen, Jacob Arkin, Yang Zhang, Nicholas Roy, and Chuchu Fan. 2024. Scalable multi-robot collaboration with large language models: Centralized or decentralized systems?. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 4311–4317.
- [31] Yuyan Chen, Qiang Fu, Yichen Yuan, Zhihao Wen, Ge Fan, Dayiheng Liu, Dongmei Zhang, Zhixu Li, and Yanghua Xiao. 2023. Hallucination detection: Robustly discerning reliable answers in large language models. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*. 245–255.
- [32] Yangyi Chen, Lifan Yuan, Ganqu Cui, Zhiyuan Liu, and Heng Ji. 2022. A close look into the calibration of pre-trained language models. *arXiv preprint arXiv:2211.00151* (2022).
- [33] Robert Chew, John Bollenbacher, Michael Wenger, Jessica Speer, and Annice Kim. 2023. LLM-assisted content analysis: Using large language models to support deductive coding. *arXiv preprint arXiv:2306.14924* (2023).
- [34] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E Gonzalez, et al. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%\* chatgpt quality. See <https://vicuna.lmsys.org> (accessed 14 April 2023) 2, 3 (2023), 6.
- [35] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168* (2021).
- [36] Cleo Condoravdi, Dick Crouch, Valeria De Paiva, Reinhard Stolle, and Daniel Bobrow. 2003. Entailment, intensionality and text understanding. In *Proceedings of the HLT-NAACL 2003 workshop on Text meaning*. 38–45.
- [37] Robert M Cronin, Daniel Fabbri, Joshua C Denny, S Trent Rosenbloom, and Gretchen Purcell Jackson. 2017. A comparison of rule-based and machine learning approaches for classifying patient portal messages. *International journal of medical informatics* 105 (2017), 110–120.
- [38] Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. 2023. Sparse autoencoders find highly interpretable features in language models. *arXiv preprint arXiv:2309.08600* (2023).
- [39] Longchao Da, Tiejun Chen, Lu Cheng, and Hua Wei. 2024. LLM Uncertainty Quantification through Directional Entailment Graph and Claim Level Response Augmentation. *arXiv preprint arXiv:2407.00994* (2024).
- [40] Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The pascal recognising textual entailment challenge. In *Machine learning challenges workshop*. Springer, 177–190.
- [41] Shih-Chieh Dai, Aiping Xiong, and Lun-Wei Ku. 2023. LLM-in-the-loop: Leveraging large language model for thematic analysis. *arXiv preprint arXiv:2310.15100* (2023).
- [42] Sylvie Delacroix. 2024. Augmenting judicial practices with LLMs: re-thinking LLMs’ uncertainty communication features in light of systemic risks. *Available at SSRN* (2024).
- [43] Shrey Desai and Greg Durrett. 2020. Calibration of pre-trained transformers. *arXiv preprint arXiv:2003.07892* (2020).
- [44] Gianluca Detommaso, Martin Bertran, Riccardo Fogliato, and Aaron Roth. 2024. Multicalibration for confidence scoring in LLMs. *arXiv preprint arXiv:2404.04689* (2024).
- [45] Jacob Devlin. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

- [46] Vishnu Sashank Dorbala, James F Mullen Jr, and Dinesh Manocha. 2023. Can an embodied agent find your “cat-shaped mug”? Llm-based zero-shot object navigation. *IEEE Robotics and Automation Letters* (2023).
- [47] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783* (2024).
- [48] Jacob Dunefsky, Philippe Chlenski, and Neel Nanda. 2024. Transcoders Find Interpretable LLM Feature Circuits. *arXiv preprint arXiv:2406.11944* (2024).
- [49] Paul-Ambroise Duquenne, Holger Schwenk, and Benoît Sagot. 2023. SONAR: sentence-level multimodal and language-agnostic representations. *arXiv e-prints* (2023), arXiv–2308.
- [50] Yanai Elazar, Nora Kassner, Shauli Ravfogel, Abhilasha Ravichander, Eduard Hovy, Hinrich Schütze, and Yoav Goldberg. 2021. Measuring and improving consistency in pretrained language models. *Transactions of the Association for Computational Linguistics* 9 (2021), 1012–1031.
- [51] Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, et al. 2022. Toy models of superposition. *arXiv preprint arXiv:2209.10652* (2022).
- [52] Joshua Engels, Isaac Liao, Eric J Michaud, Wes Gurnee, and Max Tegmark. 2024. Not All Language Model Features Are Linear. *arXiv preprint arXiv:2405.14860* (2024).
- [53] Ekaterina Fadeeva, Aleksandr Rubashevskii, Artem Shelmanov, Sergey Petrakov, Haonan Li, Hamdy Mubarak, Evgenii Tsymbalov, Gleb Kuzmin, Alexander Panchenko, Timothy Baldwin, et al. 2024. Fact-checking the output of large language models via token-level uncertainty quantification. *arXiv preprint arXiv:2403.04696* (2024).
- [54] Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2020. Language-agnostic BERT sentence embedding. *arXiv preprint arXiv:2007.01852* (2020).
- [55] Shangbin Feng, Weijia Shi, Yike Wang, Wenxuan Ding, Vidhisha Balachandran, and Yulia Tsvetkov. 2024. Don’t Hallucinate, Abstain: Identifying LLM Knowledge Gaps via Multi-LLM Collaboration. *arXiv preprint arXiv:2402.00367* (2024).
- [56] Javier Ferrando, Oscar Obeso, Senthoran Rajamanoharan, and Neel Nanda. 2024. Do I Know This Entity? Knowledge Awareness and Hallucinations in Language Models. *arXiv preprint arXiv:2411.14257* (2024).
- [57] Yaroslav Fyodorov, Yoad Winter, and Nissim Francez. 2000. A natural logic inference system. In *Proceedings of the 2nd workshop on inference in computational semantics (ICoS-2)*.
- [58] Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *International conference on machine learning*. PMLR, 1050–1059.
- [59] Yarin Gal, Jiri Hron, and Alex Kendall. 2017. Concrete dropout. *Advances in neural information processing systems* 30 (2017).
- [60] Leo Gao, Tom Dupré la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. 2024. Scaling and evaluating sparse autoencoders. *arXiv preprint arXiv:2406.04093* (2024).
- [61] Jiahui Geng, Fengyu Cai, Yuxia Wang, Heinz Koepl, Preslav Nakov, and Iryna Gurevych. 2024. A Survey of Confidence Estimation and Calibration in Large Language Models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*. 6577–6595.
- [62] Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. 2021. Did Aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *Transactions of the Association for Computational Linguistics* 9 (2021), 346–361.
- [63] Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2020. Transformer feed-forward layers are key-value memories. *arXiv preprint arXiv:2012.14913* (2020).
- [64] Tilmann Gneiting, Fadoua Balabdaoui, and Adrian E Raftery. 2007. Probabilistic forecasts, calibration and sharpness. *Journal of the Royal Statistical Society Series B: Statistical Methodology* 69, 2 (2007), 243–268.
- [65] Tilmann Gneiting and Adrian E Raftery. 2007. Strictly proper scoring rules, prediction, and estimation. *Journal of the American statistical Association* 102, 477 (2007), 359–378.
- [66] Tobias Groot and Matias Valdenegro-Toro. 2024. Overconfidence is Key: Verbalized Uncertainty Evaluation in Large Language and Vision-Language Models. *arXiv preprint arXiv:2405.02917* (2024).
- [67] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. 2017. On calibration of modern neural networks. In *International conference on machine learning*. PMLR, 1321–1330.
- [68] Huaping Guo, Hongbing Liu, Ran Li, Changan Wu, Yibo Guo, and Mingliang Xu. 2018. Margin & diversity based ordering ensemble pruning. *Neurocomputing* 275 (2018), 237–246.
- [69] Wes Gurnee, Neel Nanda, Matthew Pauly, Katherine Harvey, Dmitrii Troitskii, and Dimitris Bertsimas. 2023. Finding neurons in a haystack: Case studies with sparse probing. *arXiv preprint arXiv:2305.01610* (2023).
- [70] Jiuzhou Han, Wray Buntine, and Ehsan Shareghi. 2024. Towards uncertainty-aware language agent. *arXiv preprint arXiv:2401.14016* (2024).
- [71] W Keith Hastings. 1970. *Monte Carlo sampling methods using Markov chains and their applications*. Oxford University Press.
- [72] Jianfeng He, Linlin Yu, Shuo Lei, Chang-Tien Lu, and Feng Chen. 2023. Uncertainty estimation on sequential labeling via uncertainty transmission. *arXiv preprint arXiv:2311.08726* (2023).

- [73] Lin He and Keqin Li. 2024. Mitigating Hallucinations in LLM Using K-means Clustering of Synonym Semantic Relevance. *Authorea Preprints* (2024).
- [74] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. DeBERTa: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654* (2020).
- [75] Dan Hendrycks, Steven Basart, Saurav Kadavath, Mantas Mazeika, Akul Arora, Ethan Guo, Collin Burns, Samir Puranik, Horace He, Dawn Song, et al. 2021. Measuring coding challenge competence with apps. *arXiv preprint arXiv:2105.09938* (2021).
- [76] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300* (2020).
- [77] Geoffrey Hinton. 2015. Distilling the Knowledge in a Neural Network. *arXiv preprint arXiv:1503.02531* (2015).
- [78] Bairu Hou, Yujian Liu, Kaizhi Qian, Jacob Andreas, Shiyu Chang, and Yang Zhang. 2023. Decomposing uncertainty for large language models through input clarification ensembling. *arXiv preprint arXiv:2311.08718* (2023).
- [79] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685* (2021).
- [80] Jun Hu, Wenwen Xia, Xiaolu Zhang, Chilin Fu, Weichang Wu, Zhaoxin Huan, Ang Li, Zuoli Tang, and Jun Zhou. 2024. Enhancing sequential recommendation via llm-based semantic embedding learning. In *Companion Proceedings of the ACM on Web Conference 2024*. 103–111.
- [81] Hsiu-Yuan Huang, Yutong Yang, Zhaoxi Zhang, Sanwoo Lee, and Yunfang Wu. 2024. A Survey of Uncertainty Estimation in LLMs: Theory Meets Practice. *arXiv preprint arXiv:2410.15326* (2024).
- [82] Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al. 2023. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *arXiv preprint arXiv:2311.05232* (2023).
- [83] Rui Huang, Andrew Geng, and Yixuan Li. 2021. On the importance of gradients for detecting distributional shifts in the wild. *Advances in Neural Information Processing Systems* 34 (2021), 677–689.
- [84] Yuheng Huang, Jiayang Song, Zhijie Wang, Shengming Zhao, Huaming Chen, Felix Juefei-Xu, and Lei Ma. 2023. Look before you leap: An exploratory study of uncertainty measurement for large language models. *arXiv preprint arXiv:2307.10236* (2023).
- [85] Conor Igoe, Youngseog Chung, Ian Char, and Jeff Schneider. 2022. How useful are gradients for ood detection really? *arXiv preprint arXiv:2205.10439* (2022).
- [86] Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *Comput. Surveys* 55, 12 (2023), 1–38.
- [87] Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7B. *arXiv preprint arXiv:2310.06825* (2023).
- [88] Mingjian Jiang, Yangjun Ruan, Sicong Huang, Saifei Liao, Silviu Pitis, Roger Baker Grosse, and Jimmy Ba. 2023. Calibrating language models via augmented prompt ensembles. (2023).
- [89] Mingjian Jiang, Yangjun Ruan, Prasanna Sattigeri, Salim Roukos, and Tatsunori Hashimoto. 2024. Graph-based Uncertainty Metrics for Long-form Language Model Outputs. *arXiv preprint arXiv:2410.20783* (2024).
- [90] Daniel D Johnson, Daniel Tarlow, David Duvenaud, and Chris J Maddison. 2024. Experts Don’t Cheat: Learning What You Don’t Know By Predicting Pairs. *arXiv preprint arXiv:2402.08733* (2024).
- [91] Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *arXiv preprint arXiv:1705.03551* (2017).
- [92] Laurent Valentin Jospin, Hamid Laga, Farid Boussaid, Wray Buntine, and Mohammed Bannamoun. 2022. Hands-on Bayesian neural networks—A tutorial for deep learning users. *IEEE Computational Intelligence Magazine* 17, 2 (2022), 29–48.
- [93] Jaehun Jung, Faeze Brahman, and Yejin Choi. 2024. Trust or Escalate: LLM Judges with Provable Guarantees for Human Agreement. *arXiv preprint arXiv:2407.18370* (2024).
- [94] Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. 2022. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221* (2022).
- [95] Gaurav Kamath, Sebastian Schuster, Sowmya Vajjala, and Siva Reddy. 2024. Scope Ambiguities in Large Language Models. *Transactions of the Association for Computational Linguistics* 12 (2024), 738–754.
- [96] Shyam Sundar Kannan, Vishnunandan LN Venkatesh, and Byung-Cheol Min. 2023. Smart-llm: Smart multi-agent robot task planning using large language models. *arXiv preprint arXiv:2309.10062* (2023).
- [97] Sanyam Kapoor, Nate Gruver, Manley Roberts, Katherine Collins, Arka Pal, Umang Bhatt, Adrian Weller, Samuel Dooley, Micah Goldblum, and Andrew Gordon Wilson. 2024. Large Language Models Must Be Taught to Know What They Don’t Know. *arXiv preprint arXiv:2406.08391* (2024).
- [98] Daniel Martin Katz, Michael James Bommarito, Shang Gao, and Pablo Arredondo. 2024. Gpt-4 passes the bar exam. *Philosophical Transactions of the Royal Society A* 382, 2270 (2024), 20230254.

- [99] Geoff Keeling and Winnie Street. 2024. On the attribution of confidence to large language models. *arXiv preprint arXiv:2407.08388* (2024).
- [100] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, et al. 2024. OpenVLA: An Open-Source Vision-Language-Action Model. *arXiv preprint arXiv:2406.09246* (2024).
- [101] Sunnie SY Kim, Q Vera Liao, Mihaela Vorvoreanu, Stephanie Ballard, and Jennifer Wortman Vaughan. 2024. "I'm Not Sure, But...": Examining the Impact of Large Language Models' Uncertainty Expression on User Reliance and Trust. In *The 2024 ACM Conference on Fairness, Accountability, and Transparency*. 822–835.
- [102] Zahra Kolagar and Alessandra Zarcone. 2024. Aligning Uncertainty: Leveraging LLMs to Analyze Uncertainty Transfer in Text Summarization. In *Proceedings of the 1st Workshop on Uncertainty-Aware NLP (UncertainNLP 2024)*. 41–61.
- [103] Lingkai Kong, Haoming Jiang, Yuchen Zhuang, Jie Lyu, Tuo Zhao, and Chao Zhang. 2020. Calibrated language model fine-tuning for in-and out-of-distribution data. *arXiv preprint arXiv:2010.11506* (2020).
- [104] Jannik Kossen, Jiatong Han, Muhammed Razzak, Lisa Schut, Shreshth Malik, and Yarin Gal. 2024. Semantic entropy probes: Robust and cheap hallucination detection in llms. *arXiv preprint arXiv:2406.15927* (2024).
- [105] Lea Krause, Wondimagnh Tufa, Selene Báez Santamaría, Angel Daza, Urja Khurana, and Piek Vossen. 2023. Confidently wrong: exploring the calibration and expression of (Un) certainty of large language models in a multilingual setting. In *Proceedings of the workshop on multimodal, multilingual natural language generation and multilingual WebNLG Challenge (MM-NLG 2023)*. 1–9.
- [106] Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. *arXiv preprint arXiv:2302.09664* (2023).
- [107] 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. *arXiv preprint arXiv:2305.18404* (2023).
- [108] Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. Race: Large-scale reading comprehension dataset from examinations. *arXiv preprint arXiv:1704.04683* (2017).
- [109] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. 2017. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems* 30 (2017).
- [110] Rémi Lebre, David Grangier, and Michael Auli. 2016. Generating text from structured data with application to the biography domain. *CoRR, abs/1603.07771* (2016).
- [111] Emanuele Ledda, Giorgio Fumera, and Fabio Roli. 2023. Dropout injection at test time for post hoc uncertainty quantification in neural networks. *Information Sciences* 645 (2023), 119356.
- [112] Jinsol Lee and Ghassan AlRegib. 2020. Gradients as a measure of uncertainty in neural networks. In *2020 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2416–2420.
- [113] Katherine Lee, Orhan Firat, Ashish Agarwal, Clara Fannjiang, and David Sussillo. 2018. Hallucinations in neural machine translation. (2018).
- [114] Chengzu Li, Han Zhou, Goran Glavaš, Anna Korhonen, and Ivan Vulić. 2024. Can Large Language Models Achieve Calibration with In-Context Learning?. In *ICLR 2024 Workshop on Reliable and Responsible Foundation Models*.
- [115] Junyi Li, Xiaoxue Cheng, Wayne Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2023. Halueval: A large-scale hallucination evaluation benchmark for large language models. *arXiv preprint arXiv:2305.11747* (2023).
- [116] Shuo Li, Sangdon Park, Insup Lee, and Osbert Bastani. 2024. TRAQ: Trustworthy Retrieval Augmented Question Answering via Conformal Prediction. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*. 3799–3821.
- [117] Kaiqu Liang, Zixu Zhang, and Jaime Fernández Fisac. 2024. Introspective Planning: Guiding Language-Enabled Agents to Refine Their Own Uncertainty. *arXiv preprint arXiv:2402.06529* (2024).
- [118] Tom Lieberum, Matthew Rahtz, János Kramár, Neel Nanda, Geoffrey Irving, Rohin Shah, and Vladimir Mikulik. 2023. Does circuit analysis interpretability scale? evidence from multiple choice capabilities in chinchilla. *arXiv preprint arXiv:2307.09458* (2023).
- [119] Tom Lieberum, Senthooan Rajamanoharan, Arthur Conmy, Lewis Smith, Nicolas Sonnerat, Vikrant Varma, János Kramár, Anca Dragan, Rohin Shah, and Neel Nanda. 2024. Gemma scope: Open sparse autoencoders everywhere all at once on gemma 2. *arXiv preprint arXiv:2408.05147* (2024).
- [120] Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*. 74–81.
- [121] Stephanie Lin, Jacob Hilton, and Owain Evans. 2021. Truthfulqa: Measuring how models mimic human falsehoods. *arXiv preprint arXiv:2109.07958* (2021).
- [122] Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. Teaching models to express their uncertainty in words. *arXiv preprint arXiv:2205.14334* (2022).
- [123] Zhen Lin, Shubhendu Trivedi, and Jimeng Sun. 2023. Generating with confidence: Uncertainty quantification for black-box large language models. *arXiv preprint arXiv:2305.19187* (2023).
- [124] Chen Ling, Xujiang Zhao, Wei Cheng, Yanchi Liu, Yiyu Sun, Xuchao Zhang, Mika Oishi, Takao Osaki, Katsushi Matsuda, Jie Ji, et al. 2024. Uncertainty decomposition and quantification for in-context learning of large language models. *arXiv preprint arXiv:2402.10189*

- (2024).
- [125] Alisa Liu, Zhaofeng Wu, Julian Michael, Alane Suhr, Peter West, Alexander Koller, Swabha Swayamdipta, Noah A Smith, and Yejin Choi. 2023. We're afraid language models aren't modeling ambiguity. *arXiv preprint arXiv:2304.14399* (2023).
  - [126] Hongfu Liu, Hengguan Huang, Hao Wang, Xiangming Gu, and Ye Wang. 2024. On Calibration of LLM-based Guard Models for Reliable Content Moderation. *arXiv preprint arXiv:2410.10414* (2024).
  - [127] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2024. Visual instruction tuning. *Advances in neural information processing systems* 36 (2024).
  - [128] Hanchao Liu, Wenyan Xue, Yifei Chen, Dapeng Chen, Xiutian Zhao, Ke Wang, Liping Hou, Rongjun Li, and Wei Peng. 2024. A survey on hallucination in large vision-language models. *arXiv preprint arXiv:2402.00253* (2024).
  - [129] Linyu Liu, Yu Pan, Xiaocheng Li, and Guanting Chen. 2024. Uncertainty Estimation and Quantification for LLMs: A Simple Supervised Approach. *arXiv preprint arXiv:2404.15993* (2024).
  - [130] Terrance Liu and Zhiwei Steven Wu. 2024. Multi-group Uncertainty Quantification for Long-form Text Generation. *arXiv preprint arXiv:2407.21057* (2024).
  - [131] Xin Liu, Muhammad Khalifa, and Lu Wang. 2024. LitCab: Lightweight Language Model Calibration over Short-and Long-form Responses. In *The Twelfth International Conference on Learning Representations*.
  - [132] Yinhan Liu. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692* (2019).
  - [133] Yuxuan Liu, Tianchi Yang, Shaohan Huang, Zihan Zhang, Haizhen Huang, Furu Wei, Weiwei Deng, Feng Sun, and Qi Zhang. 2023. Calibrating llm-based evaluator. *arXiv preprint arXiv:2309.13308* (2023).
  - [134] Yang Liu, Yuanshun Yao, Jean-Francois Ton, Xiaoying Zhang, Ruocheng Guo Hao Cheng, Yegor Klochkov, Muhammad Faaiz Taufiq, and Hang Li. 2023. Trustworthy LLMs: A survey and guideline for evaluating large language models' alignment. *arXiv preprint arXiv:2308.05374* (2023).
  - [135] Antonio Loquercio, Mattia Segu, and Davide Scaramuzza. 2020. A general framework for uncertainty estimation in deep learning. *IEEE Robotics and Automation Letters* 5, 2 (2020), 3153–3160.
  - [136] Qing Lyu, Kumar Shridhar, Chaitanya Malaviya, Li Zhang, Yanai Elazar, Niket Tandon, Marianna Apidianaki, Mrinmaya Sachan, and Chris Callison-Burch. 2024. Calibrating large language models with sample consistency. *arXiv preprint arXiv:2402.13904* (2024).
  - [137] Bill MacCartney and Christopher D Manning. 2008. Modeling semantic containment and exclusion in natural language inference. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*. 521–528.
  - [138] Matéo Mahaut, Laura Aina, Paula Czarnowska, Momchil Hardalov, Thomas Müller, and Lluís Màrquez. 2024. Factual Confidence of LLMs: on Reliability and Robustness of Current Estimators. *arXiv preprint arXiv:2406.13415* (2024).
  - [139] Andrey Malinin and Mark Gales. 2020. Uncertainty estimation in autoregressive structured prediction. *arXiv preprint arXiv:2002.07650* (2020).
  - [140] Lysimachos Maltoudoglou, Andreas Paisios, and Harris Papadopoulos. 2020. BERT-based conformal predictor for sentiment analysis. In *Conformal and Probabilistic Prediction and Applications*. PMLR, 269–284.
  - [141] Potsawee Manakul, Adian Liusie, and Mark JF Gales. 2023. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. *arXiv preprint arXiv:2303.08896* (2023).
  - [142] Zhao Mandi, Shreeya Jain, and Shuran Song. 2024. Roco: Dialectic multi-robot collaboration with large language models. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 286–299.
  - [143] Xin Mao, Feng-Lin Li, Huimin Xu, Wei Zhang, and Anh Tuan Luu. 2024. Don't Forget Your Reward Values: Language Model Alignment via Value-based Calibration. *arXiv preprint arXiv:2402.16030* (2024).
  - [144] Alejandro Martín, Javier Huertas-Tato, Álvaro Huertas-García, Guillermo Villar-Rodríguez, and David Camacho. 2022. FacTeR-Check: Semi-automated fact-checking through semantic similarity and natural language inference. *Knowledge-based systems* 251 (2022), 109265.
  - [145] Gonzalo Martínez-Munoz, Daniel Hernández-Lobato, and Alberto Suárez. 2008. An analysis of ensemble pruning techniques based on ordered aggregation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31, 2 (2008), 245–259.
  - [146] Lu Mi, Hao Wang, Yonglong Tian, Hao He, and Nir N Shavit. 2022. Training-free uncertainty estimation for dense regression: Sensitivity as a surrogate. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 10042–10050.
  - [147] 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 (2022), 857–872.
  - [148] Sewon Min, Julian Michael, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2020. AmbigQA: Answering ambiguous open-domain questions. *arXiv preprint arXiv:2004.10645* (2020).
  - [149] Shervin Minaee, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao. 2024. Large language models: A survey. *arXiv preprint arXiv:2402.06196* (2024).
  - [150] Christopher Mohri and Tatsunori Hashimoto. 2024. Language models with conformal factuality guarantees. *arXiv preprint arXiv:2402.10978* (2024).

- [151] Christof Monz and Maarten de Rijke. 2001. Light-weight entailment checking for computational semantics. In *Proc. of the third workshop on inference in computational semantics (ICoS-3)*.
- [152] James F Mullen Jr and Dinesh Manocha. 2024. Towards Robots That Know When They Need Help: Affordance-Based Uncertainty for Large Language Model Planners. *arXiv preprint arXiv:2403.13198* (2024).
- [153] Mahdi Pakdaman Naeini, Gregory Cooper, and Milos Hauskrecht. 2015. Obtaining well calibrated probabilities using bayesian binning. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 29.
- [154] Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt. 2023. Progress measures for grokking via mechanistic interpretability. *arXiv preprint arXiv:2301.05217* (2023).
- [155] Shiyu Ni, Keping Bi, Lulu Yu, and Jiafeng Guo. 2024. Are Large Language Models More Honest in Their Probabilistic or Verbalized Confidence? *arXiv preprint arXiv:2408.09773* (2024).
- [156] Alexandru Niculescu-Mizil and Rich Caruana. 2005. Predicting good probabilities with supervised learning. In *Proceedings of the 22nd international conference on Machine learning*. 625–632.
- [157] Alexander Nikitin, Jannik Kossen, Yarin Gal, and Pekka Marttinen. 2024. Kernel Language Entropy: Fine-grained Uncertainty Quantification for LLMs from Semantic Similarities. *arXiv preprint arXiv:2405.20003* (2024).
- [158] Ruijia Niu, Dongxia Wu, Rose Yu, and Yi-An Ma. 2024. Functional-level Uncertainty Quantification for Calibrated Fine-tuning on LLMs. *arXiv preprint arXiv:2410.06431* (2024).
- [159] Jeremy Nixon, Michael W Dusenberry, Linchuan Zhang, Ghassen Jerfel, and Dustin Tran. 2019. Measuring Calibration in Deep Learning. In *CVPR workshops*, Vol. 2.
- [160] Ian Osband, Seyed Mohammad Asghari, Benjamin Van Roy, Nat McAleese, John Aslanides, and Geoffrey Irving. 2022. Fine-tuning language models via epistemic neural networks. *arXiv preprint arXiv:2211.01568* (2022).
- [161] Ian Osband, Zheng Wen, Seyed Mohammad Asghari, Vikranth Dwaracherla, Morteza Ibrahimi, Xiuyuan Lu, and Benjamin Van Roy. 2023. Epistemic neural networks. *Advances in Neural Information Processing Systems* 36 (2023), 2795–2823.
- [162] Lorenzo Pacchiardi, Alex J Chan, Sören Mindermann, Ilan Moscovitz, Alexa Y Pan, Yarin Gal, Owain Evans, and Jan Brauner. 2023. How to catch an ai liar: Lie detection in black-box llms by asking unrelated questions. *arXiv preprint arXiv:2309.15840* (2023).
- [163] Alina Petukhova, Joao P Matos-Carvalho, and Nuno Fachada. 2024. Text clustering with LLM embeddings. *arXiv preprint arXiv:2403.15112* (2024).
- [164] Mohammad Taher Pilehvar, David Jurgens, and Roberto Navigli. 2013. Align, disambiguate and walk: A unified approach for measuring semantic similarity. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 1341–1351.
- [165] John Platt et al. 1999. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers* 10, 3 (1999), 61–74.
- [166] Konstantin Posch, Jan Steinbrener, and Jürgen Pilz. 2019. Variational Inference to Measure Model Uncertainty in Deep Neural Networks. *arXiv preprint arXiv:1902.10189* (2019). [arXiv:1902.10189 \[stat.ML\]](https://arxiv.org/abs/1902.10189) <https://arxiv.org/abs/1902.10189>
- [167] Xin Qiu and Risto Miikkulainen. 2024. Semantic Density: Uncertainty Quantification in Semantic Space for Large Language Models. *arXiv preprint arXiv:2405.13845* (2024).
- [168] Victor Quach, Adam Fisch, Tal Schuster, Adam Yala, Jae Ho Sohn, Tommi S Jaakkola, and Regina Barzilay. 2023. Conformal language modeling. *arXiv preprint arXiv:2306.10193* (2023).
- [169] Abdul Wahab Qurashi, Violeta Holmes, and Anju P Johnson. 2020. Document processing: Methods for semantic text similarity analysis. In *2020 international conference on INnovations in Intelligent SysTems and Applications (INISTA)*. IEEE, 1–6.
- [170] Alec Radford and Karthik Narasimhan. 2018. Improving Language Understanding by Generative Pre-Training.
- [171] Rahul Rahaman et al. 2021. Uncertainty quantification and deep ensembles. *Advances in neural information processing systems* 34 (2021), 20063–20075.
- [172] Daking Rai, Yilun Zhou, Shi Feng, Abulhair Saparov, and Ziyu Yao. 2024. A practical review of mechanistic interpretability for transformer-based language models. *arXiv preprint arXiv:2407.02646* (2024).
- [173] Vipula Rawte, Amit Sheth, and Amitava Das. 2023. A survey of hallucination in large foundation models. *arXiv preprint arXiv:2309.05922* (2023).
- [174] Siva Reddy, Danqi Chen, and Christopher D Manning. 2019. Coqa: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics* 7 (2019), 249–266.
- [175] N Reimers. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. *arXiv preprint arXiv:1908.10084* (2019).
- [176] David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. 2023. Gpqa: A graduate-level google-proof q&a benchmark. *arXiv preprint arXiv:2311.12022* (2023).
- [177] Allen Z Ren, Jaden Clark, Anushri Dixit, Masha Itkina, Anirudha Majumdar, and Dorsa Sadigh. 2024. Explore until Confident: Efficient Exploration for Embodied Question Answering. *arXiv preprint arXiv:2403.15941* (2024).
- [178] Allen Z Ren, Anushri Dixit, Alexandra Bodrova, Sumeet Singh, Stephen Tu, Noah Brown, Peng Xu, Leila Takayama, Fei Xia, Jake Varley, et al. 2023. Robots that ask for help: Uncertainty alignment for large language model planners. *arXiv preprint arXiv:2307.01928* (2023).



- [179] Jie Ren, Yao Zhao, Tu Vu, Peter J. Liu, and Balaji Lakshminarayanan. 2023. Self-Evaluation Improves Selective Generation in Large Language Models. In *Proceedings on "I Can't Believe It's Not Better: Failure Modes in the Age of Foundation Models" at NeurIPS 2023 Workshops (Proceedings of Machine Learning Research, Vol. 239)*, Javier Antorán, Arno Blaas, Kelly Buchanan, Fan Feng, Vincent Fortuin, Sahra Ghalebikesabi, Andreas Krieglér, Ian Mason, David Rohde, Francisco J. R. Ruiz, Tobias Uelwer, Yubin Xie, and Rui Yang (Eds.), PMLR, 49–64.
- [180] Pouria Rouzrokh, Shahriar Faghani, Cooper U Gamble, Moein Shariatnia, and Bradley J Erickson. 2024. CONFLARE: CONFormal LARge language model REtrieval. *arXiv preprint arXiv:2404.04287* (2024).
- [181] Mauricio Sadinle, Jing Lei, and Larry Wasserman. 2019. Least ambiguous set-valued classifiers with bounded error levels. *J. Amer. Statist. Assoc.* 114, 525 (2019), 223–234.
- [182] Patrick Schwab and Walter Karlen. 2019. Cxplain: Causal explanations for model interpretation under uncertainty. *Advances in neural information processing systems* 32 (2019).
- [183] Lei Sha, Oana-Maria Camburu, and Thomas Lukasiewicz. 2021. Learning from the best: Rationalizing predictions by adversarial information calibration. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 13771–13779.
- [184] Glenn Shafer and Vladimir Vovk. 2008. A tutorial on conformal prediction. *Journal of Machine Learning Research* 9, 3 (2008).
- [185] Dhruv Shah, Blażej Osiński, Sergey Levine, et al. 2023. Lm-nav: Robotic navigation with large pre-trained models of language, vision, and action. In *Conference on robot learning*. PMLR, 492–504.
- [186] Eric Michael Smith, Diana Gonzalez-Rico, Emily Dinan, and Y-Lan Boureau. 2020. Controlling style in generated dialogue. *arXiv preprint arXiv:2009.10855* (2020).
- [187] Claudio Spiess, David Gros, Kunal Suresh Pai, Michael Pradel, Md Rafiqul Islam Rabin, Amin Alipour, Susmit Jha, Prem Devanbu, and Toufique Ahmed. 2024. Calibration and correctness of language models for code. *arXiv preprint arXiv:2402.02047* (2024).
- [188] Sebastian Steindl, Ulrich Schäfer, Bernd Ludwig, and Patrick Levi. 2024. Linguistic Obfuscation Attacks and Large Language Model Uncertainty. In *Proceedings of the 1st Workshop on Uncertainty-Aware NLP (UncertainNLP 2024)*, 35–40.
- [189] Elias Stengel-Eskin, Peter Hase, and Mohit Bansal. 2024. LACIE: Listener-Aware Finetuning for Confidence Calibration in Large Language Models. *arXiv preprint arXiv:2405.21028* (2024).
- [190] Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. 2024. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing* 568 (2024), 127063.
- [191] Jiayuan Su, Jing Luo, Hongwei Wang, and Lu Cheng. 2024. Api is enough: Conformal prediction for large language models without logit-access. *arXiv preprint arXiv:2403.01216* (2024).
- [192] Xingpeng Sun, Yiran Zhang, Xindi Tang, Amrit Singh Bedi, and Aniket Bera. 2024. TrustNavGPT: Modeling Uncertainty to Improve Trustworthiness of Audio-Guided LLM-Based Robot Navigation. *arXiv preprint arXiv:2408.01867* (2024).
- [193] Zhongxiang Sun, Xiaoxue Zang, Kai Zheng, Yang Song, Jun Xu, Xiao Zhang, Weijie Yu, and Han Li. 2024. ReDeEP: Detecting Hallucination in Retrieval-Augmented Generation via Mechanistic Interpretability. *arXiv preprint arXiv:2410.11414* (2024).
- [194] Robert H Tai, Lillian R Bentley, Xin Xia, Jason M Sitt, Sarah C Fankhauser, Ana M Chicas-Mosier, and Barnas G Monteith. 2024. An examination of the use of large language models to aid analysis of textual data. *International Journal of Qualitative Methods* 23 (2024), 16094069241231168.
- [195] Alex Tamkin, Kunal Handa, Avash Shrestha, and Noah Goodman. 2022. Task ambiguity in humans and language models. *arXiv preprint arXiv:2212.10711* (2022).
- [196] Alex Tamkin, Mohammad Taufeeque, and Noah D Goodman. 2023. Codebook features: Sparse and discrete interpretability for neural networks. *arXiv preprint arXiv:2310.17230* (2023).
- [197] Zhisheng Tang, Ke Shen, and Mayank Kejriwal. 2024. An Evaluation of Estimative Uncertainty in Large Language Models. *arXiv preprint arXiv:2405.15185* (2024).
- [198] Sree Harsha Tanneru, Chirag Agarwal, and Himabindu Lakkaraju. 2024. Quantifying uncertainty in natural language explanations of large language models. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 1072–1080.
- [199] Shuchang Tao, Liuyi Yao, Hanxing Ding, Yuexiang Xie, Qi Cao, Fei Sun, Jinyang Gao, Huawei Shen, and Bolin Ding. 2024. When to Trust LLMs: Aligning Confidence with Response Quality. *arXiv preprint arXiv:2404.17287* (2024).
- [200] Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, Andy Jones, et al. 2024. Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet. <https://transformer-circuits.pub/2024/scaling-monosemanticity/index.html>
- [201] MTCAJ Thomas and A Thomas Joy. 2006. *Elements of information theory*. Wiley-Interscience.
- [202] James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and VERification. *arXiv preprint arXiv:1803.05355* (2018).
- [203] Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher D Manning. 2023. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. *arXiv preprint arXiv:2305.14975* (2023).

- [204] Christian Tomani, Kamalika Chaudhuri, Ivan Evtimov, Daniel Cremers, and Mark Ibrahim. 2024. Uncertainty-Based Abstention in LLMs Improves Safety and Reduces Hallucinations. *arXiv preprint arXiv:2404.10960* (2024).
- [205] SM Tonmoy, SM Zaman, Vinija Jain, Anku Rani, Vipula Rawte, Aman Chadha, and Amitava Das. 2024. A comprehensive survey of hallucination mitigation techniques in large language models. *arXiv preprint arXiv:2401.01313* (2024).
- [206] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288* (2023).
- [207] Yao-Hung Hubert Tsai, Walter Talbott, and Jian Zhang. 2024. Efficient Non-Parametric Uncertainty Quantification for Black-Box Large Language Models and Decision Planning. *arXiv preprint arXiv:2402.00251* (2024).
- [208] Dennis Ulmer, Martin Gubri, Hwaran Lee, Sangdoo Yun, and Seong Joon Oh. 2024. Calibrating Large Language Models Using Their Generations Only. *arXiv preprint arXiv:2403.05973* (2024).
- [209] Roman Vashurin, Ekaterina Fadeeva, Artem Vazhentsev, Akim Tsvigun, Daniil Vasilev, Rui Xing, Abdelrahman Boda Sadallah, Lyudmila Rvanova, Sergey Petrakov, Alexander Panchenko, et al. 2024. Benchmarking Uncertainty Quantification Methods for Large Language Models with LM-Polygraph. *arXiv preprint arXiv:2406.15627* (2024).
- [210] A Vaswani. 2017. Attention is all you need. *Advances in Neural Information Processing Systems* (2017).
- [211] Artem Vazhentsev, Ekaterina Fadeeva, Rui Xing, Alexander Panchenko, Preslav Nakov, Timothy Baldwin, Maxim Panov, and Artem Shelmanov. 2024. Unconditional Truthfulness: Learning Conditional Dependency for Uncertainty Quantification of Large Language Models. *arXiv preprint arXiv:2408.10692* (2024).
- [212] Apoorv Vyas, Nataraj Jammalamadaka, Xia Zhu, Dipankar Das, Bharat Kaul, and Theodore L Willke. 2018. Out-of-distribution detection using an ensemble of self supervised leave-out classifiers. In *Proceedings of the European conference on computer vision (ECCV)*. 550–564.
- [213] Hanjing Wang and Qiang Ji. 2024. Epistemic Uncertainty Quantification For Pre-Trained Neural Networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 11052–11061.
- [214] Jun Wang, Guocheng He, and Yiannis Kantaros. 2024. Safe Task Planning for Language-Instructed Multi-Robot Systems using Conformal Prediction. *arXiv preprint arXiv:2402.15368* (2024).
- [215] J. Wang, Jiaming Tong, Kai Liang Tan, Yevgeniy Vorobeychik, and Yiannis Kantaros. 2023. Conformal Temporal Logic Planning using Large Language Models: Knowing When to Do What and When to Ask for Help. *ArXiv abs/2309.10092* (2023).
- [216] Xi Wang, Laurence Aitchison, and Maja Rudolph. 2023. LoRA ensembles for large language model fine-tuning. *arXiv preprint arXiv:2310.00035* (2023).
- [217] Yiming Wang, Pei Zhang, Baosong Yang, Derek F Wong, and Rui Wang. 2024. Latent Space Chain-of-Embedding Enables Output-free LLM Self-Evaluation. *arXiv preprint arXiv:2410.13640* (2024).
- [218] Yu-Hsiang Wang, Andrew Bai, Che-Ping Tsai, and Cho-Jui Hsieh. 2024. CLUE: Concept-Level Uncertainty Estimation for Large Language Models. *arXiv preprint arXiv:2409.03021* (2024).
- [219] Zhiyuan Wang, Jinhao Duan, Lu Cheng, Yue Zhang, Qingni Wang, Hengtao Shen, Xiaofeng Zhu, Xiaoshuang Shi, and Kaidi Xu. 2024. ConU: Conformal Uncertainty in Large Language Models with Correctness Coverage Guarantees. *arXiv preprint arXiv:2407.00499* (2024).
- [220] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems* 35 (2022), 24824–24837.
- [221] Adina Williams, Nikita Nangia, and Samuel R Bowman. 2017. A broad-coverage challenge corpus for sentence understanding through inference. *arXiv preprint arXiv:1704.05426* (2017).
- [222] Luhuan Wu and Sinead A Williamson. 2024. Posterior Uncertainty Quantification in Neural Networks using Data Augmentation. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 3376–3384.
- [223] Yijun Xiao and William Yang Wang. 2021. On hallucination and predictive uncertainty in conditional language generation. *arXiv preprint arXiv:2103.15025* (2021).
- [224] Ziang Xiao, Xingdi Yuan, Q Vera Liao, Rania Abdelghani, and Pierre-Yves Oudeyer. 2023. Supporting qualitative analysis with large language models: Combining codebook with GPT-3 for deductive coding. In *Companion proceedings of the 28th international conference on intelligent user interfaces*. 75–78.
- [225] Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. 2023. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. *arXiv preprint arXiv:2306.13063* (2023).
- [226] Tianyang Xu, Shujin Wu, Shizhe Diao, Xiaozhe Liu, Xingyao Wang, Yangyi Chen, and Jing Gao. 2024. SaySelf: Teaching LLMs to Express Confidence with Self-Reflective Rationales. *arXiv preprint arXiv:2405.20974* (2024).
- [227] Ziwei Xu, Sanjay Jain, and Mohan Kankanhalli. 2024. Hallucination is inevitable: An innate limitation of large language models. *arXiv preprint arXiv:2401.11817* (2024).
- [228] Yasin Abbasi Yadkori, Ilja Kuzborskij, András György, and Csaba Szepesvári. 2024. To Believe or Not to Believe Your LLM. *arXiv preprint arXiv:2406.02543* (2024).
- [229] Adam X. Yang, Maxime Robeyns, Xi Wang, and Laurence Aitchison. 2024. Bayesian Low-rank Adaptation for Large Language Models. *arXiv preprint arXiv:2308.13111* (2024). [arXiv:2308.13111 \[cs.LG\] https://arxiv.org/abs/2308.13111](https://arxiv.org/abs/2308.13111)

- [230] Haoyan Yang, Yixuan Wang, Xingyin Xu, Hanyuan Zhang, and Yirong Bian. 2024. Can We Trust LLMs? Mitigate Overconfidence Bias in LLMs through Knowledge Transfer. *arXiv preprint arXiv:2405.16856* (2024).
- [231] Yuqing Yang, Ethan Chern, Xipeng Qiu, Graham Neubig, and Pengfei Liu. 2023. Alignment for honesty. *arXiv preprint arXiv:2312.07000* (2023).
- [232] Yuchen Yang, Houqiang Li, Yanfeng Wang, and Yu Wang. 2023. Improving the Reliability of Large Language Models by Leveraging Uncertainty-Aware In-Context Learning. *arXiv preprint arXiv:2310.04782* (2023).
- [233] Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. *arXiv preprint arXiv:1809.09600* (2018).
- [234] Fanghua Ye, Mingming Yang, Jianhui Pang, Longyue Wang, Derek F Wong, Emine Yilmaz, Shuming Shi, and Zhaopeng Tu. 2024. Benchmarking llms via uncertainty quantification. *arXiv preprint arXiv:2401.12794* (2024).
- [235] Gal Yona, Roei Aharoni, and Mor Geva. 2024. Can Large Language Models Faithfully Express Their Intrinsic Uncertainty in Words? *arXiv preprint arXiv:2405.16908* (2024).
- [236] Lei Yu, Meng Cao, Jackie Chi Kit Cheung, and Yue Dong. 2024. Mechanisms of non-factual hallucinations in language models. *arXiv preprint arXiv:2403.18167* (2024).
- [237] Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. Bartscore: Evaluating generated text as text generation. *Advances in Neural Information Processing Systems* 34 (2021), 27263–27277.
- [238] Zeyu Yun, Yubei Chen, Bruno A Olshausen, and Yann LeCun. 2021. Transformer visualization via dictionary learning: contextualized embedding as a linear superposition of transformer factors. *arXiv preprint arXiv:2103.15949* (2021).
- [239] Bianca Zadrozny and Charles Elkan. 2001. Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers. In *ICML*, Vol. 1. 609–616.
- [240] Bianca Zadrozny and Charles Elkan. 2002. Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*. 694–699.
- [241] Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? *arXiv preprint arXiv:1905.07830* (2019).
- [242] Qingcheng Zeng, Mingyu Jin, Qinkai Yu, Zhenting Wang, Wenye Hua, Zihao Zhou, Guangyan Sun, Yanda Meng, Shiqing Ma, Qifan Wang, et al. 2024. Uncertainty is Fragile: Manipulating Uncertainty in Large Language Models. *arXiv preprint arXiv:2407.11282* (2024).
- [243] Caiqi Zhang, Fangyu Liu, Marco Basaldella, and Nigel Collier. 2024. LUQ: Long-text Uncertainty Quantification for LLMs. *arXiv preprint arXiv:2403.20279* (2024).
- [244] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675* (2019).
- [245] Tianhang Zhang, Lin Qiu, Qipeng Guo, Cheng Deng, Yue Zhang, Zheng Zhang, Chenghu Zhou, Xinbing Wang, and Luoyi Fu. 2023. Enhancing uncertainty-based hallucination detection with stronger focus. *arXiv preprint arXiv:2311.13230* (2023).
- [246] Yuwei Zhang, Zihan Wang, and Jingbo Shang. 2023. Clusterllm: Large language models as a guide for text clustering. *arXiv preprint arXiv:2305.14871* (2023).
- [247] Haiyan Zhao, Hanjie Chen, Fan Yang, Ninghao Liu, Huiqi Deng, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, and Mengnan Du. 2024. Explainability for large language models: A survey. *ACM Transactions on Intelligent Systems and Technology* 15, 2 (2024), 1–38.
- [248] Qiwei Zhao, Xujiang Zhao, Yanchi Liu, Wei Cheng, Yiyu Sun, Mika Oishi, Takao Osaki, Katsushi Matsuda, Huaxiu Yao, and Haifeng Chen. 2024. SAUP: Situation Awareness Uncertainty Propagation on LLM Agent. *arXiv preprint arXiv:2412.01033* (2024).
- [249] Theodore Zhao, Mu Wei, J Preston, and Hoifung Poon. 2024. Pareto Optimal Learning for Estimating Large Language Model Errors. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 10513–10529.
- [250] Xinran Zhao, Hongming Zhang, Xiaoman Pan, Wenlin Yao, Dong Yu, Tongshuang Wu, and Jianshu Chen. 2024. Fact-and-Reflection (FaR) Improves Confidence Calibration of Large Language Models. *arXiv preprint arXiv:2402.17124* (2024).
- [251] Yao Zhao, Mikhail Khalman, Rishabh Joshi, Shashi Narayan, Mohammad Saleh, and Peter J Liu. 2022. Calibrating sequence likelihood improves conditional language generation. In *The eleventh international conference on learning representations*.
- [252] Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. In *International conference on machine learning*. PMLR, 12697–12706.
- [253] Zhi Zheng, Qian Feng, Hang Li, Alois Knoll, and Jianxiang Feng. 2024. Evaluating Uncertainty-based Failure Detection for Closed-Loop LLM Planners. *arXiv preprint arXiv:2406.00430* (2024).
- [254] Chiwei Zhu, Benfeng Xu, Quan Wang, Yongdong Zhang, and Zhendong Mao. 2023. On the calibration of large language models and alignment. *arXiv preprint arXiv:2311.13240* (2023).
- [255] Roland S Zimmermann, Thomas Klein, and Wieland Brendel. 2024. Scale alone does not improve mechanistic interpretability in vision models. *Advances in Neural Information Processing Systems* 36 (2024).

Received 7 December 2024; revised 16 May 2025; accepted 28 May 2025