SEMANTIC UNCERTAINTY: LINGUISTIC INVARIANCES FOR UNCERTAINTY ESTIMATION IN NATURAL LANGUAGE GENERATION

Lorenz Kuhn, Yarin Gal, Sebastian Farquhar
OATML Group, Department of Computer Science, University of Oxford
lorenz.kuhn@cs.ox.ac.uk

ABSTRACT

We introduce a method to measure uncertainty in large language models. For tasks like question answering, it is essential to know when we can trust the natural language outputs of foundation models. We show that measuring uncertainty in natural language is challenging because of 'semantic equivalence'—different sentences can mean the same thing. To overcome these challenges we introduce *semantic entropy*—an entropy which incorporates linguistic invariances created by shared meanings. Our method is unsupervised, uses only a single model, and requires no modifications to 'off-the-shelf' language models. In comprehensive ablation studies we show that the semantic entropy is more predictive of model accuracy on question answering data sets than comparable baselines.

1 INTRODUCTION

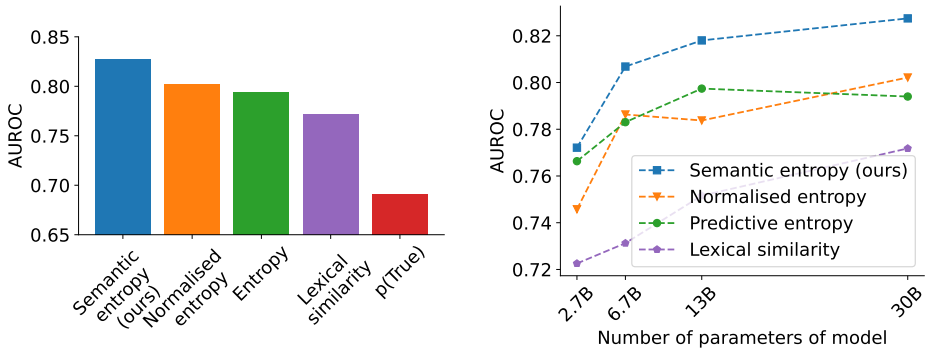
Despite progress in natural language generation (NLG) tasks like question answering or abstractive summarisation (Brown et al., 2020; Hoffmann et al., 2022; Chowdhery et al., 2022), there is little understanding of *uncertainty* in foundation models. Without measures of uncertainty in transformer-based systems it is hard to use generated language as a reliable source of information. Reliable measures of uncertainty have been identified as a key problem in building safer AI systems (Amodei et al., 2016; Hendrycks et al., 2022).

Unfortunately, uncertainty in free-form NLG faces unique challenges. This limits how much we can learn from uncertainty estimation techniques in other applications of deep learning (Gal et al., 2016; Lakshminarayanan et al., 2017; Ovadia et al., 2019) which focuses especially on image classification (Kendall & Gal, 2017) or regression in low-dimensional data spaces (Kuleshov et al., 2018).

The key challenges come from the importance in language of *meanings* and *form*. This corresponds to what linguists and philosophers call the *semantic content* of a sentence and its *syntactic* or *lexical* form. Foundation models output *token*-likelihoods—representing lexical confidence. But for almost all applications we care about meanings! For example, a model which is uncertain about whether to generate "France's capital is Paris" or "Paris is France's capital" is not uncertain in any important sense. Yet, at a token-level the model is uncertain between two *forms* of the same *meaning*. Existing unsupervised methods (e.g., Malinin & Gales (2020)) ignore this distinction.

To address semantic equivalence, we estimate semantic likelihoods—probabilities attached to *meanings* of text rather than standard sequence-likelihoods. We introduce an algorithm for clustering sequences that mean the same thing based on the principle that two sentences mean the same thing if you can infer each from the other. We then use these semantic-likelihoods to estimate semantic uncertainty—uncertainty over different meanings. In particular, we compute the entropy of the probability distribution over meanings. Adjusting for semantic equivalence in this way offers better uncertainty estimation than standard entropy and also greatly improves over methods for model self-evaluation (Kadavath et al., 2022). In addition, semantic entropy scales better with model size and makes better use of increasing numbers of samples than baselines.

We further analyse major challenges for measuring uncertainty in NLG. We show empirically how sampling a set of model answers to estimate entropies in NLG must balance sample accuracy and diversity, which significantly strengthens the baselines we compare against relative to prior imple-



(a) (b)

Figure 1: (a) Our semantic entropy (blue) predicts model accuracy better than baselines on the free-form question answering data set TriviaQA (30B parameter OPT model). Normalised entropy reimplements single-model variant of Malinin & Gales (2020), lexical similarity measures the average Rouge-L in a sampled set of answers for a given question analogously to Fomicheva et al. (2020), entropy and $p(\text{True})$ reimplement Kadavath et al. (2022). (b) Our method's outperformance increases with model size while also doing well for smaller models.

mentations. We also examine the situational heuristic of length-normalising predictive entropies. Our main contributions are thus as follows:

- We explain why uncertainty in free-form NLG is different from other settings (Section 3).
- We introduce *semantic entropy*—a novel entropy-based uncertainty measure which uses our algorithm for marginalising over semantically-equivalent samples (Section 4) and show that it outperforms comparable baselines in extensive ablations with both open- and closed-book free-form question answering using TriviaQA and CoQA (Section 6).
- Through hyperparameter ablations we suggest how to balance the trade-off between sampling diverse and accurate generations for our method as well as baselines (Section 6.2) and show that far fewer samples are needed for effective uncertainty than prior work presumes.

We focus on free-form question answering (QA) because it is a difficult and important use of NLG with high-stakes applications. At the same time, it is easier to establish a ground truth without expensive human evaluation than more nebulous tasks like summarisation.

Ultimately, we show that semantic entropy is an effective unsupervised way to estimate uncertainty in NLG. As an unsupervised method, it requires no further training or data-gathering, unlike supervised methods including Lin et al. (2022a); Kadavath et al. (2022). Semantic entropy is designed to work with existing foundation and large language models with no modifications 'out-of-the-box'. Our experiments use OPT (Zhang et al., 2022) but semantic entropy works with any similar model.

2 Background on Uncertainty Estimation

Our method draws inspiration from probabilistic tools for uncertainty estimation, which have been extensively employed in settings like deep image classification (Gal et al., 2016). Although these methods are often used in Bayesian models, we emphasise that our method does not require any special training or architectural modifications and is not limited to Bayesian settings.

The total uncertainty of a prediction can be understood as the predictive entropy of the output distribution. This measures the information one has about the output given the input. This entropy is highest when the output is minimally informative—predicting the same probability for all possible outcomes. The predictive entropy for a point x is the conditional entropy of the output random variable Y with realisation y given x

$$PE(x) = H(Y \mid x) = -\int p(y \mid x) \ln p(y \mid x) dy \tag{1}$$

One can further distinguish aleatoric uncertainty—uncertainty in the underlying data distribution—and epistemic uncertainty—resulting from missing information (Kendall & Gal, 2017). Epistemic

uncertainty, measured using a mutual information, can be useful but is hard to estimate, especially for very large models, requiring special methods and computational expense. Instead of estimating the epistemic uncertainty based on the model variance, the epistemic uncertainty can also be predicted directly using a second model (see e.g. Jain et al. (2021)). We do not use mutual information in this work, because our focus is on existing foundation models 'off-the-shelf'. Similarly, while, e.g., Malinin & Gales (2020) use ensembles of models to estimate the integral in Eq. (1) we use samples from a single model's output distribution. Prior networks (Malinin & Gales, 2018; Malinin et al., 2020) estimate model uncertainty by emulating an ensemble with a single model. This could be important for NLG because of large model sizes.

For sequence-prediction tasks like NLG, the probability of the entire sequence, $\mathbf{s}$, is the product of the conditional probabilities of new tokens given past tokens, whose resulting log-probability is $\log p(\mathbf{s} \mid x) = \sum_i \log p(s_i \mid \mathbf{s}_{<i})$, where s_i is the i'th output token and $\mathbf{s}_{<i}$ denotes the set of previous tokens. Sometimes, instead of the entropy of these probabilities, the geometric mean token-probability is used instead (Malinin & Gales, 2020) becoming an arithmetic mean log-probability $\frac{1}{N}\sum_i^N \log p(s_i \mid \mathbf{s}_{<i})$. Despite empirical success Murray & Chiang (2018), so far this has little theoretical justification.

Direct application of language models to uncertainty. In contrast to our approach using probabilistic methods, recent work has sought to use the generating language model itself to estimate its own uncertainty. For example, Lin et al. (2022a) finetune language models to verbally describe their confidence. Meanwhile, Kadavath et al. (2022) sample multiple generations and return the completion to an NLG prompt asking if a proposed answer is true (further detail in Appendix B.5). Both Lin et al. (2022a) and Kadavath et al. (2022) also propose ways to finetune predictors on the embeddings of generating models to predict models uncertainty. While promising, these approaches need task-specific labels, additional training, and seem to be unreliable out-of-distribution (as shown in Figures 13 and 14 in Kadavath et al. (2022)).

3 Challenges in Uncertainty Estimation for NLG

Approaches to NLG uncertainty might treat the language model as a black-box (e.g., asking it if its answer is correct) or alternatively focus on the probabilistic model without accounting for the special characteristics of language (e.g., measuring predictive entropy).

Our unsupervised approach instead uses the powerful tools of probabilistic modelling, but also recognises the unique challenges posed by free-form NLG. In this section, we critically analyse the probabilistic interpretation of language models in order to ground both our method and future exploration of the field.

3.1 Semantic equivalence in language outputs

Most machine learning problems have mutually exclusive outputs. An image in class 17 is not class 29 as well; a regression output of 23.1 is not anything else; an RL agent going left does not go right. In contrast, for free-form text generation an output usually means the same thing as many other outputs. For example, "The capital of France is Paris" means the same thing as "France's capital is Paris". Linguists and philosophers distinguish text's meaning—its semantic content—from its syntactic and lexical form. The syntax is the grammatical structure while its lexical form is the specific words used. Lexical equivalence entails the other two, but not the reverse.

We almost always care about the semantic content of a sentence. For decision-problems relying on NLG, *meaning* is usually an invariance in output-space which is not present in the model specification. This is true for question answering, summarisation, artificial assistants. Meanings are especially important for trustworthiness: a system can be reliable even with many different ways to say the same thing but answering with inconsistent meanings shows poor reliability.

We can formalize semantic equivalence mathematically. Let the space of tokens in a language be $\mathcal{T}$. The space of all possible sequences of tokens of length N is then $\mathcal{S}_N \equiv \mathcal{T}^N$. For some sentence $\mathbf{s} \in \mathcal{S}_N$, a sequence of tokens $s_i \in \mathcal{T}$ there is an associated meaning.[1]

Let us introduce a placeholder *semantic equivalence relation*, $E(\cdot, \cdot)$, which holds of any two sentences that mean the same thing—we operationalise this in Section 4. Recall that an equivalence

[1]Theories of meaning are contested (Speaks, 2021). However, for specific models and deployment contexts many considerations can be set aside. Care should be taken comparing very different models and contexts.