Addressing Uncertainty in LLMs: Leveraging Semantic Entropy for Predicting Conformal Sets

Ramneet Kaur[1], **Colin Samplawski**[1], **Adam D. Cobb**[1], **Anirban Roy**[1], **Brian Matejek**[1], **Manoj Acharya**[1], **Daniel Elenius**[1], **Alexander Michael Berenbeim**[2], **John A. Pavlik**[2], **Nathaniel D. Bastian**[2], **Susmit Jha**[1]

[1] Neurosymbolic Computing and Intelligence Group, Computer Science Laboratory, SRI International

[2] Army Cyber Institute, United States Military Academy, West Point

Abstract

In this paper, we introduce **ConformEnt**, which uses the entropy over a dynamic semantic clustering approach, based on a Chinese restaurant process, to quantify the uncertainty in the inference of large language models (LLMs). We show how this semantic entropy can be used as a nonconformity score for conformal prediction. We demonstrate the efficacy of our approach on two question-answering benchmarks, COQA and TriviaQA, using two LLMs, Llama-2-13b and Mistral-7b, achieving state-of-the-art results on uncertainty quantification using AUARC and AURAC as metrics. We also show that our conformal predictor generates smaller prediction sets for the same probabilistic guarantee of including correct response compared to a SOTA conformal prediction method.

1 Introduction

Large language models (LLMs) are being rapidly adopted in open-world settings where they answer questions posed to them. LLMs are known to exhibit confabulations (hallucinations) when answering questions wherein factual information could be incorrect or reasoning could be fallacious. The ability to quantify the uncertainty of the LLM on its response represents a tractable approach to model assurance. If the correctness of the LLM's response correlates with the computed uncertainty, the human user can rely on this uncertainty quantification (UQ) to determine when to trust the LLM response.

The challenge of UQ for LLMs in generative settings is different from that in regression or classification problems. While the later is well-studied (Guo et al., 2017; Jha et al., 2019; Xiao and Wang, 2019; Hu and Khan, 2021), UQ for generative models such as LLMs poses new challenges since the output is of free-form and can be of arbitrary length where the syntactic similarity of the generated token sequence does not align with semantic similarity. We build on recent observations of using embedding-based semantic similarity (Kuhn et al., 2023) to compare sampled outputs of LLMs and use semantic entropy as a measure of uncertainty of an LLM's prediction.

In this paper, we propose a novel approach towards trustworthy inference from LLMs that combines uncertainty quantification and conformal prediction. For uncertainty quantification, we extend existing approaches (Kuhn et al., 2023) by introducing a new dynamic semantic clustering algorithm based on a sequential distance dependent Chinese restaurant process (Blei and Frazier, 2011; Tuncer and Schulz, 2016). We then use this improved semantic entropy metric as the nonconformity score. Conformal prediction generates a set instead of a point prediction, where the output set is guaranteed to contain the correct label (or answer in our case) with a certain confidence level. The generative LLM models can generate free-form responses in question answering and so, the conformal prediction set in our approach is different from the usual setting of being a subset of known finite set of labels. We use semantic similarity to identify different responses when constructing the conformal set. This use of semantic entropy for conformal prediction by LLMs in question answering is the central novelty of our approach and we demonstrate its effectiveness empirically on multiple datasets (COQA and TriviaQA) and using different LLMs (Llama-2-13b: non-instruct model, and Mistral-7b: instruct model).

2 Related Work

Uncertainty quantification (UQ) for LLMs has received significant attention over the last few years. One approach is to explicitly query the model for the correctness probability (Kadavath et al., 2022). Another approach relies on utilizing the log-likelihood (Jiang et al., 2020) associated with

its generated response by taking a product or an average or other statistical aggregation over the generated tokens. LLMs are known to be not well-calibrated (Mielke et al., 2022) and consequently, methods for calibrating LLMs (Huang et al., 2024) have also been proposed. Semantic entropy or predictive uncertainty that measures the (in)consistency among multiple responses has been proposed as a metric for UQ of LLMs (Kuhn et al., 2023; Lin et al., 2023). We also use semantic entropy for UQ but adopt a novel semantic clustering approach and empirically demonstrate its effectiveness.

Conformal prediction (CP) (Balasubramanian et al., 2014) has been used for deploying deep learning models (Kaur et al., 2022; Haroush et al., 2021; Yang et al., 2024; Kaur et al., 2024) in high-assurance applications wherein the model predicts a set instead of a single prediction such that the one of the responses in the set is guaranteed to be correct with a probability higher than a given threshold. In the context of LLMs, conformal prediction has been used for providing coverage guarantees (Ye et al., 2024; Quach et al., 2023). Ye et al. (2024) concentrates on classification settings and propose non-conformity scores in the CP framework accordingly. In contrast, we focus on generative setting for LLMs in applications such as question-answering. Quach et al. (2023) propose generating diverse prediction sets based on the quality of individual responses, and a set scoring function. They utilize CP to derive parameters (λs) for diversity, quality, and set scoring function in their algorithm for coverage guarantees. We, instead propose, using semantic entropy as the non-conformity score in the CP framework for generating sets with coverage guarantees, and compare our results with Quach et al. (2023)'s approach.

3 ConformEnt

In this section, we introduce our new clustering approach for semantic entropy and describe how we then use it for building our conformal predictor.

3.1 Clustering by Semantic Equivalence

Kuhn et al. (2023) used semantic entropy, $SE(x) = -\sum_c p(c|x) \log p(c|x)$, where the probability of an equivalence class (corresponding to a cluster of embeddings), c, conditioned on the input query, x, is given by $p(c|x) = \sum_{\mathbf{s} \in c} p(\mathbf{s}|x)$ and $\mathbf{s} \in c$ denotes all the sentences (or responses) in an equivalence class for C equivalence classes. The probability of each sentence is given by the standard product of the token probabilities. The result of using the semantic entropy is to sharpen $p(c|x)$, when there are many responses with the same meaning, and therefore reduce the predicted entropy. To evaluate the semantic entropy, we need to define a function that checks semantic equivalence between sentences. In the original work, they choose the conservative approach of using Deberta-large model (He et al., 2020) to only define two sentences to be semantically equivalent if and only if entailment was classified in both directions. We empirically observe that irrelevant responses can sometimes get clustered together by this approach and we include these in Appendix A.2. Further, entailment in both directions is not necessary when one response includes additional information than another but they both contain the same relevant semantic information.

Thus, our approach computes the probability that two sentences, $\mathbf{s}_i$ and $\mathbf{s}_j$, are in the same equivalence class (that is, the same semantic cluster), by taking the maximum entailment score output by Deberta, $p_{ij} = \max(p(\mathbf{s}_i \vdash \mathbf{s}_j), (p(\mathbf{s}_j \vdash \mathbf{s}_i))$. This choice of taking a maximum can be viewed as a quantitative disjunction of entailment in either direction. We then use this approach to build a probability that a sentence, $\mathbf{s}_j$, belongs to an equivalence class, c, using the average probability across the cluster members:

$$p(\mathbf{s} \in c) = \frac{1}{|c|} \sum_{\mathbf{s}_i \in c} p_{ij} \tag{1}$$

This is also different from the existing approaches for semantic clustering where s is assigned to cluster c if it is entailed in both directions by only one member of c. We postulate that this could be the reason for semantically irrelevant responses being assigned to the same cluster because it is easier to incorrectly assign responses to the same cluster if we rely on only one member instead of all members in the cluster. In contrast, we take an average over all existing members of the semantic cluster making our assignment more robust.

A naive approach would be to greedily assign $\mathbf{s}$ to the $c \in C$ with the highest probability. However, this is not sufficient since we need a mechanism for forming new equivalence classes (or clusters). Given a set of responses, we need to decide when to form a new cluster and when to assign to an existing cluster that has highest likelihood $p(\mathbf{s} \in c)$.

We use the same mechanism as the Chinese restaurant process (CRP) for iterative clustering. We start a new cluster c^*, with $p(\mathbf{s} \in c^*) = \frac{\alpha}{\alpha+N}$, where N is the current number of clusters and $\alpha > 0$ is the rate parameter which is a prior over forming new clusters. Since the equivalence class assignment of a new sentence is related to the existing samples, we can think of our clustering algorithm as being a sequential distance dependent CRP (Blei and Frazier, 2011; Tuncer and Schulz, 2016). The final normalized probabilities are given by:

$$p(\mathbf{s} \in c_i) = \frac{p(\mathbf{s} \in c_i)}{p(\mathbf{s} \in c^*) + \sum_j^N p(\mathbf{s} \in c_j)}$$

$$p(\mathbf{s} \in c^*) = \frac{p(\mathbf{s} \in c^*)}{p(\mathbf{s} \in c^*) + \sum_j^N p(\mathbf{s} \in c_j)}.$$

Full clustering implementation is shown as Alg. 1 in Appendix A.1.

3.2 Conformal Prediction

For each cluster c, we have $p(c|x)$. We therefore use the negative log probability, $\log[1/p(c|x)]$, of the individual clusters, as the non-conformity score in conformal prediction (CP) framework (Balasubramanian et al., 2014), for generating prediction sets. Non-conformity scores of calibration datapoints is used to build a reference empirical distribution to compare against when building the prediction set. Specifically, depending on the desired significance level, ϵ, prediction set is generated by comparing scores for the test clusters with a threshold from the empirical distribution: non-conformity score of calibration set at $(1-\epsilon)^{th}$ quantile of the distribution. Intuitively, clusters with low negative log probability (high likelihood) are more likely to be included in prediction sets compared to clusters with high negative log probability. If an LLM outputs many semantically equivalent responses, then we expect the cluster's $\log[1/\sum_{\mathbf{s}\in c} p(\mathbf{s}|x)]$ to decrease due to the summation over the sentence probabilities by sharpening the cluster probability.

The use of CP for constructing the prediction sets gives us coverage guarantees on the true answer in the set with the probability greater than or equal to $1-\epsilon$ (Vovk et al., 2005). Proposed algorithm (Alg. 2) for generating prediction sets with coverage guarantees is in Appendix A.1.

4 Experimental Results

The experimental evaluation focuses on two research questions. **RQ1:** Does the novel semantic clustering approach inspired from CRP improve the UQ of LLMs? **RQ2:** How does the ConformEnt perform compared to the CP baseline?

Datasets and Models. We use two question-answer datasets: COQA (Reddy et al., 2019) and TriviaQA (Joshi et al., 2017), over which we compare the performance of two LLMs, Llama-2-13b: non-instruct model (Touvron et al., 2023), and Mistral-7b: instruct model (Jiang et al., 2023). Following existing literature (Lin et al., 2023; Kuhn et al., 2023), we deploy three evaluation methods: (1) We query GPT-4 (Achiam et al., 2023) by asking it to provide a rating on whether a response is correct with a value between 0 and 1, and label the response as correct if its rating > 0.7; (2) RougeL (Lin, 2004) score with a threshold > 0.3; (3) Deberta to check for entailment of correct answer in the response.

4.1 UQ Performance

We report Area Under Accuracy-Rejection Curve (AUARC) (Nadeem et al., 2009), and Area Under Rejection-Accuracy Curve(AURAC) for comparing our performance on UQ with Kuhn et al. (2023)'s, and Lin et al. (2023)'s with EigV as their UQ metric[1]. Details about EigV are included in Appendix A.4. While AUARC has been used as an evaluation metric previously (Lin et al., 2023), we include AURAC as a new metric. AUARC is an indicator of the accuracy of accepted (or highly certain) samples, and AURAC is an indicator of the accuracy of rejected (or highly uncertain) samples. In addition to AUARC, AURAC also indicates calibration of the UQ metric: we would like the accuracy of the model on the rejected samples by the UQ metric to be as low as possible, i.e. not rejecting samples on which LLMs are accurate.

The results are reported in Tables 1, and 2. Similar to Kuhn et al. (2023)'s, we also report our results with the UQ score unnormalized (Unnorm)/normalized (Norm) on the response's length. We outperform the baseline by Kuhn et al. (2023) in all the test cases, indicating that the proposed clustering approach performs better in UQ. We achieve competitive results in comparison to the current SOTA by Lin et al. (2023) by outperforming them in most cases. We also report AUROC, and compare with other baselines in Appendix A.3.

[1]They also propose 'Ecc', and 'Deg' as other UQ metrics. Consistent with their paper, we found that the best results in most of the cases are with 'EigV' metric, and therefore we compare our results with this metric.

		COQA Dataset				**TriviaQA** Dataset			
Model	**Eval.**	**Model Acc.**	**Sem. Ent.** Unnorm/Norm	**EigV**	**ConformEnt** Unnorm/Norm	**Model Acc.**	**Sem. Ent.** Unnorm/Norm	**EigV**	**ConformEnt** Unnorm/Norm
Llama-13b	GPT-4	73.22	85.81/86.44	**88.03**	86.35/87.47	67.03	88.13/87.94	**88.84**	88.33/88.54
Mistral-7b	GPT-4	73.38	81.91/82.68	82.82	82.22/**82.95**	60.68	80.99/81.40	**82.03**	81.23/**82.03**
Mean	GPT-4	73.30	83.86/84.56	**85.43**	84.29/85.21	63.86	84.56/84.67	**85.44**	84.78/85.29
Llama-13b	RougeL	72.75	86.03/87.05	87.92	86.84/**88.34**	64.60	85.62/85.19	85.76	85.86/**85.87**
Mistral-7b	RougeL	44.74	64.37/62.93	63.43	**64.60**/63.48	42.33	70.18/68.13	69.41	**70.26**/68.81
Mean	RougeL	58.75	75.20/74.99	75.65	75.72/**75.91**	53.47	77.90/76.66	77.59	**78.06**/77.34
Llama-13b	Deberta	63.74	80.21/79.48	**82.68**	81.04/81.37	63.33	84.92/84.34	**85.60**	85.23/85.13
Mistral-7b	Deberta	11.23	23.56/20.71	20.88	**23.53**/21.05	33.92	62.29/59.53	60.39	**62.37**/60.16
Mean	Deberta	37.49	51.89/50.10	51.78	**52.29**/51.21	48.63	73.61/71.94	73.00	**73.80**/72.65

Table 1: AUARC (↑) results in comparison to Kuhn et al. (2023)'s Semantic Entropy (Sem. Ent.) UQ metric, and SOTA EigV metric by Lin et al. (2023). Best results are in bold and second best are underlined.

		COQA Dataset				**TriviaQA** Dataset			
Model	**Eval.**	**Model Acc.**	**Sem. Ent.** Unnorm/Norm	**EigV**	**ConformEnt** Unnorm/Norm	**Model Acc.**	**Sem. Ent.** Unnorm/Norm	**EigV**	**ConformEnt** Unnorm/Norm
Llama-13b	GPT-4	73.22	58.97/56.90	**54.63**	58.42/55.32	67.03	40.09/40.27	39.42	39.92/**39.38**
Mistral-7b	GPT-4	73.38	63.06/62.02	**59.83**	62.77/61.41	60.68	35.57/35.04	**33.19**	35.13/33.29
Mean	GPT-4	73.30	61.02/59.46	**57.23**	60.60/58.37	63.86	37.83/37.66	**36.31**	37.53/36.34
Llama-13b	RougeL	72.75	56.75/55.53	53.78	55.78/**52.65**	64.60	39.12/39.39	38.93	38.81/**38.35**
Mistral-7b	RougeL	44.74	27.62/29.65	**27.12**	27.37/28.26	42.33	17.15/19.56	17.06	**16.95**/18.11
Mean	RougeL	58.75	42.19/42.59	**40.45**	41.58/40.46	53.47	28.14/29.48	28.00	**27.88**/28.23
Llama-13b	Deberta	63.74	46.07/46.91	**42.04**	45.07/43.56	63.33	37.23/37.94	**36.70**	36.88/36.84
Mistral-7b	Deberta	11.23	3.84/5.70	4.13	**3.82**/5.00	33.92	11.00/13.54	11.35	**10.89**/12.45
Mean	Deberta	37.49	24.96/26.31	**23.09**	24.45/24.28	48.63	24.12/25.74	24.03	**23.89**/24.65

Table 2: AURAC (↓) results in comparison to Kuhn et al. (2023)'s Semantic Entropy (Sem. Ent.) UQ metric, and SOTA EigV metric by Lin et al. (2023). Best results are in bold and second best are underlined.

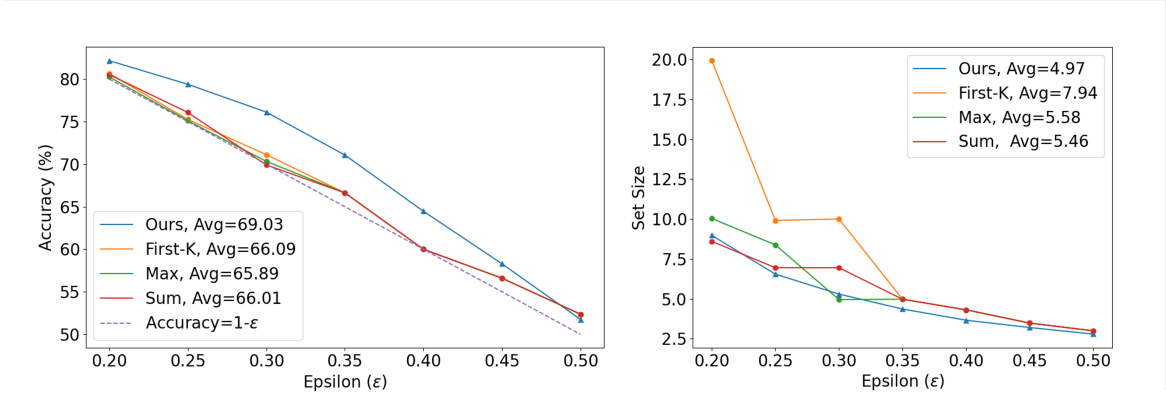

Figure 1: Comparison of Accuracy (left) and Set Size of prediction sets (right) with Conformal Prediction baseline (Quach et al., 2023).

4.2 Conformal Prediction Results

The desired properties of a prediction set is that the accuracy of the set should be as high as possible with a smaller set size. So, here we report accuracy and set size as the evaluation metrics (Fig. 1), and compare with the existing baseline (Quach et al., 2023) for using CP for generating prediction sets with coverage guarantees.

The coverage guarantee (or guarantee of the correct answer contained in the prediction set by CP) is expected to be $\geq (1 - \epsilon)$. So, as the value of ϵ increases, the accuracy and the set size is expected to decrease. This is what we observe for both approaches: ours and the baseline, with both the approaches satisfying the coverage guarantees. Quach et al. (2023) report results with different variations of their proposed algorithm (Algorithm 1 of their paper) in terms of the set scoring function ($\mathcal{F}$): First-K, Max, and Sum, and on TriviaQA with Llama-2-13b. We outperform these results for all the three variations on both evaluation metrics. ConformEnt's results on COQA with Llama-2-13b are similar and included in the Appendix A.5.

5 Limitations

Our proposed approach takes a step towards more accurate uncertainty quantification of LLMs and the use of conformal set prediction to provide guarantees on including the correct response. But practical deployment of these measures need to take into account human perception of uncertainty, and the downstream risk of using LLMs in any task. When to use the response and when to abstain is not just the function of uncertainty of the model but also the risk of wrong decision and the feasibility of abstaining. More research is needed on human-centric characterization of LLM uncertainty.

Acknowledgments

This material is based upon work supported by the United States Air Force and DARPA under Contract No. FA8750-23-C-0519. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Air Force and DARPA.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Anastasios N Angelopoulos and Stephen Bates. 2021. A gentle introduction to conformal prediction and distribution-free uncertainty quantification. *arXiv preprint arXiv:2107.07511*.

Vineeth Balasubramanian, Shen-Shyang Ho, and Vladimir Vovk. 2014. *Conformal prediction for reliable machine learning: theory, adaptations and applications*. Newnes.

David M Blei and Peter I Frazier. 2011. Distance dependent chinese restaurant processes. *Journal of Machine Learning Research*, 12(8).

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

Matan Haroush, Tzviel Frostig, Ruth Heller, and Daniel Soudry. 2021. A statistical framework for efficient out of distribution detection in deep neural networks. *arXiv preprint arXiv:2102.12967*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*.

Yibo Hu and Latifur Khan. 2021. Uncertainty-aware reliable text classification. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 628–636.

Yukun Huang, Yixin Liu, Raghuveer Thirukovalluru, Arman Cohan, and Bhuwan Dhingra. 2024. Calibrating long-form generations from large language models. *arXiv preprint arXiv:2402.06544*.

Susmit Jha, Sunny Raj, Steven Fernandes, Sumit K Jha, Somesh Jha, Brian Jalaian, Gunjan Verma, and Ananthram Swami. 2019. Attribution-based confidence metric for deep neural networks. *Advances in Neural Information Processing Systems*, 32.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Zhengbao Jiang, Frank F Xu, Jun Araki, and Graham Neubig. 2020. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *arXiv preprint arXiv:1705.03551*.

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

Ramneet Kaur, Susmit Jha, Anirban Roy, Sangdon Park, Edgar Dobriban, Oleg Sokolsky, and Insup Lee. 2022. iDECODe: In-distribution equivariance for conformal out-of-distribution detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 7104–7114.

Ramneet Kaur, Yahan Yang, Oleg Sokolsky, and Insup Lee. 2024. Out-of-distribution detection in dependent data for cyber-physical systems with conformal guarantees.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. *arXiv preprint arXiv:2302.09664*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Zhen Lin, Shubhendu Trivedi, and Jimeng Sun. 2023. Generating with confidence: Uncertainty quantification for black-box large language models. *arXiv preprint arXiv:2305.19187*.

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

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

Victor Quach, Adam Fisch, Tal Schuster, Adam Yala, Jae Ho Sohn, Tommi S Jaakkola, and Regina Barzilay. 2023. Conformal language modeling. *arXiv preprint arXiv:2306.10193*.

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

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Mehmet Ali Çağrı Tuncer and Dirk Schulz. 2016. Sequential distance dependent chinese restaurant processes for motion segmentation of 3d lidar data. In *2016 19th International Conference on Information Fusion (FUSION)*, pages 758–765. IEEE.

Vladimir Vovk, Alex Gammerman, and Glenn Shafer. 2005. *Algorithmic learning in a random world*. Springer Science & Business Media.

Yijun Xiao and William Yang Wang. 2019. Quantifying uncertainties in natural language processing tasks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 7322–7329.

Yahan Yang, Rameet Kaur, Souradeep Dutta, and Insup Lee. 2024. Memory-based distribution shit detection for learning enabled cyber-physical systems with statistical guarantees.

Fanghua Ye, Mingming Yang, Jianhui Pang, Longyue Wang, Derek F Wong, Emine Yilmaz, Shuming Shi, and Zhaopeng Tu. 2024. Benchmarking llms via uncertainty quantification. *arXiv preprint arXiv:2401.12794*.

.

A Appendix

A.1 Algorithms

Here, we provide details about both the proposed algorithms: Alg. 1 on ConformEnt Clustering inspired by Chinese restaurant process (CRP) for iterative clustering, and Alg. 2 on generating ConformEnt Prediction Sets with coverage guarantees.

Alg. 1 in a sequential fashion, starting with an empty set of clusters. After the first generation (or response from the LLM), we compute the score for forming a new cluster as: $\frac{\alpha}{\alpha+0} = 1$. This results in a new cluster probability of 1, leading a new cluster to be formed deterministically.

In subsequent rounds we compute the per cluster assignment scores for the new generation (Equation (1)). We then compute the normalized probabilities as described in Section 3.1.

In this paper, we used a rate parameter of $\alpha = 0.5$. We performed a grid search over $\alpha \in [0.2, 0.3, \ldots, 0.7]$ and observed little variance in performance. We set the value of number of generations N is set to 20, which is consistent with the existing work (Kuhn et al., 2023; Lin et al., 2023; Quach et al., 2023).

Algorithm 1 ConformEnt Clustering

Input: query x, LLM model M
Parameter: rate parameter $\alpha > 0$, number of generations N
Output: clusters C
Initialize: $C \leftarrow \emptyset$
for $i = 0$ to N **do**
  $\mathbf{s}_i = M(x)$ {generate with LLM}
  $scores \leftarrow \mathbf{0}_{|C|+1}$
  for c_j in C **do**
    $scores[j] \leftarrow P(\mathbf{s}_i \in c_j)$
  end for
  $scores[-1] \leftarrow \frac{\alpha}{\alpha+|C|}$
  $probs \leftarrow softmax(scores)$
  $k \leftarrow argmax(probs)$ {cluster assignment}

  if $k == |C|$ **then**
    $C \leftarrow C \cup \{\mathbf{s}_i\}$ {new cluster}
  else
    $C_k \leftarrow C_k \cup \{\mathbf{s}_i\}$
  end if
end for

return C

Alg. 2 shows how we build prediction sets. For an input query x (from test or calibration set), clusters are generated via Alg. 1. We use negative log probability (nlp = $\log[1/p(c|x)]$) of each generated cluster (c) for x as the non-conformity score. For the desired significance level $\epsilon \in (0, 1)$, the prediction threshold τ is decided as the score at $(1-\epsilon)^{th}$ quantile of the empirical distribution of non-conformity scores for the calibration clusters. Assuming all generations are semantically equivalent in a cluster, a single generation from the test cluster (c) is added to the prediction set if its non-conformity score (nlp((c)) is below the prediction threshold τ.

In all our experiments, we report results with the first generation from all test clusters.

Algorithm 2 ConformEnt Prediction Set

Input: query x, LLM model M, prediction threshold τ from $(1-\epsilon)^{th}$ quantile of the empirical distribution of calibration set non-conformity scores,
Output: Output Set O with predictions on x by M s.t. $Pr.$(correct answer $\in O$) $\geq 1-\epsilon$
C = set of clusters from ConformEnt Clustering(x, M)
S = set of non-conformity scores for the generated clusters: $\{\forall c \in C : \text{nlp}(c)\}$
$\mathcal{O}$ = {a generation from $C[i]$ s.t. $S[i] \leq \tau$: $i = 1, \ldots, |C|$}
return O

A.2 Qualitative comparison of Clustering Approaches

Our clustering approach makes use of semantic entropy. Therefore we provide an example from both COQA and TriviaQA datasets to analyse how our new clustering approach compares to the original approach by Kuhn et al. (2023). We look at the quality of clusters formed by both approaches.

A.2.1 An example from COQA Dataset

Story: CHAPTER XXXIV Arthur remained at the gate while Ruth climbed Maria's front steps. She heard the rapid click of the type-writer, and when Martin let her in, found him on the last page of a manuscript. She had come to make certain whether or not he would be at their table for Thanksgiving dinner; but before she could broach the subject Martin plunged into the one with which he was full. "Here, let me read you this," he cried, separating the carbon copies and running the pages

of manuscript into shape. “It’s my latest, and different from anything I’ve done. It is so altogether different that I am almost afraid of it, and yet I’ve a sneaking idea it is good. You be judge. It’s an Hawaiian story. I’ve called it ‘Wiki-wiki’.” His face was bright with the creative glow, though she shivered in the cold room and had been struck by the coldness of his hands at greeting. She listened closely while he read, and though he from time to time had seen only disapprobation in her face, at the close he asked:- “Frankly, what do you think of it?” “I–I don’t know,” she, answered. “Will it–do you think it will sell?” “I’m afraid not,” was the confession. “It’s too strong for the magazines. But it’s true, on my word it’s true.” “But why do you persist in writing such things when you know they won’t sell?” she went on inexorably. “The reason for your writing is to make a living, isn’t it?”
Question: ‘Did he answer her?’
Answer: ‘No’
Generated Responses from Llama-13b: [‘Yes’, ‘Yes’, “He didn’t”, ‘He did, only not directly’, ‘No’, ‘No’, ‘No’, ‘He asked her what she thought’, ‘He told her his latest story’, ‘Yes’, ‘No’, ‘A sneaking yes’, ‘He ran the manuscript up to Miss Lawton’, ‘No’, ‘In the affirmative’, ‘Yes’, ‘No’, ‘No’, ‘Yes’, ‘Yes’]

Results: Figures 2, and 3 show the clusters formed by Kuhn et al. (2023), and our approach respectively. For brevity, we include only unique generations in a cluster. As it can be seen, Kuhn et al. (2023) approach puts semantically different generations in the same cluster (generations 3, 4, and 6 in cluster 1 for ‘Yes’), whereas ours separate them out in different clusters.

A.2.2 An example from TriviaQA Dataset

Question: What is ’The Old Lady of Threadneedle Street’?
Answer: Bank of England
Generated Responses from Llama-13b: [‘Bank of England’, ‘Bank of England’, ‘Bank of England’, ‘The Bank of England’, ‘A nickname; what was it really?’, ‘Bank of England’, ‘The Bank of England’, ‘The Bank of England’, ‘The Bank of England’, ‘The Bank of England’, ‘The Bank of England’, ‘Bank of England’, ‘The Bank of England’, ‘Bank of England’, ‘The Bank of England’, ‘Bank of England’, ‘Bank of England’, ‘Bank of England’, ‘The Bank of England’, ‘Bank Of England’]
Results: Figures 2, and 3 show the clusters formed by Kuhn et al. (2023), and our approach respectively. Again for brevity, we include only unique generations in a cluster. As it can be seen, Kuhn et al. (2023) approach puts semantically different generations in the same cluster (generation 3 in cluster 1 for ‘Bank of England’), whereas ours separate them out in different clusters.

A.3 All UQ results

Here, we include all results on UQ performance from Section 4.1: comparison with additional baselines on AUARC, AURAC , and AUROC evaluation metrics. In addition to Kuhn et al. (2023)’s Sem. Ent. (Unnorm/Norm), and (Lin et al., 2023)’s EigV results reported in the main paper, we include “Numset”, “LexiSim”, and “SelfProb” baselines here. Numset uses the number of semantic sets as the UQ metric, and has been previously used in (Lin et al., 2023) as one of the baselines. Higher the numset, more uncertain is the LLM on the input query. LexiSim uses the average of RougeL distance between every pair of generations for UQ. Here, higher the Lexisim, lower is the uncertainty. Again, Lexisim has been used as a baseline by Kuhn et al. (2023), and Lin et al. (2023). SelfProb (Kadavath et al., 2022) estimates if the probability of a model’s generation is correct by asking the model itself, and use that as the UQ metric. We follow the same prompt format as Lin et al. (2023) for asking the model about the probability, and report average over all generations. Here, higher the SelfProb, lower is the uncertainty.

Tables 3, 5, and 7 are AUARC, AUROC, and AURAC are results on COQA. And Tables 4, 6, and 8 are AUARC, AUROC, and AURAC are results on TriviaQA. Again, we report all results from the three GT evaluation methods: GPT-4, RougeL, and Deberta. We achieve either the best or second best in all but two test cases.

Figure 2: Clusters generated by Kuhn et al. (2023)'s approach on COQA example.

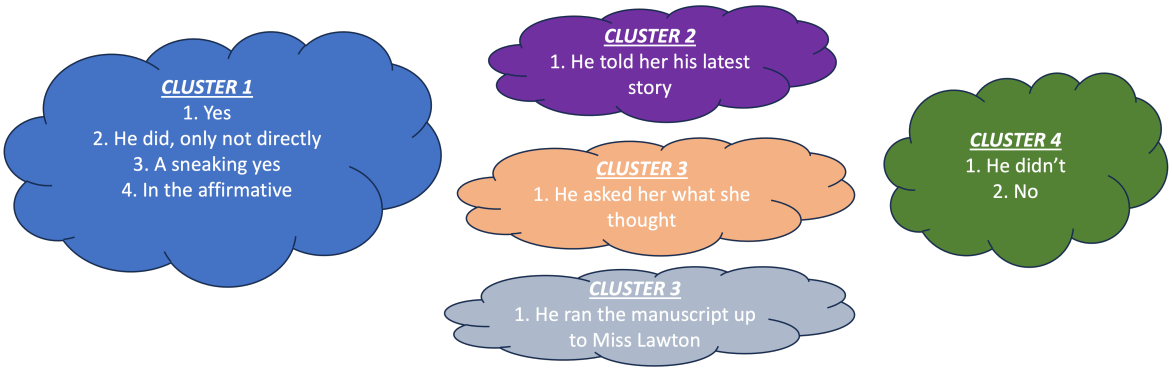

Figure 3: Clusters generated by ConformEnt Clustering (our) approach on COQA example.

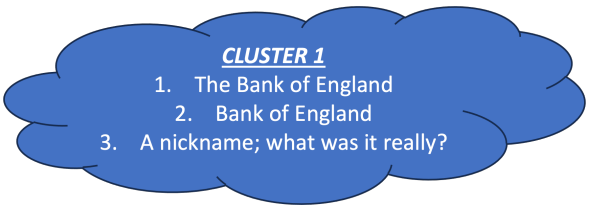

Figure 4: Clusters generated by Kuhn et al. (2023)'s approach on TriviaQA example.

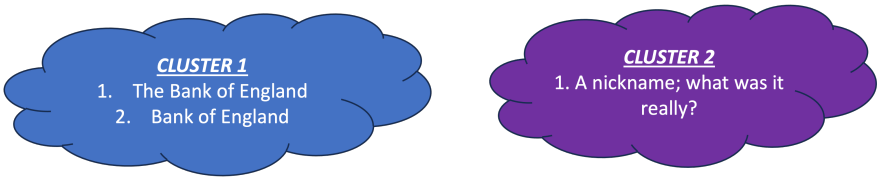

Figure 5: Clusters generated by ConformEnt Clustering (our) approach on TriviaQA example.

Model	GT	Model Acc	Sem. Ent. Unnorm/Norm	NumSet	LexiSim	SelfProb	EigV	ConformEnt Unnorm/Norm
Llama-13b	GPT-4	73.22	85.81/86.44	79.78	86.14	75.50	**88.03**	86.35/87.47
Mistral-7b	GPT-4	73.38	81.91/82.68	75.63	81.73	**85.14**	82.82	82.22/82.95
Mean	GPT-4	73.30	83.86/84.56	77.71	83.94	80.32	**85.43**	84.29/85.21
Llama-13b	RougeL	72.75	86.03/87.05	77.79	88.17	73.35	87.92	86.84/**88.34**
Mistral-7b	RougeL	44.74	64.37/62.93	46.99	59.61	52.64	63.43	**64.60**/63.48
Mean	RougeL	58.75	75.20/74.99	62.39	73.89	63.00	75.65	75.72/**75.91**
Llama-13b	Deberta	63.74	80.21/79.48	69.36	79.02	65.23	**82.68**	81.04/81.37
Mistral-7b	Deberta	11.23	23.56/20.71	11.63	16.70	12.21	20.88	**23.53**/21.05
Mean	Deberta	37.49	51.89/50.10	40.50	47.86	38.72	51.78	**52.29**/51.21

Table 3: AUARC (↑) results in comparison to all baselines on **COQA**. Best results are in bold and second best are underlined.

Model	GT	Model Acc	Sem. Ent. Unnorm/Norm	NumSet	LexiSim	SelfProb	EigV	ConformEnt Unnorm/Norm
Llama-13b	GPT-4	67.03	88.13/87.94	83.84	84.52	73.09	**88.84**	88.33/88.54
Mistral-7b	GPT-4	60.68	80.99/81.40	74.72	76.65	**84.46**	82.03	81.23/82.03
Mean	GPT-4	63.86	84.56/84.67	79.28	80.59	78.78	**85.44**	84.78/85.29
Llama-13b	RougeL	64.60	85.62/85.19	79.75	84.01	70.34	85.76	85.86/**85.87**
Mistral-7b	RougeL	42.33	70.18/68.13	54.53	61.72	62.03	69.41	**70.26**/68.81
Mean	RougeL	53.47	77.90/76.66	67.14	72.87	66.19	77.59	**78.06**/77.34
Llama-13b	Deberta	63.33	84.92/84.34	79.11	80.01	68.04	**85.60**	85.23/85.13
Mistral-7b	Deberta	33.92	62.29/59.53	44.88	51.80	50.33	60.39	**62.37**/60.16
Mean	Deberta	48.63	73.61/71.94	62.00	65.91	59.19	73.00	**73.80**/72.65

Table 4: AUARC (↑) results in comparison to all baselines on **TriviaQA**. Best results are in bold and second best are underlined.

Model	GT	Model Acc	Sem. Ent. Unnorm/Norm	NumSet	LexiSim	SelfProb	EigV	ConformEnt Unnorm/Norm
Llama-13b	GPT-4	73.22	85.19/88.69	73.06	82.63	53.74	**92.83**	87.87/91.90
Mistral-7b	GPT-4	73.38	79.91/81.99	58.00	76.57	79.46	82.77	81.48/**84.07**
Mean	GPT-4	73.3	82.55/85.34	65.53	79.6	66.6	87.80	84.68/**87.99**
Llama-13b	RougeL	72.75	80.87/86.48	68.24	**92.33**	48.40	88.45	83.90/90.28
Mistral-7b	RougeL	44.74	83.52/83.97	55.43	83.15	70.62	**86.84**	84.63/85.69
Mean	RougeL	58.75	82.20/85.23	61.84	87.74	59.51	87.65	84.27/**87.99**
Llama-13b	Deberta	63.74	85.59/88.80	69.55	87.45	48.92	**93.09**	88.38/92.02
Mistral-7b	Deberta	11.23	93.98/91.22	54.62	91.57	59.48	93.85	**94.25**/92.07
Mean	Deberta	37.485	89.79/90.01	62.09	89.51	54.2	**93.47**	91.32/92.05

Table 5: AUROC (↑) results in comparison to all baselines on **COQA**. Best results are in bold and second best are underlined.

Model	GT	Model Acc	Sem. Ent. Unnorm/Norm	NumSet	LexiSim	SelfProb	EigV	ConformEnt Unnorm/Norm
Llama-13b	GPT-4	67.03	94.74/96.85	92.04	86.64	59.38	**97.48**	95.29/97.39
Mistral-7b	GPT-4	60.68	90.58/93.59	80.93	81.00	**94.75**	93.66	91.48/94.24
Mean	GPT-4	63.86	92.66/95.22	86.49	83.82	77.07	95.57	93.39/**95.82**
Llama-13b	RougeL	64.60	92.63/94.50	88.59	97.01	59.84	**95.25**	93.27/95.23
Mistral-7b	RougeL	42.33	95.26/94.80	74.87	94.66	84.67	**96.34**	95.61/95.36
Mean	RougeL	53.47	93.95/94.65	81.73	**95.84**	72.26	95.80	94.44/95.30
Llama-13b	Deberta	63.33	92.61/94.34	87.73	86.58	55.96	**96.55**	93.49/95.30
Mistral-7b	Deberta	33.92	96.55/94.90	73.84	93.06	82.01	96.64	**96.73**/95.30
Mean	Deberta	48.63	94.58/94.62	80.79	89.82	68.99	**96.60**	95.11/95.30

Table 6: AUROC (↑) results in comparison to all baselines on **TriviaQA**. Best results are in bold and second best are underlined.

Model	GT	Model Acc	Sem. Ent. Unnorm/Norm	NumSet	LexiSim	SelfProb	EigV	ConformEnt Unnorm/Norm
Llama-13b	GPT-4	73.22	58.97/56.90	63.45	61.35	72.29	**54.63**	58.42/55.32
Mistral-7b	GPT-4	73.38	63.06/62.02	67.19	62.21	61.18	**59.83**	62.77/61.41
Mean	GPT-4	73.30	61.02/59.46	65.32	61.78	66.74	**57.23**	60.60/58.37
Llama-13b	RougeL	72.75	56.75/55.53	65.30	58.51	73.12	53.78	55.78/**52.65**
Mistral-7b	RougeL	44.74	27.62/29.65	40.13	34.11	33.56	**27.12**	27.37/28.26
Mean	RougeL	58.75	42.19/42.59	52.72	46.31	53.34	**40.45**	41.58/40.46
Llama-13b	Deberta	63.74	46.07/46.91	55.50	53.07	63.32	**42.04**	45.07/43.56
Mistral-7b	Deberta	11.23	3.84/5.70	9.84	11.83	9.45	4.13	**3.82**/5.00
Mean	Deberta	37.49	24.96/26.31	32.67	32.45	36.39	**23.09**	24.45/24.28

Table 7: AURAC (↓) results in comparison to all baselines on **COQA**. Best results are in bold and second best are underlined.

Model	GT	Model Acc	Sem. Ent. Unnorm/Norm	NumSet	LexiSim	SelfProb	EigV	ConformEnt Unnorm/Norm
Llama-13b	GPT-4	67.03	40.09/40.27	43.03	54.28	60.31	39.42	39.92/**39.38**
Mistral-7b	GPT-4	60.68	35.57/35.04	40.02	47.45	33.71	**33.19**	35.13/33.29
Mean	GPT-4	63.86	37.83/37.66	41.525	50.87	47.01	**36.31**	37.53/36.34
Llama-13b	RougeL	64.60	39.12/39.39	42.74	50.56	58.41	38.93	38.81/**38.35**
Mistral-7b	RougeL	42.33	17.15/19.56	25.09	32.12	21.46	17.06	**16.95**/18.11
Mean	RougeL	53.47	28.14/29.48	33.915	41.34	39.94	28.00	**27.88**/28.23
Llama-13b	Deberta	63.33	37.23/37.94	40.09	53.01	58.23	**36.70**	36.88/36.84
Mistral-7b	Deberta	33.92	11.00/13.54	18.51	27.93	15.97	11.35	**10.89**/12.45
Mean	Deberta	48.63	24.12/25.74	29.3	40.47	37.1	24.03	**23.89**/24.65

Table 8: AURAC (↓) results in comparison to all baselines on **TriviaQA**. Best results are in bold and second best are underlined.

A.4 Details about Lin et al. (2023)'s UQ Approach

The approach is based on a graph generated from multiple responses. Different responses are the nodes of this graph and an edge between every pair of node is assigned a weight based on the semantic similarity between the pair. Eigen value decomposition is performed on the symmetric Laplacian of the graph, and 'EigV' is the sum of the K^2 eigen values that are less than 1. They use different ways of computing semantic similarity for edge weights between every pair of graph. These are Jaccard similarity score, entailment and contradiction scores by Deberta. Consistent with their paper, we also observe that the best results are with the entailment score by Deberta. So, we report results with this score in the paper.

A.5 Comformal Prediction Results

In addition to the conformal prediction results on TriviaQA reported in main paper, we also report these results on COQA for generations from Llama-2-13b. Figure 6 shows these results for both accuracy and set size and with all the three GT evaluation approaches: GPT-4, RougeL, and Deberta.

Here, we also report the point accuracy, which is the average accuracy of the individual $N = 20$ generations. For $\epsilon \leq 0.35$, the prediction set accuracy is always higher than the point accuracy. Consistent with the results on Triviqa, here also the value of accuracy and set size decreases with the increase in the value of ϵ. GPT and RougeL evaluations satisfy the coverage guarantee $\forall \epsilon \geq 0.15$. Even though conformal prediction provides a rigorous theoretical guarantee, deviations from the coverage guarantee can occur in practice due to limited sample variability (Angelopoulos and Bates, 2021). This justifies the accuracy results with Deberta Evaluation and the other two evaluations with $\epsilon < 0.15$.

[2]They used $K = 20$ in the paper.

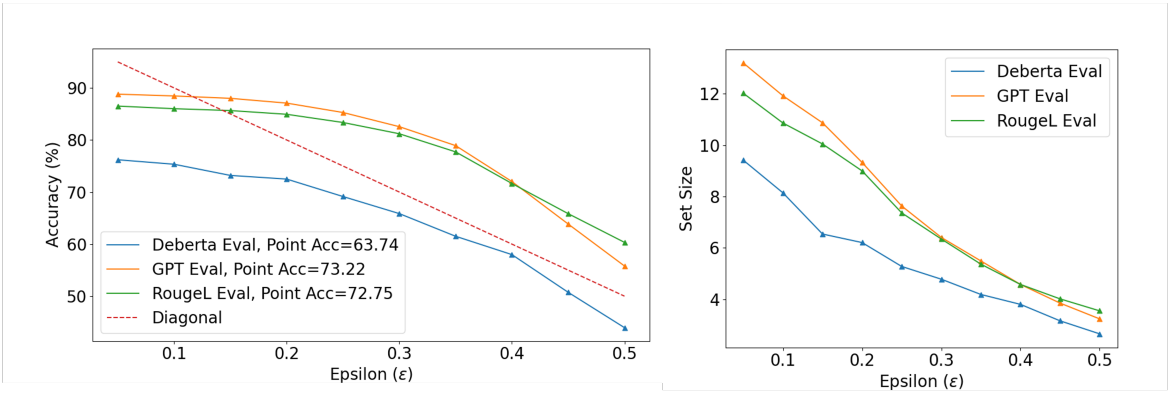

Figure 6: ConformEnt's Accuracy (left) and Set Size (right) evaluation on COQA for Llama-13b.