Evaluation Pitfalls and Sparsity Limitations in LLM-based Confidence Estimates for Classification

Elena Merdjanovska[*]
Humboldt-Universität zu Berlin
Science of Intelligence
elena.merdjanovska@hu-berlin.de

Omar Zaidan
Amazon
ozaidan@amazon.de

Andreas Rücklé
Amazon
arueckle@amazon.de

Abstract

Confidence estimation is essential when LLMs are used for classification, indicating when predictions can be trusted. However, common approaches such as verbalization produce extremely sparse outputs. For instance, Qwen3-32B verbalizes only eight unique confidence values on SST-2, with over half being exactly 95%—a pattern we observe consistently across four datasets and two LLMs. Besides limiting practical utility, we show that this sparsity critically affects evaluation: the choice of interpolation in area under the accuracy-rejection curve (AUARC) dramatically alters rankings, with consistency sampling dropping from best to worst under stepwise versus linear interpolation. We advocate for standardizing stepwise interpolation for a fairer comparison. Under such a fair evaluation, we find that weighting verbalized digits by token probabilities—a method we term *verbalization logprobs*—addresses sparsity and achieves the best AUARC (+2.3 points over vanilla verbalization) without incurring additional inference cost.

1 Introduction

LLMs are increasingly used for classification tasks, e.g., automatic evaluation (Zheng et al., 2023; Lee et al., 2025), content moderation (Yin et al., 2025; Nghiem et al., 2025), and more (Marvin Imperial and Tayyar Madabushi, 2025; Sun et al., 2025). Many applications require confidence estimates, e.g., in selective prediction, where a model can *reject* to classify examples with low confidence (Chen et al., 2023; Ren et al., 2023). Such classifiers are evaluated only on the subset of examples that exceed a specific confidence threshold, with the rest left unclassified, offering control over the risk-coverage trade-off (El-Yaniv and Wiener, 2010).

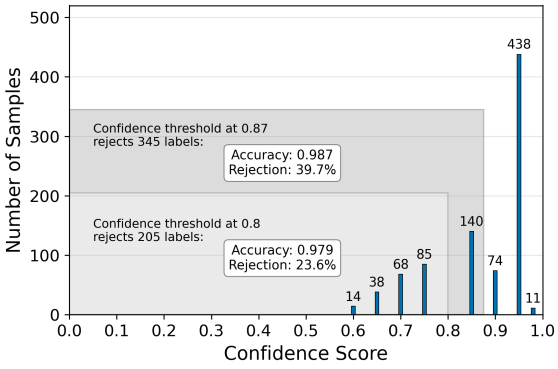
Prompting poses unique challenges for confidence estimation, as it lacks a direct mapping from outputs to class probabilities. Novel approaches such as verbalization can prompt models to express confidence for class labels in natural language (Xuan et al., 2025; Liu et al., 2025a; Zeng et al., 2025), while sampling-based approaches generate multiple predictions at a higher temperature and aggregate them (Phillips et al., 2025; Nikitin et al., 2024). These approaches differ fundamentally from traditional classifiers due to their natural language interface. For instance, the strong preference of LLMs towards generating certain numerical tokens (Coronado-Blázquez, 2025; Shao et al., 2025) can bias confidence estimates, leading to LLM overconfidence (Xiong et al., 2024). Although prior work has focused on calibration (Tonolini et al., 2024; Wang et al., 2024), we study *sparsity* as a distinct and critical challenge.

First, we demonstrate that verbalization-based approaches produce extremely sparse confidences. For instance, Qwen3-32B (Yang et al., 2025) predicts only eight unique confidence values (0.6, 0.65, 0.7, 0.75, 0.85, 0.9, 0.95, and 0.98) across SST-2 (Socher et al., 2013). This means we have *little choice when selecting decision thresholds*, which limits the practical utility of such approaches.

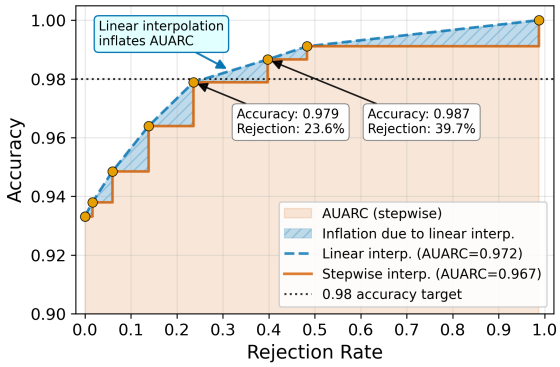
Second, we show that sparsity has *critical implications for evaluation*. Using the selective classification metric AUARC (Nadeem et al., 2009)—the area under the accuracy-rejection curve—we find that a simple and often neglected aspect of metric computation, namely the interpolation method, in fact has a major impact on system rankings. The performance of verbalization and sampling approaches decrease by 0.7–4.5 points under stepwise interpolation (vs. linear interpolation), while the continuously-distributed token logprobs approach remains unaffected. Most strikingly, the best approach using linear interpolation in our experiments becomes the worst using stepwise interpolation. Given the inconsistent usage of interpolation methods in the literature, we advocate for standard-

[*]Work done during an internship at Amazon.

Findings of the Association for Computational Linguistics: ACL 2026, pages 33424–33435



(a) Histogram showing all confidence scores on SST-2



(b) Accuracy-rejection curve on SST-2[1]

Figure 1: **Sparse confidences limit practical utility.** (a) Histogram of vanilla verbalized confidences on SST-2 with Qwen3-32B, showing extreme sparsity with only eight unique values (0.6, 0.65, 0.7, 0.75, 0.85, 0.9, 0.95, 0.98). The shaded regions highlight two rejection strategies: rejecting 205 samples (lighter gray) achieves 97.9% accuracy, while rejecting 345 samples (darker gray) achieves 98.7% accuracy. (b) Accuracy-rejection curve demonstrating limited threshold choices due to sparsity. Reaching at least 98% accuracy requires rejecting 39.7% of predictions vs. rejecting 23.6% for 97.9% accuracy (16.1pp increase). Stepwise interpolation (solid orange) is correct for sparse confidences; linear interpolation (dashed blue) incorrectly assumes performance increases gradually between thresholds, creating artificial inflation (blue shaded area).

Verbalization-based	
Vanilla verbalization	Direct self-reported confidence
Top-k verbalization	k guesses with probabilities
Sampling-based	
Consistency sampling	Frequency of class predictions
Verbalization sampling	Mean of verbalized confidences
Logit-based	
Token logprobs	Probability of class label token

Table 1: Overview of the five confidence estimation methods studied in this work.

izing stepwise interpolation for comparable and fair evaluations.

Third, we address sparsity directly: we propose *verbalization logprobs*, which weighs verbalized confidence digits by their token probabilities, transforming sparse outputs into continuous distributions. Across four datasets, verbalization logprobs achieves +2.3 AUARC points over vanilla verbalization at no additional computational cost.

The remainder of this paper reviews existing approaches (§2), analyzes the sparsity limitation (§3), and proposes verbalization logprobs (§4).

[1]All accuracy-rejection curves have an accuracy of 1.0 at rejection rate 1.0 by convention (Nadeem et al., 2009).

2 Background: Confidence Estimation for LLM Classifiers

We distinguish three types of confidence estimation approaches for classification tasks, (1) verbalization-based, (2) sampling-based, and (3) logit-based.[2] Additionally, several white-box approaches (Vashurin et al., 2025; Vazhentsev et al., 2025) leverage internal model states for confidence estimation, which we do not study in this work. While this presents an interesting line of work, such approaches require access beyond what most widely-used LLM APIs provide. Table 1 shows the approaches we study in §3—due to their wide adoption and good reported performances—with their characteristics highlighted below.

Verbalization-based methods. The simplest approach is prompting the model to output self-reported confidence, termed *vanilla verbalization* (Xiong et al., 2024). One may use different verbalized number ranges, but typically we ask the LLM to produce a confidence from 0 to 100. We can also use prompting techniques such as CoT (Kojima et al., 2022), self-probing (Xiong et al., 2024) and top-k (Tian et al., 2023). For instance, *top-2 verbalization* prompts the model to produce the *two* most likely predictions, each with an associated confidence value, even though only

[2]Note that further sub-categorization exists for generative tasks. See recent surveys by Geng et al. (2024) and Liu et al. (2025b).

the highest-confidence prediction is used. This outperforms other verbalization approaches (Xiong et al., 2024). Other interesting extensions are possible, e.g., task-specific rubrics (Kim et al., 2024), but those require task-specific prompt changes. In contrast, we focus on methods that generalize across classification tasks without requiring knowledge about the task.

Sampling-based methods. Another popular approach to confidence estimation involves sampling multiple LLM responses and aggregating their outputs. *Consistency sampling* generates multiple class predictions and considers their frequency as a confidence score. Typically, performance saturates quickly (Duan et al., 2024; Lin et al., 2024b), and therefore we sample four responses in our experiments. *Verbalization sampling* extends this by computing the mean of sampled verbalized confidences (Xiong et al., 2024). For question-answering tasks, more sophisticated similarity-based approaches perform well, including semantic entropy (Kuhn et al., 2023) and eccentricity (Lin et al., 2024b). However, those are not applicable to simply predicting class labels. One downside of sampling-based approaches is that they substantially increase the number of output tokens, especially when producing reasoning traces or explanations as part of the prediction.

Logit-based methods. As verbalization- and sampling-based methods make no use of model internals, they are black-box methods that only operate on the output token sequence. In contrast, logit-based methods occupy an interesting space between black-box and white-box. Some APIs do allow access to a limited number of log-probabilities, which we can use to estimate confidence. *Token logprobs* uses the probability of the class label token as a confidence estimate (Xiong et al., 2024). Other common methods are P(True) (Kadavath et al., 2022) or relying on entropy of the generated sequence in open-ended generation (Huang et al., 2025). While these methods are closest to traditional confidence estimates by directly accessing prediction probabilities, they cannot be used with APIs that don't return logprobs (e.g., OpenAI, Claude, Bedrock). This often leaves us with only verbalization or sampling approaches being applicable in practice.

3 The Sparsity Limitation

We now study how sparsity affects approaches from the categories outlined before. We find substantial variation and critical implications for evaluation: a methodological choice, namely interpolation, can completely reverse performance rankings.

Verbalized confidences are sparse. LLMs tend to prefer predicting certain number tokens more frequently than others (Coronado-Blázquez, 2025; Shao et al., 2025). In Figure 1a, we show this phenomenon also holds true for confidence estimation with verbalization approaches. Qwen3-32B (Yang et al., 2025) with vanilla verbalization predicts just eight unique confidence values over the entire SST-2 dataset, with *more than half* of the confidence values exactly equal to 95%. We find that this is a relatively broad phenomenon; when running different approaches on four classification datasets[3]—namely SST-2, SST-5, Amazon ESCI product classification (Reddy et al., 2022), Yahoo! answers topic classification (Zhang et al., 2015)—we find that 45–93% of confidences are concentrated in the five most common confidence values (see %t5 in Table 2). We observe the same behavior using Claude 3.7 Sonnet, and both models with or without reasoning enabled (see Appendices B and D). We also verify that these findings hold for an alternative confidence range (0–9) in Appendix E. This limits practical utility, as illustrated in Figure 1: achieving small accuracy gains often requires us to accept large increases in rejection rate.

Evaluations need to account for sparsity. Many metrics rely on measuring performance at every possible confidence threshold and integrate the area under the curve. Examples are AUROC (Hanley and McNeil, 1982), PRR (Vashurin et al., 2025), AUPRC (Ling et al., 2024), and AUARC (Nadeem et al., 2009). AUARC is closely related to use-cases where we reject low-confidence predictions (e.g., El-Yaniv and Wiener, 2010) as it computes the area under the accuracy-rejection curve. For sparse confidence estimation approaches—where only a few thresholds produce distinct outcomes—the choice of which interpolation method to use is critical, as we show in Figure 1b.

However, we find that this choice is usually overlooked, leading to inconsistencies across evaluations due to differing interpolation methods. While

[3] See Appendix A for dataset details.

	AUARC			%t5
	linear	ranks	step	
Consistency sampling	0.808	1→5	0.683	100
Vanilla verbalization	0.758	2→4	0.713	92
Top-2 verbalization	0.757	3→2	0.731	93
Verbalization sampling	0.741	4→1	0.734	45
Token logprobs	0.723	5→3	0.723	0

Table 2: Impact of interpolation method on AUARC scores and rankings for Qwen3-32B. Scores are averages over SST-2, SST-5, Amazon ESCI, and Yahoo! topic classification (see Table 5 for dataset details). %t5 refers to the concentration in top-5 confidence values.

the default in AUROC and AUPRC is stepwise interpolation (Muschelli III, 2020; Chen et al., 2024), AUARC is often computed using linear (aka trapezoidal) interpolation.[4] While this has little impact on evaluations of methods with many distinct thresholds—i.e., if there is only a small distance between subsequent points—it becomes critical when dealing with sparse confidence scores. The correct way is to use stepwise interpolation, as seen in Figure 1b.

AUARC interpolation considerably impacts evaluation ranks. Table 2 shows the impact of using stepwise rather than linear interpolation when calculating AUARC for Qwen3-32B (see Appendix B for Claude 3.7 Sonnet results). Sparse approaches show consistent and sometimes substantial performance drops, while token logprobs—which does not suffer from sparsity—remains unchanged. Specifically, verbalization approaches score 0.7–4.5 AUARC points lower with stepwise interpolation, indicating that linear interpolation typically inflates their scores. The choice of interpolation also causes dramatic rank changes: consistency sampling ranks first with linear interpolation but drops to last place with stepwise interpolation, a 12.5 point absolute decrease.

In summary, many confidence estimation approaches for prompted LLMs—particularly verbalization-based methods—produce highly sparse outputs, concentrated within a few unique confidence values. This sparsity severely limits practical utility and needs to be appropriately reflected in evaluation methodologies. Our experiments demonstrate that the choice of interpolation method for threshold-based metrics critically impacts both absolute performance scores and relative rankings. We therefore advocate for using stepwise interpolation rather than linear interpolation when reporting such metrics, as it provides fair comparisons by properly accounting for sparsity.

[4]We verified this by inspecting the publicly available code of various works (Nguyen et al., 2025; Lin et al., 2024a,b; Vashurin et al., 2025). More details in Appendix G.

4 Verbalization Logprobs

Having observed performance drops for verbalization under proper evaluation while token logprobs remains stable, we investigate whether their combination can mitigate these losses. Specifically, we propose incorporating token probabilities into vanilla verbalization to reduce sparsity. We argue that vanilla verbalization discards rich information: it obtains the sampled digit but ignores the probability the model assigned to generating it. For instance, using vanilla verbalization, we might sample the sequence "`the confidence is 95%`" and the standard approach is to take 95% *at face value*. However, each digit token ('`9`' and '`5`') has an underlying probability distribution over alternatives that we can leverage.

For *verbalization logprobs*, we compute confidence as the expected value over possible digits at each position:

$$\sum_{d=0}^{9} 10d \cdot P(x_i = d) + \sum_{d=0}^{9} d \cdot P(x_{i+1} = d)$$

where x_i and x_{i+1} are the tens and units digit tokens, respectively, and P is the LLM's token probability. This transforms verbalized confidences from discrete sampled values into continuous estimates.[5]

Table 3 shows the results using the same setup as before, with additional metrics AUROC, ECE, and cost[6]. We find that verbalization logprobs effectively combines the best of both worlds, by avoiding the sparsity of vanilla verbalization (from 92 to 1 %t5) and improving all metrics compared with both vanilla verbalization (+2.3 percentage points AUARC, +0.9 AUROC) and token logprobs (+1.3 AUARC, +4.7 AUROC). This suggests that

[5]Strictly speaking, this expectation should use conditional probabilities, since the two digit positions are not independent for autoregressive LLMs. In this paper, we treat them as independent as an approximation borne out of practical limitations: computing the exact expected value would require knowing the units digit distribution conditioned on *each* possible tens digit, i.e., running a forward pass for each one, but standard API access only returns logprobs for the actually generated token. We believe this is a reasonable approximation, since the tens digit dominates the expected value anyway.

[6]Relative cost based on input/output token counts; see Appendix C for details.

	AUROC	ECE	AUARC	%t5	Cost
Random conf.	0.482	0.223	0.652	0	1.0×
Consist. samp.	0.573	0.296	0.683	100	4.0×
Verb. sampling	0.678	0.238	0.734	45	6.8×
Top-2 verb.	0.689	0.202	0.731	93	2.3×
Vanilla verb.	0.661	0.235	0.713	92	1.7×
Token logpr.	0.623	0.283	0.723	0	1.0×
Verbalization-logprobs	0.670	0.234	0.736	1	1.7×

Table 3: Comparison of confidence estimation methods with Qwen3-32B (averages over our four datasets). %t5 indicates concentration in top-5 most frequent confidence values. ECE is the expected calibration error. Cost is a multiple relative to simply predicting the class label (no confidence).

verbalized digit distribution carries useful confidence signals beyond what token logprobs provides. Compared with verbalization sampling (the strongest baseline by AUARC), verbalization logprobs achieves comparable performance (0.736 vs. 0.734) at a fraction of the cost (1.7× vs. 6.8×), as it requires only a single inference call rather than four.

In addition to the above empirical evidence, showing the benefit of our method, we provide an analysis of how its confidence scores relate to the vanilla verbalization method, to lend some theoretical justification for why we would expect improved results. To that end, we measured the correlation between the scores of the two methods. Since both methods measure the same underlying belief of the model, it is expected that they would have high correlation. The question is (a) just how high is the correlation? (if it is "too high", say > 98%, this implies the two methods are redundant), and (b) what practical difference does it make for someone who wants to use the confidence scores for thresholding/rejection decisions?

Table 4 shows the correlation (Spearman's ρ) between verbalization logprobs and vanilla verbalization across all four datasets. We find that ρ averages 0.89 across the four datasets, meaning the two methods are, as expected, highly correlated but still with a clear divergence from each other.[7]

On the point of practical impact, we note that vanilla verbalization collapses confidence predictions into at most only 12 unique values (and as few as 8 in SST-2), while verbalization logprobs uses at least 630 unique values (and as many as 2,227 in ESCI). This is an 80× to 200× increase in resolution, which is critical and has practical consequences: as shown in Figure 1, if we desire $\geq 98\%$ accuracy, with vanilla verbalization one must go from rejecting 23.6% to rejecting 39.7% of predictions because there is no threshold in between. In contrast, verbalization logprobs provides the fine-grained thresholds to avoid such large jumps: one need only reject 28.1% of examples to achieve $\geq 98\%$ accuracy, thus salvaging a full 11.6% of the data (39.7% - 28.1%).

[7]Other divergence measures we examined were means of absolute differences, medians of absolute differences, and differences of median confidences. In all those measures we find the same pattern: the two methods are similar yet meaningfully different.

Dataset	ρ	Vanilla # unique values	Logprobs # unique values
SST-2	0.93	8	630
SST-5	0.97	11	1,466
ESCI	0.76	11	2,227
Yahoo	0.90	12	1,869

Table 4: Correlation (Spearman's ρ) between confidences of vanilla verbalization ("Vanilla") and verbalization logprobs ("Logprobs") across all four datasets (using Qwen3-32B).

5 Conclusion

Sparsity of confidence estimates, i.e., producing only a handful of unique confidence values, is a problem that can severely limit the practical usefulness of prompted LLMs for classification. It is hence important that evaluation metrics account for this sparsity by using stepwise interpolation, as opposed to linear (aka trapezoidal) interpolation. In this work, we showed how the latter artificially inflates performance scores, and thus can lead to reaching inaccurate conclusions about classifier quality.

Stepwise interpolation in AUARC was critical to understanding the limitations of the popular vanilla verbalization method. We proposed the *verbalization logprobs* method, a simple extension of vanilla verbalization that alleviates sparsity issues, improves confidence estimates, and matches sampling-based methods at much lower cost.

Limitations

Our verbalization logprobs approach assumes that each digit is consistently tokenized as a single token, and while this holds for the models we tested,

the approach may require adaptation for tokenizers that utilize different schemes. For sampling-based approaches, we fix the number of samples at four based on prior work showing early saturation, but the optimal number may vary across models and tasks. While we focus on AUARC to demonstrate the impact of interpolation methods, similar considerations apply to other threshold-based metrics such as AUPRC, which we leave for future investigation. Finally, our proposed verbalization logprobs requires access to token probabilities; we do not provide a solution for reducing sparsity in purely verbalization-based settings where logprobs are unavailable.

Acknowledgements

Elena Merdjanovska was partially funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy – EXC 2002/1 "Science of Intelligence" – project number 390523135.

References

Jiefeng Chen, Jinsung Yoon, Sayna Ebrahimi, Sercan Arik, Tomas Pfister, and Somesh Jha. 2023. Adaptation with self-evaluation to improve selective prediction in LLMs. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5190–5213, Singapore.

Wenyu Chen, Chen Miao, Zhenghao Zhang, Cathy Sin-Hang Fung, Ran Wang, Yizhen Chen, Yan Qian, Lixin Cheng, Kevin Y Yip, Stephen Kwok-Wing Tsui, and 1 others. 2024. Commonly used software tools produce conflicting and overly-optimistic auprc values. *Genome Biology*, 25(1):118.

Javier Coronado-Blázquez. 2025. Deterministic or probabilistic? the psychology of llms as random number generators. *Preprint*, arXiv:2502.19965.

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

Ran El-Yaniv and Yair Wiener. 2010. On the foundations of noise-free selective classification. *Journal of Machine Learning Research*, 11(53):1605–1641.

Jiahui Geng, Fengyu Cai, Yuxia Wang, Heinz Koeppl, Preslav Nakov, and Iryna Gurevych. 2024. A survey of confidence estimation and calibration in large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6577–6595, Mexico City, Mexico.

James A Hanley and Barbara J McNeil. 1982. The meaning and use of the area under a receiver operating characteristic (roc) curve. *Radiology*, 143(1):29–36.

Yuheng Huang, Jiayang Song, Zhijie Wang, Shengming Zhao, Huaming Chen, Felix Juefei-Xu, and Lei Ma. 2025. Look before you leap: An exploratory study of uncertainty analysis for large language models. *IEEE Transactions on Software Engineering*.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, and 17 others. 2022. Language models (mostly) know what they know. *Preprint*, arXiv:2207.05221.

Seungone Kim, Juyoung Suk, Shayne Longpre, Bill Yuchen Lin, Jamin Shin, Sean Welleck, Graham Neubig, Moontae Lee, Kyungjae Lee, and Minjoon Seo. 2024. Prometheus 2: An open source language model specialized in evaluating other language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 4334–4353, Miami, Florida, USA. Association for Computational Linguistics.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NeurIPS '22, Red Hook, NY, USA. Curran Associates Inc.

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

Yukyung Lee, JoongHoon Kim, Jaehee Kim, Hyowon Cho, Jaewook Kang, Pilsung Kang, and Najoung Kim. 2025. CheckEval: A reliable LLM-as-a-judge framework for evaluating text generation using checklists. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 15782–15809, Suzhou, China.

Zhen Lin, Shubhendu Trivedi, and Jimeng Sun. 2024a. Contextualized sequence likelihood: Enhanced confidence scores for natural language generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 10351–10368, Miami, Florida, USA.

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

Chen Ling, Xujiang Zhao, Xuchao Zhang, Wei Cheng, Yanchi Liu, Yiyou Sun, Mika Oishi, Takao Osaki, Katsushi Matsuda, Jie Ji, Guangji Bai, Liang Zhao, and Haifeng Chen. 2024. Uncertainty quantification for in-context learning of large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 3357–3370, Mexico City, Mexico.

Gabrielle Kaili-May Liu, Gal Yona, Avi Caciularu, Idan Szpektor, Tim G. J. Rudner, and Arman Cohan. 2025a. MetaFaith: Faithful natural language uncertainty expression in LLMs. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 29600–29644, Suzhou, China.

Xiaoou Liu, Tiejin Chen, Longchao Da, Chacha Chen, Zhen Lin, and Hua Wei. 2025b. Uncertainty quantification and confidence calibration in large language models: A survey. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.

Joseph Marvin Imperial and Harish Tayyar Madabushi. 2025. Scaling policy compliance assessment in language models with policy reasoning traces. *arXiv e-prints*, pages arXiv–2509.

John Muschelli III. 2020. Roc and auc with a binary predictor: a potentially misleading metric. *Journal of classification*, 37(3):696–708.

Malik Sajjad Ahmed Nadeem, Jean-Daniel Zucker, and Blaise Hanczar. 2009. Accuracy-rejection curves (arcs) for comparing classification methods with a reject option. In *Proceedings of the third International Workshop on Machine Learning in Systems Biology*, volume 8 of *Proceedings of Machine Learning Research*, pages 65–81, Ljubljana, Slovenia. PMLR.

Huy Nghiem, Advik Sachdeva, and Hal Daumé III. 2025. Smarter: A data-efficient framework to improve toxicity detection with explanation via self-augmenting large language models. *arXiv preprint arXiv:2509.15174*.

Dang Nguyen, Ali Payani, and Baharan Mirzasoleiman. 2025. Beyond semantic entropy: Boosting LLM uncertainty quantification with pairwise semantic similarity. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 4530–4540, Vienna, Austria.

Alexander Nikitin, Jannik Kossen, Yarin Gal, and Pekka Marttinen. 2024. Kernel language entropy: fine-grained uncertainty quantification for llms from semantic similarities. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, NeurIPS '24, Red Hook, NY, USA. Curran Associates Inc.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine learning in python. *J. Mach. Learn. Res.*, 12(null):2825–2830.

Edward Phillips, Sean Wu, Soheila Molaei, Danielle Belgrave, Anshul Thakur, and David Clifton. 2025. Geometric uncertainty for detecting and correcting hallucinations in llms. *arXiv preprint arXiv:2509.13813*.

Chandan K. Reddy, Lluís Màrquez, Fran Valero, Nikhil Rao, Hugo Zaragoza, Sambaran Bandyopadhyay, Arnab Biswas, Anlu Xing, and Karthik Subbian. 2022. Shopping queries dataset: A large-scale ESCI benchmark for improving product search. *Preprint*, arXiv:2206.06588.

Jie Ren, Yao Zhao, Tu Vu, Peter J. Liu, and Balaji Lakshminarayanan. 2023. Self-evaluation improves selective generation in large language models. In *Proceedings on "I Can't Believe It's Not Better: Failure Modes in the Age of Foundation Models" at NeurIPS 2023 Workshops*, volume 239 of *Proceedings of Machine Learning Research*, pages 49–64. PMLR.

Jiandong Shao, Yao Lu, and Jianfei Yang. 2025. Benford's curse: Tracing digit bias to numerical hallucination in llms. *arXiv preprint arXiv:2506.01734*.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA.

Tao Sun, Jian Xu, Yuanpeng Li, Zhao Yan, Ge Zhang, Lintao Xie, Lu Geng, Zheng Wang, Yueyan Chen, Qin Lin, Wenbo Duan, Kaixin Sui, and Yuanshuo Zhu. 2025. Bitsai-cr: Automated code review via llm in practice. In *Proceedings of the 33rd ACM International Conference on the Foundations of Software Engineering*, FSE Companion '25, page 274–285, New York, NY, USA. Association for Computing Machinery.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher Manning. 2023. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5433–5442, Singapore.

Francesco Tonolini, Nikolaos Aletras, Jordan Massiah, and Gabriella Kazai. 2024. Bayesian prompt ensembles: Model uncertainty estimation for black-box large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 12229–12272, Bangkok, Thailand.

Roman Vashurin, Ekaterina Fadeeva, Artem Vazhentsev, Lyudmila Rvanova, Daniil Vasilev, Akim Tsvigun, Sergey Petrakov, Rui Xing, Abdelrahman Sadallah, Kirill Grishchenkov, Alexander Panchenko, Timothy Baldwin, Preslav Nakov, Maxim Panov, and Artem Shelmanov. 2025. Benchmarking uncertainty quantification methods for large language models with LM-polygraph. *Transactions of the Association for Computational Linguistics*, 13:220–248.

Artem Vazhentsev, Ekaterina Fadeeva, Rui Xing, Gleb Kuzmin, Ivan Lazichny, Alexander Panchenko, Preslav Nakov, Timothy Baldwin, Maxim Panov, and Artem Shelmanov. 2025. Unconditional truthfulness: Learning unconditional uncertainty of large language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 35661–35682, Suzhou, China.

Cheng Wang, Gyuri Szarvas, Georges Balazs, Pavel Danchenko, and Patrick Ernst. 2024. Calibrating verbalized probabilities for large language models. *Preprint*, arXiv:2410.06707.

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

Weihao Xuan, Qingcheng Zeng, Heli Qi, Junjue Wang, and Naoto Yokoya. 2025. Seeing is believing, but how much? a comprehensive analysis of verbalized calibration in vision-language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 1408–1450, Suzhou, China.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Fan Yin, Philippe Laban, XIANGYU PENG, Yilun Zhou, Yixin Mao, Vaibhav Vats, Linnea Ross, Divyansh Agarwal, Caiming Xiong, and Chien-Sheng Wu. 2025. Bingoguard: LLM content moderation tools with risk levels. In *The Thirteenth International Conference on Learning Representations (ICLR 2025)*.

Qingcheng Zeng, Weihao Xuan, Leyang Cui, and Rob Voigt. 2025. Thinking out loud: Do reasoning models know when they're right? In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 1394–1407, Suzhou, China.

Mozhi Zhang, Mianqiu Huang, Rundong Shi, Linsen Guo, Chong Peng, Peng Yan, Yaqian Zhou, and Xipeng Qiu. 2024. Calibrating the confidence of large language models by eliciting fidelity. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 2959–2979, Miami, Florida, USA.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems (NeurIPS 2015)*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

A Datasets

Table 5 provides an overview of the four classification datasets used in our experiments. SST-2 and SST-5 represent sentiment analysis tasks of varying granularity on movie reviews. Yahoo! answers provides a multi-class topic classification task, while Amazon ESCI tests product-query relevance classification.

For computational efficiency, we sample subsets from the larger datasets. From SST-2 and SST-5 we use their full standard test splits (872 and 2,210 samples respectively). For Yahoo! answers, we randomly sample 6,000 examples from the original 60,000 test samples (10%). For Amazon ESCI, we sample 8,604 examples from the full dataset. These sample sizes provide sufficient statistical power for our comparisons while keeping inference costs manageable across multiple methods and models.

B Claude 3.7 Sonnet Results

We present results for Claude 3.7 Sonnet to demonstrate that our findings generalize to other models. Table 6 shows the impact of interpolation method on AUARC, mirroring our main findings: consistency sampling drops from rank 1 to rank 4 when switching from linear to stepwise interpolation. Table 7 provides comprehensive metrics. Note that token logprobs are unavailable via the Bedrock API.

	AUARC			
	linear	ranks	step	%t5
Consistency Sampling	0.814	1→4	0.703	100
Vanilla Verbalization	0.801	2→3	0.771	96
Verbalization Sampling	0.795	3→1	0.790	42
Top-2 Verbalization	0.795	4→2	0.775	95
Token Logprobs	/		/	/

Table 6: Comparison of different interpolations when calculating AUARC for Claude 3.7 Sonnet.

	AUROC	ECE	AUARC	%t5	Cost
Random	0.494	0.225	0.665	0	1.0×
Consist. Samp.	0.579	0.266	0.703	100	4.0×
Vanilla Verb.	0.691	0.184	0.771	96	1.9×
Verb. Sampling	0.705	0.187	0.790	42	7.4×
Top-2 Verb.	0.710	0.131	0.775	95	2.6×

Table 7: Comparison of confidence estimation methods with Claude 3.7 Sonnet (averages over our four datasets).

Approach	w/o reasoning	w/ reasoning
Consistency Sampling	4.0×	4.0×
Vanilla Verbalization	1.8×	1.3×
Verbalization Sampling	7.1×	5.0×
Top-2 Verbalization	2.5×	1.5×
Logprobs	1.0×	1.0×

Table 8: Relative cost multipliers for each confidence estimation method compared to a baseline that predicts only the class label. This uses Qwen3-32B with and without reasoning. Sampling-based approaches incur higher costs due to multiple inference calls; verbalization adds modest overhead from confidence tokens.

C Cost Calculation

We compute relative cost as the ratio of total tokens (input + output) compared to a baseline that predicts only the class label without confidence estimation. Table 8 shows the cost multipliers for each method using Qwen3-32B with and without reasoning. Sampling-based approaches incur higher costs due to multiple inference calls. Verbalization adds modest overhead from the confidence tokens. Token logprobs has no additional cost as it uses the same inference call.

D Reasoning Results

Table 9 presents results with reasoning mode enabled (1024 reasoning budget tokens for Claude). Sparsity patterns persist even with reasoning, confirming that the phenomenon is not limited to direct prediction settings.

E Sparsity Investigation for Alternative Confidence Range 0–9

In our main experiments, we instruct the model to verbalize its confidence from 0–100, which is arguably the most natural range as it aligns well with percentages and is consistent with prior work. In this appendix, we present results using an alternative confidence range of 0–9 to verify that our findings are not an artifact of the chosen confidence range.

We run vanilla verbalization and verbalization sampling with this new confidence range, following the setup in Section 3. For direct comparison with Figure 1, we show in Figure 2 confidence histograms and accuracy-rejection curves for the same dataset and model. As expected, we again find extreme sparsity of confidence values—the narrower range offers even fewer distinct values. Consistent

Dataset	Task	Text	# Classes	Classes	Test Size
SST-2 (Socher et al., 2013)	sentiment	movie review	2	[positive, negative]	872
SST-5 (Socher et al., 2013)	sentiment	movie review	5	[very positive, positive, neutral...]	2210
Yahoo (Zhang et al., 2015)	topic	question title	10	[Society & Culture, Health...]	6000
ESCI (Reddy et al., 2022)	relevance	query-product pair	4	[exact, substitute, complement...]	8604

Table 5: Overview of the four classification datasets used in our experiments. SST-2 and SST-5 are sentiment analysis tasks of varying granularity on movie reviews. Yahoo! answers provides multi-class topic classification. Amazon ESCI tests product-query relevance classification.

		Qwen3-32B					*Claude 3.7 Sonnet*				
	Approach	AUROC↑	ECE↓	AUARC↑	%t5↓	Cost↓	AUROC↑	ECE↓	AUARC↑	%t5	Cost↓
baselines	Random confidences	0.498	0.223	0.674	0	1.00×	0.507	0.225	0.716	0	1.00×
	Consistency sampling	0.622	0.223	0.706	100	4.08×	0.569	0.238	0.732	100	4.00×
	Vanilla verbalization	0.659	0.217	0.739	89	1.21×	0.707	0.182	0.794	90	1.30×
	Verbalization sampling	**0.688**	0.221	**0.768**	43	4.87×	0.719	0.183	**0.807**	49	5.19×
	Top-2 verbalization	0.682	**0.174**	0.757	85	1.60×	**0.732**	**0.136**	0.803	85	1.50×
	Token Logprobs	0.529	0.280	0.711	0	1.00×	/	/	/	/	/

Table 9: Confidence estimation results with reasoning mode enabled. Claude uses 1024 reasoning budget tokens. Sparsity (%t5) remains high for verbalization approaches even with reasoning, confirming that the phenomenon persists across inference settings. Token logprobs are unavailable for Claude via Bedrock API.

with our prior findings, the majority of confidence scores are concentrated in just two values, $\frac{8}{9}$ and $\frac{9}{9}$.

Table 10 compares both confidence ranges across both models. As expected, concentration in the top-5 values (%t5) increases substantially with the narrower range, while AUARC scores remain comparable. Importantly, linear interpolation continues to inflate AUARC scores for both ranges. Our results confirm that both the sparsity limitation and the inflation from linear interpolation persist with the alternative range.

F Confidence Estimation Details

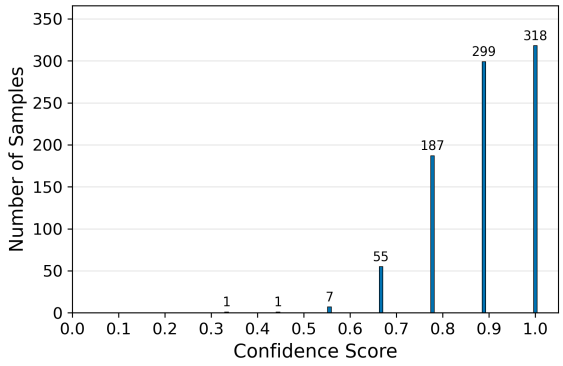
Below we describe the confidence estimation methods included in our experimental comparison. Table 11 shows example prompts for each approach on the SST-2 dataset. Claude uses the Bedrock API, and Qwen3-32B uses vLLM. We generate tokens at temperature=0 if not otherwise mentioned with 1000 maximum tokens.

- **Vanilla verbalization** (Tian et al., 2023; Xiong et al., 2024): The model is prompted to output a self-reported confidence score as a percentage between 0 and 100 (e.g., "Label and Confidence: positive, 85%").

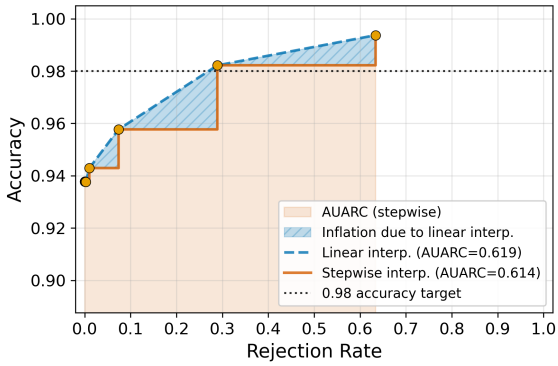
- **Top-K verbalization** (Tian et al., 2023): The model outputs its k best guesses along with confidence scores for each. We use $k = 2$ and take the confidence of the top-ranked guess.

	AUARC		
	linear	step	%t5
Qwen3 32B			
Vanilla verbalization			
conf. range 0–100	0.758	0.713	92
conf. range 0–9	0.761	0.716	99
Verbalization sampling			
conf. range 0–100	0.741	0.734	45
conf. range 0–9	0.754	0.728	66
Claude 3.7 Sonnet			
Vanilla verbalization			
conf. range 0–100	0.801	0.771	96
conf. range 0–9	0.805	0.773	99
Verbalization sampling			
conf. range 0–100	0.795	0.790	42
conf. range 0–9	0.800	0.788	59

Table 10: Impact of confidence scoring range on AUARC and sparsity. Scores are averages over SST-2, SST-5, Amazon ESCI, and Yahoo! topic classification (see Table 5 for dataset details). %t5 refers to the concentration in top-5 confidence values. The narrower range 0–9 increases concentration substantially while AUARC scores remain comparable.



(a) Histogram showing all confidence scores on SST-2 with an alternative confidence range of 0–9



(b) Accuracy-rejection curve on SST-2 with an alternative confidence range of 0–9

Figure 2: **Sparse confidences for confidence scoring range 0–9.** (a) Histogram of vanilla verbalized confidences on SST-2 with Qwen3-32B using the confidence range 0–9 (normalized to 0–1). Only seven unique values appear, with the majority concentrated at $\frac{8}{9}$ and $\frac{9}{9}$. (b) Accuracy-rejection curve demonstrating limited threshold choices due to sparsity.

- **Consistency sampling** (Xiong et al., 2024): The same prompt is executed multiple times without requesting confidence scores. Confidence is computed as the frequency of the most common predicted label. We sample 4 responses (Xiong et al., 2024) at temperature 1.0 (Zhang et al., 2024).
- **Verbalization sampling** (Xiong et al., 2024): Similar to consistency sampling, but using the vanilla verbalization prompt. Confidence is the mean of the sampled verbalized values. We use the same sampling parameters (4 responses, temperature 1.0).
- **Token logprobs** (Huang et al., 2025): We use a prompt without explicit confidence requests and compute confidence from the log-probability of the predicted class label token. For multi-token labels, we average the log-probabilities. This approach requires API access to token probabilities.
- **Random baseline**: We assign uniformly random confidence values between 0 and 1 to establish a lower bound for comparison and to calculate relative cost increases.

G Linear Interpolation in Prior Work

We verified that prior work uses linear interpolation when calculating AUARC, by inspecting publicly available code. We found that researchers commonly use `scikit-learn`'s (Pedregosa et al., 2011) `auc()` function (Nguyen et al., 2025; Lin et al., 2024a,b), which uses trapezoidal integration (i.e. linear interpolation) per its public documentation. It is not surprising that the use of this function became the default; unlike AUROC and AUPRC—which have dedicated `scikit-learn` functions with correct interpolation: `roc_auc_score()` and `average_precision_score()`, respectively—there is no specialized function for AUARC.

Other implementations compute AUARC without calling `auc()` but still approximate linear interpolation. For instance, LM-Polygraph (Vashurin et al., 2025) can compute AUARC (via the Prediction-Rejection Ratio—PRR, equivalent to AUARC for classification tasks) but does so by thresholding at every prediction in order of confidence scores, regardless of whether multiple predictions share the same confidence. With random tie breaking, it approximates linear interpolation between the confidence-thresholded anchors.

While these implementations are valid for continuously-distributed confidence scores, they compute overly optimistic AUARC scores for sparse verbalized confidences.

Prompt Type	Baseline prompt without confidence
Example Prompt	```` Given the sentence, assign a sentiment label from ['negative', 'positive']. Use the following format: ```Label [ONLY the sentiment label; not a complete sentence]``` Only the label, don't give me the explanation. Sentence: too much of the humor falls flat . ````
Used in	Random, Logprobs, Consistency Sampling
Prompt Type	Vanilla Verbalization
Example Prompt	```` Given the sentence, assign a sentiment label from ['negative', 'positive'] and your confidence in this answer. The confidence indicates how likely you think your answer is true. Use the following format: ``` Label and Confidence (0-100): [ONLY the sentiment label; not a complete sentence], [Your confidence level, please only include the number in the range of 0-100]% ``` Only the label and confidence, don't give me the explanation. Sentence: too much of the humor falls flat . ````
Used in	Vanilla Verbalization, Verbalization Sampling
Prompt Type	Top-2 Verbalization
Example Prompt	```` Given the sentence, assign a sentiment label from ['negative', 'positive']. Provide your 2 best guesses and the probability that each is correct (0\% to 100\%). Give only the sentiment label and probabilities, no other words or explanation. Example: G1: <ONLY the sentiment label of first most likely guess; not a complete sentence, the guess!> P1: <ONLY the probability that G1 is correct, without any extra commentary whatsoever; the probability!> ... G2: <ONLY the sentiment label of 2-th most likely guess> P2: <ONLY the probability that G2 is correct, without any extra commentary whatsoever; the probability!> Sentence: too much of the humor falls flat . ````
Used in	Top-2 Verbalization

Table 11: Example Prompts for SST-2, with a list of approaches that uses each prompt.