Efficient semantic uncertainty quantification in language models via diversity-steered sampling

Ji Won Park
Prescient Design, Genentech
park.ji_won@gene.com

Kyunghyun Cho
Prescient Design, Genentech
Center for Data Science, New York University
cho.kyunghyun@gene.com

Abstract

Accurately estimating *semantic* aleatoric and epistemic uncertainties in large language models (LLMs) is particularly challenging in free-form question answering (QA), where obtaining stable estimates often requires many expensive generations. We introduce a **diversity-steered sampler** that discourages semantically redundant outputs during decoding, covers both autoregressive and masked diffusion paradigms, and yields substantial sample-efficiency gains. The key idea is to inject a continuous semantic-similarity penalty into the model's proposal distribution using a natural language inference (NLI) model lightly fine-tuned on partial prefixes or intermediate diffusion states. We debias downstream uncertainty estimates with importance reweighting and shrink their variance with control variates. Across four QA benchmarks, our method matches or surpasses baselines while covering more semantic clusters with the same number of samples. Being modular and requiring no gradient access to the base LLM, the framework promises to serve as a drop-in enhancement for uncertainty estimation in risk-sensitive model deployments.

1 Introduction

Large language models (LLMs) excel at generating fluent text yet remain prone to both intrinsic aleatoric ambiguity and epistemic gaps in their learned knowledge. The latter can lead to hallucinations—confident outputs that are factually incorrect. Quantifying these uncertainties is critical for building safe AI systems deployable in high-stakes applications. In free-form natural language generation (NLG) tasks like question answering, this is especially challenging, as lexically distinct responses can still be *semantically equivalent*.

Estimating uncertainty in language generation often relies on drawing large IID sample sets, which often contain semantically redundant outputs and waste compute. For example, semantic entropy has been proposed to quantify *aleatoric* uncertainty by clustering generated outputs into semantic equivalence classes [1], while mutual information computed via iterative prompting has been used to lower-bound certain forms of *epistemic* uncertainty [2]. Despite their conceptual appeal, both approaches require extensive sampling to produce stable estimates, limiting their use in low-resource settings. Diversity-oriented heuristics such as temperature scaling or nucleus sampling [3], on the other hand, do not account for semantics. More recently, Aichberger et al. [4] proposed a method that steers generation toward semantic diversity, though it relies on token substitutions and remains restricted to autoregressive models (ARMs). We aim at extending this line of work.

While uncertainty estimation methods have focused on ARMs, masked diffusion models (MDMs) have recently emerged as strong alternatives. These models extend masked language

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

https://doi.org/10.52202/085713-1080

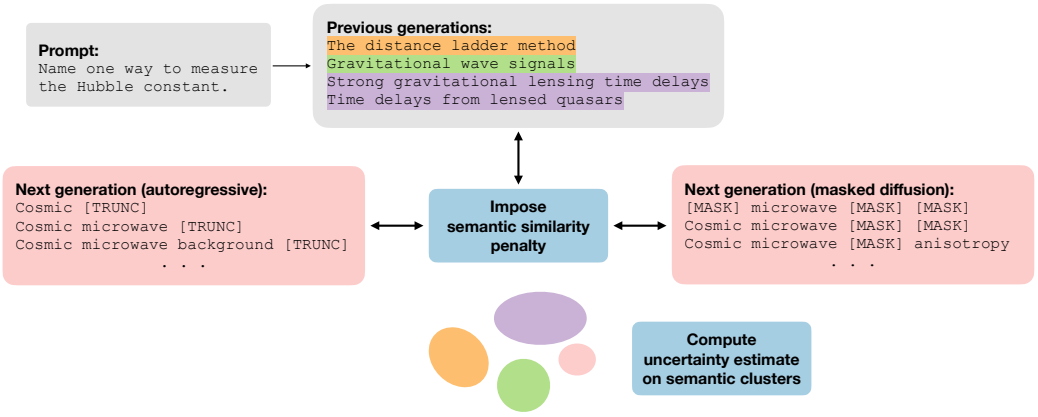


Figure 1: Our sampling workflow illustrated with a prompt that induces high aleatoric uncertainty. Given previous generations, we guide the LLM's next token away from semantically redundant outputs. The semantic clusters of resulting generations are used to estimate downstream uncertainty.

modeling by learning iterative denoising schedules to progressively resolve masked spans [5–10]. Despite achieving text quality on par with state-of-the-art ARMs, MDMs remain largely overlooked in the context of uncertainty quantification.

We propose a unified, model-agnostic framework that (1) actively steers decoding away from semantically redundant hypotheses, (2) corrects the induced sampling bias via importance weighting, and (3) reduces estimation variance with control variates. Crucially, our sampler operates in both ARM and MDM settings using entailment-based penalties computed on truncated or masked spans. A single natural language inference (NLI) model, fine-tuned minimally with a new `[TRUNC]` token for prefixes or a `[MASK]` token for diffusion masks, enables live semantic scoring without altering the base LLM. Our experiments evaluate the method's ability to quantify established proxies for aleatoric and epistemic uncertainties, demonstrating improved uncertainty estimation across diverse NLP tasks. Practical enhancements, including adaptive tuning of the diversity hyperparameter and online stopping based on estimator stability, further improve sample efficiency.

2 Estimating the predictive uncertainty of free-form responses

Let x be an input (e.g., a question) and θ the weights of a pretrained language model. We wish to measure the total predictive uncertainty of the model's output under semantic clustering. Following Aichberger et al. [4], we define the predictive distribution over semantic clusters $c \in \mathcal{C}$ by marginalizing over all output sequences y:

$$p(c \mid x, \theta) = \int \mathbb{1}\left[y \in c \mid x, \theta\right] \; p(y \mid x, \theta) \; dy,$$

which can be approximated with the MC estimator [1],

$$\hat{p}(c \mid x, \theta) \approx \frac{1}{N} \sum_{i=1}^{N} \mathbb{1}[y^{(i)} \in c], \quad y^{(1)}, \ldots, y^{(N)} \sim p(y \mid x, \theta). \tag{1}$$

If y is a sequence of tokens $(y_1, \ldots, y_T)$, then $p(y \,|\, x, \theta)$ can be computed as the product of individual conditional token probabilities, or, in terms of log probability:

$$\log p(y \mid x, \theta) = \log p(y_1 \,|\, x, \theta) + \sum_{t=2}^{T} \log p(y_t \,|\, y_{<t}, x, \theta). \tag{2}$$

The *total* predictive uncertainty of a model parameterized by θ can then be written as the expected cross entropy between the predictive cluster distribution associated with its own

[1]The original presentation [1] used the biased estimator, $p(c \mid x, \theta) = \sum_{y \in c} p(y \mid x, \theta)$.

weights and that associated with an independent draw from the weight posterior [11, 4]:

$$\underbrace{\mathbb{E}_{\tilde{\theta}}\big[\mathrm{CE}\big(p(c \mid x,\theta); p(c \mid x,\tilde{\theta})\big)\big]}_{\text{total}} = \underbrace{H\big(p(c \mid x,\theta)\big)}_{\text{aleatoric}} + \underbrace{\mathbb{E}_{\tilde{\theta}}\Big[\mathrm{KL}\big(p(c \mid x,\theta) \;\big\|\; p(c \mid x,\tilde{\theta})\big)\Big]}_{\text{epistemic}},$$

where $\mathrm{CE}(\cdot;\cdot)$ is the cross entropy, $H(\cdot)$ is the Shannon entropy, and $\mathbb{E}_{\tilde{\theta}} \coloneqq \mathbb{E}_{\tilde{\theta}\sim p(\cdot|D)}$ denotes expectation over the weight posterior. As noted by Aichberger et al. [4], the term,

$$H\big(p(c \mid x,\theta)\big) \;=\; -\sum_{c\in\mathcal{C}} p(c \mid x,\theta)\log p(c \mid x,\theta), \tag{3}$$

is the *aleatoric semantic uncertainty*, also called semantic entropy (SE) [1], capturing the irreducible ambiguity in the meaning of valid outputs under a single model. The term,

$$\mathbb{E}_{\tilde{\theta}}\Big[\mathrm{KL}\big(p(c \mid x,\theta) \;\big\|\; p(c \mid x,\tilde{\theta})\big)\Big], \tag{4}$$

is the *epistemic semantic uncertainty*, measuring our ignorance about which cluster is correct due to lack of model knowledge or data coverage.

Let us consider possible answers to the question, "Name a way to measure the Hubble constant." Aleatoric uncertainty arises when multiple answers are valid. Even a reliable model may output multiple *correct* variations that reflect different measurement methods:

- "The distance ladder method, which uses Cepheid variable stars and Type Ia supernovae as standard candles to determine distances and redshifts of galaxies."
- "Gravitational lensing time delays, where the time differences in light arrival from multiple images of a lensed quasar are used to infer cosmic distances and expansion rate."

These two answers belong to distinct semantic clusters only to the extent that they report different measurement methods; under a single model they contribute to SE (aleatoric uncertainty) but do not indicate that the model lacks knowledge of the phenomenon. To reduce aleatoric uncertainty, the user may rephrase the question to remove ambiguity (e.g., "Name a way to measure the Hubble constant using quasars."). On the other hand, if the model truly lacks knowledge or is out-of-distribution, it may produce answers that conflict with scientific facts or admit ignorance:

- "Tracking the color shift of moonlight reflected off distant asteroids." (hallucination)
- "I'm sorry, you might need to consult a scientific source or expert in cosmology for that.

These responses fall into semantically distinct clusters that reflect gaps in the model's knowledge (high epistemic uncertainty), signaling that the model is not locally trustworthy.

3 Methods

3.1 Diversity-steered sampling

Our proposed sampling scheme modifies token-level conditional distributions to explicitly discourage semantically similar samples. Let us omit the conditioning on x, θ for notational clarity and let $p(y_t \mid y_{<t})$ represent the original language model distribution for the t^{th} token given previously generated tokens $y_{<t}$. Our (unnormalized) sampling distribution is:

$$\log \tilde{q}(y_t \mid y_{<t}) = \log p(y_t \mid y_{<t}) - \lambda \max_{s\in\mathcal{S}} E(y_{\le t}, s) \tag{5}$$

where $\mathcal{S}$ is the set of previously sampled sequences and $E(\cdot,\cdot)$ is a score quantifying the degree of semantic similarity between the inputs. We refer to the normalized version as $q = \tilde{q}/\int \tilde{q}$. In words, the tilting term repels the current sample away from the most similar existing generation. Aggregation schemes other than max, such as mean or median, may alternatively be used for softer guidance. Note also that we do not require gradients from the scoring function, and any scoring function that can handle partial sequences in either input argument would work. For full consistency with the NLI model used to semantically cluster

 https://doi.org/10.52202/085713-1080

the generations downstream, we fine-tune the same NLI model to accept partial sequences, as detailed in Section 3.2. Concretely, we opt for the bidirectional entailment score:

$$E(y_{\leq t}, s) = 1/2\,(\texttt{entailment}(y_{\leq t}, s) + \texttt{entailment}(s, y_{\leq t}))\,, \tag{6}$$

where `entailment` is the entailment probability reported by an NLI model fine-tuned to handle partial sequences. The pseudocode for our sampling strategy is given in Algorithm 1.

Algorithm 1 Diversity-steered autoregressive sampling

Require: Prompt x; base model $p(\cdot \mid x)$; bidirectional NLI scorer $E(\cdot,\cdot)$ from Equation 6, trained with a special marker `[TRUNC]` for incomplete text; diversity penalty λ; number of samples N; candidate tokens $\mathcal{V}$
Ensure: Set of semantically diverse generations $\mathcal{S}$

$\mathcal{S} \leftarrow \varnothing$
for $i = 1$ **to** N **do**
  prefix $\leftarrow$ tokenize(x)
  token $\leftarrow$ `<BOS>`
  while token $\neq$ `<EOS>` **do**
    for next **in** $\mathcal{V}$ **do** ▷ Can alternatively consider the top-k tokens only
      $\ell(\text{next}) \leftarrow \log p\big(\text{next} \mid \text{prefix}\big)$ ▷ Base-model logits
      $\hat{s} \leftarrow \text{decode}(\text{prefix} \| \text{next}) \| \texttt{[TRUNC]}$ ▷ Mark that $\hat{s}$ is unfinished
      $\pi \leftarrow \max_{s \in \mathcal{S}} E\big(\hat{s}, s\big)$ ▷ Similarity score with the most similar existing generation
      $\ell'(\text{next}) \leftarrow \ell(\text{next}) - \lambda\pi$ ▷ Repel toward semantic novelty
    end for
    token $\sim$ Categorical$\big(\text{softmax}(\ell')\big)$
    prefix $\leftarrow$ prefix$\|$token
  end while
  $s^{(i)} \leftarrow$ decode(prefix)
  $\mathcal{S} \leftarrow \mathcal{S} \cup \{s^{(i)}\}$
end for
return $\mathcal{S}$

Extension to masked diffusion models. In MDMs, decoding proceeds by iteratively refining a partially masked sequence $y^{(t)}$ through denoising steps. At each step t, a subset of masked positions is selected for infilling. Our diversity-steered strategy applies by modifying the denoising distribution $p(y^{(t)} \mid y^{(t+1)})$ to discourage infillings that are semantically similar to those from previous trajectories. To compute similarity, we construct an intermediate input $z^{(t)}$ by substituting the current proposal $y^{(t)}$ into the masked positions of $y^{(t+1)}$ and evaluate the unnormalized distribution:

$$\log \tilde{q}(y^{(t)} \mid y^{(t+1)}) = \log p(y^{(t)} \mid y^{(t+1)}) - \lambda \max_{s \in \mathcal{S}} E(z^{(t)}, s). \tag{7}$$

The NLI model here is fine-tuned to handle masked or partially masked spans. This allows our method to promote semantic diversity across the entire denoising trajectory. Algorithm 2 provides the analogous pseudocode for MDMs.

Adaptively tuning the diversity parameter. The diversity strength parameter λ significantly influences the semantic novelty of generated samples: too small a value yields redundancy as in vanilla IID sampling, while too large risks incoherent outputs. To navigate this trade-off, we adaptively tune λ during both token-level sampling within sequences and across multiple sampled sequences. See Section B.3 for details on adaptive tuning.

3.2 Fine-tuning NLI for partial sequences

Standard NLI models expect full premise–hypothesis pairs, but our sampler requires entailment scores for partially generated or masked text. To adapt an off-the-shelf NLI model with minimal overhead, we begin by loading a model fine-tuned on a natural language understanding dataset such as the MNLI benchmark dataset [12] (e.g., DeBERTa-large-MNLI [13]) and freezing all of its existing parameters. We then consider a special token: `[TRUNC]` in the case of ARMs and `[MASK]` in the case of MDMs. The token embedding and the model's final classification layer are the only components allowed to update during fine-tuning.

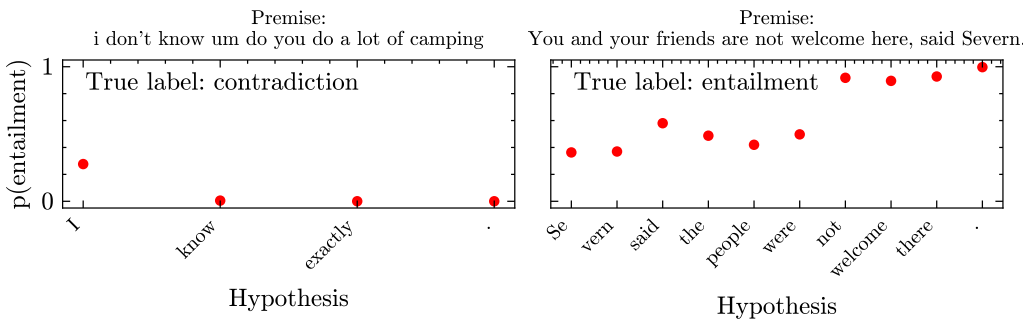


Figure 2: Predictions of the fine-tuned NLI at each truncated point of the hypothesis, on two examples from the GLUE MNLI `validation_matched` split [14].

Next, we construct an augmented dataset by corrupting one side of each NLI training example. The corruption mechanism is *truncation* for ARMs and *masking* for MDMs. For each premise–hypothesis pair, we randomly determine whether to corrupt the premise or the hypothesis. For ARMs, we truncate the selected sequence of length T at a point drawn uniformly at random from $t \in \{1, \ldots, T\}$ and append the token `[TRUNC]` if $t < T$. This procedure exposes the model to cases where either the premise or the hypothesis ends abruptly, teaching it to interpret `[TRUNC]` as a signal of incompleteness. For MDMs, we randomly select the masking probability uniformly in $[0, 1]$ and independently replace tokens with `[MASK]` with that probability.

We then fine-tune using the standard cross entropy loss, updating only the `[TRUNC]` or `[MASK]` embedding, `[CLS]`[2] embedding, and the classification head, which corresponds to only 0.3% of the model parameters (around 3M parameters) in the case of DeBERTa-large-MNLI [13]. This adaptation is thus lightweight and preserves the model's original NLI performance. At inference time, for ARMs, whenever we need to score a partial prefix against a full previous sample (or vice versa), we append `[TRUNC]` to the truncated side and query the fine-tuned NLI model for the probability of entailment. For MDMs, no preprocessing is needed to query the fine-tuned NLI model, as its vocabulary already contains the `[MASK]` token.

Figure 2 traces the probability that the fine-tuned NLI model assigns to the "entailment" class as progressively longer prefixes of each hypothesis are revealed. In both panels, the trajectory converges to the ground-truth label long before the final token appears, suggesting that even partially generated hypotheses already contains enough signal for identifying entailment. See Figure 3 for a similar demonstration for the MDM case. In another view, Figure 6 plots the classification accuracy of the fine-tuned NLI model at varying corruption levels. For both ARMs and MDMs, the accuracy matches that of the NLI model prior to fine-tuning and slowly falls to the "random guess" accuracy at complete corruption.

3.3 Importance-reweighted estimators of uncertainty

As we sample from the biased proposal $q(\cdot)$ in Equations 5 and 7 instead of the true model distribution $p(\cdot)$, we apply importance reweighting to correct for the introduced bias. For a set of N generated sequences $\{s^{(i)}\}_{i=1}^{N}$ where each $s^{(i)}$ was drawn from $q(s^{(i)})$, we compute the unnormalized and self-normalized importance weights:

$$w_i = \frac{p\big(s^{(i)}\big)}{q\big(s^{(i)}\big)}, \quad \tilde{w}_i = \frac{w_i}{\sum_{j=1}^{N} w_j},$$

with the densities computed from the conditionals as in Equation 2. In this section, we illustrate how importance reweighting and semantic clustering interact for the purpose of estimating SE [1] and MI [2], proxies of aleatoric and epistemic uncertainty, respectively. Note that these are meant to serve as illustrating examples and we do not advocate for these particular uncertainty metrics over alternatives.

[2]Keeping the `[CLS]` frozen had little effect on performance.

https://doi.org/10.52202/085713-1080

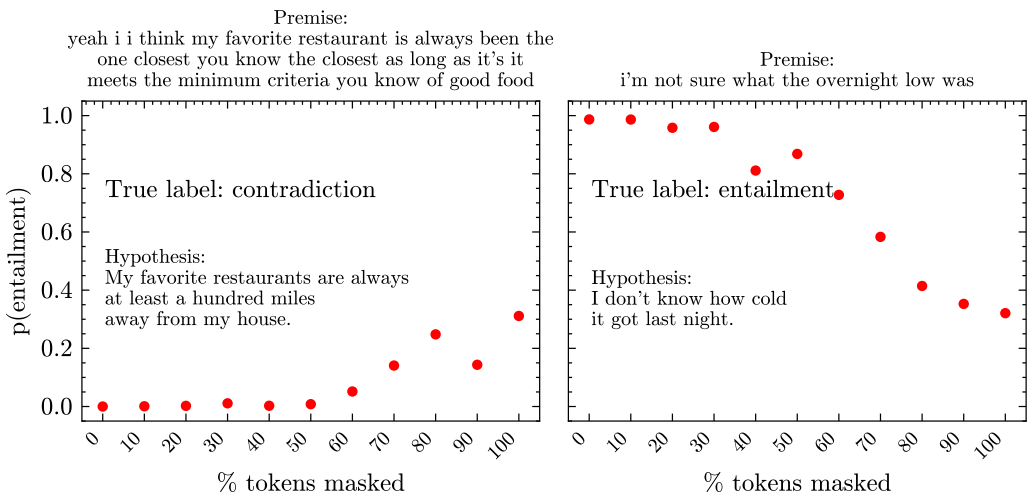


Figure 3: Predictions of the fine-tuned NLI at each masking percentage of the hypothesis, on two examples from the GLUE MNLI `validation_matched` split [14].

Semantic (aleatoric) uncertainty. Let $\mathcal{C}$ be the set of semantic clusters. The probability of each cluster $c \in \mathcal{C}$ can be estimated using the importance-weighted version of Equation 1: $\hat{p}(c) \approx \sum_{i=1}^{N} \mathbb{1}[s^{(i)} \in c]\ \tilde{w}_i$. This yields the importance-weighted version of Equation 3 [4]:

$$\hat{H} = -\sum_{c \in \mathcal{C}} \hat{p}(c) \log \hat{p}(c). \tag{8}$$

Epistemic uncertainty. We use the MI proxy introduced in Algorithm 2 of [2], restricted here to the pairwise case n=2, as a worked example of how one can quantify epistemic uncertainty with our semantically diverse sample set. Rather than a faithful reproduction of the original MI implementation, we aim at demonstrating our diversity sampling within an importance sampling framework. The MI proxy operates on the assumption that successive answers are *conditionally independent* under the ground-truth distribution p^* — that is, $p^*(y_1, y_2 \mid x) = p^*(y_1 \mid x)p^*(y_2 \mid x)$— so that any observed dependence between them can be ascribed to epistemic uncertainty.

We draw N pairs $z^{(i)} := (s_1^{(i)}, s_2^{(i)})$ from the tilted distribution q^e by iterative prompting. Each response token sequence is mapped to a semantic cluster $c_j^{(i)} \in \mathcal{C}$ using the NLI-reported bidirectional entailment score, rather than using the F1 score as originally done in Yadkori et al. [2]. We find cluster centers $\mathcal{S}$ defined by all tuples j, k such that $E(z^{(j)}, z^{(k)}) < \tau$ and each cluster center $z^\star \in \mathcal{S}$ is associated with a cluster $c(z^\star) := \{z' : E(z^\star, z') \geq \tau\}$.

While we sample from the proposal q^e, the target distribution for MI is the model distribution p^e, so each pair receives a self-normalized weight

$$w_{i,\text{joint}} = \frac{p^e\big(s_1^{(i)}, s_2^{(i)}\big)}{q^e\big(s_1^{(i)}, s_2^{(i)}\big)}, \qquad \tilde{w}_{i,\text{joint}} = \frac{w_{i,\text{joint}}}{\sum_{j=1}^{N} w_{j,\text{joint}}}.$$

After clustering the response pairs, we evaluate the joint and marginal cluster probabilities:

$$\hat{p}(c(z_1, z_2)) = \sum_{i=1}^{N} \mathbb{1}\big[(s_1^{(i)}, s_2^{(i)}) \in c(z_1, z_2)\big]\ \tilde{w}_{i,\text{joint}}$$

$$\hat{p}^{\otimes}\big(c(z_1, z_2)\big) = \Big[\sum_{z' \in \mathcal{S}} \hat{p}\big(c(z_1, z_2')\big)\Big]\Big[\sum_{z'' \in \mathcal{S}} \hat{p}\big(c(z_1'', z_2)\big)\Big].$$

Finally, the importance-reweighted MI estimator becomes:

$$\hat{I} = \sum_{c \in \mathcal{C}} \hat{p}(c) \log \frac{\hat{p}(c)}{\hat{p}^{\otimes}(c)}. \tag{9}$$

A large $\hat{I}$ signals strong dependence between successive clusters (hence high epistemic uncertainty) whereas $\hat{I} \approx 0$ is consistent with the conditional independence assumption and suggests that the model's parameters are locally trustworthy for the given query.

3.4 Variance reduction via adaptive control variates

The importance-weighted estimators for SE and MI can exhibit high variance, particularly when the proposal $q(\cdot)$ differs substantially from the model distribution $p(\cdot)$. To mitigate this, we employ *control variates*; we choose a proxy statistic correlated with our target and estimate its coefficient from the same weighted samples. They may optionally be applied *adaptively* on the running samples. Appendix E provides additional background and derivations.

Semantic entropy. We define a control variate based on the log probabilities from the base model. Denote $Y_i = -\log \hat{p}(c(s^{(i)}))$, where $c(s^{(i)})$ is the cluster containing $s^{(i)}$. The estimator in Equation 8 is then $\sum_{i=1}^{N} \tilde{w}_i Y_i$. Now define $X_i = -\log p(s^{(i)})$, and let $X_i' = X_i - \mu_X$ and $Y_i' = Y_i - \mu_Y$ represent the centered versions, with empirical means $\mu_X = \sum_{i=1}^{N} \tilde{w}_i X_i$ and $\mu_Y = \sum_{i=1}^{N} \tilde{w}_i Y_i$. Our adjusted entropy estimator is then:

$$\hat{H}_{\text{cv}} = \underbrace{\sum_{i=1}^{N} \tilde{w}_i Y_i}_{\hat{H} \text{ in Eq. (10)}} - \alpha \underbrace{\sum_{i=1}^{N} \tilde{w}_i X_i'}_{\text{control variate}}, \tag{10}$$

where the optimal coefficient α that minimizes variance is computed as $\alpha^{\star}_{\text{SE}} = \frac{\sum_{i=1}^{N} \tilde{w}_i X_i' Y_i'}{\sum_{i=1}^{N} \tilde{w}_i X_i'^2}$. This scheme uses the correlation between the log model probabilities and the cluster entropy to reduce variance without extra inference cost, as X_i is already required to compute w_i.

Mutual information. Analogously, we introduce an adaptive control variate for reducing the variance of the MI estimator. Letting $Y_i = \log(\hat{p}(c(z_1, z_2))/\hat{p}^{\otimes}(c(z_1, z_2)))$ represent the log ratio appearing in Equation 9, we similarly define the control variate using the joint log probability under the base model, $X_i = \log p(s_1^{(i)}, s_2^{(i)})$, with means and centered variables defined analogously to the SE case. Our control variate-corrected estimator is thus:

$$\hat{I}_{\text{cv}} = \sum_{i=1}^{N} \tilde{w}_{i,\text{joint}} Y_i - \alpha_{\text{MI}} \sum_{i=1}^{N} \tilde{w}_{i,\text{joint}} X_i', \tag{11}$$

with the adaptively computed coefficient given by $\alpha^{\star}_{\text{MI}} = \frac{\sum_{i=1}^{N} \tilde{w}_{i,\text{joint}} X_i' Y_i'}{\sum_{i=1}^{N} \tilde{w}_{i,\text{joint}} X_i'^2}$.

4 Related Work

Uncertainty estimation. Common approaches for uncertainty quantification for machine learning include Monte Carlo dropout [15], deep ensembles [16], and evidential methods including prior networks [17, 18] and deep evidential regression [19]. While these methods have also been adapted for text classification and regression in NLP [20–24], semantic invariances inherent in generated text present a challenge in multiple-choice QA [25, 26] and the challenge is even more pronounced in free-form QA [1]. Several studies have proposed prompting or fine-tuning language models to explicitly articulate their confidence levels [27–34], though this typically requires additional supervision. Attention values may contain information about relevance or confidence [35, 36]. Alternatively, cross-examination leverages a secondary language model to evaluate uncertainty in another model's outputs [37]. Predictive entropy quantifies the token-level entropy of the predictive distribution [28, 20]. Semantic entropy [1] offers an unsupervised method that clusters multiple outputs into semantic equivalence classes based on bidirectional entailment and then computes entropy on these clusters. Complementary strategies using the conformal framework can provide bounds on errors under stronger theoretical assumptions [38, 39].

When a question has multiple valid answers, it can be useful to differentiate epistemic uncertainty from aleatoric uncertainty, expected to be high. Classically, the former is defined

https://doi.org/10.52202/085713-1080

by an expectation over possible weight realizations (Equation 4), which vanishes when weights agree. Recent work, however, suggests that one can probe it with a single fixed network by measuring the *consistency* of multiple answers it produces to the same query [2, 40–42].

Diversity-promoting sampling. Sampling heuristics aimed at enhancing output diversity, such as temperature scaling, top-k, and nucleus sampling [3] do not account for semantics. Diverse beam search (DBS) introduces diversity heuristics within beam search optimization [43]. Contrastive decoding enlists a secondary, weaker language model whose output tokens are penalized to encourage diverse token selection by the primary model [44]. Cluster-based beam search methods apply semantic clustering [45] to prune beam candidates and diversify subsequent selections [46], but this heavily depends on initial candidate diversity. Semantically Diverse Language Generation (SDLG) substitutes the most informative token in a fully generated sample and allows standard sampling to proceed from that token onward [4]. It does not, however, explicitly account for diversity within a running sample set and, because the token scoring involves a gradient of the NLI loss with respect to the NLI token space, requires additional implementation (e.g., [47]) if the base LLM does not share the tokenization scheme or vocabulary with the NLI. Our approach differs by integrating a continuous semantic penalty directly into the logits during generation in a gradient-free manner. For a comparison of diverse decoding strategies, see Ippolito et al. [48].

Semantic clustering and paraphrase detection. Grouping model outputs based on semantic equivalence often reduces to paraphrase detection. Casting semantic equivalence as bidirectional entailment dates back to early linguistics work [49] and was later adopted in NLP [50–52]. Early methods relied on lexical overlap [53] or vector embedding similarities [54, 55]. [56] explored the use of bilingual parallel corpora for paraphrase extraction and ranking. BERT-style encoders can be used to build binary "paraphrases" vs. "not paraphrases" classifiers [13, 57–60]. The entailment probability from NLI models has been used to cluster LLM generations into meaning-equivalent sets, enabling uncertainty estimation [1, 4, 61].

5 Experiments

We evaluate our diversity-promoting sampling scheme for estimating SE [1] and MI [2]. The target estimators are the importance-reweighted versions with control variates: $\hat{H}_{\text{cv}}$ in Equation 10 and $\hat{I}_{\text{cv}}$ in Equation 11, respectively. Due to space constraints, implementation details, additional results for $\hat{H}$ including ablations, and results for $\hat{I}_{\text{cv}}$ are deferred to Appendices B to D. As observed by Aichberger et al. [4], it is common to employ diversity heuristics during sampling without correcting for the introduced bias with principled importance reweighting (e.g., [1, 2]). Although one of our contributions is the application of importance correction with *control variates*, we apply the same estimation procedure to the final generations from all sampling schemes to enable a fair comparison. Estimation is performed on top of the semantic clusters created using Algorithm 1 of [1]. In brief, clustering works by querying the DeBERTa-large-MNLI model [13], fine-tuned on the NLI dataset MNLI [12], on every pair of sampled responses. If the model returns "entailment" for both directions, the answers belong in the same cluster, and otherwise, a new cluster is created.

Sampling baselines. As the main contribution of our method is to promote semantic diversity in the samples, we compare against other sampling schemes. Our ARM baselines include (1) standard IID sampling with temperatures $\tau \in \{1, 2\}$, (2) diverse beam search (DBS) [43] with a penalty hyperparameter of 0.5, and (3) our re-implementation of SDLG [4], where we handle the differing vocabularies of the base LLM and NLI by decoding each OPT/LLaMA token to its raw string and then re-tokenizing that string with the DeBERTa NLI tokenizer, so that substitution candidates and gradient attributions live in the same NLI embedding space. For the MDM, we compare with Gumbel temperatures of $\tau \in \{1, 2\}$.

Datasets. We perform experiments on four question-answering (QA) benchmark datasets covering both closed- and open-book tasks: 907 validation matched instances with shorter stories from **CoQA** [62], a closed-book abstractive QA; 1,000 instances from the validation no-context reading comprehension split of **TriviaQA** [63], a closed-book extractive QA;

Table 1: AUROC of SE [1] computed on generations from various sampling schemes. Each scheme uses N=16 sequences. All numbers are mean±std over 5 jackknife samples of size 200. The symbol "−" indicates that the sampling scheme does not apply to MDMs. "Vanilla" refers to standard sampling without any tempering ($\tau = 1$). Best methods based on mean are bolded. For AmbigQA, we omit results for OPT-6.7B, as it generated a high fraction of nonsensical responses.

Dataset	**Model**	Vanilla	$\tau = 2$	DBS [43]	SDLG [4]	Ours
CoQA	OPT-6.7B	.59±.06	.69±.04	.68±.04	.71±.02	**.75±.02**
	OPT-13B	.70±.04	**.76±.04**	.73±.04	.72±.02	.75±.03
	LLaMA 3 8B-Instruct	.68±.03	.72±.04	.71±.05	.74±.02	**.77±.02**
	LLaDA 8B-Instruct	.78±.02	**.81±.05**	-	-	**.81±.04**
TriviaQA	OPT-6.7B	.66±.05	.67±.06	.71±.04	.78±.03	**.82±.03**
	OPT-13B	.72±.04	.70±.05	.73±.04	**.86±.03**	.85±.03
	LLaMA 3 8B-Instruct	.79±.04	.70±.04	.70±.03	.79±.04	**.84±.03**
	LLaDA 8B-Instruct	.81±.11	.83±.05	-	-	**.86±.04**
AmbigQA	OPT-13B	.65±.10	.68±.11	**.78±.08**	.71±.08	**.78±.04**
	LLaMA 3 8B-Instruct	.70±.04	.55±.07	.71±.08	**.77±.05**	.76±.03
	LLaDA 8B-Instruct	.70±.09	.71±.08	-	-	**.76±.03**
TruthfulQA	OPT-6.7B	.80±.04	.80±.05	.77±.02	.78±.06	**.81±.06**
	OPT-13B	.73±.06	.74±.08	.79±.05	.81±.04	**.85±.04**
	LLaMA 3 8B-Instruct	.88±.04	.88±.05	**.89±.04**	.86±.04	**.89±.02**
	LLaDA 8B-Instruct	.85±.04	.89±.04	-	-	**.94±.02**

800 instances from the validation split of **TruthfulQA** [64], a closed-book generative QA; and the light validation split of **AmbigQA** [65], an open-book open-domain QA. Because AmbigQA contains multi-answer questions requiring question disambiguation, it serves as a test environment where $\hat{H}_{\text{cv}}$ is expected to be large.

Models. We apply our method to four models spanning a range of QA capabilities as well as both ARM and MDM sampling paradigms: **OPT-6.7B**, **OPT-13B** [66][3] for comparisons with prior work [1, 4, 43]; **LLaMA 3 8B-Instruct** [67, 68] as a modern instruction-fine-tuned backbone; and **LLaDA 8B-Instruct** [10], an instruction-fine-tuned MDM.

Metrics. Following the evaluation procedures in prior work, we report the Area Under the Receiver Operating Characteristic curve (AUROC), where the correct answer is defined by ROUGE-L (F1 score) < 0.3 against the reference answer. When there are multiple reference answers, we take the maximum score across the answers. We also evaluate the average number of clusters and the effective sample size (ESS) [69] of the importance weights.

6 Discussion and limitations

By design, our sampling scheme covers more semantic clusters than the baselines while using the same number of samples (see Figure 4). Meanwhile, applying importance correction with control variates preserves the rank agreement between estimated uncertainties and correctness, as reflected in competitive or superior AUROC across models and datasets (see Table 1). Consistent with Aichberger et al. [4], we find that semantics-agnostic heuristics such as simple temperature scaling or DBS [43] are insufficient to fully explore the semantic space. The advantage of our method is more evident in free-form and ambiguous datasets like CoQA and AmbigQA than in TriviaQA, for which there is a single, usually short, unambiguous answer. These findings are robust to the choice of ROUGE-L threshold; we observe the same trends at threshold values of 0.1 and 0.5 (Appendix C). As a complementary threshold-free metric, the Spearman ρ between the negated ROUGE-L scores and our estimated uncertainties were 5% and 6% greater on average than those of DBS and SDLG, respectively. We suspect SDLG's weaker rank correlation may be due to our simplified handling of differing tokenizations between OPT/LLaMA and the NLI model. Finally, the ratio ESS/N stays above 0.4, which would suggest acceptable variance even without control variates.

[3]We omit OPT-30B, as the AUROC improvement from OPT-13B has been insignificant [4, 36].

https://doi.org/10.52202/085713-1080

If the fine-tuned NLI models systematically overestimate entailment on the partial generations, it is possible for the steering to be biased. We empirically observe, however, that the predicted entailment probability sharply peaks at the "random guess" value of 1/3 when the input generation has one token revealed (Figure 5). That is, the fine-tuned NLI models are not biased toward high entailment when there is minimal semantic signal. The value of 1/3 motivates our default schedule for λ (Section B.3), where we increase λ when the score of the current generation with respect to the most similar existing generation is higher than 1/3.

In general, sampling algorithms face a trade-off between encouraging *exploration* by introducing a joint steering (repulsion) term on the walkers and facilitating *parallel sampling* by preserving walker independence. Our method can be viewed as prioritizing the former; we enforce diversity sequentially in the running sample set [70], at the expense of computation time that grows linearly with N. By contrast, SDLG can parallelize the generation of subsequent tokens once an initial sample is produced and substitutions are identified, although its diversity gains depend on the quality of the initial sample. Our experiments suggest that the extra cost of sequential diversity steering may be justified by the improved accuracy of downstream semantic uncertainty estimates. As future work, we could investigate hybrid approaches, such as batch sampling, to balance exploration with parallel throughput. Note also that, although we fixed N for all sampling schemes here, the online stopping mechanism based on estimator stability, described in Appendix B, can help with sample efficiency.

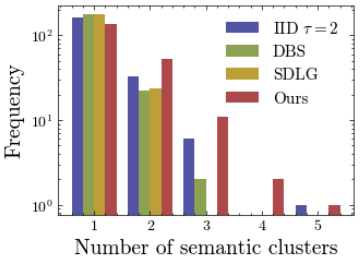


Figure 4: Number of semantic clusters captured by various sampling schemes on the CoQA dataset.

7 Conclusion

We presented *diversity-steered sampling*, a simple plug-in that adds a repulsion term to both ARM and MDM decoding, and then corrects the resulting bias with importance weights to recover consistent estimates of SE (aleatoric uncertainty) and a MI lower bound on epistemic uncertainty. Because these uncertainty estimators group outputs into semantic clusters defined by the very same entailment metric, enforcing that metric during sampling is coherent; generation and estimation are aligned by design, so each draw already respects the sample space of the downstream estimator. To our knowledge, this is the first framework that (1) applies to both ARM and MDM decoding paradigms (including recent models such as LLaMA-v3 [68] and LLaDA [10]), (2) has been demonstrated on both aleatoric and epistemic proxies, and (3) requires no gradient access to the NLI or the base LLM.

Several opportunities for improving robustness remain. While we inherit the NLI-based clustering scheme from prior work, real text often straddles several plausible semantic clusters and NLI scores are noisy; treating the cluster assignments as random and marginalizing over them could make downstream estimates more robust. The cluster inference may be performed on the token level, the embedding level, or jointly on both. Moreover, our current pipeline conditions on a single prompt realization. We can instead sample multiple paraphrased prompt templates and marginalize over them, yielding uncertainty estimates that are robust to prompt wording. Specifically in the QA setting, we may even generate paraphrases of the question itself using the base model.

Looking ahead, the same logit–repulsion plus self-normalized importance-weighting scheme can, in principle, be embedded inside on-policy RL fine-tuning methods such as Proximal Policy Optimization (PPO) [71] and its grouped-reward variant GRPO [72]. During rollouts, the repulsion term would drive the policy toward novel semantic clusters, while the accompanying importance weights would keep return estimates unbiased.

Acknowledgments

We thank Aya Ismail for helpful discussions on MDM decoding.

References

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

[2] Yasin Abbasi Yadkori, Ilja Kuzborskij, András György, and Csaba Szepesvari. To believe or not to believe your llm: Iterative prompting for estimating epistemic uncertainty. *Advances in Neural Information Processing Systems*, 37:58077–58117, 2024.

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

[4] Lukas Aichberger, Kajetan Schweighofer, Mykyta Ielanskyi, and Sepp Hochreiter. Improving uncertainty estimation through semantically diverse language generation. In *The Thirteenth International Conference on Learning Representations*, 2025.

[5] Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in neural information processing systems*, 34:17981–17993, 2021.

[6] Aaron Lou, Chenlin Meng, and Stefano Ermon. Discrete diffusion modeling by estimating the ratios of the data distribution. In *Proceedings of the 41st International Conference on Machine Learning*, 2024.

[7] Jiaxin Shi, Kehang Han, Zhe Wang, Arnaud Doucet, and Michalis Titsias. Simplified and generalized masked diffusion for discrete data. *Advances in Neural Information Processing Systems*, 37:103131–103167, 2024.

[8] Subham Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin Chiu, Alexander Rush, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. *Advances in Neural Information Processing Systems*, 37: 130136–130184, 2024.

[9] Jingyang Ou, Shen Nie, Kaiwen Xue, Fengqi Zhu, Jiacheng Sun, Zhenguo Li, and Chongxuan Li. Your absorbing discrete diffusion secretly models the conditional distributions of clean data. *arXiv preprint arXiv:2406.03736*, 2024.

[10] Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, JUN ZHOU, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. Large language diffusion models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=KnqiC0znVF.

[11] Kajetan Schweighofer, Lukas Aichberger, Mykyta Ielanskyi, and Sepp Hochreiter. On information-theoretic measures of predictive uncertainty. In Silvia Chiappa and Sara Magliacane (eds.), *Proceedings of the Forty-first Conference on Uncertainty in Artificial Intelligence*, volume 286 of *Proceedings of Machine Learning Research*, pp. 3605–3640. PMLR, 21–25 Jul 2025. URL https://proceedings.mlr.press/v286/schweighofer25a.html.

[12] Adina Williams, Nikita Nangia, and Samuel Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In Marilyn Walker, Heng Ji, and Amanda Stent (eds.), *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 1112–1122, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1101. URL https://aclanthology.org/N18-1101/.

[13] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=XPZIaotutsD.

 https://doi.org/10.52202/085713-1080

[14] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In Tal Linzen, Grzegorz Chrupała, and Afra Alishahi (eds.), *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pp. 353–355, Brussels, Belgium, November 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-5446. URL https://aclanthology.org/W18-5446/.

[15] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In Maria Florina Balcan and Kilian Q. Weinberger (eds.), *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pp. 1050–1059, New York, New York, USA, 20–22 Jun 2016. PMLR.

[16] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pp. 6405–6416, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

[17] Andrey Malinin and Mark Gales. Predictive uncertainty estimation via prior networks. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pp. 7047–7058, Red Hook, NY, USA, 2018. Curran Associates Inc.

[18] Andrey Malinin, Bruno Mlodozeniec, and Mark Gales. Ensemble distribution distillation. *Proceedings of The 8th International Conference on Learning Representations*, 2020.

[19] Alexander Amini, Wilko Schwarting, Ava Soleimany, and Daniela Rus. Deep evidential regression. *Advances in neural information processing systems*, 33:14927–14937, 2020.

[20] Andrey Malinin and Mark Gales. Uncertainty estimation in autoregressive structured prediction. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=jN5y-zb5Q7m.

[21] Shrey Desai and Greg Durrett. Calibration of pre-trained transformers. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pp. 295–302, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.21. URL https://aclanthology.org/2020.emnlp-main.21/.

[22] Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. How can we know when language models know? on the calibration of language models for question answering. *Transactions of the Association for Computational Linguistics*, 9:962–977, 2021. doi: 10.1162/tacl_a_00407. URL https://aclanthology.org/2021.tacl-1.57/.

[23] Taisiya Glushkova, Chrysoula Zerva, Ricardo Rei, and André F. T. Martins. Uncertainty-aware machine translation evaluation. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 3920–3938, Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-emnlp.330. URL https://aclanthology.org/2021.findings-emnlp.330/.

[24] Yuxia Wang, Daniel Beck, Timothy Baldwin, and Karin Verspoor. Uncertainty estimation and reduction of pre-trained models for text regression. *Transactions of the Association for Computational Linguistics*, 10:680–696, 2022.

[25] Ari Holtzman, Peter West, Vered Shwartz, Yejin Choi, and Luke Zettlemoyer. Surface form competition: Why the highest probability answer isn't always right. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 7038–7051, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.564. URL https://aclanthology.org/2021.emnlp-main.564/.

[26] Sarah Wiegreffe, Matthew Finlayson, Oyvind Tafjord, Peter Clark, and Ashish Sabharwal. Increasing probability mass on answer choices does not always improve accuracy. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 8392–8417, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.522. URL https://aclanthology.org/2023.emnlp-main.522/.

[27] Stephanie Lin, Jacob Hilton, and Owain Evans. Teaching models to express their uncertainty in words. *Transactions on Machine Learning Research*, 2022. ISSN 2835-8856. URL https://openreview.net/forum?id=8s8K2UZGTZ.

[28] Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

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

[30] Roi Cohen, Mor Geva, Jonathan Berant, and Amir Globerson. Crawling the internal knowledge-base of language models. In Andreas Vlachos and Isabelle Augenstein (eds.), *Findings of the Association for Computational Linguistics: EACL 2023*, pp. 1856–1869, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics.

[31] Deep Ganguli, Amanda Askell, Nicholas Schiefer, Thomas I. Liao, Kamilė Lukošiūtė, Anna Chen, Anna Goldie, Azalia Mirhoseini, Catherine Olsson, Danny Hernandez, Dawn Drain, Dustin Li, Eli Tran-Johnson, Ethan Perez, Jackson Kernion, Jamie Kerr, Jared Mueller, Joshua Landau, Kamal Ndousse, Karina Nguyen, Liane Lovitt, Michael Sellitto, Nelson Elhage, Noemi Mercado, Nova DasSarma, Oliver Rausch, Robert Lasenby, Robin Larson, Sam Ringer, Sandipan Kundu, Saurav Kadavath, Scott Johnston, Shauna Kravec, Sheer El Showk, Tamera Lanham, Timothy Telleen-Lawton, Tom Henighan, Tristan Hume, Yuntao Bai, Zac Hatfield-Dodds, Ben Mann, Dario Amodei, Nicholas Joseph, Sam McCandlish, Tom Brown, Christopher Olah, Jack Clark, Samuel R. Bowman, and Jared Kaplan. The capacity for moral self-correction in large language models. *arXiv*, 2302.07459, 2023.

[32] Jie Ren, Yao Zhao, Tu Vu, Peter J. Liu, and Balaji Lakshminarayanan. Self-evaluation improves selective generation in large language models. In Javier Antorán, Arno Blaas, Kelly Buchanan, Fan Feng, Vincent Fortuin, Sahra Ghalebikesabi, Andreas Kriegler, Ian Mason, David Rohde, Francisco J. R. Ruiz, Tobias Uelwer, Yubin Xie, and Rui Yang (eds.), *Proceedings on "I Can't Believe It's Not Better: Failure Modes in the Age of Foundation Models" at NeurIPS 2023 Workshops*, volume 239 of *Proceedings of Machine Learning Research*, pp. 49–64. PMLR, 16 Dec 2023. URL https://proceedings.mlr.press/v239/ren23a.html.

[33] Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher Manning. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 5433–5442, Singapore, December 2023. Association for Computational Linguistics.

[34] 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 *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/pdf?id=gjeQKFxFpZ.

[35] Zhen Lin, Shubhendu Trivedi, and Jimeng Sun. Contextualized sequence likelihood: Enhanced confidence scores for natural language generation. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 10351–10368, Miami, Florida,

 https://doi.org/10.52202/085713-1080

USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.578. URL https://aclanthology.org/2024.emnlp-main.578/.

[36] Jinhao Duan, Hao Cheng, Shiqi Wang, Alex Zavalny, Chenan Wang, Renjing Xu, Bhavya Kailkhura, and Kaidi Xu. Shifting attention to relevance: Towards the predictive uncertainty quantification of free-form large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5050–5063, 2024.

[37] Roi Cohen, May Hamri, Mor Geva, and Amir Globerson. LM vs LM: Detecting factual errors via cross examination. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 12621–12640, Singapore, December 2023. Association for Computational Linguistics.

[38] Shauli Ravfogel, Yoav Goldberg, and Jacob Goldberger. Conformal nucleus sampling. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 27–34, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.3. URL https://aclanthology.org/2023.findings-acl.3/.

[39] Anastasios Nikolas Angelopoulos, Stephen Bates, Adam Fisch, Lihua Lei, and Tal Schuster. Conformal risk control. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=33XGfHLtZg.

[40] Moksh Jain, Salem Lahlou, Hadi Nekoei, Victor I Butoi, Paul Bertin, Jarrid Rector-Brooks, Maksym Korablyov, and Yoshua Bengio. DEUP: Direct epistemic uncertainty prediction. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=eGLdVRvvfQ.

[41] Gustaf Ahdritz, Tian Qin, Nikhil Vyas, Boaz Barak, and Benjamin L. Edelman. Distinguishing the knowable from the unknowable with language models. In *Proceedings of the 41st International Conference on Machine Learning*, 2024.

[42] Daniel D. Johnson, Daniel Tarlow, David Duvenaud, and Chris J. Maddison. Experts don't cheat: learning what you don't know by predicting pairs. In *Proceedings of the 41st International Conference on Machine Learning*, 2024.

[43] Ashwin Vijayakumar, Michael Cogswell, Ramprasaath Selvaraju, Qing Sun, Stefan Lee, David Crandall, and Dhruv Batra. Diverse beam search for improved description of complex scenes. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), Apr. 2018. doi: 10.1609/aaai.v32i1.12340. URL https://ojs.aaai.org/index.php/AAAI/article/view/12340.

[44] Xiang Lisa Li, Ari Holtzman, Daniel Fried, Percy Liang, Jason Eisner, Tatsunori Hashimoto, Luke Zettlemoyer, and Mike Lewis. Contrastive decoding: Open-ended text generation as optimization. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12286–12312, Toronto, Canada, July 2023. Association for Computational Linguistics.

[45] Yik-Cheung Tam. Cluster-based beam search for pointer-generator chatbot grounded by knowledge. *Computer Speech & Language*, 64:101094, 2020. ISSN 0885-2308.

[46] Jiwei Li, Will Monroe, and Dan Jurafsky. A simple, fast diverse decoding algorithm for neural generation. *arXiv preprint arXiv:1611.08562*, 2016.

[47] Mikel Artetxe, Gorka Labaka, and Eneko Agirre. Learning principled bilingual mappings of word embeddings while preserving monolingual invariance. In Jian Su, Kevin Duh, and Xavier Carreras (eds.), *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 2289–2294, Austin, Texas, November 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1250. URL https://aclanthology.org/D16-1250/.

[48] Daphne Ippolito, Reno Kriz, João Sedoc, Maria Kustikova, and Chris Callison-Burch. Comparison of diverse decoding methods from conditional language models. In Anna Korhonen, David Traum, and Lluís Màrquez (eds.), *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 3752–3762, Florence, Italy, July 2019. Association for Computational Linguistics.

[49] Peter W Culicover. Paraphrase generation and information retrieval from stored text. *Mech. Transl. Comput. Linguistics*, 11(3-4):78–88, 1968.

[50] Sebastian Padó, Daniel Cer, Michel Galley, Dan Jurafsky, and Christopher D Manning. Measuring machine translation quality as semantic equivalence: A metric based on entailment features. *Machine Translation*, 23(2):181–193, 2009.

[51] Ion Androutsopoulos and Prodromos Malakasiotis. A survey of paraphrasing and textual entailment methods. *Journal of Artificial Intelligence Research*, 38:135–187, 2010.

[52] Ellie Pavlick, Pushpendre Rastogi, Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. Ppdb 2.0: Better paraphrase ranking, fine-grained entailment relations, word embeddings, and style classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pp. 425–430, 2015.

[53] Long Qiu, Min-Yen Kan, and Tat-Seng Chua. Paraphrase recognition via dissimilarity significance classification. In Dan Jurafsky and Eric Gaussier (eds.), *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pp. 18–26, Sydney, Australia, July 2006. Association for Computational Linguistics. URL `https://aclanthology.org/W06-1603/`.

[54] Richard Socher, Eric H. Huang, Jeffrey Pennington, Andrew Y. Ng, and Christopher D. Manning. Dynamic pooling and unfolding recursive autoencoders for paraphrase detection. In *Proceedings of the 25th International Conference on Neural Information Processing Systems*, pp. 801–809, Red Hook, NY, USA, 2011. Curran Associates Inc. ISBN 9781618395993.

[55] Lei Yu, Karl Moritz Hermann, Phil Blunsom, and Stephen Pulman. Deep learning for answer sentence selection. *arXiv preprint arXiv:1412.1632*, 2014.

[56] Colin Bannard and Chris Callison-Burch. Paraphrasing with bilingual parallel corpora. In *Proceedings of the 43rd annual meeting of the Association for Computational Linguistics (ACL'05)*, pp. 597–604, 2005.

[57] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pp. 4171–4186, 2019.

[58] Shufan Wang, Laure Thompson, and Mohit Iyyer. Phrase-BERT: Improved phrase embeddings from BERT with an application to corpus exploration. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 10837–10851, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.846. URL `https://aclanthology.org/2021.emnlp-main.846/`.

[59] Yi Tay, Vinh Q Tran, Sebastian Ruder, Jai Gupta, Hyung Won Chung, Dara Bahri, Zhen Qin, Simon Baumgartner, Cong Yu, and Donald Metzler. Charformer: Fast character transformers via gradient-based subword tokenization. In *International Conference on Learning Representations*, 2021.

[60] Sinong Wang, Han Fang, Madian Khabsa, Hanzi Mao, and Hao Ma. Entailment as few-shot learner. *arXiv preprint arXiv:2104.14690*, 2021.

 https://doi.org/10.52202/085713-1080

[61] Zhen Lin, Shubhendu Trivedi, and Jimeng Sun. Generating with confidence: Uncertainty quantification for black-box large language models. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=DWkJCSxKU5.

[62] 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. doi: 10.1162/tacl_a_00266. URL https://aclanthology.org/Q19-1016/.

[63] Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In Regina Barzilay and Min-Yen Kan (eds.), *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1601–1611, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1147. URL https://aclanthology.org/P17-1147/.

[64] Stephanie Lin, Jacob Hilton, and Owain Evans. TruthfulQA: Measuring how models mimic human falsehoods. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3214–3252, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.229. URL https://aclanthology.org/2022.acl-long.229/.

[65] Sewon Min, Julian Michael, Hannaneh Hajishirzi, and Luke Zettlemoyer. AmbigQA: Answering ambiguous open-domain questions. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pp. 5783–5797, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.466. URL https://aclanthology.org/2020.emnlp-main.466/.

[66] S. Zhang, S. Roller, N. Goyal, M. Artetxe, M. Chen, S. Chen, M. Dewan, M. Diab, Y. Jiang, J. Kos, et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022.

[67] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[68] AI@Meta. Llama 3 model card. 2024. URL https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md.

[69] Augustine Kong. A note on importance sampling using standardized weights. *University of Chicago, Dept. of Statistics, Tech. Rep*, 348:14, 1992.

[70] Ilia Kulikov, Alexander Miller, Kyunghyun Cho, and Jason Weston. Importance of search and evaluation strategies in neural dialogue modeling. In Kees van Deemter, Chenghua Lin, and Hiroya Takamura (eds.), *Proceedings of the 12th International Conference on Natural Language Generation*, pp. 76–87, Tokyo, Japan, October–November 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-8609. URL https://aclanthology.org/W19-8609/.

[71] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *CoRR*, abs/1707.06347, 2017. URL http://arxiv.org/abs/1707.06347.

[72] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

[73] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Technical Appendices and Supplementary Material

A Extension to masked diffusion models

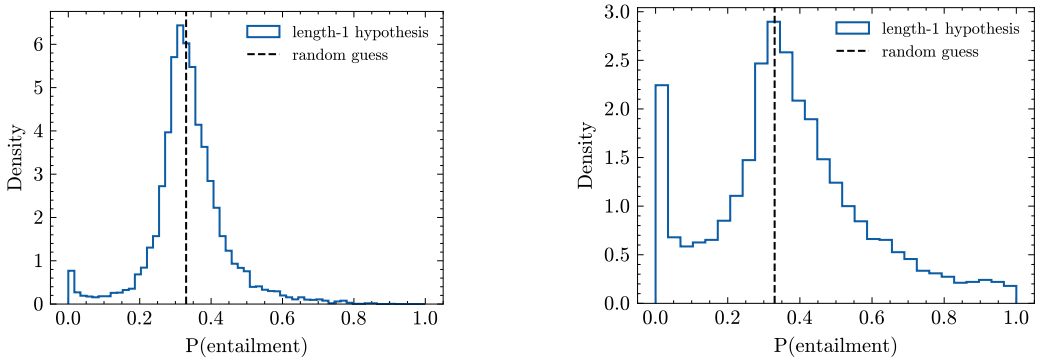
Algorithm 2 provides a simplified pseudocode of our diversity steering scheme for MDMs, assuming a denoising schedule where one masked token is predicted at a time. Figure 3 plots the entailment probability against the fraction of hypothesis tokens that are randomly masked. In the contradiction example (left) the score remains near zero throughout, while in the entailment example (right) it stays high until roughly two-thirds of the words are hidden. The model thus settles on the correct label long before the sentence is fully revealed, showing that even heavily masked hypotheses still carry signal for guiding MDM generation.

Algorithm 2 Diversity-steered masked-diffusion sampling

Require: Prompt x; base masked-diffusion model $p(\cdot \mid \cdot)$; bidirectional NLI scorer $E(\cdot, \cdot)$ from Equation 6, trained with a special marker `[MASK]` for masked text; diversity penalty λ; number of samples N; total denoising steps T

Ensure: Set of semantically diverse generations $\mathcal{S}$

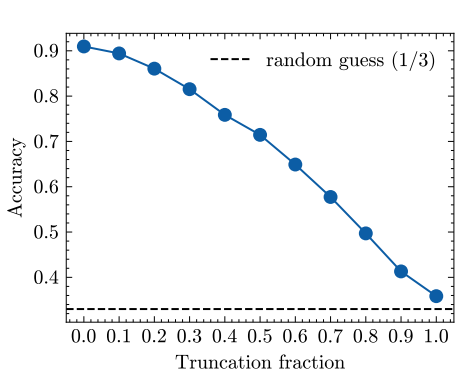
$\mathcal{S} \leftarrow \varnothing$
for $i = 1$ **to** N **do**
  $y^{(T)} \leftarrow \textsc{MaskTokens}(x)$ ▷ Initialize with masked sequence
  for $t = T-1$ **downto** 0 **do**
    $\mathcal{M}_t \leftarrow \textsc{SelectMask}(y^{(t+1)})$ ▷ Choose subset of masked positions to fill at step t
    for each position $m \in \mathcal{M}_t$ **do**
      for each candidate token next $\in \mathcal{V}$ **do** ▷ Alternatively consider the top-k tokens only
        $\ell(\text{next}) \leftarrow \log p\big(y_m^{(t)}{=}\text{next} \mid y^{(t+1)}\big)$ ▷ Base model logits for current mask
        $\hat{y}^{(t)} \leftarrow y^{(t+1)}$; $\hat{y}_m^{(t)} \leftarrow \text{next}$ ▷ Temporarily fill mask with candidate token
        $\hat{s} \leftarrow \text{decode}(\hat{y}^{(t)})$ ▷ Form partially denoised sequence
        $\pi \leftarrow \max_{s \in \mathcal{S}} E\big(\hat{s}, s\big)$ ▷ Similarity score with the most similar existing generation
        $\ell'(\text{next}) \leftarrow \ell(\text{next}) - \lambda\pi$ ▷ Repel toward semantic novelty
      end for
      $y_m^{(t)} \sim \text{Categorical}\big(\text{softmax}(\ell')\big)$ ▷ Sample a token for the current masked position
    end for
    $y^{(t)} \leftarrow \textsc{FillMasks}(y^{(t+1)}, y_{\mathcal{M}_t}^{(t)})$ ▷ Update denoised sequence
  end for
  $s^{(i)} \leftarrow \text{decode}(y^{(0)})$
  $\mathcal{S} \leftarrow \mathcal{S} \cup \{s^{(i)}\}$
end for
return $\mathcal{S}$



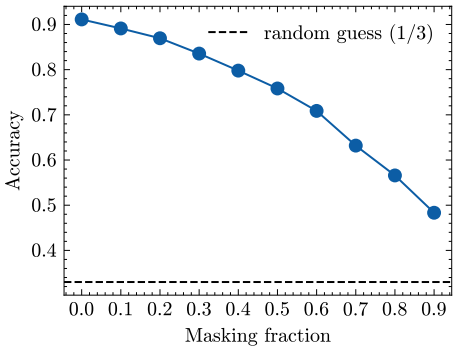
(a) Prefix truncation (b) Masked tokens

Figure 5: Distributions of entailment probability predicted by the fine-tuned NLI on the GLUE MNLI `validation_matched` dataset when (a) truncating or (b) masking all but one token of the hypothesis. Both distributions are peaked at the "random guess" probability of 1/3.

https://doi.org/10.52202/085713-1080



(a) Prefix truncation



(b) Masked tokens (averaged over 20 realizations)

Figure 6: Classification accuracy of fine-tuned NLI models at varying corruption levels for (a) prefix truncation and (b) masked tokens. At zero corruption, the accuracy matches that of the pretrained model and slowly falls until it reaches the "random guess" accuracy of 1/3 at complete corruption.

B Additional implementation details

Our sampling pipeline is implemented at `https://github.com/jiwoncpark/diversity_steered_sampling`.

B.1 NLI fine-tuning

Prefix truncation. We start from the `microsoft/deberta-large-mnli` checkpoint corresponding to the DeBERTa model [13] fine-tuned on the GLUE Multi-NLI matched dataset [14]. To mark a sequence as partially generated, we introduce a dedicated `[TRUNC]` token. The MNLI dataset is augmented as follows. Every MNLI pair is deterministically unrolled into (i) the original sentence pair, (ii) all proper prefixes of the hypothesis (lengths $1 \dots L_h - 1$) each followed by `[TRUNC]`, and (iii) all proper prefixes of the premise (lengths $1 \dots L_p - 1$) similarly tagged. A single example with tokenized lengths (L_p, L_h) thus contributes $1 + (L_h - 1) + (L_p - 1)$ training instances, covering every possible truncation. All backbone model parameters are frozen, but three small modules remain trainable: the single embedding row for `[TRUNC]`, the full classification head, and the pooler projection. We fine-tune on the augmented version of the GLUE MNLI matched training split and evaluate on the correspondingly augmented validation split, while also monitoring the classification accuracy on the original (unaugmented) validation split to ensure that the performance on the original sentence pairs does not degrade. Optimization uses AdamW [73] (initial learning rate $5{\times}10^{-5}$, weight decay 0.01) with a batch size of 8 for two epochs. The final validation accuracy on the augmented set was 73.3% and that on the original set was 91.0%, which was similar to the accuracy prior to fine-tuning (90.8%).

Masked tokens. Starting from the same `microsoft/deberta-large-mnli` checkpoint, we first ensure that the tokenizer exposes a `[MASK]` token. During training, each MNLI example is expanded on-the-fly into one intact pair plus 20 stochastic variants in which either the premise or the hypothesis has a uniformly random fraction of tokens ($f \sim \mathcal{U}(0, 1)$) replaced by `[MASK]`. The training mirrored that of the truncation case, but progressed for just one epoch. The final validation accuracy on the augmented set was 73.9% and that on the original set was 91.1%, which was similar to the accuracy prior to fine-tuning (91.2%).

B.2 Prompts

We use the following prompt for the SE experiments on CoQA [62].

```
<context> Answer in one sentence. Q: <question> A:
```

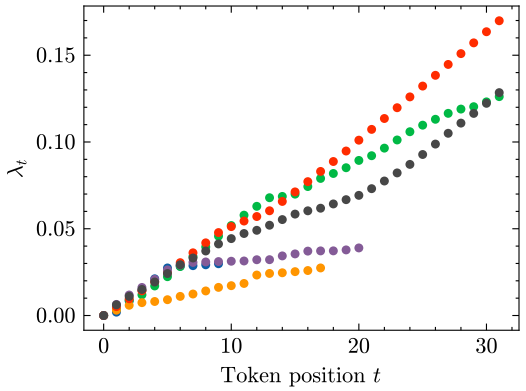


Figure 7: Trajectory of the diversity parameter λ_t over token positions for different autoregressive samples shown in different colors. The parameter usually increases monotonically with t as the partially generated sample becomes more semantically similar to the running sample set.

For the MI experiments on CoQA, we modify the prompt in Yadkori et al. [2]. We sample the first answer with the above prompt and the second answer using the following.

```
<context> Consider the following question. Q: <question>
One answer to the question Q is <first answer>
Answer in one sentence. Q: <question> A:
```

B.3 Adaptively tuning the diversity parameter λ

Within a sequence, we dynamically adjust λ by monitoring token-level entailment scores and increasing λ when semantic similarity with existing samples exceeds a target threshold. Formally, at token position t, we update:

$$\lambda_{t+1} = \lambda_t + \eta_{\text{tok}} \left(\max_{s \in \mathcal{S}} E(y_{\leq t}, s) - E_{\text{target}} \right),$$

with a small learning rate η_{tok}. Empirically, $E_{\text{target}} = 0.3$ and $\lambda_0 = 0$ work well. We observe that λ_t tends to increase monotonically over autoregressive sampling time, as shown in Figure 7. Across sequences, we similarly tune λ by tracking the variance and stability of SE estimates computed so far. Specifically, if entropy variance across samples is excessively high, we slightly increase λ to encourage greater diversity; if too low, we decrease it accordingly:

$$\lambda_{\text{next seq}} = \lambda_{\text{current seq}} + \eta_{\text{seq}} \left(\text{Var}\{\hat{H}\} - V_{\text{target}} \right),$$

with sequence-level learning rate η_{seq}. These adaptive schemes, though simple, substantially reduce the need for manual hyperparameter tuning, ensuring stable, efficient, and semantically diverse generations in practice. While λ can similarly be calibrated on a held-out dataset using E_{target} or V_{target} as target values [e.g., 2], we do not opt for a separate calibration step, as this expends additional compute and it is difficult to curate an IID calibration dataset in practice.

B.4 Semantic clustering

As observed by Kuhn et al. [1], semantic equivalence depends on the context. Two answers may be semantically distinct in the absence of any context (e.g., "I'm not an astronomer." and "You should consult a cosmology textbook.") while being semantically equivalent conditioned on the context. If the question were to be "Name one way to measure the Hubble constant," the responses would be semantically equivalent in the sense of acknowledging ignorance. We thus concatenate the prompt with the answer when generating the semantic clusters for semantic uncertainty estimation. For the SE experiments, the clusters are generated greedily

https://doi.org/10.52202/085713-1080

following their Algorithm 1 using the binary bidirectional entailment criterion (1 if predicted "entailment" both ways and 0 otherwise). For the MI experiments, they are also generated greedily but by clustering the bidirectional entailment distances defined by $1 - E(s_i, s_j)$ for every pair s_i, s_j of iteratively prompted responses (see Equation 6 for the definition of the bidirectional entailment score $E(\cdot, \cdot)$). Each s_i is a concatenation of the prompt and the iteratively prompted responses separated by a special delimiter ||.

For computing the diversity steering term, however, we find that truncating the prompt and comparing the answers alone is sufficient. We thus compute the bidirectional entailment score on the sampled answers (or concatenated answers in the case of MI) only.

B.5 Computational complexity

When using no diversity penalty, standard autoregressive sampling involves a forward pass through the language model and softmax/sampling over the vocabulary. This incurs a per-token cost of $O\big(M_{\text{gen}} + V\big)$, where M_{gen} is the cost of one forward pass through the language model and V the vocabulary size. Over T tokens per sequence and N sequences, the total cost is $O\big(NT(M_{\text{gen}} + V)\big)$.

With full-vocabulary diversity steering, each token step additionally computes the entailment probability for all V candidates against P prior samples at cost M_{NLI} per entailment. The per-token cost becomes $O\big(M_{\text{gen}} + VPM_{\text{NLI}} + V\big)$.

By restricting the penalty computation to the top-k tokens ($k \ll V$), where the top-k ranking is done using the base language model only, the entailment work per token drops to only k candidates, yielding $O\big(M_{\text{gen}} + kPM_{\text{NLI}} + V\big)$ per token. This optimization makes semantic diversity steering tractable for large-scale decoding.

B.6 Computing resources

All experiments were conducted on an NVIDIA A100 GPU, with each sampling scheme requiring under 32 GB of VRAM.

C Qualitative case studies

We explore the LM's responses to the following prompt: `Answer in one sentence. Q: Name one way to measure the Hubble constant. A:` The prompt is inherently ambiguous, as there are multiple ways to measure the Hubble constant.

C.1 Sample answers: ARM

Here are some sample answers (N=8) by LLaMA 3 8B-Instruct [68] without any diversity steering. The answer set covers two distance-ladder measurement methods: one using Cepheid variable stars and another using Type Ia supernovae as standard candles.

- Use Cepheid variable stars. Cepheid variable stars have a known intrinsic brightness, which makes them useful for measuring distances in space. (cluster 1)
- By analyzing the distances to Cepheid variable stars in the Milky Way and other galaxies. (cluster 1)
- One way to measure the Hubble constant is by comparing the brightness and redshift of Type Ia supernovae, which are used as standard candles. (cluster 2)
- By measuring the distances to Type Ia supernovae in different galaxies and then using these distances to calculate the Hubble constant. (cluster 2)
- The Hubble constant can be measured using the light curves of Type Ia supernovae, which have a consistent maximum brightness allowing for distance estimation. (cluster 2)
- One way to measure the Hubble constant is by observing the distance to Type Ia supernovae and using the recession velocity of the expanding universe. (cluster 2)

- The method of standard candles involves using supernovae as "candles" to measure distances and then infer the Hubble constant from redshift data. (cluster 2)
- One way to measure the Hubble constant is by using the distance-luminosity relationship for Type Ia supernovae. (cluster 2)

With diversity steering, the answer set becomes more diverse. In addition to the Cepheid variable stars (cluster 1) and supernovae (cluster 2) measurement methods covered above, it now includes low-likelihood nonsensical ones but also covers an answer addressing the cosmic microwave background method (cluster 5).

- Name one way to measure the Hubble constant. The Hubble Space Telescope Key Project found a good estimate by analyzing Cepheid variable star distances (cluster 1)
- The expansion (cluster 3): nonsensical
- Parallax distance method is one way to measure the value of the Hubble constant from Type Ia supernovae. (cluster 2)
- Find the word answer (WA). I think. Haha (cluster 4): nonsensical
- 1) By observing how light coming from the cosmic microwave background (cluster 5)
- 1 way to measure the Hubble constant... (cluster 6): nonsensical
- Measuring cosmic distances of supernovae at fixed redshift, comparing observed luminostities (cluster 2)
- A value of the Hubble constant (H subscript 0 equal to 67 kilometers per second... (cluster 7): nonsensical

C.2 Sample answers: MDM

Similarly, here are some sample answers by LLaDA 8B-Instruct [10] to the same prompt as the above. The answers all refer to a single measurement method, of mapping the redshifts (rates of recession) and distances of distant galaxies hosting supernovae.

- One way to measure the Hubble constant is by observing the distance and apparent work of celestial objects/standard candles such as Type Ia supernovae.
- One way to measure the Hubble constant is by observing the redshift of supernovae, which provides direct evidence of the expansion of the universe.
- One way to measure the Hubble constant is by observing the expansion of the universe through the measurement of the redshift of the light from distant supernovae.
- One way to measure the Hubble constant is by observing the redshift of distant supernovae, as their distance changes relates to the expansion of the universe.
- One way to measure the Hubble constant is by observing the rate of recession of distant galaxies, which allows astronomers to determine the expansion rate of the universe.
- One way to measure the Hubble constant is by observing the redshift of distant supernovae, which allows astronomers to determine the expansion rate of the universe.
- One way to measure the Hubble constant is by observing the redshift of distant supernovae, which allows astronomers to determine the expansion rate of the universe.
- One way to measure the Hubble constant is by observing the redshifts of distant galaxies, which indicate stretched light due to the expansion of the universe.

Diversity steering makes the answer set more diverse for MDMs as well. It consists mostly of answers addressing the same supernovae-based measurement method covered above (cluster 2) and also includes low-likelihood vague or nonsensical ones, but also covers an answer addressing the cosmic microwave background method (cluster 6).

- One of the better methods is to measure the Hubble constant (known as H_0) using modern techniques, resulting in better accuracy than distance-ladder observations. (cluster 1): vague

https://doi.org/10.52202/085713-1080

- One method around the red complex within the past few years of water resources is to estimate it using radiation and other methods. (cluster 5): nonsensical
- The Hubble constant can be measured using standard galaxy-distance measurements by comparing the brightness of distant supernovae to their redshift. (cluster 2)
- One way to measure the Hubble constant is by observing Type Ia supernovae and creating the standard distance ladder (LZ ladder diagram). (cluster 2)
- One way to measure the Hubble constant is through observations of remote galaxies, such as with the Hubble Telescope or Lemlint. (cluster 3): vague
- One way to measure the Hubble constant involves measuring the distance between seemingly related astronomical objects. (cluster 4): vague
- One method to measure the Hubble constant is by observing the cosmic microwave background using microwave receivers and other sophisticated techniques. (cluster 6)
- To measure the Hubble constant, one can use observations and discoveries of Type Ia supernovae, also known as supernovae H0. (cluster 2)

D Mutual information results

As shown in Table 2, AUROC gains over baselines for MI are modest, and the results exhibit high variance. One possible explanation is that the clustering based on NLI (which applies to all methods in the comparison) introduces noise.

Table 2: AUROC of MI [2] computed on generations from various sampling schemes. Each scheme uses N=8 pairs of iteratively generated answers. The correctness metric Rouge-L (F1 score) was thresholded at 0.2. All numbers are mean±std over 5 jackknife samples of size 200. "Vanilla" refers to standard sampling without any tempering ($\tau = 1$). Best methods based on mean are bolded.

Dataset	**Model**	Vanilla	$\tau = 2$	DBS [43]	SDLG [4]	Ours
CoQA	LLaMA 3 8B-Instruct	.61±.07	.66±.09	.58±.12	.64±.08	**.68±.07**
	LLaDA 8B-Instruct	.51±.03	**.54±.06**	-	-	**.56±.06**
TriviaQA	LLaMA 3 8B-Instruct	.49±.07	.51±.04	.54±.08	**.57±.11**	.54±.07
	LLaDA 8B-Instruct	.52±.08	.53±.06	-	-	**.55±.06**
AmbigQA	LLaMA 3 8B-Instruct	**.56±.04**	.55±.07	.51±.08	**.56±.08**	**.56±.06**
	LLaDA 8B-Instruct	**.53±.09**	.51±.12	-	-	**.53±.05**
TruthfulQA	LLaMA 3 8B-Instruct	.58±.06	.57±.07	.59±.08	**.63±.08**	**.63±.07**
	LLaDA 8B-Instruct	.60±.06	.67±.10	-	-	**.69±.08**

E Variance reduction with control variates

In this section, we provide additional justification for using control variates to reduce the variance of importance-weighted estimators in Section 3.3. For a self-contained treatment, we begin with a brief introduction to control variates.

Let $Z \sim p$ be a random variable and assume we wish to estimate its moment

$$\mu = \mathbb{E}_p\big[h(Z)\big],$$

using N iid samples $Z_{1:N}$. The classical estimator takes the form

$$\hat{\mu} = \frac{1}{N}\sum_{i=1}^{N} h(Z_i),$$

which has variance $\mathrm{Var}[\hat{\mu}] = \frac{\mathrm{Var}[h(Z)]}{N}$.

The idea of control variates is to pick any auxiliary function g whose mean $\mu_g = \mathbb{E}_p[g(Z)]$ is known in closed form. For any coefficient $\alpha \in \mathbb{R}$,

$$\hat{\mu}_{\mathrm{cv}}(\alpha) = \frac{1}{N}\sum_{i=1}^{N}\Big\{h(Z_i) - \alpha\big[g(Z_i) - \mu_g\big]\Big\}$$

is unbiased, because $\mathbb{E}_p[g(Z) - \mu_g] = 0$.

It then remains to choose the optimal coefficient $\alpha^\star$ that minimizes the variance. Define $\sigma_h^2 = \mathrm{Var}[h(Z)]$, $\sigma_g^2 = \mathrm{Var}[g(Z)]$, and $\sigma_{hg} = \mathrm{Cov}[h(Z), g(Z)]$. Writing $H = h(Z) - \mu$ and $G = g(Z) - \mu_g$, we have

$$\mathrm{Var}\big[\hat{\mu}_{\mathrm{cv}}(\alpha)\big] = \frac{1}{N}\mathrm{Var}\big[H - \alpha G\big] = \frac{1}{N}\big(\sigma_h^2 - 2\alpha\sigma_{hg} + \alpha^2\sigma_g^2\big).$$

Differentiating with respect to α and setting the result to zero,

$$\frac{\partial}{\partial\alpha}\mathrm{Var} = 0 \implies -2\sigma_{hg} + 2\alpha\sigma_g^2 = 0 \implies \alpha^\star = \frac{\sigma_{hg}}{\sigma_g^2} = \frac{\mathrm{Cov}[h(Z), g(Z)]}{\mathrm{Var}[g(Z)]}.$$

Substituting $\alpha^\star$ back yields

$$\mathrm{Var}\big[\hat{\mu}_{\mathrm{cv}}(\alpha^\star)\big] = \frac{1}{N}\sigma_h^2(1 - \rho^2),$$

where $\rho = \frac{\sigma_{hg}}{\sigma_h\sigma_g}$ is the correlation between h and g, so variance is reduced by the factor $1 - \rho^2$ such that the closer $\mid \rho \mid$ is to 1, the greater the reduction in variance.

Semantic entropy The estimator proposed in Equation 10 uses

$$\sum_{i=1}^{N}\tilde{w}_i X_i'$$

as the control variate. The centered negative log probabilities $X_i' = X_i - \mu_X$ with $X_i = -\log p(s^{(i)})$ are strongly correlated with the negative cluster probabilities $Y_i = -\log\hat{p}(c(s^{(i)}))$, being based on the same sample $s^{(i)}$. In natural language generation, likely samples with low X_i tend to be mapped to dominant clusters, so Y_i becomes more negative when X_i is more negative.

The proposed control variate does not incur extra inference cost, as the log probabilities were already computed for evaluating the importance weights w_i. Particularly in the context of self-normalized importance sampling, log probabilities behave nicer than, e.g., a function of the ratio p/q, which can take higher moments.

https://doi.org/10.52202/085713-1080

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We state the main methodological contributions (diversity steering using a NLI model and importance sampling with control variates) in the abstract and Section 1.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the computational limitations in Section 6 and the dependence on NLI clustering and prompt wording in Section 7.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.

- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification:

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Section 5 and Appendix B contain all the main implementation details.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.

 https://doi.org/10.52202/085713-1080

(b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.

(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Appendix B contains an anonymized codebase that implements our method and experiments, with plans to open-source it following the paper's acceptance. All the pretrained models and datasets used in our experiments are open-source.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: In the main text, Section 5 lays out the data splits, baselines, and metrics. Additional details regarding hyperparameters, NLI finetuning, and diversity-steered sampling are deferred to Appendix A and Appendix B.

   Guidelines:

   - The answer NA means that the paper does not include experiments.

- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Due to computational resource constraints, we report the error bars from jackknife sampling.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We include compute details in Appendix B.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We conform to the NeurIPS Code of Ethics.

https://doi.org/10.52202/085713-1080

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: As mentioned in the Introduction, accurate uncertainty characterization is critical for deploying LLMs in safety-critical applications. As this work represents foundational research on uncertainty quantification of LLMs, we do not foresee immediate negative societal impacts.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: We do not pretrain a language model nor release a dataset.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.

- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We properly cite all pretrained models and benchmark datasets used in our experiments.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification:

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification:

    Guidelines:

https://doi.org/10.52202/085713-1080

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification:

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: This paper proposes a new sampling scheme for LLMs that encourages semantic diversity in their generation. We describe the usage of all LLMs used in the experiments.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.