Semantic Entropy Probes: Robust and Cheap Hallucination Detection in LLMs

Jannik Kossen [*1] **Jiatong Han** [*1 2] **Muhammed Razzak** [*1] **Lisa Schut** [1] **Shreshth Malik** [1] **Yarin Gal** [1]

Abstract

We propose semantic entropy probes (SEPs), a cheap and reliable method for uncertainty quantification in Large Language Models (LLMs). Hallucinations, which are plausible-sounding but factually incorrect and arbitrary model generations, present a major challenge to the practical adoption of LLMs. Recent work by Farquhar et al. (2024) proposes semantic entropy (SE), which can reliably detect hallucinations by quantifying the uncertainty over different generations by estimating entropy over semantically equivalent sets of outputs. However, the 5-to-10-fold increase in computation cost associated with SE computation hinders practical adoption. To address this, we propose SEPs, which directly approximate SE from the hidden states of a single generation. SEPs are simple to train and do not require sampling multiple model generations at test time, reducing the overhead of semantic uncertainty quantification to almost zero. We show that SEPs retain high performance for hallucination detection and generalize better to out-of-distribution data than previous probing methods that directly predict model accuracy. Our results across models and tasks suggest that model hidden states capture SE, and our ablation studies give further insights into the token positions and model layers for which this is the case.

1. Introduction

Large Language Models (LLMs) have demonstrated impressive capabilities across a wide variety of natural language processing tasks (Touvron et al., 2023a;b; OpenAI, 2023; Team, 2023; Brown et al., 2020). They are increasingly deployed in real-world settings, including in high-stakes domains such as medicine, journalism, or legal services (Singhal et al., 2022; Weiser, 2023; Opdahl et al., 2023; Shen et al., 2023). It is therefore paramount that we can *trust* the outputs of LLMs. Unfortunately, LLMs have a tendency to *hallucinate*. Originally defined as "content that is nonsensical or unfaithful to the provided source" (Maynez et al., 2020; Filippova, 2020; Ji et al., 2023), the term is now used to refer to nonfactual, arbitrary content generated by LLMs. For example, when asked to generate biographies, even capable LLMs such as GPT-4 will often fabricate facts entirely (Min et al., 2023; Tian et al., 2023).

Various approaches have been proposed to address hallucinations in LLMs (see Appendix A). An effective strategy for detecting hallucinations is to sample multiple responses for a given prompt and check if the different samples convey the same meaning (Farquhar et al., 2024; Kuhn et al., 2023; Kadavath et al., 2022; Duan et al., 2023; Cole et al., 2023; Chen & Mueller, 2023; Elaraby et al., 2023; Manakul et al., 2023b; Min et al., 2023). The core idea is that if the model knows the answer, it will consistently provide the same answer. If the model is hallucinating, its responses may vary across generations.

One explanation for why this works is that LLMs have calibrated uncertainty (Kadavath et al., 2022; OpenAI, 2023), i.e., "language models (mostly) know what they know" (Kadavath et al., 2022). When an LLM is certain about an answer, it consistently provides the correct response. Conversely, when uncertain, it generates arbitrary answers. This suggests that we can leverage model uncertainty to detect hallucinations. However, we cannot use token-level probabilities to estimate uncertainty directly. This is because different sequences of tokens may convey the same meaning. To address this, Farquhar et al. (2024) proposed *semantic entropy* (SE). SE estimates uncertainty across different generations by identifying sets of semantically equivalent responses. These sets are then used to estimate the entropy over the generations (see Appendix B for SE definition).

A major limitation of SE and other sampling-based approaches is that they require multiple model generations for each input query, typically between 5 and 10. This results in a 5-to-10-fold higher cost compared to naive generation without SE, presenting a major hurdle to practical adoption.

We propose *Semantic Entropy Probes* (SEPs), linear probes that capture semantic uncertainty from the hidden states of

[*]Equal contribution [1]OATML, Department of Computer Science, University of Oxford [2]Work done during my affiliation with OATML. Correspondence to: Jannik Kossen <jannik.kossen@cs.ox.ac.uk>.

Published at ICML 2024 Workshop on Foundation Models in the Wild.

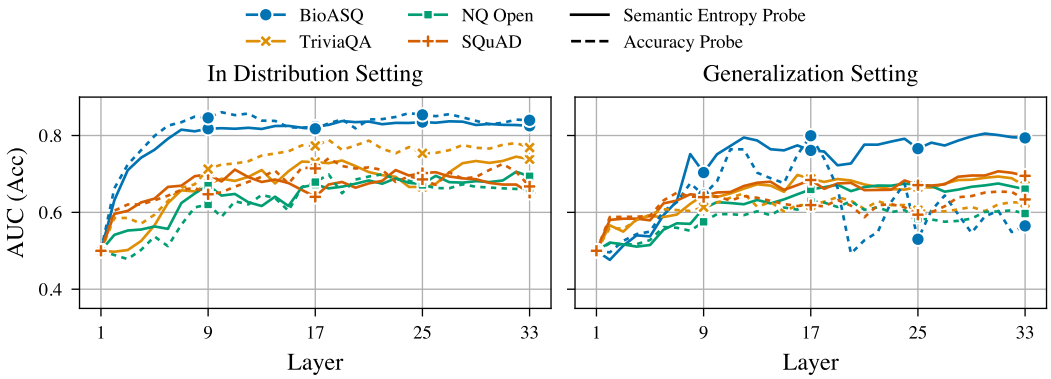

Figure 1: SEPs predict model hallucinations better than accuracy probes when generalizing to unseen tasks. In-distribution, accuracy probes perform better. Short generation setting with Llama-2-7B, SEPs trained on the second-last-token (SLT). For the generalization setting, probes are trained on all tasks except the one that we evaluate on.

LLMs, presenting a cost-effective and reliable hallucination detection method. Similar to sampling-based hallucination detection, SEPs capture the *semantic uncertainty* of the model. Furthermore, they address some of the shortcomings of previous approaches. Contrary to sampling-based hallucination detection, SEPs act directly on a *single* model hidden state and do not require generating multiple samples at test time. SEPs are trained to predict *semantic entropy* (Kuhn et al., 2023) rather than model accuracy, which can be computed without access to ground truth accuracy labels that can be expensive to curate.

2. Semantic Entropy Probes

Training SEPs. SEPs are constructed as linear logistic regression models, trained on the hidden states of LLMs to predict semantic entropy. We create a dataset of $(h_p^l(x), H_{\text{SE}}(x))$ pairs, where x is an input query, $h_p^l(x) \in \mathbb{R}^d$ is the model hidden state at token position p and layer l, d is the hidden state dimension, and $H_{\text{SE}}(x) \in \mathbb{R}$ is the semantic entropy (as defined in Appendix B). Given an input query x, we first generate a high-likelihood model response via greedy sampling and store the hidden state at a particular layer and token position, $h_p^l(x)$. We then sample $N = 10$ responses from the model at high temperature $(T = 1)$ and compute the likelihood that high semantic entropy is high. For inputs, we rely on questions from popular QA datasets (see Section 3 for details), although we do not need the ground-truth labels provided by these datasets and could alternatively compute semantic entropy for any unlabeled set of LLM inputs.

Binarization. Semantic entropy scores are real numbers. However, for the purposes of this paper, we convert them into binary labels, indicating whether semantic entropy is high or low, and then train a logistic regression classifier to predict these labels. Our motivation for doing so is two-fold. For one, we ultimately want to use our probes for predicting binary model correctness, so we eventually need to construct a binary classifier regardless. Additionally, we would like to compare the performance of semantic entropy probes and accuracy probes. This is easier if both probes target binary classification problems. We note that the logistic regression classifier returns probabilities, such that we can always recover fine-grained signals even after transforming the problem into binary classification. See Appendix B for more details.

Probing Locations. We collect hidden states, $h_p^l(x)$, across all layers, l, of the LLM to investigate which layers best capture semantic entropy. We consider two different token positions, p. Firstly, we consider the hidden state at the last token of the *input* x, i.e. the token before generating (TBG) the model response. Secondly, we consider the last token of the *model response*, which is the token before the end-of-sequence token, i.e. the second last token (SLT).

3. Experiment Setup

Tasks. We evaluate SEPs on four datasets: TriviaQA (Joshi et al., 2017), SQuAD (Rajpurkar et al., 2018), BioASQ (Tsatsaronis et al., 2015), and NQ Open (Kwiatkowski et al., 2019). We use the input queries of these tasks to derive training sets for SEPs and evaluate the performance of each method on the validation/test sets, creating splits if needed. We consider a short- and a long-form setting: Short-form answers are generated by few-shot prompting the LLM to answer "as briefly as possible" and long-form answers asks for a "single brief but complete sentence", leading to an approximately six-fold increase in the number of generated tokens. For short-form generations, we follow Kuhn et al. (2023) and assess model accuracy via the SQuAD F1 score, and for long-form generations, we use GPT-4 (OpenAI, 2023) to compare model answers to ground truth labels. We

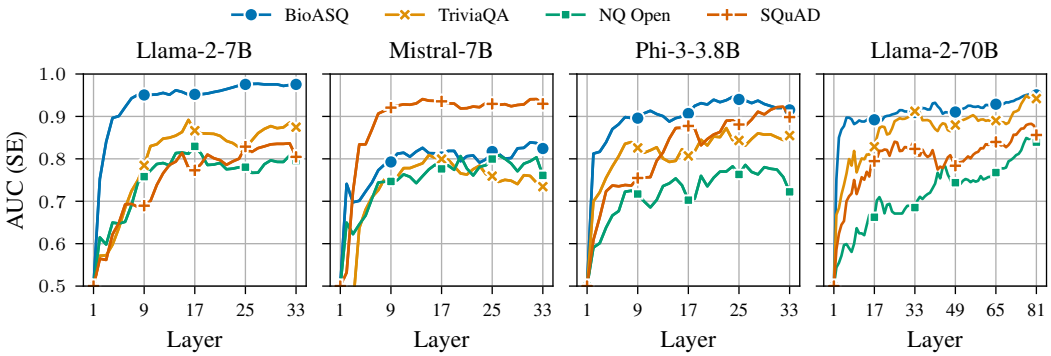

Figure 2: Semantic Entropy Probes (SEPs) achieve high fidelity for predicting semantic entropy. Across datasets and models, SEPs are consistently able to capture semantic entropy from hidden states of mid-to-late layers. Short generation scenario with probes trained on second-last token (SLT).

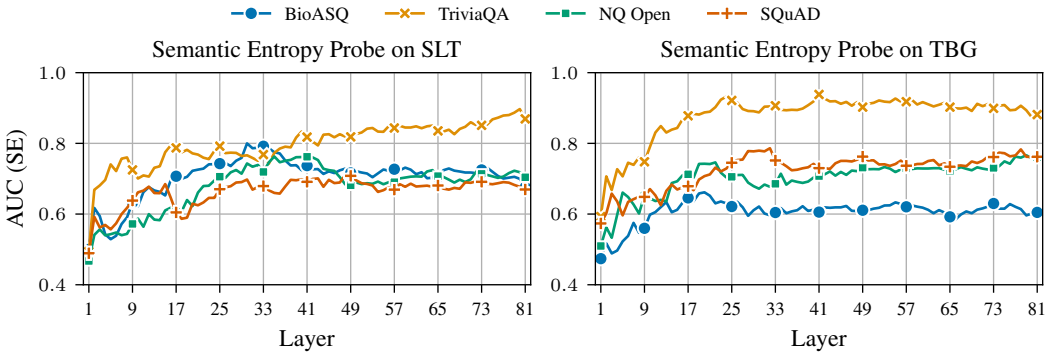

Figure 3: SEPs successfully capture semantic entropy in Llama-2-70B for long generations across layers and for both SLT and TBG token positions.

provide prompt templates in Appendix D.1.

Models. We evaluate SEPs on four different models. For short generations, we generate hidden states and answers for Llama-2 7B and 70B (Touvron et al., 2023b), Mistral 7B (Jiang et al., 2023), and Phi-3 Mini (Abdin et al., 2024), and use DeBERTa-Large (He et al., 2021) as the entailment model for calculating semantic entropy (Kuhn et al., 2023). For long generations, we generate hidden states and answers with Llama-2-70B (Touvron et al., 2023b) and use GPT-3.5 (Brown et al., 2020) to predict entailment.

Baselines. We compare SEPs against the ground truth semantic entropy, accuracy probes supervised with model correctness labels, naive entropy, log likelihood, and the $p(\text{True})$ method of Kadavath et al. (2022). For naive entropy, following Farquhar et al. (2024), we compute the length-normalized average log token probabilities across the same number of generations as for SE. For log likelihood, we use the length-normalized log likelihood of a single model generation. The $p(\text{True})$ method works by constructing a custom few-shot prompt that contains a number of examples – each consisting of a training set input, a corresponding low-temperature model answer, high-temperature model samples, and a model correctness score. Essentially, $p(\text{True})$ treats sampling-based truthfulness detection as an in-context learning task, where the few-shot prompt teaches the model that model answers with high semantic variety are likely incorrect. We refer to Kadavath et al. (2022) for more details.

Linear Probe. For both SEPs and our accuracy probe baseline, we use the logistic regression model from scikit-learn (Pedregosa et al., 2011) with default hyperparameters for L_2 regularization and the LBFGS optimizer.

Evaluation. We evaluate SEPs both in terms of their ability to capture semantic entropy as well as their ability to predict model hallucinations. In both cases, we compute the area under the receiver operating characteristic curve (AUROC), with gold labels given by binarized SE or model accuracy.

4. Experiments

We evaluate Semantic Entropy Probes (SEPs) across various models and datasets. We compare SEPs against the ground truth semantic entropy, accuracy probes supervised with model correctness labels, naive entropy, and the $p(\text{True})$ method by Kadavath et al. (2022). See Section 3 for tasks, baselines, and evaluation details.

4.1. LLM Hidden States Implicitly Capture Semantic Entropy

We explore whether LLM hidden states encode semantic entropy. We study SEPs across different tasks, models, and layers, and compare them to accuracy probes both in- and out-of-distribution.

Hidden States Capture Semantic Entropy. Figure 2 shows that SEPs are consistently able to capture semantic entropy across different models and tasks. Here, probes are trained on hidden states of the second-last-token for the short-form generation setting (see Section 3). In general, we observe that AUROC values increase for later layers in the model, reaching values between 0.7 and 0.95.

Semantic Entropy Can Be Predicted Before Generating. Next, we investigate if semantic entropy can be predicted before even generating the output. Similar to before, Figure C.1 shows AUROC values for predicting binarized semantic entropy from the SEP probes. Perhaps surprisingly (although in line with related work, cf. Appendix A), we find that SEPs can capture semantic entropy even before generation. SEPs consistently achieve good AUROC values, with performance slightly below the SLT experiments in Figure 2. Further, the performance is consistently better at layer 1. The TBG variant provides even larger cost savings than SEPs already do, as it allows us to quantify uncertainty before generating any novel tokens, i.e. with a single forward pass through the model.

SEPs Capture Semantic Uncertainty for Long Generations. While experiments with short generations are popular even in the recent literature (Kuhn et al., 2023; Kadavath et al., 2022; Duan et al., 2023; Cole et al., 2023; Chen & Mueller, 2023), this scenario is increasingly disconnected from popular use cases of LLMs as free-form natural language generators. In recognition of this, we also study our probes in a long-form generation setting, which increases the average length of model responses from $\sim$15 characters in the short-length scenario to $\sim$100 characters.

Figure 3 shows that, even in the long-form setting, SEPs are able to capture semantic entropy well in both the second-last-token and token-before-generation scenarios for Llama-2-70B. Compared to the short-form generation scenario, we now observe more often that AUROC values peak for intermediate layers. This makes sense as hidden states closer to the final layer will likely be preoccupied with predicting the next token. In the long-form setting, the next token is more often unrelated to the semantic uncertainty of the overall answer, and instead concerned with syntax or lexis.

Counterfactual Context Addition Experiment. To confirm that SEPs capture SE rather than relying on spurious correlations, we perform a counterfactual intervention experiment for Llama-2-7B on TriviaQA. For each input question of TriviaQA, the dataset contains a "context", from which the ground truth answer can easily be predicted. We usually exclude this context, because including it makes the task too easy. However, for the purpose of this experiment, we add the context and study how this affects SEP predictions. Figure 5 shows a kernel density estimate of the distribution over the predicted probability for high semantic entropy, $p(\text{high SE})$, for Llama-2-7B on the TriviaQA dataset with context (blue) and without context (orange) in the short generation setting using the SLT. Without context, the distribution for $p(\text{high SE})$ from the SEP is concentrated around 0.9. However, as soon as we provide the context, $p(\text{high SE})$ decreases, as shown by the shift in distribution. As the task becomes much easier – accuracy increases from 26% to 78% – the model becomes more certain – ground truth SE decreases from 1.84 to 0.50. This indicates SEPs accurately capture model behavior for the context addition experiment, with predictions for $p(\text{high SE})$ following ground truth SE behavior when context is added, despite never being trained on inputs with context.

4.2. SEPs Are Cheap and Reliable Hallucination Detectors

We explore the use of SEPs to predict hallucinations, comparing them to accuracy probes and other baselines. Crucially, we also evaluate probes in a challenging generalization setting, testing them on tasks that they were not trained for. This setup is much more realistic than evaluating probes in-distribution, as, for most deployment scenarios, inputs will rarely match the training distribution exactly.

Table 1: ΔAUROC (x100) of SEPs and acc. probes over tasks in-distribution and for task generalization. Avg $\pm$ std error, (S)hort- and (L)ong-form gens.

Model	**In-distribution** (SEP $-$ Acc Pr.)	**Generalization** (SEP $-$ Acc Pr.)
Mistral-7B (S)	2.8 ± 1.4	10.5 ± 3.5
Phi-3-3.8B (S)	2.1 ± 0.8	9.9 ± 2.9
Llama-2-7B (S)	-0.5 ± 2.6	7.7 ± 1.3
Llama-2-70B (S)	1.3 ± 0.7	7.9 ± 3.0
Llama-2-70B (L)	-1.9 ± 7.5	2.2 ± 0.4

Figure 1 shows both in-distribution and generalization performance of SEPs and accuracy probes across different layers for Llama-2-7B in a short-form generation setting trained on the SLT. In-distribution, accuracy probes outperform SEPs across most layers and tasks, with the exception of NQ Open. In Table 1 (**In-distribution**), we report the average difference in AUROC between SEP and accuracy probes for predicting model hallucinations, taking a representative set of high-performing layers (see Appendix D). We find that SEPs and accuracy probes perform similarly on

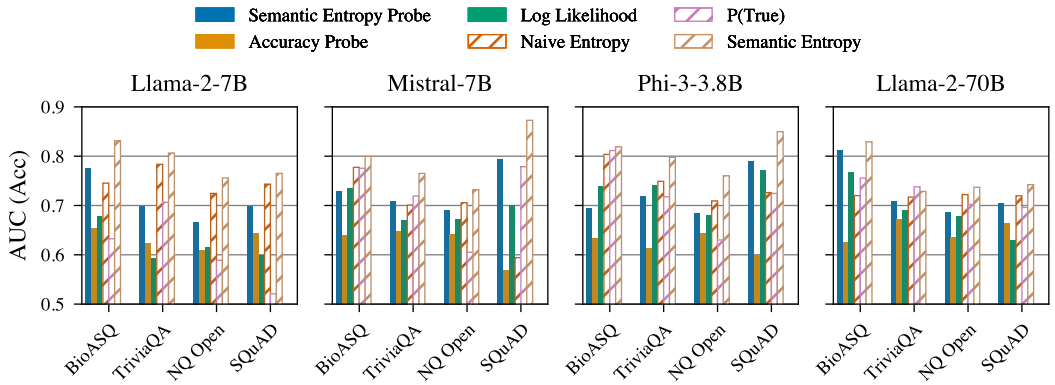

Figure 4: SEPs generalize better to new tasks than accuracy probes across models and tasks. They approach, but do not match, the performance of other, 10x costlier baselines (hatched). Short generation setting, SLT, performance for a selection of representative layers.

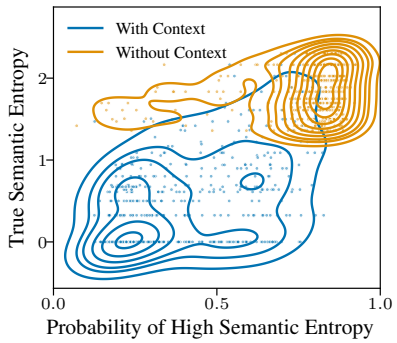

Figure 5: SEPs capture drop in SE due to added context.

in-distribution data across models. We report unaggregated results in Figure C.9. The performance of SEPs here is commendable: SEPs are trained without any ground truth answers or accuracy labels, and yet can capture truthfulness.

When evaluating probe *generalization to new tasks*, SEPs show their true strength. We evaluate probes in a leave-one-out fashion – training on all datasets except one, which we evaluate on. As shown in Figure 1 (right), SEPs consistently outperform accuracy probes across various layers and tasks for short-form generations in the generalization setting. For BioASQ, the difference is particularly large. SEPs clearly generalize better to unseen tasks than accuracy probes.

In Figure 4 and Table 1 (**Generalization**), we report results for more models, taking a representative set of high-performing layers. We again find that SEPs generalize better than accuracy probes to novel tasks. We additionally compare to the sampling-based semantic entropy, naive entropy, and $p(\text{True})$ methods. While SEPs cannot match the performance of these methods, it is important to note the significantly higher cost these baselines incur, requiring 10 additional model generations, whereas SEPs and accuracy probes operate on single generations at test time.

We further evaluate SEPs for long-form generations. As shown in Figure C.10 (right), SEPs outperform accuracy probes for Llama-2-70B for the generalization setting. We also provide in-distribution results for long generations on Llama-2-70B in Figure C.10 (left). Both results confirm the trend discussed above. Overall, our results clearly suggest that SEPs are the best choice for cost-effective uncertainty quantification in LLMs, especially if the distribution of the query data is unknown.

5. Discussion

Our experiments show that SEPs generalize better than accuracy probes – in terms of detecting hallucinations – to inputs from unseen tasks. One potential explanation for this is that semantic uncertainty is a better probing target than correctness, because semantic uncertainty is a more model-internal characteristic that can be better predicted from model hidden states. Model correctness labels required for accuracy probing on the other hand are external and can be noisy, which may make more them difficult to predict from hidden states. We can see evidence for this by comparing in-distribution AUROC for SEPs (for predicting binarized SE) with the AUROC of the accuracy probes for predicting accuracy in Figures C.6 and C.7.

Another possible explanation for the gap in OOD generalization could be that accuracy probes capture model correctness that is *specific* to the training dataset. For example, the probe may latch on to discriminative features for model correctness that relate to the task at hand but do not generalize. Conversely, semantic probes may capture more inherent model states – e.g., uncertainty from failure to gather relevant facts or attributes for the query. The literature on mechanistic interpretability (Nanda et al., 2023) supports the idea that such information is likely contained in model hidden states. We believe that concretizing these links is a fruitful area for future research.

Author Contributions and Acknowledgements. JK conceived the project idea, wrote the initial version of this paper, and, together with MR, provided close mentoring for JH throughout the project. JH wrote the code for SEPs, carried out all of the experiments in the paper, wrote this workshop paper by making edits to the initial version, and created some of the figures. JK, MR, LS, and SM explored SEPs in a hackathon, refining the idea and collecting positive preliminary results. LS provided expertise on model interpretability and suggested extensive improvements to the writing. SM created all plots in the initial version of main paper and appendix. YG provided high level guidance. All authors provided critical feedback on writing.

The authors further thank Kunal Handa, Gunshi Gupta, and all members of the OATML lab for insightful discussions, in particular for the feedback given during the hackathon. SM and LS acknowledge funding from the EPSRC Centre for Doctoral Training in Autonomous Intelligent Machines and Systems (Grant No: EP/S024050/1)

References

Abdin, M., Jacobs, S. A., Awan, A. A., Aneja, J., Awadallah, A., Awadalla, H., Bach, N., Bahree, A., Bakhtiari, A., Behl, H., Benhaim, A., Bilenko, M., Bjorck, J., Bubeck, S., Cai, M., Mendes, C. C. T., Chen, W., Chaudhary, V., Chopra, P., Giorno, A. D., de Rosa, G., Dixon, M., Eldan, R., Iter, D., Garg, A., Goswami, A., Gunasekar, S., Haider, E., Hao, J., Hewett, R. J., Huynh, J., Javaheripi, M., Jin, X., Kauffmann, P., Karampatziakis, N., Kim, D., Khademi, M., Kurilenko, L., Lee, J. R., Lee, Y. T., Li, Y., Liang, C., Liu, W., Lin, E., Lin, Z., Madan, P., Mitra, A., Modi, H., Nguyen, A., Norick, B., Patra, B., Perez-Becker, D., Portet, T., Pryzant, R., Qin, H., Radmilac, M., Rosset, C., Roy, S., Ruwase, O., Saarikivi, O., Saied, A., Salim, A., Santacroce, M., Shah, S., Shang, N., Sharma, H., Song, X., Tanaka, M., Wang, X., Ward, R., Wang, G., Witte, P., Wyatt, M., Xu, C., Xu, J., Yadav, S., Yang, F., Yang, Z., Yu, D., Zhang, C., Zhang, C., Zhang, J., Zhang, L. L., Zhang, Y., Zhang, Y., Zhang, Y., and Zhou, X. Phi-3 technical report: A highly capable language model locally on your phone, 2024.

Agrawal, A., Mackey, L., and Kalai, A. T. Do language models know when they're hallucinating references? *arXiv preprint arXiv:2305.18248*, 2023.

Alain, G. and Bengio, Y. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*, 2016.

Azaria, A. and Mitchell, T. The internal state of an llm knows when it's lying, 2023.

Band, N., Li, X., Ma, T., and Hashimoto, T. Linguistic calibration of language models. *arXiv preprint arXiv:2404.00474*, 2024.

Belinkov, Y. Probing classifiers: Promises, shortcomings, and advances, 2021.

Belrose, N., Furman, Z., Smith, L., Halawi, D., Ostrovsky, I., McKinney, L., Biderman, S., and Steinhardt, J. Eliciting latent predictions from transformers with the tuned lens, 2023.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. Language models are few-shot learners. *NeurIPS*, 33:1877–1901, 2020. URL https://arxiv.org/abs/2005.14165.

Burns, C., Ye, H., Klein, D., and Steinhardt, J. Discovering latent knowledge in language models without supervision, 2022.

Cao, M., Dong, Y., and Cheung, J. C. K. Hallucinated but factual! inspecting the factuality of hallucinations in abstractive summarization. *arXiv preprint arXiv:2109.09784*, 2021.

Chen, J. and Mueller, J. Quantifying uncertainty in answers from any language model and enhancing their trustworthiness. 2023.

Chuang, Y.-S., Xie, Y., Luo, H., Kim, Y., Glass, J., and He, P. Dola: Decoding by contrasting layers improves factuality in large language models. *arXiv preprint arXiv:2309.03883*, 2023.

Cole, J. R., Zhang, M. J., Gillick, D., Eisenschlos, J. M., Dhingra, B., and Eisenstein, J. Selectively answering ambiguous questions. *Conference on Empirical Methods in Natural Language Processing*, 2023.

Deutsch, D., Bedrax-Weiss, T., and Roth, D. Towards question-answering as an automatic metric for evaluating the content quality of a summary. *Transactions of the Association for Computational Linguistics*, 9:774–789, 2021.

Dhuliawala, S., Komeili, M., Xu, J., Raileanu, R., Li, X., Celikyilmaz, A., and Weston, J. Chain-of-verification reduces hallucination in large language models. *arXiv preprint arXiv:2309.11495*, 2023.

Duan, J., Cheng, H., Wang, S., Wang, C., Zavalny, A., Xu, R., Kailkhura, B., and Xu, K. Shifting attention to relevance: Towards the uncertainty estimation of large language models. *arXiv preprint arXiv:2307.01379*, 2023.

Durmus, E., He, H., and Diab, M. Feqa: A question answering evaluation framework for faithfulness assessment in abstractive summarization. *arXiv preprint arXiv:2005.03754*, 2020.

Dziri, N., Madotto, A., Zaïane, O., and Bose, A. J. Neural path hunter: Reducing hallucination in dialogue systems via path grounding. *arXiv preprint arXiv:2104.08455*, 2021.

Elaraby, M., Lu, M., Dunn, J., Zhang, X., Wang, Y., and Liu, S. Halo: Estimation and reduction of hallucinations in open-source weak large language models. *arXiv preprint arXiv:2308.11764*, 2023.

Farquhar, S., Kossen, J., Kuhn, L., and Gal, Y. Personal Communication.

Farquhar, S., Kossen, J., Kuhn, L., and Gal, Y. Detecting Hallucinations in Large Language Models Using Semantic Entropy. *Nature*, 2024.

Feldman, P., Foulds, J. R., and Pan, S. Trapping llm hallucinations using tagged context prompts. *arXiv preprint arXiv:2306.06085*, 2023.

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

Gao, L., Dai, Z., Pasupat, P., Chen, A., Chaganty, A. T., Fan, Y., Zhao, V. Y., Lao, N., Lee, H., Juan, D.-C., et al. Rarr: Researching and revising what language models say, using language models. *arXiv preprint arXiv:2210.08726*, 2022.

He, P., Liu, X., Gao, J., and Chen, W. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=XPZIaotutsD.

Hernandez, E., Li, B. Z., and Andreas, J. Measuring and manipulating knowledge representations in language models. *arXiv preprint arXiv:2304.00740*, 2023.

Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y. J., Madotto, A., and Fung, P. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38, 2023.

Jiang, A. Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D. S., de las Casas, D., Bressand, F., Lengyel, G., Lample, G., Saulnier, L., Lavaud, L. R., Lachaux, M.-A., Stock, P., Scao, T. L., Lavril, T., Wang, T., Lacroix, T., and Sayed, W. E. Mistral 7b. *arXiv*, 2023.

Joshi, M., Choi, E., Weld, D. S., and Zettlemoyer, L. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pp. 1601–1611, May 2017.

Kadavath, S., Conerly, T., Askell, A., Henighan, T., Drain, D., Perez, E., Schiefer, N., Dodds, Z. H., DasSarma, N., Tran-Johnson, E., et al. Language models (mostly) know what they know. *arXiv:2207.05221*, 2022. URL https://arxiv.org/abs/2207.05221.

Kossen, J., Han, J., Razzak, M., Schut, L., Malik, S., and Gal, Y. Semantic entropy probes: Robust and cheap hallucination detection in llms. *arXiv:2406.15927*, 2024. URL https://arxiv.org/abs/2406.15927.

Kuhn, L., Gal, Y., and Farquhar, S. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *International Conference on Learning Representations (ICLR)*, 2023.

Kwiatkowski, T., Palomaki, J., Redfield, O., Collins, M., Parikh, A., Alberti, C., Epstein, D., Polosukhin, I., Kelcey, M., Devlin, J., Lee, K., Toutanova, K. N., Jones, L., Chang, M.-W., Dai, A., Uszkoreit, J., Le, Q., and Petrov, S. Natural questions: a benchmark for question answering research. *Transactions of the Association of Computational Linguistics*, 2019.

Lanham, T., Chen, A., Radhakrishnan, A., Steiner, B., Denison, C., Hernandez, D., Li, D., Durmus, E., Hubinger, E., Kernion, J., et al. Measuring faithfulness in chain-of-thought reasoning. *arXiv preprint arXiv:2307.13702*, 2023.

Lee, N., Ping, W., Xu, P., Patwary, M., Fung, P. N., Shoeybi, M., and Catanzaro, B. Factuality enhanced language models for open-ended text generation. *Advances in Neural Information Processing Systems*, 35:34586–34599, 2022.

Li, K., Patel, O., Viégas, F., Pfister, H., and Wattenberg, M. Inference-time intervention: Eliciting truthful answers from a language model. *Advances in Neural Information Processing Systems*, 36, 2024.

Li, X., Zhao, R., Chia, Y. K., Ding, B., Joty, S., Poria, S., and Bing, L. Chain-of-knowledge: Grounding large language models via dynamic knowledge adapting over heterogeneous sources. In *The Twelfth International Conference on Learning Representations*, 2023.

Lin, S., Hilton, J., and Evans, O. Teaching models to express their uncertainty in words. *TMLR*, 2023. URL https://arxiv.org/abs/2205.14334.

Loh, W.-Y. Classification and regression trees. *Wiley interdisciplinary reviews: data mining and knowledge discovery*, 1(1):14–23, 2011.

Luo, J., Xiao, C., and Ma, F. Zero-resource hallucination prevention for large language models. *arXiv preprint arXiv:2309.02654*, 2023.

MacDiarmid, M., Maxwell, T., Schiefer, N., Mu, J., Kaplan, J., Duvenaud, D., Bowman, S., Tamkin, A., Perez, E., Sharma, M., Denison, C., and Hubinger, E. Simple probes can catch sleeper agents, 2024. URL https://www.anthropic.com/news/probes-catch-sleeper-agents.

Malinin, A. and Gales, M. Uncertainty estimation in autoregressive structured prediction. *ICLR*, 2021.

Manakul, P., Liusie, A., and Gales, M. J. Mqag: Multiple-choice question answering and generation for assessing information consistency in summarization. *arXiv preprint arXiv:2301.12307*, 2023a.

Manakul, P., Liusie, A., and Gales, M. J. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. In *Conference on Empirical Methods in Natural Language Processing*, 2023b.

Marks, S. and Tegmark, M. The Geometry of Truth: Emergent Linear Structure in Large Language Model Representations of True/False Datasets, 2023.

Maynez, J., Narayan, S., Bohnet, B., and McDonald, R. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 1906–1919, Online, July 2020. Association for Computational Linguistics.

Meta. Introducing meta llama 3: The most capable openly available llm to date, 2024. URL https://ai.meta.com/blog/meta-llama-3/. [Online; accessed June 16 2024].

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

Min, S., Krishna, K., Lyu, X., Lewis, M., Yih, W.-t., Koh, P. W., Iyyer, M., Zettlemoyer, L., and Hajishirzi, H. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. *arXiv preprint arXiv:2305.14251*, 2023.

Mündler, N., He, J., Jenko, S., and Vechev, M. Self-contradictory hallucinations of large language models: Evaluation, detection and mitigation. *arXiv preprint arXiv:2305.15852*, 2023.

Murray, K. and Chiang, D. Correcting length bias in neural machine translation. *Proceedings of the Third Conference on Machine Translation*, pp. 212–223, 2018.

Nan, F., Santos, C. N. d., Zhu, H., Ng, P., McKeown, K., Nallapati, R., Zhang, D., Wang, Z., Arnold, A. O., and Xiang, B. Improving factual consistency of abstractive summarization via question answering. *arXiv preprint arXiv:2105.04623*, 2021.

Nanda, N., Rajamanoharan, S., Kramar, J., and Shah, R. Fact finding: Attempting to reverse-engineer factual recall on the neuron level, Dec 2023. URL https://www.alignmentforum.org/s/hpWHhjvjn67LJ4xXX/p/iGuwZTHWb6DFY3sKB.

Opdahl, A. L., Tessem, B., Dang-Nguyen, D.-T., Motta, E., Setty, V., Throndsen, E., Tverberg, A., and Trattner, C. Trustworthy journalism through AI. *Data Knowl. Eng.*, pp. 102182, April 2023.

OpenAI. GPT-4 technical report, 2023.

Pal, K., Sun, J., Yuan, A., Wallace, B., and Bau, D. Future lens: Anticipating subsequent tokens from a single hidden state. In *Proceedings of the 27th Conference on Computational Natural Language Learning (CoNLL)*. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.conll-1.37. URL http://dx.doi.org/10.18653/v1/2023.conll-1.37.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2011.

Peng, B., Galley, M., He, P., Cheng, H., Xie, Y., Hu, Y., Huang, Q., Liden, L., Yu, Z., Chen, W., et al. Check your facts and try again: Improving large language models with external knowledge and automated feedback. *arXiv preprint arXiv:2302.12813*, 2023.

Petroni, F., Rocktäschel, T., Lewis, P., Bakhtin, A., Wu, Y., Miller, A. H., and Riedel, S. Language models as knowledge bases? *arXiv preprint arXiv:1909.01066*, 2019.

Rajpurkar, P., Jia, R., and Liang, P. Know what you don't know: Unanswerable questions for squad. *arXiv preprint arXiv:1806.03822*, 2018.

Rawte, V., Sheth, A., and Das, A. A survey of hallucination in large foundation models. *arXiv preprint arXiv:2309.05922*, 2023.

Rimsky, N., Gabrieli, N., Schulz, J., Tong, M., Hubinger, E., and Turner, A. M. Steering llama 2 via contrastive activation addition. *arXiv preprint arXiv:2312.06681*, 2023.

Roberts, A., Raffel, C., and Shazeer, N. How much knowledge can you pack into the parameters of a language model? *arXiv preprint arXiv:2002.08910*, 2020.

Shen, Y., Heacock, L., Elias, J., Hentel, K. D., Reig, B., Shih, G., and Moy, L. ChatGPT and other large language models are double-edged swords. *Radiology*, 307(2): e230163, April 2023.

Shi, W., Han, X., Lewis, M., Tsvetkov, Y., Zettlemoyer, L., and Yih, S. W.-t. Trusting your evidence: Hallucinate less with context-aware decoding. *arXiv preprint arXiv:2305.14739*, 2023.

Singhal, K., Azizi, S., Tu, T., Mahdavi, S. S., Wei, J., Chung, H. W., Scales, N., Tanwani, A., Cole-Lewis, H., Pfohl, S., Payne, P., Seneviratne, M., Gamble, P., Kelly, C., Scharli, N., Chowdhery, A., Mansfield, P., y Arcas, B. A., Webster, D., Corrado, G. S., Matias, Y., Chou, K., Gottweis, J., Tomasev, N., Liu, Y., Rajkomar, A., Barral, J., Semturs, C., Karthikesalingam, A., and Natarajan, V. Large language models encode clinical knowledge, 2022.

Su, D., Li, X., Zhang, J., Shang, L., Jiang, X., Liu, Q., and Fung, P. Read before generate! faithful long form question answering with machine reading. *arXiv preprint arXiv:2203.00343*, 2022.

Subramani, N., Suresh, N., and Peters, M. E. Extracting latent steering vectors from pretrained language models. *arXiv:2205.05124*, 2022.

Team, T. G. Gemini: a family of highly capable multimodal models. 2023.

Tian, K., Mitchell, E., Yao, H., Manning, C. D., and Finn, C. Fine-tuning language models for factuality. *arXiv*, 2023.

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al. Llama: Open and efficient foundation language models. *arXiv:2302.13971*, 2023a. URL https://arxiv.org/abs/2302.13971.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv:2307.09288*, 2023b. URL https://arxiv.org/abs/2307.09288.

Tsatsaronis, G., Balikas, G., Malakasiotis, P., Partalas, I., Zschunke, M., Alvers, M. R., Weissenborn, D., Krithara, A., Petridis, S., Polychronopoulos, D., Almirantis, Y., Pavlopoulos, J., Baskiotis, N., Gallinari, P., Artiéres, T., Ngomo, A.-C. N., Heino, N., Gaussier, E., Barrio-Alvers, L., Schroeder, M., Androutsopoulos, I., and Paliouras, G. An overview of the BIOASQ large-scale biomedical semantic indexing and question answering competition. *BMC Bioinformatics*, 16:138, April 2015.

Turpin, M., Michael, J., Perez, E., and Bowman, S. Language models don't always say what they think: unfaithful explanations in chain-of-thought prompting. *Advances in Neural Information Processing Systems*, 36, 2024.

Varshney, N., Yao, W., Zhang, H., Chen, J., and Yu, D. A stitch in time saves nine: Detecting and mitigating hallucinations of llms by actively validating low-confidence generation.

Wang, A., Cho, K., and Lewis, M. Asking and answering questions to evaluate the factual consistency of summaries. *arXiv preprint arXiv:2004.04228*, 2020.

Weiser, B. Lawyer who used ChatGPT faces penalty for made up citations. *The New York Times*, June 2023.

Zhang, S., Pan, L., Zhao, J., and Wang, W. Y. Mitigating language model hallucination with interactive question-knowledge alignment. *arXiv preprint arXiv:2305.13669*, 2023a.

Zhang, Y., Li, Y., Cui, L., Cai, D., Liu, L., Fu, T., Huang, X., Zhao, E., Zhang, Y., Chen, Y., et al. Siren's song in the ai ocean: a survey on hallucination in large language models. *arXiv preprint arXiv:2309.01219*, 2023b.

Zou, A., Phan, L., Chen, S., Campbell, J., Guo, P., Ren, R., Pan, A., Yin, X., Mazeika, M., Dombrowski, A.-K., et al. Representation engineering: A top-down approach to ai transparency. *arXiv preprint arXiv:2310.01405*, 2023.

A. Related Work

LLM Hallucinations. We refer to Rawte et al. (2023); Zhang et al. (2023b) for extensive surveys on hallucinations in LLMs and here briefly review the most relevant related work to this paper. Early work on hallucinations in language models typically refers to issues in summarization tasks where models "hallucinate" content that was not faithful to the provided source text (Maynez et al., 2020; Deutsch et al., 2021; Durmus et al., 2020; Cao et al., 2021; Wang et al., 2020; Manakul et al., 2023a; Nan et al., 2021). Around the same time, research emerged that showed LLMs themselves could store and retrieve factual knowledge (Petroni et al., 2019), leading to the currently popular closed-book setting, where LLMs are queried without any additional context (Roberts et al., 2020). Since then, a large variety of work has focused on detecting hallucinations in LLMs for general natural language generation tasks. These can typically be classified into one of two directions: sampling-based and retrieval-based approaches.

Sampling-Based Hallucination Detection. For sampling-based approaches, a variety of methods have been proposed that sample multiple model completions for a given query and then quantify the semantic difference between the model generations (Kuhn et al., 2023; Kadavath et al., 2022; Duan et al., 2023; Cole et al., 2023; Chen & Mueller, 2023; Elaraby et al., 2023). For this paper, Kuhn et al. (2023) is particularly relevant, as we use their semantic entropy measure to supervise our hidden state probes (we summarize their method in Appendix B). A different line of work does not directly re-sample answers for the same query, but instead asks follow-up questions to uncover inconsistencies in the original answer (Dhuliawala et al., 2023; Agrawal et al., 2023). Recent work has also proposed to detect hallucinations in scenarios where models generate entire paragraphs of text by decomposing the paragraph into individual facts or sentences, and then validating the uncertainty of those individual facts separately (Luo et al., 2023; Mündler et al., 2023; Manakul et al., 2023b; Dhuliawala et al., 2023).

Retrieval-Based Methods. A different strategy to mitigate hallucinations is to rely on external knowledge bases, e.g. web search, to verify the factuality of model responses (Feldman et al., 2023; Zhang et al., 2023a; Peng et al., 2023; Dziri et al., 2021; Gao et al., 2022; Li et al., 2023; Varshney et al.; Su et al., 2022). An advantage of such approaches is that they do not rely on good model uncertainties and can be used directly to fix errors in model generations. However, retrieval-based approaches can add significant cost and latency. Further, they may be less effective for domains such as reasoning, where LLMs are also prone to produce unfaithful and misleading generations (Turpin et al., 2024; Lanham et al., 2023). Thus, retrieval- and uncertainty-based methods should be combined for maximum effect.

Sampling and Finetuning Strategies. A number of different strategies exist to reduce, rather than detect, the number of hallucinations that LLMs generate. Previous work has proposed simple adaptations to LLM sampling schemes (Lee et al., 2022; Chuang et al., 2023; Shi et al., 2023), preference optimization targeting factuality (Tian et al., 2023), or finetuning to align "verbal" uncertainties of LLMs with model accuracy (Mielke et al., 2022; Lin et al., 2023; Band et al., 2024).

Understanding Hidden States. Recent work suggests that simple operations on LLM hidden states can qualitatively change model behavior (Zou et al., 2023; Subramani et al., 2022; Rimsky et al., 2023) manipulate knowledge (Hernandez et al., 2023), or reveal deceitful intent (MacDiarmid et al., 2024). Probes can be a valuable tool to better understand the internal representations of neural networks like LLMs (Alain & Bengio, 2016; Belinkov, 2021). Previous work has shown that hidden state probes can predict LLM outputs one or multiple tokens ahead with high accuracy (Belrose et al., 2023; Pal et al., 2023). Relevant to our paper is recent work that suggests there is a "truthfulness" direction in latent space that predicts correctness of statements and generations (Marks & Tegmark, 2023; Azaria & Mitchell, 2023; Burns et al., 2022; Li et al., 2024; Azaria & Mitchell, 2023). Our work extends this – we are also interested in predicting if the model is hallucinating nonfactual responses, however, rather than directly supervising probes with accuracy labels, we argue that capturing semantic entropy is key for generalization performance.

B. Semantic Entropy

Measuring uncertainty in free-form natural language generation tasks is challenging. The uncertainties over tokens output by the language model can be misleading because they conflate semantic uncertainty, uncertainty over the meaning of the generation, with lexical and syntactic uncertainty, uncertainty over how to phrase the answer (see the example in Section 1). To address this, Farquhar et al. (2024) propose *semantic entropy*, which aggregates token-level uncertainties over clusters of semantic equivalence. Semantic entropy is important in the context of this paper because we use it as the supervisory signal to train our hidden state SEP probes.

Semantic entropy is calculated in three steps: (1) for a given query x, sample model completions from the LLM, (2) aggregate the generations into clusters $(C_1, \ldots, C_K)$ of equivalent semantic meaning, (3) calculate semantic entropy, H_{SE},

by aggregating uncertainties within each cluster. Step (1) is trivial, and we detail steps (2) and (3) below.

Semantic Clustering. To determine if two generations convey the same meaning, Kuhn et al. (2023) use natural language inference (NLI) models, such as DeBERTa (He et al., 2021), to predict entailment between the generations. Concretely, two generations s_a and s_b are identical in meaning if s_a entails s_b and s_b entails s_a, i.e. they entail each other bi-directionally. Kuhn et al. (2023) then propose a greedy algorithm to cluster generations semantically: for each sample s_a, we either add it to an existing cluster C_k if bi-directional entailment holds between s_a and a sample $s_b \in C_k$, or add it to a new cluster if the semantic meaning of s_a is distinct from all existing clusters. After processing all generations, we obtain a clustering of the generations by semantic meaning.

Semantic Entropy. Given an input context x, the joint probability of a generation s consisting of tokens $(t_1, \ldots, t_n)$ is defined as the product of conditional token probabilities in the sequence,

$$p(s \mid x) = \prod_{i=1}^{n} p(t_i \mid t_{1:i-1}, x). \tag{B.1}$$

The probability of the semantic cluster C is then the aggregate probability of all possible generations s which belong to that cluster,

$$p(C \mid x) = \sum_{s \in C} p(s \mid x). \tag{B.2}$$

The uncertainty associated with the distribution over semantic clusters is the semantic entropy,

$$H[C \mid x] = \mathbb{E}_{p(C|x)}[-\log p(C \mid x)]. \tag{B.3}$$

Estimating SE in Practice. In practice, we cannot compute the above exactly. The expectations with respect to $p(s|x)$ and $p(C|x)$ are intractable, as the number of possible token sequences grows exponentially with sequence length. Instead, Kuhn et al. (2023) sample N generations $(s_1, \ldots, s_N)$ at non-zero temperature from the LLM (typically and also in this paper $N = 10$). They then treat $(C_1, \ldots, C_K)$ as Monte Carlo samples from the true distribution over semantic clusters $p(C|x)$, and approximate semantic entropy as

$$H[C \mid x] \approx -\frac{1}{K} \sum_{k=1}^{K} \log p(C_k|x). \tag{B.4}$$

We use an additional approximation, employing a *discrete* variant of semantic entropy that yields equal performance without access to token probabilities, making it compatible with black-box models (Farquhar et al.). For the discrete SE variant, we estimate cluster probabilities $p(C|x)$ as the fraction of generations in that cluster, $p(C_k|x) = \sum_{j=1}^{N} \mathbb{1}[s_j \in C_k]/K$, and then compute semantic entropy as the entropy of the resulting categorical distribution, $H_{\text{SE}}(x) \coloneqq -\sum_{k=1}^{K} p(C_k|x) \log p(C_k|x)$. Discrete SE further avoids problems when estimating Equation (B.4) for generations of different lengths (Malinin & Gales, 2021; Murray & Chiang, 2018; Kuhn et al., 2023).

Binarization. We discuss SE binarization as introduced in Section 2 in more technical detail.

More formally, we compute $\tilde{H}_{\text{SE}}(x) = \mathbb{1}[H_{\text{SE}}(x) > \gamma^\star]$, where $\gamma^\star$ is a threshold that optimally partitions the raw SE scores into high and low values according to the following objective:

$$\gamma^\star = \arg\min_\gamma \sum_{j \in \text{SE}_{\text{low}}} (H_{\text{SE}}(x_j) - \hat{H}_{\text{low}})^2 + \sum_{j \in \text{SE}_{\text{high}}} (H_{\text{SE}}(x_j) - \hat{H}_{\text{high}})^2, \tag{B.5}$$

where

$$\text{SE}_{\text{low}} = \{j : H_{\text{SE}}(x_j) < \gamma\}, \qquad \text{SE}_{\text{high}} = \{j : H_{\text{SE}}(x_j) \geq \gamma\},$$
$$\hat{H}_{\text{low}} = \frac{1}{|\text{SE}_{\text{low}}|} \sum_{j \in \text{SE}_{\text{low}}} H_{\text{SE}}(x_j), \qquad \hat{H}_{\text{high}} = \frac{1}{|\text{SE}_{\text{high}}|} \sum_{j \in \text{SE}_{\text{high}}} H_{\text{SE}}(x_j).$$

This procedure is inspired by splitting objectives used in regression trees (Loh, 2011) and we have found it to perform well in practice compared to alternatives such as soft labelling, cf. Appendix D.

In summary, given a input dataset of queries, $\{x_j\}_{j=1}^{Q}$, we compute a training set of hidden state – binarized semantic entropy pairs, $\{(h_p^l(x_j), \tilde{H}_{\text{SE}}(x_j))\}_{j=1}^{Q}$, and use this to train a linear classifier, which is our semantic entropy probe (SEP). At test time, SEPs predict the probability that a model generation for a given input query x has high semantic entropy.

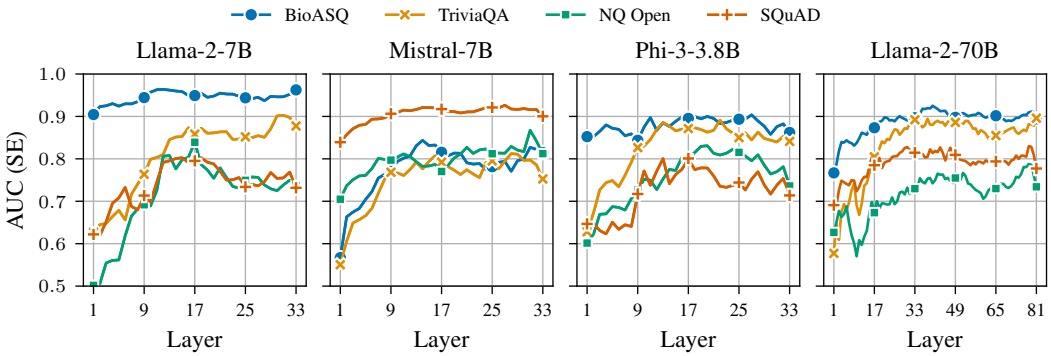

Figure C.1: Semantic entropy can be predicted from the hidden states of the last input token, without generating any novel tokens. Short generations with SEPs trained on the token before generating (TBG).

Table C.1: Task accuracy of models across datasets, for (S)hort- and (L)ong-form Generation.

Model	BioASQ (%)	TriviaQA (%)	NQ Open (%)	SQuAD (%)
Llama-2-70B (L)	60.3	85.0	58.3	43.9
Llama-2-70B (S)	48.4	75.7	49.5	31.4
Llama-2-7B (S)	43.3	64.8	38.3	23.5
Mistral-7B (S)	39.3	52.3	28.3	20.7
Phi-3-3.8B (S)	45.5	48.3	26.1	24.3

C. Additional Results

High AUROC on Predicting BioASQ. AUROC values for Llama-2-7B on BioASQ, in both Figure 2 and Figure C.1, reach very high values, even for early layers. We investigated this and believe it is likely related to the particularities of BioASQ. Concretely, it is the only of our tasks to contain a significant number of yes-no questions, which are generally associated with lower semantic entropy as the possible number of semantic meanings in outcome space is limited. For a model with relatively low accuracy such as Llama-2-7B, simply identifying whether or not the given input is a yes-no question, will lead to high AUROC values.

Model Task Accuracies. We report the accuracies achieved by the models on the various datasets used in this work in Table C.1.

Predicting Model Correctness from Hidden States. Figures C.2 and C.3 give additional results that show we can predict model correctness from hidden states using SEPs trained on the second-last-token (SLT) or token-before-generating (TBG) in the short-form in-distribution scenario across models and tasks. In Figures C.4 and C.5, we further demonstrate that accuracy probes also perform similarly when trained on the SLT or TBG in the short-form in-distribution scenario across models and tasks.

Predicting Correctness vs. Semantic Entropy. Figures C.6 and C.7 show that predicting semantic entropy from hidden states is generally easier than directly predicting model correctness, suggesting that semantic entropy is implicitly encoded in the hidden states.

Additional Comparisons to Baselines. In, Figure C.8 we additionally report results comparing SEPs to accuracy probes across layers for Mistral-7B for the in-distribution and generalization settings. In Figure C.9, we compare the performance of SEPs to baselines for the in-distribution setting across models and datasets, finding that SEPs and accuracy probes perform similarly, with SEPs performing slightly better for 3 out of 5 models. In Figure C.10 we report in- and out-of-distribution results for Llama-2-70B in the long-form generation setting.

Hidden State Alternatives. In addition to investigating the performance of probes on the hidden states, we study whether residual stream or MLP outputs can also be used for semantic entropy prediction. Figure C.11 shows that probing the hidden states results in consistently higher performance across layers.

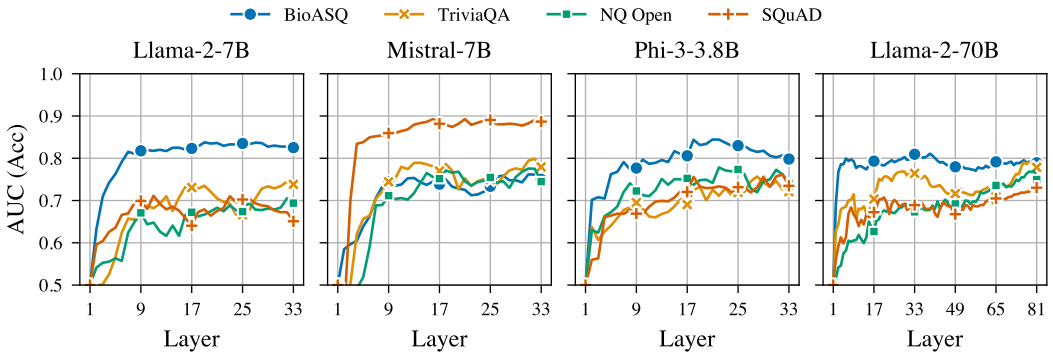

Figure C.2: Semantic Entropy Probes (SEPs) capture model hallucinations. Short generations with SEPs trained on the hidden states of the model at the second-last-token (SLT).

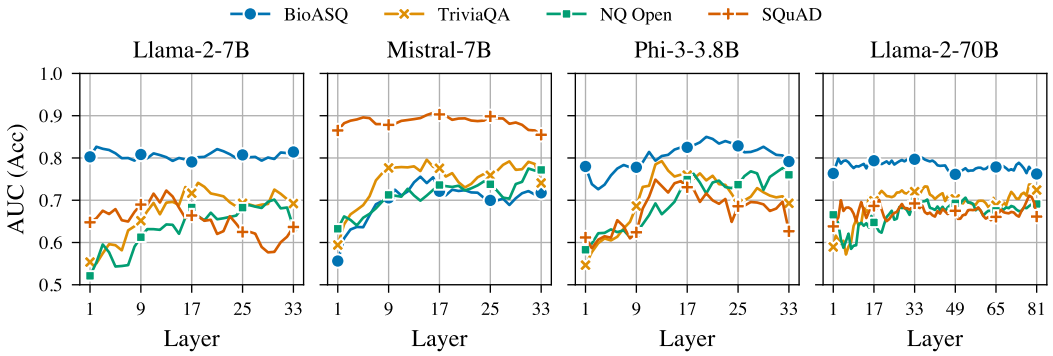

Figure C.3: Semantic Entropy Probes (SEPs) capture model hallucinations. Short generations with SEPs trained on the hidden states of the model at the token-before-generation (TBG).

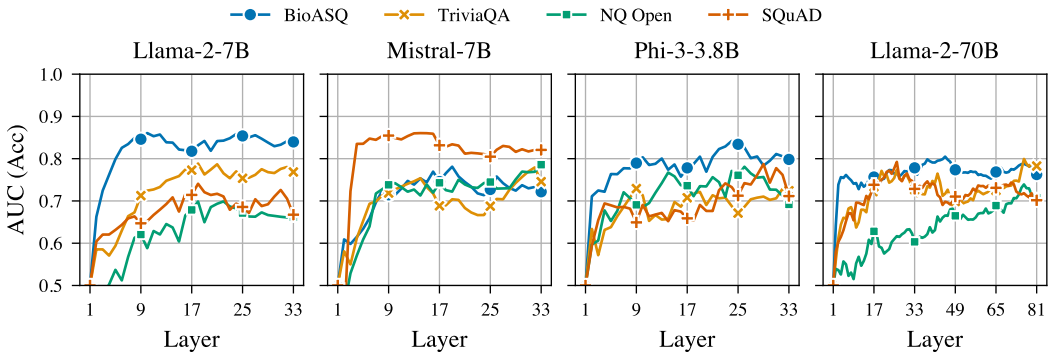

Figure C.4: Accuracy probes for in-distribution short-form generation trained on the second-last-token (SLT).

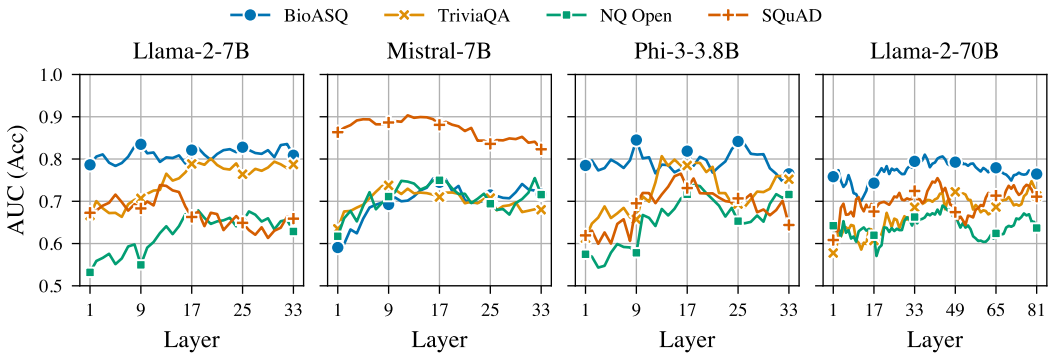

Figure C.5: Accuracy probes for in-distribution short-form generation trained on the token-before-generation (TBG).

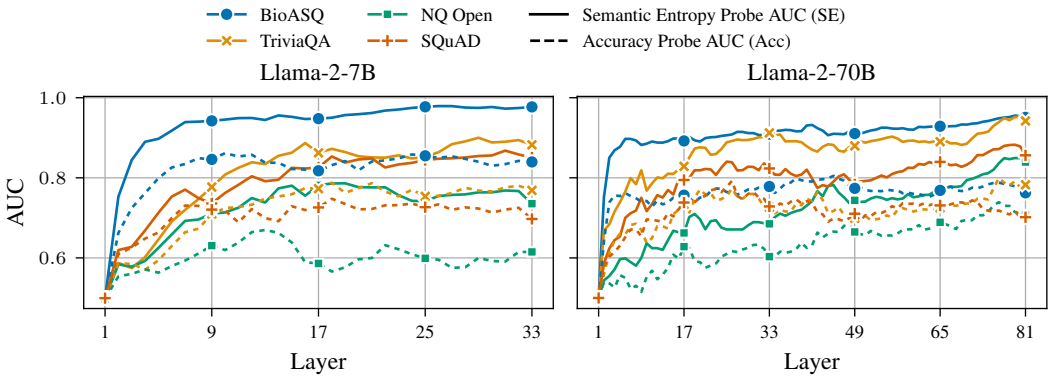

Figure C.6: Predicting semantic entropy from hidden states with SEPs works better than predicting accuracy from the hidden states with accuracy probes. Llama-2-7B and 70B in the short generation setting with probes trained on hidden states of the SLT, evaluated in-distribution.

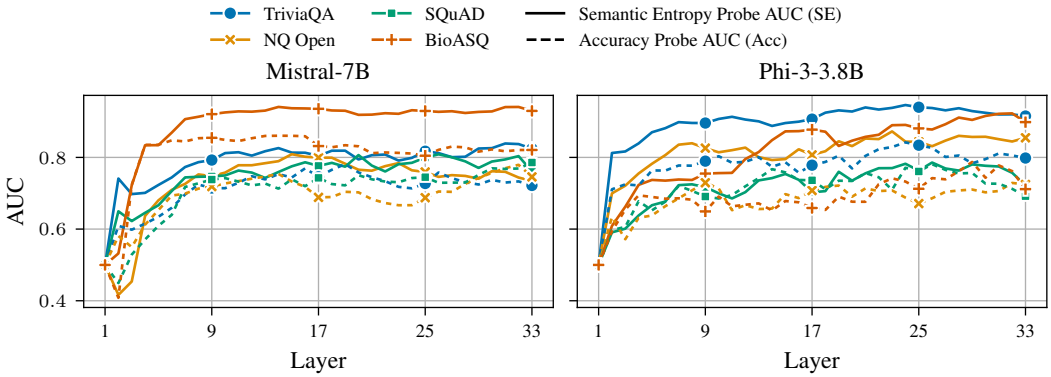

Figure C.7: Predicting semantic entropy from hidden states with SEPs works better than predicting accuracy from the hidden states with accuracy probes. Mistral-7B and Phi-3 Mini in short generation setting with probes trained on hidden states of the SLT, evaluated in-distribution.

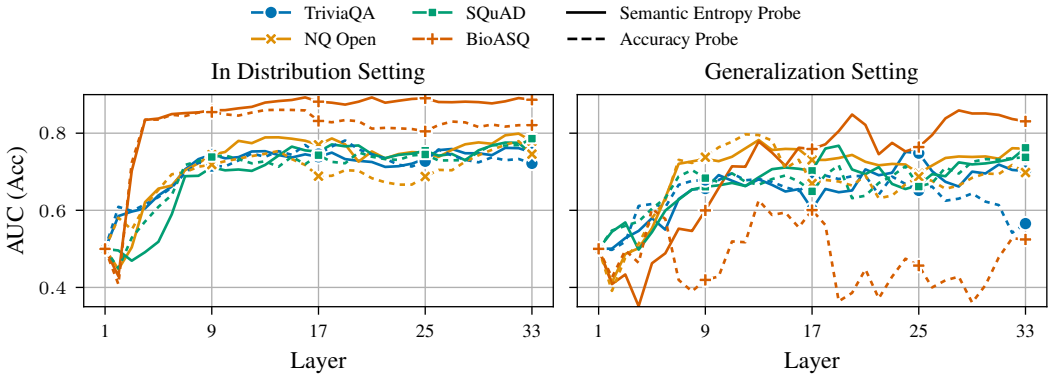

Figure C.8: SEPs predict model hallucinations better than accuracy probes when generalizing to unseen tasks (right). In-distribution, accuracy probes have comparable performance (left). Mistral-7B in the short generations setting with probes trained hidden states from the SLT. For the generalization setting, probes are trained on all tasks except the one that we evaluate on.

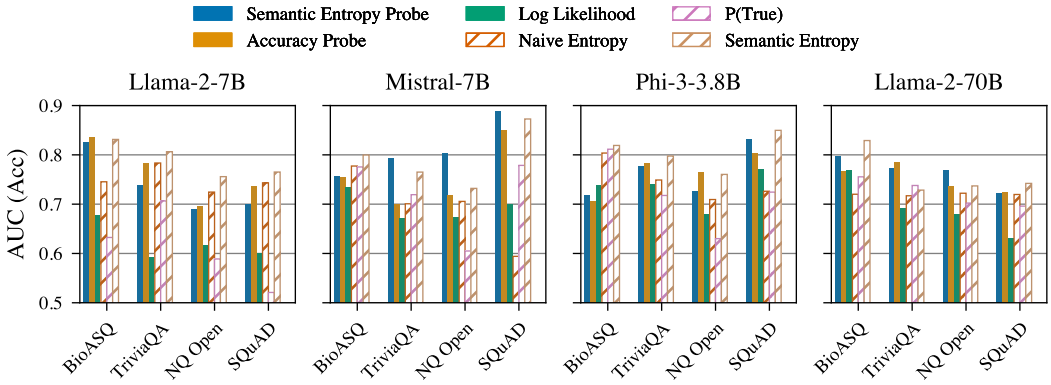

Figure C.9: Short generation performance for the **in-distribution** setting across models compared to baseline methods. Hatched bars indicate more computationally expensive methods.

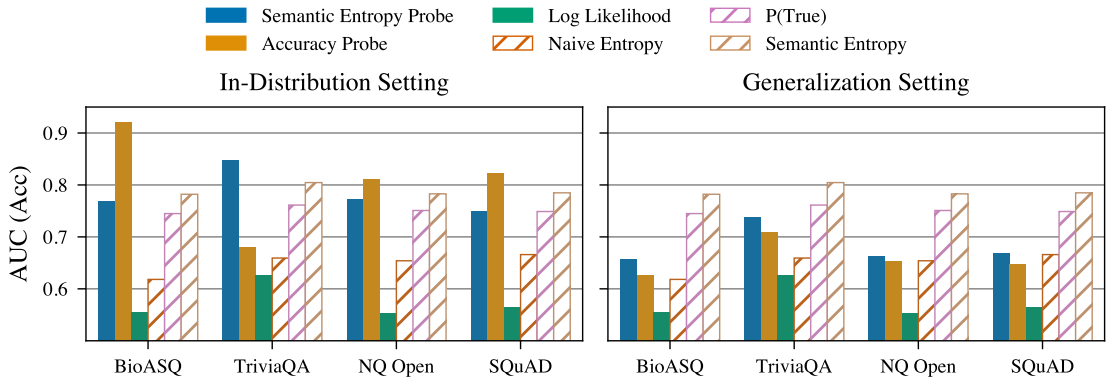

Figure C.10: Semantic entropy probes outperform accuracy probes for hallucination detection in the long-form generation generalization setting with Llama-2-70B. In-distribution, accuracy probes sometimes outperform and sometimes underperform. Probes cannot match the performance of the significantly more expensive baselines.

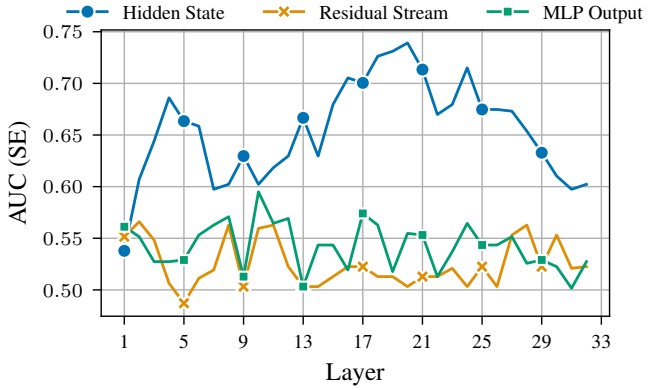

Figure C.11: Probing different model components for SEPs. The hidden states are more predictive than residual streams and MLP outputs. TriviaQA, Llama-2-7B, in-distribution, short-form generations, SLT.

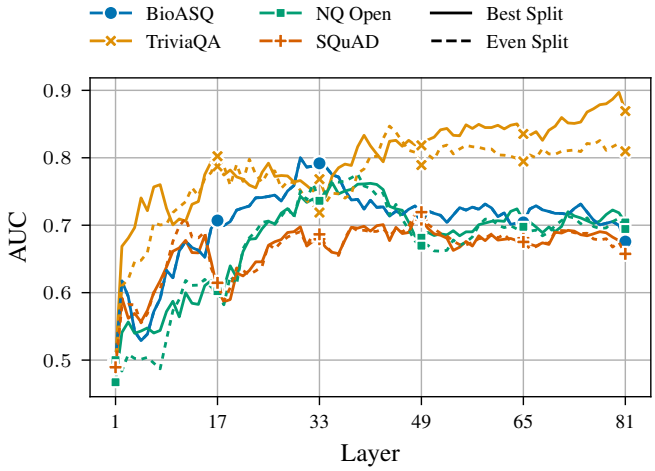

Figure C.12: Comparing binarization methods for semantic entropy. Our "best split" procedure slightly outperforms the "even split" strategy, although SEPs do not appear overly sensitive to the binarization procedure. Long-form generations for Llama-2-70B, SLT, in-distribution.

Different Binarization Procedures. In addition to the "best split" procedure discussed in Section 2 and used in all of our experiments, we here explore the performance of a simple "even split" alternative, which splits semantic entropy into high and low classes such that there are an equal number samples in both classes. Figure C.12 shows that performance is similar, with our optimal splitting procedure slightly outperforming the even split ablation. For illustration purposes, Figure C.13 shows the behavior of the best split objective Equation (B.5) across different thresholds. We have also explored a "soft labelling" strategy as an alternative to hard binarization, for which we obtain soft labels by transforming raw semantic entropies into probabilities with a sigmoid function centered around the best-split threshold, and then train SEPs on the resulting soft labels. Early results did not improve performance.

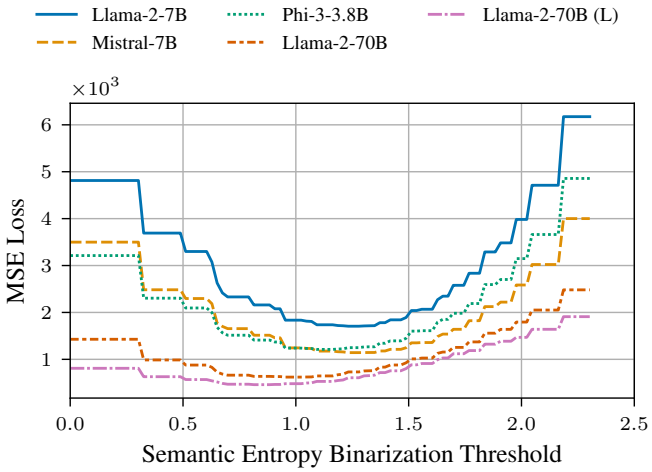

Figure C.13: MSE of the best-split objective Equation (B.5) for different binarization thresholds γ for models in either short-form generation or (L)ong-form generation settings (SLT).

D. Experiment Details

Here we provide additional details to reproduce the experiments of the main paper.

D.1. Prompt Templates

We use the following prompt templates across experiments.

For long-form generations, we use the following prompt template:

```
Answer the following question in a single brief but complete sentence.
Question: [query question]
Answer:
```

For short-form generations, we adjust the instruction and additionally provide 5 demonstration examples with short ground truth answers, to elicit a short answer from the model:

```
Answer the following question as briefly as possible.
Question: [example question 1]
Answer:   [example answer 1]
...
Question: [example question 5]
Answer:   [example answer 5]
Question: [query question]
Answer:
```

Finally, for the counterfactual context addition experiment, we prepend the context, prior to the question:

```
Context: [query context]
Question: [query question]
Answer:
```

D.2. Semantic Entropy Calculation

We compute semantic entropy with $N = 10$ generations sampled at temperature $T = 1.0$ and using default values of top-p ($p = 0.9$) and top-K ($K = 50$).

For short-form generations, we predict entailment using DeBERTa-Large (He et al., 2021) and assess model accuracy via the SQuAD F1 score.

For long-form generations, we predict entailment with GPT-3.5 (Brown et al., 2020) and the following prompt:

```
Here are two possible answers:
Possible Answer 1: [model generation a]
Possible Answer 2: [model generation b]
Does Possible Answer 1 semantically entail Possible Answer 2?
Respond with entailment, contradiction, or neutral.
```

To assess the correctness of long-form generations, we prompt GPT-4 (OpenAI, 2023) as follows:

```
We are assessing the quality of answers to the following question: [query question]
The expected answer is: [ground truth label].
The proposed answer is: [model generation].
Within the context of the question,
does the proposed answer mean the same as the expected answer?
Respond only with yes or no.
Response:
```

D.3. Semantic Entropy Probes

SEPs are trained on the hidden states, which vary in dimensionality between models. We detail the dimensionality of the hidden states, and number of layers in Table D.1.

Table D.1: Models properties and selected layers for concatenation for SEPs and (Acc)uracy (P)robe, in (L)ong-form and (S)hort-form generation settings.

Model Name	No. of Layers	Hidden Dim.	Layers for SEPs	Layers for Acc. P.
Llama-2-70B (L)	80	8192	[74, 75, 76, 77, 78]	[76, 77, 78, 79, 80]
Llama-2-70B (S)	80	8192	[76, 77, 78, 79, 80]	[75, 76, 77, 78, 79]
Llama-2-7B (S)	32	4096	[28, 29, 30, 31, 32]	[18 ,19 ,20 ,21, 22]
Mistral-7B (S)	32	4096	[28, 29, 30, 31, 32]	[12 ,13 ,14 ,15, 16]
Phi-3-3.8B (S)	32	3072	[21, 22, 23, 24, 25]	[25, 26, 27, 28, 29]

Layer Concatenation. For any aggregate results presented in the main paper or appendix, i.e. any barplots or tables, we report SEP and accuracy probe performance on a representative set of high-performing layers. Concretely, we select a set of adjacent layers and concatenate their hidden states to train both types of probes based on the highest mean AUROC value achieved in the interval (on un-concatenated hidden states) in the in-distribution setting. We report the layers across which we concatenate in Table D.1.

Filtering for Long-form Generations. In order to provide a clearer signal to the SEP on what constitutes high and low semantic entropy inputs, we filter out training samples with semantic entropy in between the 55% and 80% quantiles for long generations, as we have found this to give a mild increase in performance. Note that this filtering did not improve performance for the accuracy probes, and we report results for the accuracy probes without filtering.

Training Set Size. For long-generation experiments, we collect 1000 samples across tasks. For short-generation experiments, we collect 2000 samples of hidden state–semantic entropy pairs across tasks. We match the training set sizes between accuracy probes and SEPs.

D.4. Baselines

For the $p(\text{True})$ baseline, we construct a few-shot prompt with 10 examples, where each example is formatted as below:

```
Question: [example question 1]
Brainstormed Answers: [model generation a]
[model generation b]
[model generation c]
..
[model generation j]
Possible answer: [greedy model generation]
Is the possible answer:
A) True
B) False
The possible answer is: [A / B depending on correctness of possible answer]
```

We give an illustrative example below for what this could look like in practice:

```
Question: What is the capital of France?
Brainstormed Answers: The capital of France is Paris.
Paris is the capital of France.
It's Paris.
Possible answer: The capital of France is Paris.
Is the possible answer:
A) True
```

```
B) False
The possible answer is: A
```

For $p(\text{True})$, we obtain the probability of model truthfulness by measuring the token probability of A at the end of the prompt.

D.5. Evaluation

To evaluate the performance of the probes in the generalization setting for both long-form and short-form generations, we employ the following leave-one-out procedure for the aggregate results reported in the barplots and tables.

First, each probe is trained on a single dataset. Then, the trained probes are evaluated on all other datasets in terms of AUROC of detecting hallucinations, excluding the dataset used for training. We then report the mean across all probes evaluated on that specific dataset. This allows us to assess the generalization capability of the probes by measuring their performance on datasets that were not used during the training phase. This scenario is important in practice, as the distribution of the query data will rarely be known.

E. Future Work

We believe it should be possible to fully close the performance gap between sampling-based approaches, such as semantic entropy, and SEPs. One avenue to achieve this could be to increase the scale of the training datasets used to train SEPs. In this work, we relied on established QA tasks to train SEPs to allow for easy comparison to accuracy probes. However, future work could explore training SEPs on unlabelled data, such as inputs generated from another LLM or natural language texts used for general model training or finetuning.

This could massively increase in the amount of training data for SEPs, which should improve probe accuracy, and also allow us to explore other more complex probing techniques that require more training data. We have further proven the usefulness of SEPs in long-form generation settings (Kossen et al., 2024) on Llama-3-70B (Meta, 2024).

F. Compute Resources

We make use of an internal cluster of 24 Nvidia A100 80GB GPUs. We use GPT 3.5 and 4 to calculate calculate the semantic uncertainty and correctness of an answer.

For experiments requiring the use of Llama 70B models, we require 2 A100s to do inference and calculate the hidden states. The smaller models require only a slice of an A100 80GB. However, once the training data for the semantic entropy probes has been created, a CPU-only computing resource is sufficient to fit the logistic regression models.

Based on tracked finished runs, we estimate $\sim$270 GPU-hours plus $\sim$280 CPU-hours to obtain the results in the paper.