Enhancing Uncertainty Quantification in Large Language Models through Semantic Graph Density

Zhaoye Li[1] **Siyuan Shen**[1] **Wenjing Yang**[1] **Ruochun Jin**[1] **Huan Chen**[1] **Ligong Cao**[1] **Jing Ren**[*1]

[1]College of Computer Science and Technology, National University of Defense Technology, Changsha, China

Abstract

Large Language Models (LLMs) excel in language understanding but are susceptible to "confabulation," where they generate arbitrary, factually incorrect responses to uncertain questions. Detecting confabulation in question answering often relies on Uncertainty Quantification (UQ), which measures semantic entropy or consistency among sampled answers. While several methods have been proposed for UQ in LLMs, they suffer from key limitations, such as overlooking fine-grained semantic relationships among answers and neglecting answer probabilities. To address these issues, we propose Semantic Graph Density (SGD). SGD quantifies semantic consistency by evaluating the density of a semantic graph that captures fine-grained semantic relationships among answers. Additionally, it integrates answer probabilities to adjust the contribution of each edge to the overall uncertainty score. We theoretically prove that SGD generalizes the previous state-of-the-art method, Deg, and empirically demonstrate its superior performance across four LLMs and four free-form question-answering datasets. In particular, in experiments with Llama3.1-8B, SGD outperformed the best baseline by 1.52% in AUROC on the CoQA dataset and by 1.22% in AUARC on the TriviaQA dataset.

1 INTRODUCTION

Large language models (LLMs) have shown impressive performance in language understanding and text generation across various domains [Zhao et al., 2023, Chang et al., 2024, Wei et al., 2022, Chi et al., 2024]. However, these models often encounter a critical issue known as "hallucination," where the generated content is either nonsensical or unfaithful to the provided source [Ji et al., 2023, Maynez et al., 2020, Filippova, 2020]. Hallucinations manifest in various forms and exhibit different characteristics across different tasks [Huang et al., 2023, Farquhar et al., 2024]. In this paper, we focus exclusively on one type of hallucination—confabulation—and limit the scope to short-form question answering (QA). Confabulation occurs when LLMs generate arbitrary, factually incorrect responses to uncertain questions [Farquhar et al., 2024], often arising when a query exceeds the model's knowledge boundaries [Huang et al., 2023]. For example, when asked "Which programming language has been used for implementing GWAR?", LLMs may answer "C++" or "Perl" inconsistently, even when the same question is posed.

Detecting confabulation in LLM-generated answers can be approached through Uncertainty Quantification (UQ), which assesses the likelihood that an LLM will generate a confabulated response to a given question [Farquhar et al., 2024]. Existing UQ methods include entropy-based [Farquhar et al., 2024, Nikitin et al., 2024] and graph-based approaches [Lin et al., 2024, Da et al., 2024]. Both of these approaches share the commonality of first sampling multiple possible answers and then evaluating their entropy or consistency. The most well-known entropy-based method is Semantic Entropy (SE) [Farquhar et al., 2024], which makes a notable contribution by using semantic equivalence clustering to mitigate lexical uncertainty. However, SE only considers whether two responses are semantically equivalent, overlooking finer semantic similarities [Nikitin et al., 2024]. Existing graph-based methods achieve effective modeling of semantic consistency through graphs, but they neglect answer probabilities. This introduces bias, as answers with higher probabilities are generally more representative and reliable for uncertainty quantification compared to answers with lower probabilities[1] [Geng et al., 2024].

[*] Corresponding author.

[1] In Appendix A, we present a detailed example illustrating the consequences of current graph-based methods overlooking answer probabilities.

To address these issues, we propose a novel UQ method, Semantic Graph Density (SGD). For the first issue, SGD measures semantic consistency by evaluating the density of a semantic graph that reflects fine-grained semantic relationships. A denser graph indicates higher consistency and lower uncertainty. For the second issue, we adjust the contribution of each edge in the semantic graph to the density based on the probability of the answers it connects. The higher the probabilities of the two answers forming an edge, the greater their contribution to the final uncertainty score. We theoretically demonstrate that where certain hyperparameters of SGD are specified, it generalizes the previous state-of-the-art graph-based method, Deg [Lin et al., 2024].

We evaluate the performance of SGD across various question-answering domains, including conversational settings (CoQA [Reddy et al., 2019]), trivia knowledge (TriviaQA [Joshi et al., 2017]), biomedical science (BioASQ [Tsatsaronis et al., 2015]), and natural questions (NQ [Kwiatkowski et al., 2019]) derived from real-world Google Search queries. Our evaluation focuses on two key aspects: (1) its capability to distinguish between correct and fabricated responses, and (2) the improvement in question-answering accuracy when high-uncertainty questions are rejected to answer. Experimental results across four LLMs demonstrate that SGD outperforms baseline methods, achieving enhanced efficiency and robustness in UQ.

2 RELATED WORK

Hallucinations and Confabulations in LLMs Hallucinations in LLMs manifest in various forms, including factual fabrication [Huang et al., 2023, Farquhar et al., 2024], instruction inconsistency [Huang et al., 2023], and reasoning failure [Berglund et al., 2024, Zheng et al., 2023]. This paper focuses on one specific type: confabulation, also known as fabrication, particularly in short-form QA scenarios. Fabrication in QA refers to the phenomenon where the LLM-generated answer is arbitrary and incorrect [Farquhar et al., 2024]. For example, when asked, "Which programming language has been used for implementing GWAR?" the model may confabulate by answering "C++" at one time and "Perl" at another, despite the same input question. A more intuitive and concrete example is shown in Table 5. One possible cause of this phenomenon is the model's tendency to generate an answer even when the query exceeds its knowledge boundaries [Huang et al., 2023]. This behavior aligns with the training objective of maximizing rewards for providing answers, resulting in an "overeager" tendency to respond instead of abstaining [Farquhar et al., 2024]. Detecting whether an LLM-generated answer is accurate or fabricated is an important research problem. Approaches include leveraging external knowledge sources, where LLM outputs are cross-referenced with established databases [Sui et al., 2024], or using an external LLM as a judge [Cohen et al., 2023]. Another method involves supervised learning, training classifiers on the LLM's internal states to distinguish between accurate and fabricated content [Azaria and Mitchell, 2023]. A different approach is uncertainty quantification, as discussed in the following paragraph.

Uncertainty Quantification in LLMs Uncertainty Quantification (UQ) is an effective method for assessing whether LLMs generate hallucinations [Farquhar et al., 2024, Lin et al., 2024]. Notably, the term "uncertainty" in this paper specifically refers to the degree of dispersion in the LLM's predicted distribution, rather than response confidence. The former depends exclusively on the input prompt, while the latter is influenced by both the input prompt and the output response. A higher uncertainty score indicates a greater likelihood of hallucinations [Farquhar et al., 2024]. Specifically for QA, it measures how likely an LLM is to produce a confabulated answer for a given question [Farquhar et al., 2024]. Recently, numerous UQ methods have emerged, differing in their approaches to uncertainty modeling and the types of information utilized, such as output text and token probabilities. UQ methods are classified into white-box and black-box categories based on their access to the LLM's internal workings and numerical outputs. Black-box methods only access the LLM's output text, whereas white-box methods have full access [Lin et al., 2024]. Semantic Entropy [Kuhn et al., 2023, Farquhar et al., 2024] serves as a gold standard for quantifying uncertainty in LLMs and represents a prominent white-box approach. Early methods combined lexical and semantic uncertainty, overlooking the fact that different lexical expressions can convey the same meaning. Semantic Entropy, by focusing exclusively on semantics, addresses this limitation and marks a significant advancement in UQ.

Representative black-box methods include Deg, Ecc, and EigV [Lin et al., 2024]. These methods construct a graph where edge weights reflect semantic similarity. EigV estimates the number of connected components in the graph by analyzing the eigenvalues of the graph Laplacian. In contrast, Deg and Ecc measure output diversity using the graph's degree matrix and the spectral embedding of its nodes, respectively. Although SGD, as well as Deg, Ecc, and EigV, all construct semantic graphs, the key difference lies in our approach's use of graph density in the semantic graph as a consistency proxy. Additionally, we address the bias issues inherent in these methods by incorporating token probability.

Beyond these key methods, additional approaches include Discrete Semantic Entropy [Farquhar et al., 2024], a black-box approximation of Semantic Entropy; Kernel Language Entropy [Nikitin et al., 2024], which extends Semantic Entropy by incorporating fine-grained semantic relations beyond equivalence; D-UE [Da et al., 2024], which addresses limitations in using average entailment probabilities from bidirectional Natural Language Inference (NLI) models to evaluate response similarity; and SEU [Grewal et al., 2024], which utilizes transformer-based sentence embeddings to provide a smoother and more robust estimation of semantic similarities in UQ.

There is an increasing focus on quantifying uncertainty in long-form answers [Zhang et al., 2024, Jiang et al., 2024, Fang et al., 2025]. However, this paper specifically focuses on short-form answers, which are defined as single-proposition responses to a question [Farquhar et al., 2024]. These answers are typically concise, comprising only a few words or, at most, a single sentence, in contrast to more extensive paragraphs.

Complementary methods Kuhn et al. [2023] and Aichberger et al. [2025] emphasized that UQ benefits from semantically diverse yet likely output sequences. Aichberger et al. [2025] experimentally demonstrated that Diverse Beam Search [Vijayakumar et al., 2018] and the Semantically Diverse Language Generation (SDLG) method improve the performance of sample-based UQ methods. These techniques can be incorporated into SGD to further enhance its effectiveness.

3 SEMANTIC GRAPH DENSITY

In question answering (QA), given an input prompt x (i.e., a question with or without context) and an LLM, the goal is to evaluate how likely the LLM is to generate a fabricated answer for the given prompt. It is important to emphasize that the objective is to derive a *relative* score indicating the potential for confabulated output, rather than calculating the exact probability of the model's correctness (which is related to *model calibration* [Zhu et al., 2023, Guo et al., 2017], an orthogonal research topic). A higher uncertainty score corresponds to a greater potential for confabulation.

The first two steps of Semantic Graph Density (SGD) involve sampling multiple possible answers (Step 1) and measuring the fine-grained semantic relationships among them (Step 2). In Step 3, these relationships are used to construct a semantic graph. Based on this graph, we compute the graph density and adjust each edge's contribution according to the probabilities of the connected answers. Edges linking answers with higher probabilities contribute more to the final uncertainty score.

Step 1. Sample N possible answers. Following Farquhar et al. [2024] and Nikitin et al. [2024], for an input x, we sample N possible answers $\{y^{(i)}\}_{i=1}^{N}$, where each answer is represented as a sequence of tokens $y^{(i)} = [y_1^{(i)}, y_2^{(i)}, \dots, y_{L_i}^{(i)}]$, with $y_j^{(i)}$ denoting the j-th output token of $y^{(i)}$. We then compute the corresponding length-normalized probability [2] [Murray and Chiang, 2018] for each answer: $\{P(y^{(i)}|x)\}_{i=1}^{N}$, with $P(y^{(i)}|x) = \prod_{j=1}^{L_i} P(y_j^{(i)}|y_{<j}^{(i)}, x)^{1/L_i}$, where $y_{<j}^{(i)}$ represents the sequence of tokens preceding $y_j^{(i)}$.

Step 2. Compute pairwise semantic similarities among N possible answers. The output logits of Natural Language Inference (NLI) models have been demonstrated to effectively measure the semantic similarity between two responses within a given textual context [Lin et al., 2024]. We follow the best practice in [Lin et al., 2024] to measure the semantic similarity between any two sampled answers $y^{(i)}$ and $y^{(j)}$ within the context x. The NLI model we employ is DeBERTa-Large-MNLI[3] [He et al., 2021].

We concatenate x with $y^{(i)}$ and $y^{(j)}$ to yield $x \oplus y^{(i)}$ and $x \oplus y^{(j)}$ (see Appendix D for the input format used to generate the output logits of the NLI model) and feed them into the NLI model twice. In the first pass, $x \oplus y^{(i)}$ is regarded as the premise and $x \oplus y^{(j)}$ as the hypothesis. Conversely, in the second pass, $x \oplus y^{(j)}$ is regarded as the premise, and $x \oplus y^{(i)}$ as the hypothesis. We apply the softmax function to the predicted logits from the NLI model and take the average of the entailment logits from the two passes as the similarity score.

$$s_{i,j} = \frac{1}{2}\big(\hat{p}_{entail}(x \oplus y^{(i)}, x \oplus y^{(j)}) + \hat{p}_{entail}(x \oplus y^{(j)}, x \oplus y^{(i)})\big) \tag{1}$$

Step 3. Construct a semantic graph and compute the semantic graph density. We first construct a semantic graph to capture fine-grained semantic relationships between answers and then adapt graph density to quantify semantic consistency. Given an undirected simple graph $G = (V, E)$, graph density is defined as the ratio of the number of edges $|E|$ to the maximum possible number of edges [ERDdS and R&wi, 1959].

$$D = \frac{|E|}{\binom{|V|}{2}} = \frac{|E|}{|V|(|V|-1)/2} \tag{2}$$

Each answer is treated as a node in the graph. The adjacency matrix is represented as $W = [w_{ij}]_{N \times N}$. In practice, an edge is established between nodes i and j if their similarity score s_{ij} exceeds a threshold δ. This relationship is formally defined as follows:

$$w_{ij} := \mathbb{1}_{s_{ij} > \delta}. \tag{3}$$

Under this design, *a denser graph signifies greater semantic consistency and reduced uncertainty*. We define SGD as

[2] In this paper, all probabilities refer to length-normalized probabilities [Murray and Chiang, 2018], a commonly used method to correct for length bias in sequence probabilities.

[3] https://huggingface.co/microsoft/deberta-large-mnli

follows:

$$\mathrm{SGD}_\delta(x) = \frac{-|E|}{|V|(|V|-1)/2} = -\sum_{\substack{i,j\in[N],\\ i<j}} \frac{\mathbb{1}_{s_{ij}>\delta}}{N(N-1)/2}. \tag{4}$$

Another approach is to directly define the weight w_{ij} of each edge as the similarity s_{ij}, i.e., $w_{ij} := s_{ij}$. However, graph density traditionally applies only to binary graphs. Since the weights defined by the similarity are non-negative and bounded within the range $[0, 1]$, we extend the definition of graph density by calculating the total sum of all edge weights divided by the total sum of the maximum possible weights of all edges. Under this design, SGD is defined as:

$$\begin{aligned}\mathrm{SGD}_s(x) &= -\sum_{i,j\in[N],i<j} \frac{2w_{ij}}{|V|(|V|-1)\cdot \sup_{\substack{m,n\in[N]\\ m<n}} w_{mn}} \\ &= -\sum_{i,j\in[N],i<j} \frac{2s_{ij}}{N(N-1)\cdot 1} \\ &= -\sum_{i,j\in[N],i<j} \frac{s_{ij}}{N(N-1)/2}.\end{aligned} \tag{5}$$

Here, $\sup_{m,n,m<n} w_{mn}$ represents the maximum weight that can be assigned to any edge, which corresponds to the maximum pairwise similarity that can be assigned between responses $\{y^{(i)}\}_{i=1}^{N}$. Since the similarities are computed from the output logits of the NLI model, which are transformed via softmax and bounded within the range of $[0, 1]$, we have $\sup_{m,n,m<n} w_{mn} = \sup_{m,n,m<n} s_{mn} = 1$.

Probability Incorporation In Equations 4 and 5, the numerator (s_{ij} or $\mathbb{1}_{s_{ij}>\delta}$) represents the weights of the edges. Each edge contributes equally to the uncertainty score at $1/(N(N-1)/2)$, as determined by the denominator $N(N-1)/2$. However, the sampled answers are not necessarily equally probable, meaning the edges formed by pairs of answers may not have equal probabilities. The length-normalized probability of an answer reflects the LLM's confidence in that answer [Geng et al., 2024]. If an edge $\{y^{(i)}, y^{(j)}\}$ has a higher probability compared to other edges, this indicates that the LLM has higher confidence in producing $\{y^{(i)}, y^{(j)}\}$. As such, $\{y^{(i)}, y^{(j)}\}$ should carry more weight in UQ, contributing more significantly to the final uncertainty score.

We define the contribution of each edge as $\mu(i, j)$, $\sum_{i,j\in[N],i<j} \mu(i, j) = 1$. SGD is further calculated as:

$$\begin{aligned}\mathrm{SGD}_{\delta+P}(x) &= -\sum_{i,j\in[N],i<j} \mathbb{1}_{s_{ij}>\delta}\cdot \mu(i,j), \\ \mathrm{SGD}_{s+P}(x) &= -\sum_{i,j\in[N],i<j} s_{ij}\cdot \mu(i,j)\end{aligned} \tag{6}$$

We use the notation "$+P$" to denote probability incorporation. As each sampling is mutually independent, the categorical distribution of the relative occurrence of $\{y^{(i)}, y^{(j)}\}$ is estimated as:

$$P(y^{(i)}, y^{(j)}|x) = \frac{P(y^{(i)}|x)\cdot P(y^{(j)}|x)}{\sum_{\substack{i,j\in[N]\\ i<j}} P(y^{(i)}|x)\cdot P(y^{(j)}|x)} \tag{7}$$

We define $\mu(i, j)$ as a convex combination of $1/(N(N-1)/2)$ and Equation 7.

$$\begin{aligned}\mu(i,j) = &\frac{\theta}{N(N-1)/2} + (1-\theta)\cdot P(y^{(i)}, y^{(j)}|x), \\ &i,j\in[N], i<j.\end{aligned} \tag{8}$$

Algorithm 1 summarizes the SGD procedure.

Generalization towards Deg [Lin et al., 2024] Deg constructs a semantic graph based on the pairwise similarity of answers, where the edge weights are defined as $w_{ij} := s_{ij}$, $i, j \in [N]$. The degree matrix is defined as $D = \mathrm{diag}(d_1, d_2, \ldots, d_N)$, where $d_i = \sum_{j\in[N]} s_{ij}$ represents the degree of node i. Next, we explain how to derive an approximation of Deg [Lin et al., 2024] from SGD_s (equivalent to SGD_{s+P} when $\theta = 1$).

$$\begin{aligned}\mathrm{Deg}(x) &= \mathrm{tr}(N\mathbb{I} - D)/N^2 \\ &= \left(N\,\mathrm{tr}(\mathbb{I}) - \mathrm{tr}(D)\right)/N^2 \\ &= \frac{1}{N^2}\left(N^2 - \sum_{i,j\in[N]} s_{ij}\right) \\ &= 1 - \frac{1}{N^2}\left(\sum_{\substack{i,j\in[N]\\ i=j}} s_{ij} + 2\sum_{\substack{i,j\in[N]\\ i<j}} s_{ij}\right)\end{aligned} \tag{9}$$

In Deg, s_{ij} $(i, j \in [N], i = j)$ is calculated using the output logits of the NLI model. When $y^{(i)}$ is identical to $y^{(j)}$, the softmax logit for entailment is nearly equal to 1 (e.g., 0.998), though it can never be exactly 1. Thus, s_{ij} $(i, j \in [N], i = j)$ can be approximated as 1. By further combining the following equation,

$$\begin{aligned}2\sum_{\substack{i,j\in[N]\\ i<j}} s_{ij} &= N(N-1)\sum_{\substack{i,j\in[N]\\ i<j}} \frac{s_{ij}}{N(N-1)/2} \\ &= -N(N-1)\,\mathrm{SGD}_s(x),\end{aligned} \tag{10}$$

we can derive the following:

$$\begin{aligned}\mathrm{Deg}(x) &\approx 1 - \frac{1}{N} + \frac{N-1}{N}\mathrm{SGD}_s(x) \\ &= \frac{N-1}{N}(1 + \mathrm{SGD}_s(x)).\end{aligned} \tag{11}$$

It is evident that the approximation of $\mathrm{Deg}(x)$ is a linear combination of $\mathrm{SGD}_s(x)$, and linearly scaling the uncertainty scores does not alter their relative discriminative capability (as previously mentioned, we require a relative score to determine whether a prompt will generate a correct or confabulated response). Ablation experiments (refer to Section 4.3) demonstrate that SGD_s and Deg achieve nearly identical performance.

Algorithm 1: Semantic Graph Density

```
Input: An input prompt $x$ (i.e. a question with/without
       context), an LLM, the number of possible
       responses $N$, hyperparameters $\delta$ and $\theta$.
Output: Semantic Graph Density.
Step 1: Sample $N$ possible answers.
Sample $N$ possible answers $\{y^{(i)}\}_{i=1}^{N}$ based on the
  input prompt $x$, and compute the length-normalized
  probability for each answer, resulting in
  $\{P(y^{(i)}|x)\}_{i=1}^{N}$.
Step 2: Compute pairwise semantic similarities
  among $N$ possible answers.
Compute the pairwise semantic similarities for any pair
  of $N$ possible answers, resulting in $s_{ij}$, where
  $s_{i,j} = \big(\hat{p}_{entail}(x \oplus y^{(i)}, x \oplus y^{(j)}) + \hat{p}_{entail}(x \oplus y^{(j)}, x \oplus y^{(i)})\big)/2$
Step 3: Compute the semantic graph density.
$\mathrm{SGD}_{\delta}(x) = -\sum_{i,j\in[N],i<j} \frac{\mathbf{1}_{s_{ij}>\delta}}{N(N-1)/2}$,
$\mathrm{SGD}_{s}(x) = -\sum_{i,j\in[N],i<j} \frac{s_{ij}}{N(N-1)/2}$,
$\mathrm{SGD}_{\delta+P}(x) = -\sum_{i,j\in[N],i<j} \mathbf{1}_{s_{ij}>\delta} \cdot \mu(i,j)$,
$\mathrm{SGD}_{s+P}(x) = -\sum_{i,j\in[N],i<j} s_{ij} \cdot \mu(i,j)$,
where $\mu(i,j) = \frac{\theta}{N(N-1)/2} + (1-\theta) \cdot P(y^{(i)}, y^{(j)}|x)$,
$P(y^{(i)}, y^{(j)}|x) = \frac{P(y^{(i)}|x) \cdot P(y^{(j)}|x)}{\sum_{i,j\in[N],i<j} P(y^{(i)}|x) \cdot P(y^{(j)}|x)}$,
  $i, j \in [N], i < j$.
```

Computational Cost We focus solely on the resource consumption arising from language model inference, as no other operation is more computationally expensive. Reviewing the entire process of calculating SGD, model inferences are only required in the first and second steps. The first step involves sampling N possible answers, requiring N LLM inferences, which can be executed in parallel. The second step demands $2 \cdot \binom{N}{2}$ inferences by the NLI model to calculate the semantic similarity s_{ij} for each pair $\{y^{(i)}, y^{(j)}\}$ ($i < j$). Computing s_{ij} requires two inferences to obtain $\hat{p}_{entail}(x \oplus y^{(i)}, x \oplus y^{(j)})$ and $\hat{p}_{entail}(x \oplus y^{(j)}, x \oplus y^{(i)})$. These $N(N-1)$ inferences can also be performed in parallel. The NLI model used in this study is DeBERTa-Large-MNLI, with approximately 150 million parameters. In contrast to LLMs, which process over a billion parameters to generate a single token, the computational cost of NLI models is relatively minimal.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUPS

Datasets and LLMs We consider four generative question-answering tasks for evaluation, including the open-book conversational QA dataset CoQA [Reddy et al., 2019], the closed-book QA dataset TriviaQA [Joshi et al., 2017], the biomedical QA dataset BioASQ [Tsatsaronis et al., 2015], and Natural Questions [Kwiatkowski et al., 2019]. We use the development split of CoQA, which contains 7,983 questions, the deduplicated validation split of TriviaQA (rc.noncontext subset) with 9,960 questions, the validation split of NQ with 3,610 questions, and the training split of BioASQ with 2,814 questions[4]. We utilize four popular off-the-shelf instruction-tuned LLMs for evaluation, with model sizes ranging from 1B to 12B parameters. These models include Llama-3.2-1B[5], Llama-3.1-8B[6], Mistral-7B-v0.3[7] and Mistral-Nemo-12B[8].

Evaluation Metric We use AUROC (Area Under the Receiver Operating Characteristic Curve) to evaluate how well the uncertainty scores distinguish between correct and incorrect answers. AUROC indicates the probability that a randomly selected correct generation has a lower uncertainty score than a randomly selected incorrect generation. An AUROC of 0.5 indicates that the assigned uncertainty score is no better than random guessing, meaning it cannot effectively differentiate between correct and incorrect answers. An AUROC of 1 signifies perfect discrimination, where all correct answers are assigned lower uncertainty scores than all incorrect answers.

Additionally, QA accuracy can be improved by rejecting questions with high uncertainty. This improvement is quantified using the AUARC (Area Under the Accuracy-Rejection Curve) [Nadeem et al., 2009], which measures the area under the accuracy-rejection curve at various thresholds. The rejection accuracy at a given threshold is determined by the accuracy of the remaining answers after rejecting those with uncertainty scores above this threshold.

Answer Generation For each question, we generated 10 answers using nucleus sampling ($P = 0.9$) and top-K sampling ($K = 50$) at a temperature of $T = 1$, following Farquhar et al. [2024] and Nikitin et al. [2024]. To assess

[4] http://participants-area.bioasq.org/Tasks/10b/trainingDataset/

[5] https://huggingface.co/meta-llama/Llama-3.2-1B-Instruct

[6] https://huggingface.co/meta-llama/Llama-3.1-8B-Instruct

[7] https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.3

[8] https://huggingface.co/mistralai/Mistral-Nemo-Instruct-2407

model accuracy, we generated a single answer at $T = 0.1$ and prompted GPT-4-0613 to verify whether this response aligned with any ground truths provided by the datasets, as adopted by Farquhar et al. [2024]. The prompts used for generating the answers and conducting correctness checks are detailed in Appendix B and Appendix E, respectively.

Baselines We included nine UQ methods for comparison. These baselines are categorized as follows: (1) entropy-based methods, including Semantic Entropy (**SE**) [Farquhar et al., 2024], Discrete Semantic Entropy (**DSE**) [Farquhar et al., 2024] and Kernel Language Entropy (**KLE**) [Nikitin et al., 2024]; (2) graph-based methods, including **Ecc** [Lin et al., 2024], **EigV** [Lin et al., 2024], **Deg** [Lin et al., 2024] and **D-UE** [Da et al., 2024]; and (3) consistency-based methods, including Number of Semantic Sets (**NSS**) [Kuhn et al., 2023] and Semantic Embedding Uncertainty (**SEU**) [Grewal et al., 2024]. For SE and DSE, we used GPT-3.5-Turbo-0125 for entailment prediction as recommended by Farquhar et al. [2024]. For KLE, we employ $\mathrm{KLE}(K_{HEAT})$, as it demonstrates the best performance among all variants of KLE. To ensure fairness, the NLI model utilized in KLE, Ecc, EigV, Deg, and D-UE is identical to ours, specifically DeBERTa-Large-MNLI. For further details, refer to Appendix F.

Implementation Details We repeated the experiment five times, each time randomly selecting 10 QA pairs from the dataset as context examples and subsequently dividing the remaining dataset into a validation set (1,000 QA pairs; 400 QA pairs for BioASQ) and a test set. Hyperparameter tuning for our method was conducted on the validation set. The hyperparameters δ and θ were selected from the set {0.01, 0.1, 0.2, ..., 0.9, 0.99} to maximize validation set performance. Finally, we evaluated the performance on the test data. Since both KLE and Ecc require hyperparameter tuning, to ensure fairness, all methods, including ours, were tuned using the same subset of the dataset. All experiments were conducted on a server equipped with one NVIDIA A100 (80GB) GPU.

4.2 MAIN RESULTS

Table 1 and Table 2 present the AUROC and AUARC scores for various uncertainty metrics across 16 model-dataset combinations. SGD_{s+P} outperforms baseline methods in 15 out of 16 cases for both AUROC and AUARC. For instance, when evaluated on the CoQA dataset using Llama-3.1-8B, SGD_{s+P} achieved an average AUROC of 81.52%, surpassing the best baseline result by 1.52%. SGD_{s+P} consistently outperforms SE due to its utilization of fine-grained semantic relations rather than relying solely on semantic equivalence. Compared to graph-based methods (e.g., Deg, Ecc) and KLE, both these baselines and SGD_{s+P} utilize semantic similarity measured by the NLI model. However, the advantage of SGD_{s+P} lies in its incorporation of probability to adjust the contribution of each edge of the semantic graph to the uncertainty score, which our subsequent ablation experiments validate as effective (see Section 4.3). Among $\mathrm{SGD}_{\delta+P}$ and SGD_{s+P}, the latter is the best as it has superior performance and requires only one hyperparameter, while the former needs two.

4.3 ABLATION EXPERIMENTS

Number of Possible Answers In the main experiment, we sampled $N = 10$ possible answers for each question. In this section, we investigate how the performance changes as the number of possible answers increases. We selected SE, KLE, and Deg as baselines, as these three methods have demonstrated the strongest performance among numerous baseline approaches. We compared our best-performing method, SGD_{s+P}, with these three baselines. The experiments were conducted using Mistral-Nemo-12B on the TriviaQA dataset. In these experiments, N varied from 3 to 10. Except for the variation in N, all other experimental settings were identical to those in the main experiment. The results, as shown in Figure 1, demonstrate that SGD_{s+P} is more generation-efficient compared to the baseline methods. Specifically, to achieve comparable AUROC or AUARC scores, SGD_{s+P} generates fewer possible answers, resulting in reduced computational resource consumption.

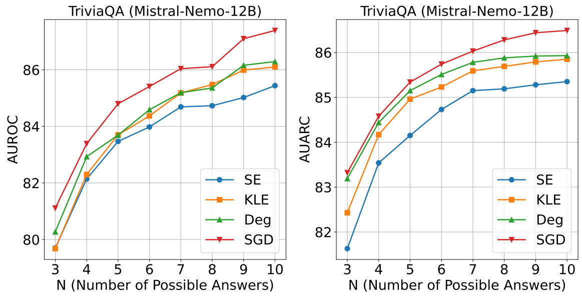

Figure 1: Performance of different uncertainty metrics with increasing numbers of possible answers. The number of possible answers ranges from 3 to 10, incremented by 1 at each step. We only included competitive baseline methods SE, KLE and Deg for comparison.

Effectiveness of Probability Incorporation In Section 3, we presented how to adjust the contribution of each edge in the semantic graph to the uncertainty based on the probability of the paired nodes, namely, the probability of the paired answers. Subsequently, we verified its effectiveness through experiments. Specifically, we compared the performance of SGD_{δ} and $\mathrm{SGD}_{\delta+P}$, as well as that of SGD_{s} and SGD_{s+P}. We verified the effectiveness of probability incorporation using Mistral-7B-v0.3 on the NQ and BioASQ datasets. Results are shown in Table 3. The results on AUROC and AUARC indicate that the incorporation of probability is ef-

Table 1: Performance (AUROC) comparison of various uncertainty metrics. All results are presented as percentages. For each model-dataset combination, the best average result from both baseline methods and our proposed methods is underlined, while the overall best result among all methods is highlighted in **bold**.

Datasets	**Entropy-based Methods**			**Graph-based Methods**				**Consistency-based Methods**		**Ours**	
	SE	DSE	KLE	Ecc	EigV	Deg	D-UE	NSS	SEU	$\text{SGD}_{\delta+P}$	SGD_{s+P}
Llama-3.2-1B											
NQ	$\underline{77.48\pm0.55}$	76.50±0.53	75.36±0.50	76.66±0.62	75.87±0.57	77.18±0.51	71.93±0.63	76.43±0.56	67.38±0.72	76.89±0.55	**$\underline{78.29\pm0.55}$**
CoQA	73.73±0.21	73.22±0.22	74.59±0.28	73.84±0.29	70.82±0.25	$\underline{75.73\pm0.27}$	73.95±0.28	72.51±0.22	69.80±0.27	75.66±0.44	**$\underline{76.35\pm0.29}$**
BioASQ	86.87±0.45	86.76±0.47	86.73±0.40	86.79±0.45	85.53±0.42	$\underline{87.25\pm0.39}$	85.62±0.44	86.36±0.44	78.78±0.51	**$\underline{87.56\pm0.39}$**	87.32±0.38
TriviaQA	$\underline{82.17\pm0.18}$	81.15±0.16	80.41±0.18	81.13±0.18	78.84±0.16	81.64±0.16	79.25±0.13	80.51±0.15	77.04±0.14	82.24±0.18	**$\underline{82.44\pm0.20}$**
Average	80.06	79.41	79.27	79.61	77.77	$\underline{80.45}$	77.69	78.95	73.25	80.59	**$\underline{81.10}$**
Llama-3.1-8B											
NQ	78.30±0.43	77.88±0.47	77.55±0.44	77.73±0.46	76.26±0.41	$\underline{78.64\pm0.44}$	75.00±0.42	77.48±0.47	71.03±0.44	78.84±0.39	**$\underline{78.87\pm0.42}$**
CoQA	75.26±0.36	74.89±0.35	78.92±0.27	76.97±0.40	71.75±0.33	$\underline{80.04\pm0.24}$	77.90±0.30	74.14±0.35	72.71±0.33	78.59±0.44	**$\underline{81.56\pm0.28}$**
BioASQ	83.40±0.47	83.35±0.47	84.28±0.45	83.03±0.58	81.32±0.42	$\underline{84.73\pm0.46}$	82.59±0.57	82.45±0.51	74.81±0.74	84.93±0.42	**$\underline{85.42\pm0.48}$**
TriviaQA	85.95±0.11	85.23±0.13	85.67±0.12	84.97±0.24	83.27±0.12	$\underline{86.23\pm0.12}$	84.51±0.13	84.42±0.13	81.95±0.13	86.47±0.13	**$\underline{87.17\pm0.12}$**
Average	80.73	80.34	81.61	80.68	78.15	$\underline{82.41}$	80.00	79.62	75.13	82.21	**$\underline{83.26}$**
Mistral-7B-v0.3											
NQ	76.88±0.60	76.88±0.60	$\underline{77.58\pm0.55}$	77.24±0.46	76.62±0.37	77.42±0.56	76.15±0.45	76.67±0.57	71.85±0.43	77.68±0.60	**$\underline{78.39\pm0.60}$**
CoQA	75.82±0.33	75.76±0.29	77.60±0.21	78.11±0.35	72.18±0.27	$\underline{79.61\pm0.28}$	78.44±0.26	75.32±0.28	73.47±0.25	78.81±0.55	**$\underline{80.58\pm0.25}$**
BioASQ	80.86±0.53	80.90±0.50	$\underline{83.66\pm0.41}$	83.05±0.50	82.66±0.50	83.57±0.53	80.54±0.55	80.98±0.49	67.84±0.41	83.88±0.60	**$\underline{84.56\pm0.54}$**
TriviaQA	83.76±0.29	83.53±0.28	83.86±0.28	83.74±0.11	82.80±0.12	$\underline{85.04\pm0.28}$	83.58±0.28	82.98±0.28	79.59±0.12	85.48±0.26	**$\underline{86.27\pm0.27}$**
Average	79.33	79.27	80.68	80.54	78.57	$\underline{81.41}$	79.68	78.99	73.19	81.46	**$\underline{82.45}$**
Mistral-Nemo-12B											
NQ	76.78±0.59	76.35±0.57	$\underline{77.78\pm0.58}$	76.55±0.47	76.28±0.58	76.92±0.52	73.04±0.44	75.84±0.56	69.53±0.39	**$\underline{78.34\pm0.50}$**	77.14±0.54
CoQA	76.08±0.19	75.72±0.24	78.09±0.19	77.25±0.19	71.11±0.24	$\underline{79.10\pm0.14}$	77.01±0.16	75.05±0.23	72.41±0.20	78.26±0.73	**$\underline{80.39\pm0.18}$**
BioASQ	81.66±0.48	81.58±0.56	$\underline{84.54\pm0.44}$	82.20±0.39	81.90±0.61	83.60±0.38	79.55±0.44	80.91±0.57	69.64±0.49	84.03±0.43	**$\underline{84.54\pm0.37}$**
TriviaQA	85.44±0.10	84.88±0.19	86.10±0.10	84.61±0.14	83.31±0.11	$\underline{86.29\pm0.11}$	84.29±0.11	84.07±0.09	81.47±0.11	86.83±0.17	**$\underline{87.39\pm0.11}$**
Average	79.99	79.63	$\underline{81.63}$	80.15	78.15	81.48	78.47	78.97	73.26	81.87	**$\underline{82.37}$**

fective, as demonstrated by the fact that the performance of SGD_δ is inferior to that of $\text{SGD}_{\delta+P}$, and the performance of SGD_s is inferior to that of SGD_{s+P}. In addition, we draw the following conclusions: (1) SGD_s and Deg exhibit nearly identical performance, consistent with the theoretical analysis in Section 3; and (2) the superior performance of SGD_{s+P} over baseline methods KLE and Deg can be primarily attributed to the incorporation of probability. This is evident from the fact that SGD_s yields nearly identical results to Deg but performs worse than KLE in the Mistral-7B experiment[9]; after incorporating probability, SGD_{s+P} outperforms both KLE and Deg.

Results of Diverse Beam Search Previous studies predominantly utilized multinomial sampling to generate multiple answers, often setting $T = 1$ to increase diversity. Few studies explored alternative sampling methods. In this paper, we conducted an ablation study using Diverse Beam Search [Vijayakumar et al., 2018] because it tends to produce diverse yet highly probable responses [Vijayakumar et al., 2018], which is crucial for UQ [Aichberger et al., 2025]. We sampled 10 answers for each question by configuring 10 groups, with each group containing one beam. The answer from the first group, generated via greedy beam search, was used to evaluate model accuracy. The subsequent groups, designed to introduce greater diversity, provided the possible answers used to assess consistency. We compared our best method, SGD_{s+P}, with several competitive baseline methods, including SE, DSE, KLE, and Deg. The results, shown in Figure 2, indicate that under Diverse Beam Search, KLE performs worse than SE, which contrasts with the multinomial sampling experiments where KLE outperforms SE in most cases. However, our method, SGD_{s+P}, still outperforms other competitive baseline methods, demonstrating the stronger robustness of our approach.

4.4 COMPARISON OF COMPUTATIONAL RESOURCE CONSUMPTION

Previous experiments have demonstrated that SGD_{s+P} outperforms baseline methods. In this section, we specifically compare the resource consumption of different UQ methods, concentrating solely on the consumption of language model inferences, as these operations are significantly more computationally intensive than other components. All methods are sampling-based and first require sampling N possible answers, which incurs the same consumption. To improve the accuracy of SE and DSE, we use GPT-3.5-turbo-0125 for entailment prediction, following Farquhar et al. [2024]. This approach requires at most $O(N^2)$ model inferences, where N is the number of statements to be compared. We also apply this strategy to NSS. KLE and all graph-based methods need to use the output logits of the NLI model to obtain the

[9] In the experiments with the Mistral-7B-BioASQ combination, KLE outperforms Deg. However, as shown in Tables 1 and 2, Deg generally outperforms KLE across most cases.

Table 2: Performance (AUARC) comparison of various uncertainty metrics. All results are presented as percentages. For each model-dataset combination, the best average result from both baseline methods and our proposed methods is underlined, while the overall best result among all methods is highlighted in **bold**.

Datasets	Entropy-based Methods			Graph-based Methods				Consistency-based Methods		Ours	
	SE	DSE	KLE	Ecc	EigV	Deg	D-UE	NSS	SEU	SGD$_{\delta+P}$	SGD$_{s+P}$
Llama-3.2-1B											
NQ	27.54±0.63	27.43±0.63	27.08±0.58	27.75±0.49	26.50±0.53	28.10±0.57	25.91±0.52	27.20±0.54	23.70±0.50	27.62±0.67	**28.56±0.58**
CoQA	86.97±0.28	86.32±0.16	87.65±0.11	87.25±0.14	85.28±0.12	87.98±0.28	87.45±0.10	86.08±0.26	85.80±0.11	87.87±0.17	**88.34±0.29**
BioASQ	71.25±0.86	71.01±0.85	70.64±0.93	70.63±0.90	69.37±0.88	70.88±0.83	70.11±0.86	70.62±0.91	65.72±0.81	**71.27±0.84**	71.22±0.83
TriviaQA	52.04±0.24	51.48±0.23	51.39±0.37	51.55±0.24	49.35±0.21	52.22±0.24	50.47±0.21	50.89±0.23	48.30±0.22	52.37±0.27	**53.62±0.23**
Average	59.45	59.06	59.19	59.30	57.63	59.80	58.49	58.70	55.88	59.78	**60.44**
Llama-3.1-8B											
NQ	51.74±1.02	51.10±1.11	51.39±1.06	51.11±0.90	49.51±1.00	52.10±0.92	49.66±0.97	50.71±1.20	46.83±0.81	51.62±0.97	**53.18±0.91**
CoQA	94.74±0.29	94.79±0.30	96.02±0.19	95.80±0.17	94.13±0.24	96.30±0.15	95.92±0.16	94.69±0.18	95.15±0.15	95.99±0.24	**96.39±0.15**
BioASQ	82.48±0.77	82.30±0.80	83.57±0.67	82.97±0.63	81.05±0.79	83.94±0.53	82.82±0.54	83.21±0.37	81.84±0.36	83.44±0.66	**84.90±0.53**
TriviaQA	84.12±0.31	83.60±0.33	84.18±0.32	83.85±0.32	82.50±0.33	84.49±0.29	83.57±0.33	83.21±0.37	81.84±0.36	84.21±0.38	**85.71±0.30**
Average	78.27	77.95	78.79	78.43	76.80	79.21	77.99	77.96	76.42	78.82	**80.05**
Mistral-7B-v0.3											
NQ	51.96±0.61	51.49±0.71	52.53±0.50	52.22±0.50	51.16±0.56	52.75±0.48	52.01±0.51	51.27±0.70	49.52±0.64	51.98±0.86	**53.71±0.48**
CoQA	92.83±0.26	93.11±0.22	94.29±0.20	94.17±0.16	91.95±0.32	94.47±0.13	94.32±0.16	93.02±0.18	93.28±0.17	94.05±0.31	**95.97±0.13**
BioASQ	80.46±0.77	80.07±0.66	82.27±0.70	81.07±0.59	80.34±0.86	81.65±0.63	80.18±0.61	80.05±0.71	73.21±0.62	81.75±0.72	**82.64±0.64**
TriviaQA	82.58±0.25	82.53±0.31	83.02±0.25	82.31±0.26	81.81±0.33	83.11±0.32	82.11±0.30	82.23±0.37	79.48±0.35	83.39±0.22	**84.19±0.32**
Average	76.96	76.80	78.03	77.44	76.32	78.00	77.16	76.64	73.87	77.79	**79.13**
Mistral-Nemo-12B											
NQ	51.32±1.30	51.27±1.17	52.12±1.16	51.18±1.10	50.47±1.28	51.70±1.17	49.83±1.12	50.82±1.39	47.12±1.02	51.82±1.05	**53.46±1.19**
CoQA	93.35±0.24	93.15±0.24	94.24±0.18	94.15±0.17	91.82±0.17	94.54±0.13	94.10±0.14	93.02±0.23	93.03±0.18	94.17±0.20	**95.60±0.13**
BioASQ	82.31±0.56	82.00±0.66	83.55±0.53	82.33±0.53	81.63±0.70	83.50±0.45	81.49±0.56	81.73±0.59	76.35±0.54	83.50±0.68	**84.50±0.46**
TriviaQA	85.35±0.26	85.07±0.38	85.85±0.32	85.14±0.26	84.14±0.32	85.93±0.27	85.14±0.27	84.63±0.29	83.31±0.27	85.93±0.27	**86.49±0.25**
Average	78.08	77.87	78.94	78.20	77.02	78.92	77.64	77.55	74.95	78.86	**80.01**

Table 3: Performance comparison among four variants of SGD (SGD$_\delta$, SGD$_{\delta+P}$, SGD$_s$, and SGD$_{s+P}$) and competitive baseline methods including SE, KLE, and Deg. Experiments were conducted on Mistral-7B-v0.3.

Methods	NQ		BioASQ	
	AUROC	AUARC	AUROC	AUARC
SE	76.88±0.60	51.96±0.61	80.86±0.53	80.46±0.77
KLE	77.58±0.42	52.53±0.50	83.66±0.41	82.27±0.70
Deg	77.42±0.56	52.75±0.48	83.57±0.53	81.65±0.63
SGD$_{\delta+P}$	77.68±0.60	51.98±0.86	83.88±0.60	81.75±0.72
SGD$_\delta$	76.71±0.58 $\downarrow$	50.88±0.88 $\downarrow$	82.94±0.55 $\downarrow$	80.92±0.70 $\downarrow$
SGD$_{s+P}$	78.39±0.60	53.71±0.48	84.56±0.54	82.64±0.64
SGD$_s$	77.41±0.56 $\downarrow$	52.74±0.48 $\downarrow$	83.58±0.54 $\downarrow$	81.66±0.63 $\downarrow$

similarity between texts, which incurs the same resource consumption as ours. SEU requires $N(N-1)$ inferences of the all-mpnet-base-v2 model [10] with 109M parameters to obtain the embeddings of answers. In summary, compared to SE, DSE, and NSS, our method outperforms them in terms of performance and has lower resource consumption; our method outperforms graph-based methods in terms of performance and has the same resource consumption; compared to SEU, although its resource consumption is lower than that of other methods, its performance is inferior to that of other baseline methods and our method.

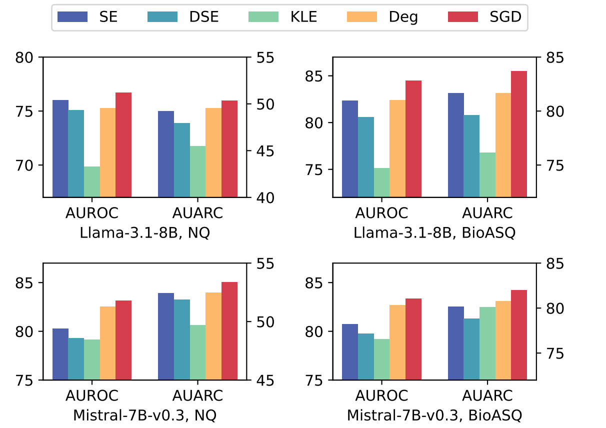

Figure 2: Performance comparison of different uncertainty metrics when generating possible answers using diverse beam search. We compare our best method SGD$_{s+P}$ with competitive baseline methods SE, DSE, KLE, and Deg. All results are shown as percentages.

5 CONCLUSION

In this study, we proposed Semantic Graph Density (SGD), a novel method for Uncertainty Quantification (UQ) in short-form QA. SGD incorporates fine-grained semantic relationships and adjusts edge contributions within semantic graphs, enabling more precise uncertainty quantification. Our experiments on four free-form QA datasets demonstrate that

[10] https://huggingface.co/sentence-transformers/all-mpnet-base-v2

SGD outperforms baseline methods. These results highlight the potential of SGD to enhance the reliability of LLMs, providing a promising direction for addressing hallucination in LLMs. Future work may explore extending SGD to other NLP tasks and integrating it with real-time decision-making systems.

Acknowledgements

We thank the area chair and reviewers for their constructive feedback, which significantly improved the clarity and quality of this paper. This work is supported by the General Program of the National Natural Science Foundation of China (No. 62372459).

References

Lukas Aichberger, Kajetan Schweighofer, Mykyta Ielanskyi, and Sepp Hochreiter. Semantically diverse language generation for uncertainty estimation in language models. In *International Conference on Learning Representations*, 2025.

Amos Azaria and Tom Mitchell. The internal state of an LLM knows when it's lying. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 967–976, 2023.

Lukas Berglund, Meg Tong, Maximilian Kaufmann, Mikita Balesni, Asa Cooper Stickland, Tomasz Korbak, and Owain Evans. The reversal curse: LLMs trained on "A is B" fail to learn "B is A". In *The Twelfth International Conference on Learning Representations*, 2024.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, et al. A survey on evaluation of large language models. *ACM Transactions on Intelligent Systems and Technology*, 15(3):1–45, 2024.

Haoang Chi, He Li, Wenjing Yang, Feng Liu, Long Lan, Xiaoguang Ren, Tongliang Liu, and Bo Han. Unveiling causal reasoning in large language models: Reality or mirage? *Advances in Neural Information Processing Systems*, 2024.

Roi Cohen, May Hamri, Mor Geva, and Amir Globerson. LM vs LM: Detecting factual errors via cross examination. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12621–12640, 2023.

Longchao Da, Tiejin Chen, Lu Cheng, and Hua Wei. LLM uncertainty quantification through directional entailment graph and claim level response augmentation. *arXiv preprint arXiv:2407.00994*, 2024.

P ERDdS and A R&wi. On random graphs i. *Publ. math. debrecen*, 6(290-297):18, 1959.

Xinyue Fang, Zhen Huang, Zhiliang Tian, Minghui Fang, Ziyi Pan, Quntian Fang, Zhihua Wen, Hengyue Pan, and Dongsheng Li. Zero-resource hallucination detection for text generation via graph-based contextual knowledge triples modeling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 23868–23877, 2025.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630, 2024.

Katja Filippova. Controlled hallucinations: Learning to generate faithfully from noisy data. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 864–870, 2020.

Jiahui Geng, Fengyu Cai, Yuxia Wang, Heinz Koeppl, Preslav Nakov, and Iryna Gurevych. 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)*, pages 6577–6595, 2024.

Yashvir S Grewal, Edwin V Bonilla, and Thang D Bui. Improving uncertainty quantification in large language models via semantic embeddings. *arXiv preprint arXiv:2410.22685*, 2024.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330, 2017.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. DeBERTa: Decoding-enhanced BERT with disentangled attention. In *International Conference on Learning Representations*, 2021.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 2023.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. *ACM computing surveys*, 55(12):1–38, 2023.

Mingjian Jiang, Yangjun Ruan, Prasanna Sattigeri, Salim Roukos, and Tatsunori Hashimoto. Graph-based uncertainty metrics for long-form language model outputs. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

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

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

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466, 2019.

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

Zhen Lin, Shubhendu Trivedi, and Jimeng Sun. Generating with confidence: Uncertainty quantification for black-box large language models. *Transactions on Machine Learning Research*, 2024.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1906–1919, 2020.

Kenton Murray and David Chiang. Correcting length bias in neural machine translation. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 212–223, 2018.

Malik Sajjad Ahmed Nadeem, Jean-Daniel Zucker, and Blaise Hanczar. Accuracy-rejection curves (ARCs) for comparing classification methods with a reject option. In *Machine Learning in Systems Biology*, pages 65–81, 2009.

Alexander Nikitin, Jannik Kossen, Yarin Gal, and Pekka Marttinen. Kernel language entropy: Fine-grained uncertainty quantification for LLMs from semantic similarities. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Siva Reddy, Danqi Chen, and Christopher D Manning. CoQA: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266, 2019.

Yize Sui, Jing Ren, Huibin Tan, Huan Chen, Zhaoye Li, and Ji Wang. Enhancing LLM's reliability by iterative verification attributions with keyword fronting. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 251–268, 2024.

George Tsatsaronis, Georgios Balikas, Prodromos Malakasiotis, Ioannis Partalas, Matthias Zschunke, Michael R Alvers, Dirk Weissenborn, Anastasia Krithara, Sergios Petridis, Dimitris Polychronopoulos, et al. An overview of the BIOASQ large-scale biomedical semantic indexing and question answering competition. *BMC bioinformatics*, 16:1–28, 2015.

Ashwin Vijayakumar, Michael Cogswell, Ramprasaath Selvaraju, Qing Sun, Stefan Lee, David Crandall, and Dhruv Batra. Diverse beam search for improved description of complex scenes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. Emergent abilities of large language models. *Transactions on Machine Learning Research*, 2022.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, YIFEI LI, Jie Fu, Junxian He, and Bryan Hooi. Can LLMs express their uncertainty? an empirical evaluation of confidence elicitation in LLMs. In *International Conference on Learning Representations*, 2024.

Caiqi Zhang, Fangyu Liu, Marco Basaldella, and Nigel Collier. LUQ: Long-text uncertainty quantification for LLMs. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 5244–5262, 2024.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 2023.

Shen Zheng, Jie Huang, and Kevin Chen-Chuan Chang. Why does ChatGPT fall short in answering questions faithfully. *arXiv preprint arXiv:2304.10513*, 2023.

Chiwei Zhu, Benfeng Xu, Quan Wang, Yongdong Zhang, and Zhendong Mao. On the calibration of large language models and alignment. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 9778–9795, 2023.

Enhancing Uncertainty Quantification in Large Language Models through Semantic Graph Density (Supplementary Material)

Zhaoye Li[1] Siyuan Shen[1] Wenjing Yang[1] Ruochun Jin[1] Huan Chen[1] Ligong Cao[1] Jing Ren[*][1]

[1]College of Computer Science and Technology, National University of Defense Technology, Changsha, China

A BIAS ARISING FROM NEGLECT OF ANSWER PROBABILITIES IN GRAPH-BASED METHODS

Different sampled answers may have varying probabilities, which can be calculated using the numeric outputs (token-level logits) provided by LLMs. Answers with higher probabilities are considered more representative in UQ, compared to answers with lower probabilities [Geng et al., 2024].

Table 4 shows the responses of the same LLM to two questions. For both cases, the sample answers consisted of one "**No**" and two "**Yes**" responses. However, the model's confidence in the "**Yes**" sampled answer is ***notably higher in the first case***, where the probabilities for "**Yes**" and "**No**" are **0.95** and **0.15**, respectively. In contrast, for the second case, the probabilities are **0.55** and **0.45**. It can be inferred that the model exhibits greater confidence in answering Question 1 compared to Question 2. Consequently, the uncertainty score assigned to Question 1 should be lower than that assigned to Question 2.

Existing graph-based methods [Lin et al., 2024, Da et al., 2024] overlook answer probabilities and focus exclusively on LLM output text. Consequently, these methods would assign identical uncertainty scores to both questions, despite the sampled answers having different probabilities, which is problematic.

Table 4: Example Responses from BioASQ.

Question 1 Has Denosumab (Prolia) been approved by FDA ? **Ground Truth** Yes	**Question 2** Is the ACE inhibitor indicated for lung cancer treatment? **Ground Truth** No
Model Answer **Yes** ✓	**Model Answer** **Yes** ✗
Possible Answer, Probability • $y^{(1)}$: **No, 0.15** • $y^{(2)}$: **Yes, 0.95** • $y^{(3)}$: **Yes, 0.95**	**Possible Answer, Probability** • $y^{(1)}$: **No, 0.45** • $y^{(2)}$: **Yes, 0.55** • $y^{(3)}$: **Yes, 0.55**

[*] Corresponding author.

B PROMPTS FOR OBTAINING ANSWERS

We follow Farquhar et al. [2024] and use the following prompt template to obtain answers, including both the target answer (to evaluate the model's accuracy) and the possible answers (to measure the model's uncertainty), for datasets without context, such as NQ, TriviaQA, and BioASQ:

```
Question: {Example 1 Question}
Answer: {Example 1 Answer}
[Additional Examples]
Question: {question}
Answer:
```

For CoQA, a dataset that includes context, we adopt a modified prompt template provided by Lin et al. [2024]:

```
[The provided context paragraph]
[Additional question-answer pairs]
Q: [Provided question]
A:
```

Below are some example prompts for each dataset, formatted according to the template described above.

NQ

Question: Can exosomes be detected in urine?
Answer: yes.
Question: Who wrote the music for Annie Get Your Gun?
Answer: Irving Berlin.
Question: What is the name of the speaker of parliament in Ghana?
Answer: Aaron Mike Oquaye.
Additional Question-Answer Pairs
Question: When did Stevie Wonder release his first album?
Answer:

CoQA

Once upon a time, in a barn near a farm house, there lived a little white kitten named Cotton. Cotton lived high up in a nice warm place above the barn where all of the farmer's horses slept. **Some contents are omitted here.** "Don't ever do that again, Cotton!" they all cried. "Next time you might mess up that pretty white fur of yours and we wouldn't want that!" Then Cotton thought, "I change my mind. I like being special".

Q: What color was Cotton?
A: white.
Additional Question-Answer Pairs
Q: Where did she live?
A:

BioASQ

Question: In which cell organelle is the SAF-A protein localized?
Answer: the nucleus.
Question: What is the role of IL-18BP?
Answer: IL-18 binding protein (IL-18BP) is a natural inhibitor of IL-18. The balance between IL-18 and IL-18BP has an important role in the inflammatory setting.
Question: Which is the genetic lesion associated with Huntington's disease?
Answer: A CAG trinucleotide repeat expansion in the HD gene.
Additional Question-Answer Pairs
Question: What is a exposome?
Answer:

TriviaQA

Question: Who discovered electromagnetic induction, so facilitating the transformer and dynamo?
Answer: Michael Faraday.
Question: Which card game, originating in Spain and introduced to England in 1861, is played between 2 persons with 2 packs of cards (with sixes and below removed) who are dealt 8 cards each?
Answer: Bezique.
Question: Whose twelfth studio album Magna Carta Holy Grail released in July has topped the charts on both sides of the Atlantic?
Answer: JAY-Z.
Additional Question-Answer Pairs
Question: Which American won the Nobel Peace Prize in 2002?
Answer:

C EXAMPLE RESPONSES

Table 5 includes two examples showing the responses of the Llama-3.1-8B model to two questions, each corresponding to a question from either the NQ (left) or CoQA (right) dataset.

D INPUT FORMAT FOR NLI MODELS TO MEASURE CONTEXTUAL SEMANTIC SIMILARITY

Following Lin et al. [2024], we use the following format to obtain the output logits of the NLI model:

```
[x] [yi] [SEP] [x] [yj]
```

E PROMPTS FOR LLM-BASED CORRECTNESS CHECKING

We automatically determined whether the target answer is *correct or incorrect* by using GPT-4-0613 to compare the given answer to the corresponding ground truth answers provided by the dataset. Following Farquhar et al. [2024], we set the temperature to 0.01 in order to minimize randomness. We utilized the template provided by Farquhar et al. [2024] as follows:

```
We are assessing the quality of answer to the following question: {question}
The expected answer is(are): {reference answers}
The proposed answer is: {target answer}
Within the context of the question, does the proposed answer mean the same as the
expected answer? Respond only with yes or no.
```

Table 5: Example Responses from NQ and CoQA.

Example Responses from NQ	**Example Responses from CoQA**
Question In the honour of which god is anant chaturdashi celebrated ? **Ground Truth** Ganesh **Model Answer** 4 gods ✘	**Context** Once upon a time, in a barn near a farm house ... **Question** Whose paint was it? **Ground Truth** the old farmer's **Model Answer** the old farmer's ✔
Possible Answers • $y^{(1)}$: Hindu Lord Vishnu • $y^{(2)}$: 4 Avatars of Vishnu • $y^{(3)}$: Chaturbhuja or Kartikeya • $y^{(4)}$: 10 incarnations of Lord Vishnu • $y^{(5)}$: 12 Avatars of Lord Vishnu • $y^{(6)}$: the god of prosperity • $y^{(7)}$: 12-year-old Ganesh • $y^{(8)}$: 16 forms of Vishnu • $y^{(9)}$: 9 different deities of Hinduism • $y^{(10)}$: 16 forms of Vishnu	**Possible Answers** • $y^{(1)}$: it was the old farmer's paint • $y^{(2)}$: the old farmer's paint • $y^{(3)}$: It was the old farmer's orange paint • $y^{(4)}$: the old farmer's paint • $y^{(5)}$: the old farmer's paint • $y^{(6)}$: the old farmer's paint • $y^{(7)}$: the farmer's • $y^{(8)}$: the old farmer's • $y^{(9)}$: the old farmer's paint • $y^{(10)}$: the old farmer's

Performance of Correctness Checking. In the supplementary materials of Note 6 in [Farquhar et al., 2024], the authors evaluated the agreement between GPT-4-0613 and human raters in answer correctness assessment. The results showed that (1) GPT-4-0613 agreed with two human raters at an average rate of 93%, while the two raters agreed with each other at a similar rate of 92%. (2) Compared to Llama-2-Chat-70B and GPT-3.5, GPT-4 demonstrated performance most closely aligned with human-level judgment.

F BASELINE IMPLEMENTATION DETAILS

- **Semantic Entropy (SE)** [Farquhar et al., 2024], **Discrete Semantic Entropy (DSE)** [Farquhar et al., 2024]. SE measures the entropy of the meaning distribution of free-form responses to questions. Specifically, SE groups the sampled answers with the same semantic and computes cluster-wise predictive entropy to quantify semantic uncertainty. For our study, we utilize the SE implementation from the Nature publication [Farquhar et al., 2024], rather than the version from the ICLR publication [Kuhn et al., 2023]. In accordance with the recommendation in [Farquhar et al., 2024], we use GPT-3.5-turbo-0125 to determine whether two answers entail one another. If the answers are deemed to entail each other, it indicates that they share the same semantic. DSE is an approximation of SE, designed for black-box LLMs. Unlike SE, DSE estimates the probability of the meaning by counting the frequency of each answer in the samples.
- **Kernel Language Entropy (KLE)** [Nikitin et al., 2024]. KLE treats each answer as a mixed state in quantum mechanics, where each pure state represents a distinct "semantic meaning". It then calculates the von Neumann entropy of the mixed state that corresponds to all sampled answers. Specifically, KLE constructs a semantic graph based on the fine-grained semantic relationships among answers, computes a semantic kernel K over this graph, and calculates the von Neumann entropy of the kernel. We particularly employ the variant KLE(K_{HEAT}), which has demonstrated superior performance compared to other KLE variants.
- **Sum of Eigenvalues of the Graph Laplacian (EigV)** [Lin et al., 2024], **The Degree Matrix (Deg)** [Lin et al., 2024], and **Eccentricity (Ecc)** [Lin et al., 2024]. These metrics construct a weighted semantic graph based on the semantic similarity among the sampled answers. EigV approximates the number of connected components by analyzing the eigenvalues of the graph Laplacian. Deg uses the degree matrix to quantify the diversity of the answers, while Ecc calculates the eccentricity of the spectral embedding, capturing the spread of the answers in the semantic space.
- **Directed Uncertainty Evaluation (D-UE)** [Da et al., 2024]. D-UE captures the semantic relationships between

answers using a bidirectional approach and quantifies uncertainty by analyzing directional instability. Specifically, D-UE constructs a directional graph based on the entailment logits from the NLI model. It then applies Random Walk Laplacian analysis, considering the asymmetric properties of the constructed directed graph. Finally, uncertainty is aggregated through the eigenvalues derived from the Laplacian process.

- **Number of Semantic Sets (NSS)** [Kuhn et al., 2023]. NSS clusters the sampled answers that share the same meaning and uses the number of clusters, which represents the number of distinct meanings conveyed by the prompt, as an uncertainty measurement.
- **Semantic Embedding Uncertainty (SEU)** [Grewal et al., 2024]. SEU calculates the average pairwise cosine similarity of the embeddings of possible answers. SEU calculates the negative average pairwise cosine similarity of the embeddings of possible answers as an uncertainty measure. The embeddings are obtained using a pretrained embedding model; in this implementation, we use all-mpnet-base-v2[1]. SEU argues that leveraging the embedding model to capture semantic similarities can achieve smoother and more robust estimation of semantic uncertainty than NLI models.

G COMPARISON WITH NON-SAMPLING-BASED METHODS

In the main comparison, we focus on sampling-based baselines. In this section, we introduce three non-sampling-based methods as baselines for comparison. The baselines and their brief descriptions are as follows. The last two methods are verbalization-based, in which LLMs are prompted to express their uncertainty in words.

- Perplexity: Perplexity is calculated as the exponentiation of the average negative log-likelihood of the predicted probabilities for each token in a sequence, normalized by the length of the sequence.
- Post-hoc Verbalized Uncertainty (PH-VU) [Lin et al., 2022]: Instructing an LLM to evaluate the confidence in the accuracy of its previously generated answer, and using the negative value of this confidence as the uncertainty score. We use the following prompt: *Q: question A: the best generation. The proposed answer is true with a confidence value (0-100) of*
- In-line Verbalized Uncertainty (IL-VU) [Xiong et al., 2024]: Directing an LLM to provide an answer along with its confidence score, and using the negative value of this score as the uncertainty score. We use the following prompt provided by Xiong et al. [2024]: *Read the question, provide your answer, and your confidence in this answer. Note: The confidence indicates how likely you think your answer is true. Use the following format to answer: Answer and Confidence (0-100): [Your answer], [Your confidence level, please only include the numerical number in the range of 0-100] Question: question Answer and Confidence (0-100):*

We follow the setup of the main experiments in our paper. The experimental results are as shown in Table 6 and Table 7. The results indicate that each variant of our method consistently and significantly outperforms these non-sampling-based approaches.

Table 6: Results with Llama-3.1-8B (AUROC).

	Perplexity	PH-VU	IL-VU	SGD$_{\delta+P}$ (Ours)	SGD$_{s+P}$ (Ours)
NQ	62.02±0.32	67.83±0.82	64.35±0.91	78.84±0.39	**78.87±0.42**
BioASQ	63.96±0.51	68.24±0.48	68.52±0.62	84.93±0.42	**85.42±0.48**
TriviaQA	62.08±0.21	67.38±0.35	70.25±0.41	86.47±0.13	**87.17±0.12**

Table 7: Results with Mistral-7B-v0.3 (AUROC).

	Perplexity	PH-VU	IL-VU	SGD$_{\delta+P}$ (Ours)	SGD$_{s+P}$ (Ours)
NQ	60.42±0.58	64.91±1.04	63.06±0.88	77.68±0.60	**78.39±0.60**
BioASQ	63.74±0.46	66.05±0.62	68.10±0.74	83.88±0.60	**84.56±0.54**
TriviaQA	62.21±0.31	70.68±0.53	73.69±0.69	85.48±0.26	**86.27±0.27**

[1] `https://huggingface.co/sentence-transformers/all-mpnet-base-v2`