universität wien

DISSERTATION | DOCTORAL THESIS

Titel | Title

Identifying Influences on Language Model Trustworthiness

verfasst von | submitted by

Yuxi Xia MSc

angestrebter akademischer Grad | in partial fulfilment of the requirements for the degree of

Doktorin der technischen Wissenschaften (Dr.techn.)

Wien | Vienna, 2026

Studienkennzahl lt. Studienblatt | Degree programme code as it appears on the student record sheet:	UA 786 880
Dissertationsgebiet lt. Studienblatt | Field of study as it appears on the student record sheet:	Informatik
Betreut von | Supervisor:	Univ.-Prof. Dr. -Ing. Benjamin Roth B.Sc. M.Sc. Univ.-Prof. Dipl.-Inform.Univ. Dr. Claudia Plant

Acknowledgements

This thesis would not have been possible without the support, guidance, and encouragement of many people.

First and foremost, I would like to express my deepest gratitude to my supervisor, **Prof. Benjamin Roth**, for his invaluable mentorship throughout my doctoral journey. I am especially thankful for the freedom to explore ideas, the trust he placed in my work, and the many thoughtful discussions that continuously sharpened my thinking and writing. I am also grateful to **Prof. Claudia Plant** for her insightful advice on research directions and broader perspectives. Finally, I sincerely thank my thesis reviewers, **Prof. Torsten Zesch** and **Prof. Christian Hardmeier**, for agreeing to evaluate my thesis and for their time, expertise, and valuable feedback.

I am grateful to my collaborators and co-authors, **Anastasiia, Dennis, Kinga, Klim, Lisa, Loris, Pedro, Terra, Vasiliki, Yihong, and Prof. Hinrich Schütze**, for the inspiring and productive collaborations that contributed to the research presented in this thesis. Working with them has been both intellectually rewarding and personally enriching. I would also like to thank my colleagues **Andreas, Lukas, Lena, Timour, Vanja, and Nurcan** for creating a stimulating and supportive research environment, and for the many insightful conversations that shaped my research journey. Finally, I appreciate the short but wonderful time spent with my newer colleagues, **Marie, Nick, Pingjun, and Sarah**, and wish them a successful and fulfilling PhD journey ahead.

I would like to acknowledge the broader research community whose work laid the foundation for this thesis. Engaging with their ideas has been a constant source of motivation and inspiration. I am also thankful for the reviewers and organizers of the conferences and journals where parts of this work were presented, whose feedback helped improve the quality and clarity of my research.

I gratefully acknowledge the financial and institutional support that made this research possible, including the **University of Vienna** and **WWTF**, which supported my doctoral studies.

Finally, I would like to thank my family and friends for their unwavering support, patience, and encouragement throughout this journey. Special thanks to my boyfriend, **Mārcis**, whose belief in me provided the strength to persevere through challenges and celebrate milestones along the way.

Abstract

The growing deployment of language models in high-stakes applications places trustworthiness at the forefront, including aspects such as privacy preservation, reliable confidence communication, and robust generalization. Despite strong performance, those models exhibit systematic failures along these dimensions, often driven by external factors that are overlooked or underestimated in standard performance evaluation.

This thesis investigates influences on language model trustworthiness across three interrelated areas: privacy, confidence estimation and calibration, and generalization under distribution shift. The unifying insight is that trustworthiness failures frequently arise from contextual factors such as linguistic variation.

For privacy, the thesis examines prompt-induced memorization in masked language models fine-tuned for named entity recognition, showing that privacy leakage varies substantially with prompt formulation and cannot be treated as a fixed model property.

For confidence and calibration, the thesis studies confidence estimation in large language models, proposing calibration methods based on model agreement and auxiliary confidence estimators. Beyond calibration, the thesis reveals that confidence estimates are often unstable under prompt perturbation and insensitive to semantic changes in estimated responses. Further analysis of training data influence indicates that models may learn to express verbalized confidence from linguistic patterns rather than from content-based evidence.

For generalization, the thesis analyzes AI-text detectors, demonstrating sharp generalization degradation under cross-prompt, cross-model, and cross-domain shifts. Linguistic analysis links these failures to shifts in surface-level syntactic and stylistic features, suggesting that those detectors may rely on fragile cues rather than robust generalizable signals.

Overall, this thesis shows that language model trustworthiness is highly sensitive to external influences such as prompt strategies, language variations and linguistic features. By identifying and analyzing these influences, it provides guidance for more robust evaluation and more trustworthy deployment of language models.

Kurzfassung

Der zunehmende Einsatz von Sprachmodellen in sicherheitskritischen Anwendungsbereichen rückt deren Vertrauenswürdigkeit in den Mittelpunkt. Dazu zählen insbesondere Datenschutz, die verlässliche Kommunikation von Unsicherheit sowie robuste Generalisierungsfähigkeit. Trotz ihrer hohen Leistungsfähigkeit zeigen diese Modelle systematische Schwächen in allen drei Bereichen, die häufig durch externe Einflussfaktoren verursacht werden, welche in Standardbewertungen oft übersehen oder unterschätzt werden.

Diese Dissertation untersucht Einflüsse auf die Vertrauenswürdigkeit von Sprachmodellen in drei miteinander verknüpften Bereichen: Datenschutz, Konfidenzschätzung und Kalibrierung sowie Generalisierung unter Verteilungsverschiebung. Die zentrale Erkenntnis besteht darin, dass Vertrauenswürdigkeitsprobleme häufig durch kontextuelle Faktoren wie sprachliche Variation entstehen.

Im Bereich Datenschutz analysiert die Arbeit prompt-induzierte Memorisation in für Named Entity Recognition feinabgestimmten Masked Language Models und zeigt, dass Privacy Leakage stark von der Promptformulierung abhängt und nicht als feste Modelleigenschaft betrachtet werden kann.

Im Bereich Konfidenz und Kalibrierung untersucht die Dissertation Konfidenzschätzung in großen Sprachmodellen und entwickelt Kalibrierungsverfahren auf Basis von Modellübereinstimmung sowie zusätzlicher Konfidenzschätzer. Über die numerische Kalibrierung hinaus zeigt die Arbeit, dass Konfidenzschätzungen häufig instabil gegenüber Prompt-Variationen sind und nur unzureichend auf semantische Veränderungen in Antwortalternativen reagieren. Weiterführende Analysen des Einflusses von Trainingsdaten deuten darauf hin, dass Modelle verbalisierte Sicherheit eher aus sprachlichen Mustern als aus inhaltlich begründeter Evidenz ableiten.

Im Bereich Generalisierung analysiert die Arbeit Detektoren für KI-generierte Texte und zeigt deutliche Leistungsabfälle unter Prompt-, Modell- und Domänenverschiebungen. Sprachwissenschaftliche Analysen führen diese Generalisierungsprobleme auf Veränderungen syntaktischer und stilistischer Oberflächenmerkmale zurück und legen nahe, dass solche Detektoren häufig fragile Hinweise statt robust generalisierbarer Signale nutzen.

Insgesamt zeigt diese Dissertation, dass die Vertrauenswürdigkeit von Sprachmodellen stark von externen Einflüssen wie Promptstrategien, sprachlicher Variation und linguistischen Merkmalen abhängt. Durch die Identifikation und Analyse dieser Einflussfaktoren liefert sie Ansätze für robustere Evaluationsverfahren und eine vertrauenswürdigere Anwendung von Sprachmodellen.

Contents

Acronyms

AI Artificial Intelligence. 1

API Application Programming Interface. 16, 23

AUROC Area Under the Receiver Operating characteristic Curve. 12, 18, 27, 37, 38

BCE Binary Cross-Entropy. 26, 28

CE Confidence estimation. 15, 35–38

ECE Expected Calibration Error. 18

FL Focal Loss. 27, 28

GPT Generative Pre-trained Transformer. 9

LLMs Large Language Models. 1–3, 15, 16, 19, 20, 23–27, 30, 31, 33, 34, 38, 39

LSTM Long Short-Term Memory. 9

MLMs Mask Language Models. 2, 8–11, 13, 45

NER Named Entity Recognition. 2, 10–12, 46

TQA Textual Question Answering. 23, 24

VQA Visual Question Answering. 23, 24

1. Introduction

1.1. Motivation

Large Language Models (LLMs) have rapidly become foundational components of modern Artificial Intelligence (AI) systems, underpinning applications ranging from information retrieval (Zhu et al., 2025) and decision support to education, creative writing, and scientific discovery (Chakrabarty et al., 2024; Rane et al., 2023). Their growing deployment in high-stakes and user-facing settings has shifted research attention from purely optimizing predictive performance to ensuring the **trustworthiness** of these models. Trustworthiness encompasses a constellation of properties, including privacy preservation, reliable confidence communication, robustness and generalization, that collectively determine whether users and downstream systems can safely and appropriately rely on model outputs (Fan et al., 2025).

Despite impressive advances in scale and capability, contemporary language models exhibit systematic failures along these dimensions. They may (1) unintentionally memorize and reveal sensitive information from their training data, (2) express confidence that is poorly aligned with factual correctness, or (3) behave unpredictably when faced with distribution shifts such as new prompts, domains, or model families. These shortcomings challenge the assumption that larger or more accurate models are inherently more trustworthy, and motivate a deeper, more principled investigation into the mechanisms that shape user trust and model reliability.

This doctoral thesis investigates **influences on language model trustworthiness** through three interrelated lenses: *privacy leakage*, *confidence estimation and calibration*, and *generalization* in language models. Together, these perspectives capture key requirements for trustworthy deployment: protecting sensitive information, communicating uncertainty reliably, and maintaining robust behavior beyond the training distribution. Although these topics are often studied in isolation, this thesis argues that they are tightly connected by a common theme: **the gap between what a language model appears to communicate and what its outputs are reliably grounded in or able to generalize**. By combining empirical analysis, methodological contributions, and diagnostic frameworks, this work aims to characterize better and ultimately mitigate this gap.

1.2. Trustworthiness and Privacy in Language Models

One fundamental threat to trustworthiness arises from the potential for language models to memorize and expose specific details from their training data. Such memorization poses

a privacy risk because models can be prompted to reveal information about individuals, entities, or proprietary sources present during training (Neel and Chang, 2024). While prior work has demonstrated memorization in auto-regressive language models by directly probing training data using a pre-fixed prompt (Carlini et al., 2021), far less is known about how prompting strategies interact with memorization in fine-tuned Mask Language Models (MLMs), where standard generation-based probing methods are not directly applicable. This gap is particularly relevant because such models remain widely used and achieve state-of-the-art performance in many domain-specific applications and tasks (Darji et al., 2023; Nguyen et al., 2023).

In this thesis, privacy leakage is studied through the lens of **training data memorization detection**. We design the memorization detection framework and examine how different prompt formulations can amplify or obscure memorized signals in MLMs fine-tuned for Named Entity Recognition (NER). By systematically varying prompt properties and analyzing their effect on prediction confidence for in-sample versus out-of-sample entities, this work reveals that memorization is not a static model property, but one that is highly sensitive to prompt design and model choice. These findings underscore the need to consider prompting as a crucial influence factor in privacy risk assessments.

1.3. Confidence, Calibration, and the Communication of Uncertainty

Beyond privacy, trust in language models critically depends on how models communicate uncertainty (Kim et al., 2024). Users frequently rely on model-provided confidence, explicitly or implicitly, when deciding whether to accept, verify, or act upon a model response (Zhao et al., 2023). However, LLMs are known to be systematically miscalibrated, often exhibiting overconfidence even when their predictions are incorrect (Tian et al., 2023; Xiong et al., 2024). This misalignment between confidence and accuracy can mislead users and inflate perceived reliability. This thesis advances the study of confidence estimation and calibration in language models along three dimensions.

1. Improving LLM calibration. It introduces methods for improving calibration by leveraging response agreement across multiple models and training auxiliary confidence estimators with task-appropriate loss functions. This work demonstrates consistent improvements over standard confidence estimation methods such as verbalized certainty or token-level probabilities.

2. Explaining LLM confidence. This thesis also examines the origins of verbalized confidence in language models. By tracing confidence expressions back to influential training data, we show that models may learn to mimic generic linguistic markers of certainty without grounding confidence in task-relevant evidence. This disconnect highlights a deeper limitation in current training paradigms: language models can learn how to sound confident without learning when confidence is warranted.

3. Evaluating LLM confidence. The thesis argues that evaluating confidence estimates with only their calibration with accuracy is insufficient. Real-world usage demands confidence estimates that are robust to prompt variations, stable across semantically equivalent answers, and sensitive to meaningful changes in answer content. To address this, we propose a comprehensive evaluation framework that exposes critical failure modes of existing confidence estimation approaches, revealing that methods performing well on traditional calibration metrics may still behave unreliably under linguistic variation.

Taken together, these contributions provide a broader perspective on confidence estimation in language models by addressing how confidence estimation can be improved, what shapes LLMs' confidence expression, and where current evaluation remains incomplete.

1.4. Generalization and Trust in AI-Text Detection

A third pillar of language model trustworthiness concerns generalization, particularly for systems that detect AI-generated text. As generative models become increasingly widespread, these detectors are being adopted in areas such as academic writing, content moderation, and media forensics (Khlaif et al., 2023; Saqib and Zia, 2025). While many detectors achieve high accuracy under controlled, in-domain conditions, their performance often degrades sharply when confronted with new prompts, unseen model families, or different domains (He et al., 2024).

This thesis investigates the causes of such generalization failures through a large-scale study of AI-text detectors combined with systematic linguistic analysis. It introduces a benchmark dataset containing AI-generated texts produced with diverse prompting strategies by multiple LLMs (generators) across domains, enabling generalization evaluation under cross-prompt, cross-model, and cross-domain conditions. Beyond reporting performance, the analysis links generalization behavior to measurable shifts in linguistic features between training and test data. The results show that detector robustness is closely associated with surface-level patterns such as tense usage and pronoun frequency, suggesting that many detectors rely on shortcut linguistic cues rather than on signals that generalize reliably across settings.

1.5. Publication Overview and Thesis Structure

Taken together, the contributions of this thesis provide a multifaceted view of language model trustworthiness.

- **Privacy.** Demonstrates how prompt design influences privacy leakage and memorization detection in fine-tuned language models.

(P-1) **Yuxi Xia**, Anastasiia Sedova, Pedro Henrique Luz de Araujo, Vasiliki Kougia, Lisa Nußbaumer, and Benjamin Roth. "*Exploring prompts to elicit memorization in masked language model-based named entity recognition*". In: PLoS One 20.9 (2025), e0330877.

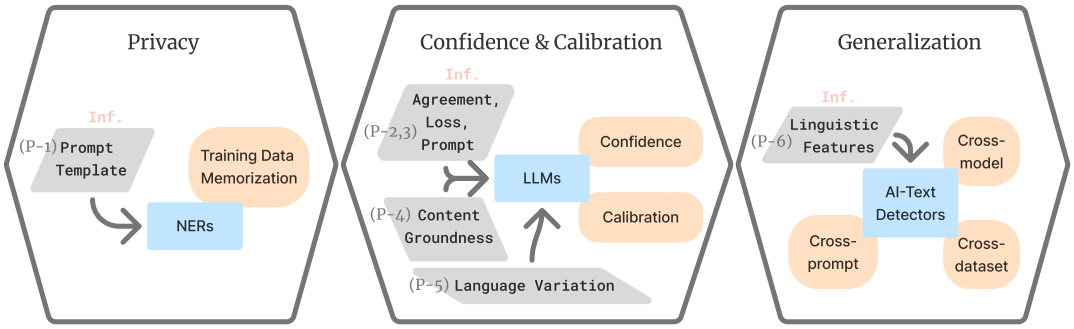


Figure 1.1.: The thesis is mainly divided into three parts: influences (Inf.) on privacy leakage in NERs, confidence calibration in LLMs, and generalization of AI-text detectors.

- **Confidence and Calibration.** Proposes methods for improving confidence estimation and calibration (*P-2, P-3*), analyzes the training data influences underlying verbalized confidence, revealing systematic grounding deficiencies (*p-4*), and introduces new evaluation criteria aligned with real-world usage (*P-5*).

  (P-2) **Yuxi Xia**, Klim Zaporojets, and Benjamin Roth. "*Learn to pick the winner: Black-box ensembling for textual and visual question answering*". In: Proceedings of the 21st Conference on Natural Language Processing (KONVENS 2025): Long Papers. Sept. 2025, pp. 12–26.

  (P-3) **Yuxi Xia**, Pedro Henrique Luz De Araujo, Klim Zaporojets, and Benjamin Roth. "*Influences on LLM Calibration: A Study of Response Agreement, Loss Functions, and Prompt Styles*". In: Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL 2025): Long Papers. Oral presentation. July 2025, pp. 3740–3761.

  (P-4) **Yuxi Xia**, Loris Schoenegger, and Benjamin Roth. "*Influential Training Data Retrieval for Explaining Verbalized Confidence of LLMs*". In: Proceedings of the 48th European Conference on Information Retrieval (ECIR 2026): Long papers.

  (P-5) **Yuxi Xia**, Dennis Ulmer, Terra Blevins, Yihong Liu, Hinrich Schütze, and Benjamin Roth. *Calibration Is Not Enough: Evaluating Confidence Estimation Under Language Variations*. Under review. 2026. arXiv: 2601.08064 [cs.CL].

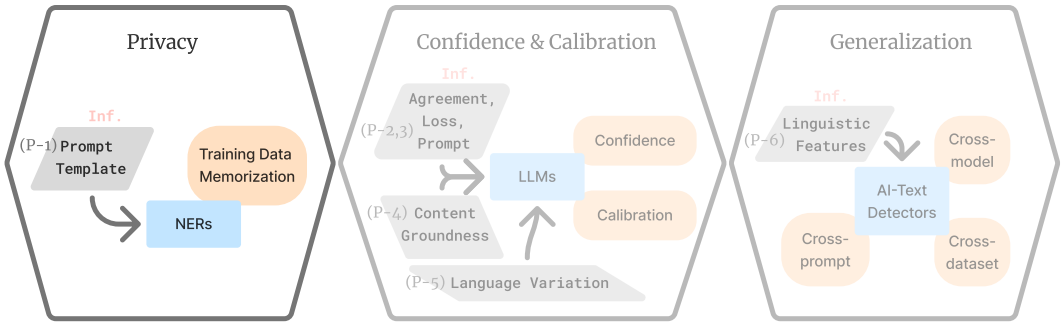
- **Generalization.** Explains generalization gaps in AI-text detection through linguistic analysis, offering insights into why detectors fail under distribution shift.

  (P-6) **Yuxi Xia**, Kinga Stańczak, and Benjamin Roth. "*Explaining Generalization of AI-Generated Text Detectors Through Linguistic Analysis*". In: Proceed-

ings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2026): Long papers.

The structure and content of this thesis are visualized in Figure 1.1. The remaining chapters are organized as follows. Chapters 2, 3, and 4 provide the background, review related work, and present our contributions to privacy, confidence estimation and calibration, and AI-text detection, respectively. Finally, Chapter 5 summarizes our findings, discusses their broader implications for designing trustworthy language models, and outlines directions for future research.

2. Prompt Influences on the Memorization of Named Entity Recognition Models



Privacy Leakage

Privacy leakage refers to the unintended exposure of training data or private user information through the model's outputs or internal representations.

Trustworthy language models should handle sensitive information in a privacy-preserving manner. As models grow in scale and capability, concerns about privacy leakage have become increasingly prominent. Privacy leakage refers to the unintended disclosure of training data, sensitive attributes, or private user information through a model's outputs or internal representations, even when such information is not explicitly provided in the input (Jegorova et al., 2022). From a trustworthiness perspective, privacy leakage is particularly problematic because it undermines user safety, violates data protection expectations, and raises legal and ethical concerns related to data governance (Neel and Chang, 2024).

2.1. Privacy Leakage Manifestation

Language models are trained on large corpora that may contain personally identifiable information, proprietary content, or copyrighted material (Kim et al., 2023b; Jegorova

et al., 2023). Even when datasets undergo deduplication or filtering, residual sensitive content may remain. Empirical studies (Kim et al., 2023b; Lukas et al., 2023; Ali et al., 2022) have demonstrated that MLMs and autoregressive language models can memorize individual data points when those points appear repetitively or exhibit atypical linguistic patterns. This memorization increases the risk that querying the model, intentionally or unintentionally, elicits sensitive fragments from its training data. As such, understanding privacy leakage is essential for assessing a model's trustworthiness, especially in high-risk domains such as healthcare, education, customer service, or enterprise knowledge management (Zhang et al., 2020; Nastoska et al., 2025).

Privacy leakage can manifest in several forms:

- **Membership inference attacks** (Hu et al., 2022; Shokri et al., 2017; Carlini et al., 2022) attempt to determine whether a specific record appeared in a model's training data.

- **Attribute inference attacks** (Jia and Gong, 2018; Jayaraman and Evans, 2022) probe the model to deduce the latent user's private attributes (e.g., location, sexual orientation, political view) or details not provided in the input.

- **Training data extraction attacks** (Ishihara, 2023; Carlini et al., 2023a, 2021) aim to recover sensitive sequences directly from the model by crafting specific prompts. Prompt formulation has therefore become a critical factor, as the extent of private or memorized content revealed can vary substantially depending on phrasing, structure, and contextual cues. This connection between prompting and privacy leakage is particularly important given the growing role of prompt engineering in downstream applications of language models.

Trustworthiness frameworks commonly treat privacy as a foundational dimension alongside robustness, fairness, and interpretability (Chander et al., 2025). Unlike robustness or fairness, which can often be evaluated through standardized benchmarks (Roth et al., 2024), privacy leakage is harder to detect because harmful memorization may remain hidden under conventional evaluation settings. Models that perform well on downstream tasks can still memorize and disclose sensitive training content (Wei et al., 2025). Addressing this risk therefore requires methods that explicitly probe memorization and account for how interaction conditions, such as prompt formulation, affect the extraction of memorized content.

In summary, privacy leakage represents a central challenge for trustworthy language models. As language models are increasingly deployed in public and commercial systems, understanding how prompt formulation affects the disclosure of memorized content becomes essential for reliable evaluation and safer deployment.

2.2. Training Data Extraction and Memorization in Language Models

Memorization in machine learning models has been primarily motivated by privacy concerns arising from the unintended disclosure of sensitive training data.

Auto-regressive Model

Auto-regressive models are machine learning models that generate sequences by iteratively predicting the next element based on the preceding context.

Training Data Extraction in Auto-regressive Models. Early work by Carlini et al. (2019) demonstrated that recurrent neural language models, such as LSTMs (Hochreiter and Schmidhuber, 1997), memorize a non-trivial fraction of their training data, including rare and potentially sensitive sequences. This work established memorization as a measurable and systematic phenomenon. Recent work on training data extraction mainly focus on auto-regressive models, especially GPT models. Carlini et al. (2021) introduced extractability as an operational definition of memorization, showing that sensitive training examples can be recovered from GPT-2 using carefully designed prefixed prompts. Later work further refined this perspective by quantifying memorization across model sizes and datasets, demonstrating that larger models and rarer training sequences are more susceptible to memorization (Carlini et al., 2023b). Similar behaviors have been observed in other autoregressive models such as GPT-Neo (Black et al., 2021), confirming that memorization is a general property of large generative language models.

A defining characteristic of extractability-based approaches is that they rely on the generative nature of auto-regressive models: memorized content is detected by prompting the model with an input context that resembles a training sequence and observing whether it reproduces the corresponding continuation or missing content. Consequently, these methods assume that memorized information becomes observable when the input sufficiently activates stored training patterns. While powerful, this assumption limits the applicability of extractability-based methods to auto-regressive models and makes them difficult to directly apply to masked language models or task-specific fine-tuned architectures.

Masked Language Model

Mask Language Models (MLMs) refer to language models trained or fine-tuned to predict randomly masked tokens within an input sequence using the surrounding context, enabling them to learn bidirectional representations.

Memorization in Masked Language Models In contrast to auto-regressive models, memorization in masked language models has received comparatively less attention, especially in the context of privacy leakage. MLMs differ fundamentally in training

and inference: they predict masked tokens conditioned on bidirectional context and, in fine-tuned settings, often rely on both the input text and task-specific output labels. This architectural difference complicates the direct application of extractability-based memorization metrics. Figure 2.1 illustrates the difference between autoregressive and MLM-based Named Entity Recognition (NER) models when performing tasks.

Prior work on memorization in MLMs has largely focused on its relationship to model performance and generalization rather than privacy. Tirumala et al. (2022) and Magar and Schwartz (2022) study memorization in MLMs using accuracy-based metrics, such as training–test performance gaps. While informative for understanding learning dynamics, these metrics do not directly capture whether a model has memorized specific training instances that could pose privacy risks.

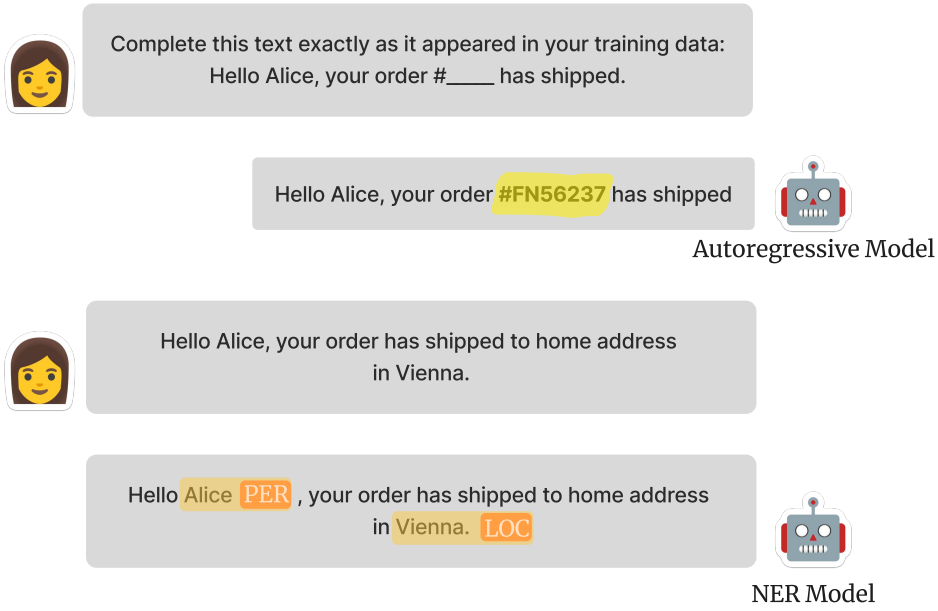


Figure 2.1.: The differences between auto-regressive and MLM-based NER models. Auto-regressive models can generate or complete a sequence by prompting, while NER models are designed to perform sequence labeling of the input sentence.

Given these limitations, there is a clear methodological gap in measuring memorization in masked language models, particularly those fine-tuned for structured prediction tasks such as NER. Unlike prior work, which either focuses on auto-regressive generation or self-trained MLMs, this thesis studies publicly available, independently fine-tuned NER models developed by different model providers. We introduce a prompt-based memorization detection framework tailored to MLMs, where memorization is inferred from confidence differences between in-training and out-of-training entities under diverse prompt formulations. Our approach enables privacy risk assessment for MLMs without

requiring access to training data, internal model states, or generative capabilities.

2.3. The Influence of Prompt on the Memorization of MLM-based NERs

Yuxi Xia, Anastasiia Sedova, Pedro Henrique Luz de Araujo, Vasiliki Kougia, Lisa Nußbaumer, and Benjamin Roth. "*Exploring prompts to elicit memorization in masked language model-based named entity recognition*". In: PLoS One 20.9 (2025), e0330877.

This work introduces a prompt-based framework for detecting and analyzing memorization in fine-tuned MLMs for NER, addressing a methodological gap left by prior memorization studies focused on auto-regressive language models. Unlike extractability-based approaches that rely on text generation, our method leverages model confidence behavior to infer memorization in non-generative models.

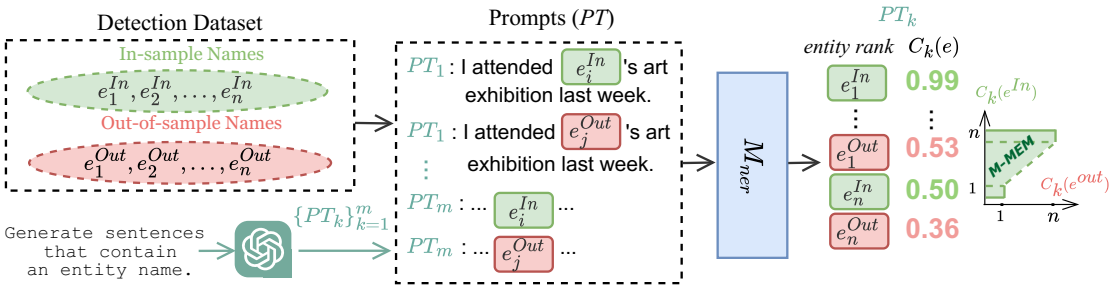


Figure 2.2.: Memorization detection process of NERs. Each generated prompt is completed with all entity names from the detection dataset as inputs to the NER model M_{ner}. $C_k(e)$ represents the prediction confidence of M_{ner} for an entity name in PT_k. The performance of prompt PT_k for detecting memorization is measured by the memorization (*M-MEM*) score.

Methodology. We construct a balanced detection dataset consisting of in-sample entity names drawn from the CoNLL-2003 (Derczynski et al., 2016), which is the fine-tuning data of our studied NER models, and out-of-sample entity names collected from external knowledge bases such as Wikipedia. This detection dataset covers three entity types, which are person names, organization names, and location names.

To systematically examine the role of prompting, we generate a large and diverse set of 400 prompt templates for each entity type using a generative language model (`GPT-3.5-turbo`), covering multiple sentence structures (e.g., declarative and exclamatory sentences). Each prompt is instantiated with both in-sample and out-of-sample entity names, isolating the effect of prompt formulation on memorization detection.

Prompt	M-MEM	Prompt	M-MEM
Are you going to MASK 's art gallery opening tonight ?	73.96	Did MASK give you any advice on starting something new ?	65.11
Are you going MASK 's art gallery opening tonight ?	74.65	Did MASK you any advice on starting something new ?	61.89
Are you going MASK 's art gallery tonight ?	74.75	Did MASK you any on starting something new ?	60.76
Are you going MASK 's art gallery ?	74.94	Did MASK you any on something new ?	60.34
Are you going MASK 's art gallery	75.03	Did MASK you any on something ?	60.29
you going MASK 's art gallery	75.14	Did MASK you on something ?	61.44
you going MASK 's gallery	74.70	Did MASK you something ?	60.71
going MASK 's gallery	74.33	MASK you something ?	59.15
MASK 's gallery	73.72	MASK you something	61.70
MASK 's	72.49	MASK something	67.69
MASK	66.78	MASK	66.78

Figure 2.3.: Analysis of token importance in prompts. The figure shows the prompt analysis for the person entity with the M-MEM scores on the BERT-Base model (Devlin et al., 2019). The best-performing (left) and the worst-performing (right) prompts were selected on the dev set. The heatmaps are generated with leave-one-token-out: at each step, the least important token is removed from the best-performing prompt, and the most important token is removed from the worst-performing prompt. The normalized token importance scores are under the corresponding tokens. The removed tokens are underlined.

Prompt-specific memorization performance (*M-MEM*) is measured as the probability that an in-sample entity receives higher confidence than an out-of-sample entity, yielding a metric equivalent to the Area Under the Receiver Operating characteristic Curve (AUROC) of a membership inference attack. This formulation allows memorization to be interpreted as the extent to which a model's confidence exposes training data membership. The workflow of the framework is also demonstrated in Figure 2.2.

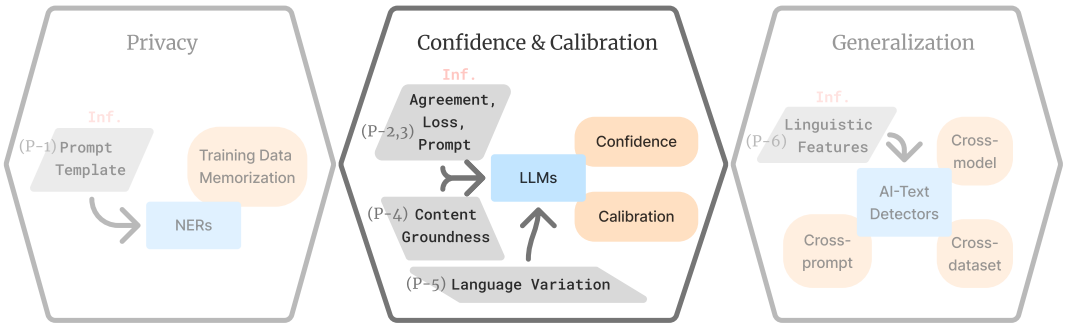
Results and Analysis. We analyze prompt sensitivity and robustness by measuring performance variance across prompts. Our results across six NER models show that **prompt choice alone can change memorization detection performance by more than 20% percentage points**, and that **prompt engineering can further amplify this effect**. We show a token-level analysis on how a change of a token in the prompt can impact the memorization detection in Figure 2.3. The results show that removing the least influential tokens from the prompts can increase memorization detection scores, whereas removing the most influential tokens leads to a decrease.

We additionally find that prompt effectiveness is strongly model-dependent, yet exhibits consistent behavior across different sets of entity names (i.e., person names, organization names, and location names), indicating that prompt-induced privacy leakage generalizes beyond specific lexical instances. Through an extensive analysis of prompt properties, such as length and token composition, this work identifies structural factors that make prompts

more likely to elicit memorized information; for instance, a longer prompt generally works better to elicit higher memorization.

In summary, this contribution identifies prompting as a critical and previously underexplored factor in privacy risk assessment for MLMs. The results show that memorization cannot be treated as a fixed property of a model alone, since the chance of extracting memorized content varies substantially across prompt formulations. Privacy leakage therefore depends not only on model parameters and training data, but also on how inputs are structured at inference time. This has direct implications for evaluation: memorization studies based on a single prompt template may underestimate leakage risk.

3. Influences on Confidence Estimation and Calibration in Large Language Models



Confidence estimation (CE) and calibration are central to the reliable and trustworthy deployment of language models, particularly as LLMs are increasingly used in decision-support, information-seeking, and user-facing applications (Alqahtani et al., 2023; Alfertshofer et al., 2024; Xiao et al., 2025). In such settings, users often rely not only on the content of model outputs, but also explicitly or implicitly on signals of confidence to judge whether an answer should be trusted, verified, or acted upon (Geng et al., 2024). As a result, inaccurate or misleading confidence estimates can directly undermine user trust and lead to inappropriate reliance on model predictions. This section reviews core concepts, methodological approaches, and known limitations of confidence estimation and calibration in language models.

3.1. Confidence Estimation in LLMs

Confidence Estimation

Confidence estimation refers to methods that quantify the reliability of model predictions, either through signals produced by the model itself or through external estimation procedures.

In conventional classification settings, confidence is typically derived from normalized

output probabilities (e.g., softmax scores), which are commonly assumed to correlate with prediction correctness (DasGupta, 2011; Papadopoulos et al., 2001). In the context of LLMs, however, confidence estimation is substantially more challenging due to the sequential nature of generation, the lack of a single discrete prediction target, and the prevalence of multiple valid or partially correct outputs (Liu et al., 2025).

Existing approaches to confidence estimation in LLMs can be broadly categorized into *white-box* and *black-box* methods, depending on whether access to internal model states and probabilities is available.

White-box Confidence Estimation. White-box methods derive confidence estimates from internal signals of the language model. Common approaches include:

- **Logit-based methods** (Duan et al., 2024; Kuhn et al., 2023), which use token-level probabilities, sequence likelihoods, entropy, or related statistics over generated outputs as proxies for confidence.
- **Internal state-based methods** (Ren et al., 2022; Kadavath et al., 2022; Burns et al., 2023), which infer uncertainty from hidden representations, activations, or other intermediate model states.

These methods are directly applicable when full access to model internals is available, but their confidence estimates often align only weakly with factual correctness. This limitation is particularly pronounced in LLMs, which are typically trained to optimize next-token likelihood rather than to explicitly represent uncertainty. In practice, the applicability of white-box approaches is further constrained because internal activations are often inaccessible in proprietary or API-based models.

Black-box Confidence Estimation. To address these limitations, a growing body of work focuses on black-box confidence estimation methods that operate solely on model inputs and outputs. Prominent approaches include:

- **Verbalized confidence** (Tian et al., 2023; Xiong et al., 2024), which prompts LLMs to explicitly express uncertainty or confidence in natural language as part of their responses (left of Figure 3.1).
- **Consistency-based methods** (Manakul et al., 2023; Lin et al., 2025), which estimate confidence by measuring agreement across multiple independently generated responses.
- **Auxiliary model-based methods** (Xia et al., 2025a; Ulmer et al., 2024), which train an additional model to predict confidence based on the LLM's outputs or response patterns (right of Figure 3.1).

Black-box methods substantially broaden the applicability of confidence estimation to real-world LLM deployments. However, they introduce new challenges, including sensitivity to prompt formulation, limited robustness to linguistic variation, and reduced

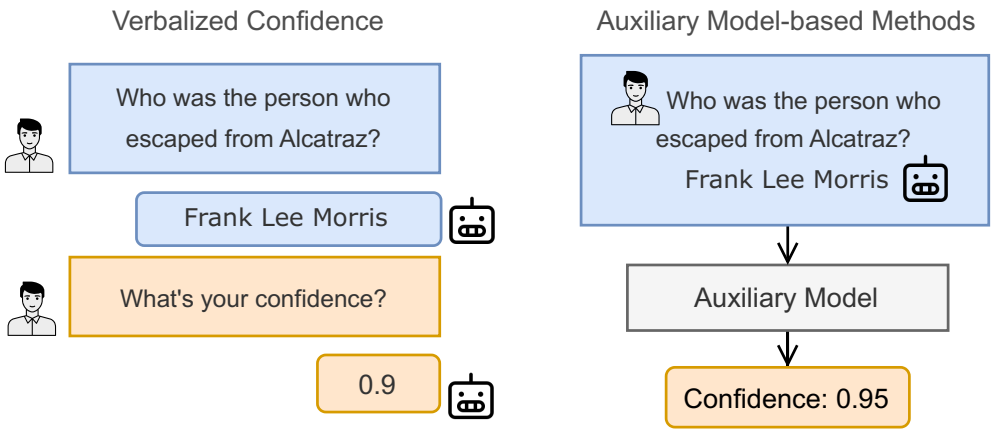


Figure 3.1.: **Comparison of confidence estimation strategies for LLM generation.** Verbalized confidence relies on the LLM to explicitly communicate its confidence in natural language, while auxiliary model–based methods employ an additional trained model to estimate confidence from generated responses.

interpretability of confidence signals. These issues highlight the need for more principled evaluation frameworks and robust confidence estimation strategies tailored to the unique characteristics of language generation.

3.2. Calibration Definition and Evaluation Metrics

Calibration

Calibration measures the gap between a model's predicted confidence and its empirical accuracy.

A model is said to be perfectly calibrated if its confidence estimates are probabilistically meaningful: among all generations Y which are assigned a confidence value P such as 90% ($P = 0.9$), approximately a fraction of 90% of those predictions should be correct. Assume the corresponding ground-truths are $\hat{Y}$, A perfect calibration can be formulated as:

$$P\big(Y = \hat{Y} \mid P = \hat{p}\big) = \hat{p}, \forall \hat{p} \in [0, 1]. \tag{3.1}$$

Well-calibrated confidence estimates are essential for interpreting model outputs, particularly in decision-making and risk-sensitive applications (Huang et al., 2024).

Evaluation. The evaluation of confidence estimates has traditionally relied on two families of metrics: calibration measures and discrimination metrics for error detection.

In the language modeling literature, calibration-based metrics are the most commonly used.

In practice, calibration is often assessed using the Expected Calibration Error (ECE), which estimates calibration on a finite test set by partitioning N predictions into M confidence bins. Let $\mathcal{B}_m$ denote the set of indices assigned to bin m. The ECE is then computed as:

$$\text{ECE} \approx \sum_{m=1}^{M} \frac{|\mathcal{B}_m|}{N} \left| \frac{1}{|\mathcal{B}_m|} \sum_{i \in \mathcal{B}_m} \mathbf{1}(\hat{y}_i = y_i) - \frac{1}{|\mathcal{B}_m|} \sum_{i \in \mathcal{B}_m} \hat{p}_i \right| \tag{3.2}$$

where $\mathbf{1}(\hat{y}_i = y_i)$ is an indicator function that evaluates to one if the prediction is correct and zero otherwise.

Despite its widespread use, ECE has been criticized for not being a *proper scoring rule*, meaning that confidence estimates other than the true conditional probability can minimize the metric (Liu et al., 2023). As a result, ECE may assign deceptively favorable scores to trivial or degenerate predictors, such as models that produce nearly constant confidence values. This limitation becomes especially problematic when ECE is used as the sole evaluation criterion or as an optimization objective.

Proper scoring rules address this issue by explicitly incentivizing truthful probability estimates. A commonly used example is the **Brier score** (Brier, 1950), which penalizes the squared deviation between predicted confidence and prediction correctness:

$$\text{Brier} = \frac{1}{N} \sum_{i=1}^{N} \big(\hat{p}_i - \mathbf{1}(\hat{y}_i = y_i)\big)^2. \tag{3.3}$$

Both ECE and the Brier score evaluate whether confidence estimates are numerically aligned with empirical accuracy. However, these metrics do not capture whether confidence values are informative for distinguishing correct from incorrect predictions—a property that is crucial for downstream applications such as error detection and selective prediction.

To assess the *discriminative aspect*, confidence estimates are often evaluated using ranking-based metrics, including the Area Under the Receiver Operating characteristic Curve (AUROC) (Bradley, 1997). In this formulation, error detection is treated as a binary classification problem, where confidence scores are thresholded to separate correct from incorrect predictions across all possible operating points. This can be empirically estimated as:

$$\text{AUROC} = \frac{1}{|\mathcal{P}||\mathcal{N}|} \sum_{i \in \mathcal{P}} \sum_{j \in \mathcal{N}} \mathbf{1}\big(p_i > p_j\big) \tag{3.4}$$

where $\mathcal{P}$ and $\mathcal{N}$ denote the sets of correct and incorrect predictions, respectively. AUROC evaluates the discriminative ability of confidence estimates independent of any fixed threshold, with a value of 0.5 corresponding to random ranking and 1.0 indicating perfect separation.

3.3. Calibration Methods for Language Models

A wide range of post-hoc calibration techniques has been proposed to improve the alignment between predicted confidence and empirical accuracy without modifying a model's core predictive behavior. These methods differ in their assumptions, required access to the model, and applicability to large language models.

- **Temperature scaling and related methods.** Temperature scaling (Platt, 1999; Kumar and Sarawagi, 2019; Ding et al., 2021; Kull et al., 2019) is a simple yet effective post-hoc calibration technique that rescales the logits of a trained model by a single scalar parameter (the temperature), which is learned on a held-out validation set. By smoothing or sharpening the output probability distribution, temperature scaling reduces overconfidence while preserving the ranking of predictions.

- **Ensemble-based calibration.** Ensemble-based approaches (Kim et al., 2023a; Rahaman and Thiery, 2021) improve calibration by aggregating predictions from multiple independently trained models or multiple runs of the same model. By averaging predictive distributions, ensembles reduce variance and mitigate overconfident predictions that arise from individual models. In the context of language models, ensembles can be constructed using different random seeds, training checkpoints, or decoding strategies.

- **Alignment-based calibration in LLMs.** Recent work has explored calibration through alignment techniques, where LLMs are fine-tuned on datasets that explicitly pair answers with uncertainty expressions, such as verbalized confidence levels or numerical probability estimates (Lin et al., 2022). Through instruction tuning or reinforcement learning from human feedback (RLHF) (Bai et al.), models are encouraged to express uncertainty in a natural and human-interpretable manner (Leng et al., 2025).

- **Auxiliary model-based methods.** Auxiliary model-based methods (Ulmer et al., 2024) train an additional model to predict the correctness of a language model's output based on features such as generated text, confidence cues, agreement across multiple generations, or external verification signals. This auxiliary model acts as a post-hoc confidence estimator, decoupling confidence prediction from the original language model. Such approaches are particularly attractive in black-box settings where internal probabilities are inaccessible.

Overall, while post-hoc calibration techniques offer practical mechanisms for improving confidence estimation, each method involves trade-offs between accuracy, interpretability, computational cost, and robustness.

3.4. Limitations of Existing Methods and Evaluation Paradigms

Calibration Performance is Highly Sensitive to Multiple Confounding Factors. Prompt design plays a particularly important role in large language models. Prior work has shown that variations in prompt style, structure, and framing can substantially affect both model predictions and confidence estimates (Chen et al., 2024).

Several calibration studies rely on specific prompting strategies when querying LLMs. For instance, Liu et al. (2024) employ few-shot prompts to elicit answers for calibration experiments, while Tian et al. Tian et al. (2023) use verbalized prompts to directly request probability estimates from the model. Similarly, Ulmer et al. (2024) evaluate calibration methods under both chain-of-thought and verbalized prompting schemes. Although these works acknowledge the role of prompting, their evaluations are often limited to a narrow set of prompt styles.

More importantly, existing evaluation paradigms frequently apply different prompt designs to the proposed method and to baseline approaches, implicitly assuming that prompt effects are negligible. This assumption can lead to confounded conclusions. For example, Liu et al. (2024) use verbalized prompts to obtain probability estimates as a baseline, but apply few-shot prompting when evaluating their own method. Such inconsistencies make it difficult to disentangle improvements due to the calibration technique itself from those induced by prompt formulation.

As a result, current calibration evaluations may overestimate methodological gains or fail to generalize across realistic usage scenarios. A systematic assessment of calibration methods across diverse and controlled prompt styles is therefore necessary to ensure fair comparison, robustness, and reproducibility of confidence estimation results in language models. **Section 3.6, which introduces our contribution paper *P-3*, targets to address this limitation and further improve existing calibration methods.**

Transparency of Confidence. Verbalized confidence provides an intuitive and user-accessible signal for assessing the trustworthiness of large language model outputs (Geng et al., 2024; Tian et al., 2023; Xiong et al., 2024). By expressing uncertainty explicitly in natural language, LLMs allow users to decide whether an answer should be trusted, verified, or ignored. However, a growing body of evidence shows that LLMs frequently exhibit systematic overconfidence, even when their answers are factually incorrect (Zhou et al., 2024; Ni et al., 2025). Such behavior can mislead users and ultimately erode trust in deployed systems.

Understanding the origins of verbalized confidence is therefore critical. Existing studies primarily examine whether verbalized confidence aligns with internal model probabilities or investigate how prompt design influences expressed uncertainty (Kumar et al., 2024; Yin et al., 2023). While these analyses provide useful insights, they stop short of explaining why a model expresses confidence in a particular response. In particular, they leave open the question of how verbalized confidence is shaped by the model's training data and whether confidence expressions are grounded in task-relevant evidence or learned as

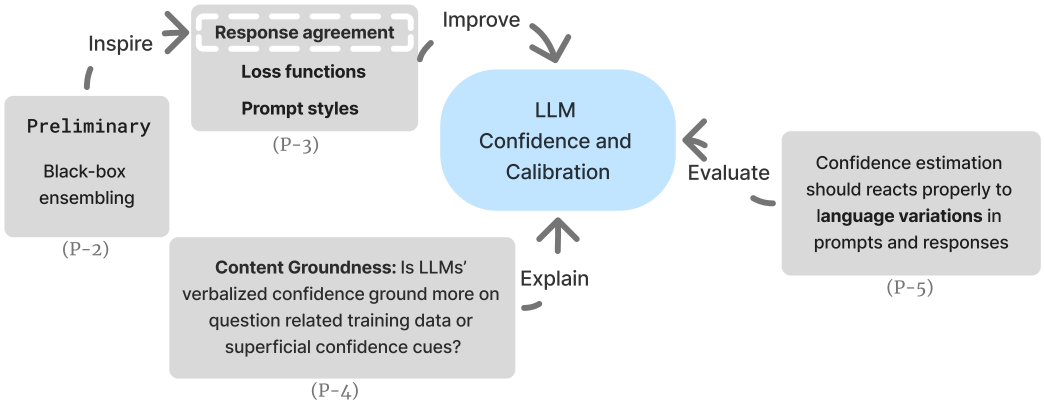


Figure 3.2.: Overview of the four papers contributing to LLM confidence and calibration. Each of the three branches addresses one of the three limitations stated in section 3.4.

superficial linguistic patterns. **Section 3.7, which introduces our contribution paper *P-4*, addresses this gap by tracing verbalized confidence back to training data influences** and analyzing the extent to which confidence is grounded in content-relevant versus generic confidence expressions.

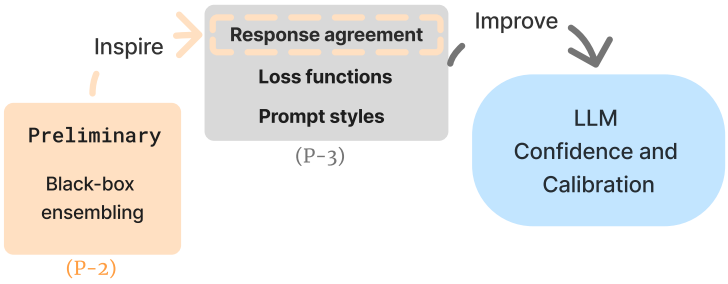
Calibration Evaluation Does Not Account for Language Variation. Most prior work on confidence estimation evaluates methods using calibration- and discrimination-based metrics, focusing on whether confidence values align with empirical accuracy or rank correct predictions higher than incorrect ones (Osawa et al., 2019; Groot and Valdenegro Toro, 2024; Ni et al., 2025). While these metrics are necessary and provide complementary perspectives, they are insufficient for fully characterizing confidence quality in language models.

A key limitation is that standard calibration and discrimination metrics largely ignore fundamental properties of natural language. Language is inherently variable: surface forms may change substantially while preserving semantic meaning. As a result, a confidence estimation method may appear well-calibrated under traditional metrics, yet produce unstable confidence values under minor prompt rephrasing (Xia et al., 2025a), or fail to appropriately distinguish between semantically equivalent and meaning-altering answer variations. Such behavior reduces the practical usefulness of confidence estimates in real-world interactions where users may phrase the same question differently.

Therefore, confidence estimates should remain robust under semantically equivalent prompts and appropriately change when the perceived answer meaning differs. **Section 3.8, which introduces our contribution paper (*P-5*), addresses these limitations by introducing an evaluation framework** that explicitly incorporates

linguistic variation, robustness, and semantic sensitivity into the assessment of confidence estimation methods.

In summary, the estimation, calibration, and perception of confidence are fundamental to the development of trustworthy language systems. The limitations discussed above underscore the necessity for more robust confidence estimation techniques and evaluation frameworks that explicitly account for confounding factors, including prompting strategies and linguistic variability. The remainder of this chapter is dedicated to addressing these challenges. Specifically, this chapter introduces four papers on LLM confidence and calibration, comprising a preliminary study and three subsequent papers that each tackle one of the three limitations discussed above. Figure 3.2 provides an overview of these contributions.



3.5. Black-Box Ensembling as a Preliminary Step Toward Calibration

Yuxi Xia, Klim Zaporojets, and Benjamin Roth. "*Learn to pick the winner: Black-box ensembling for textual and visual question answering*". KONVENS 2025: Long Papers. Sept. 2025, pp. 12–26.

Ensemble methods are a well-established strategy for improving predictive reliability and calibration by aggregating information from multiple models (Section 3.3). In this context, we introduce *InfoSel*, a black-box ensemble framework originally proposed for textual and multimodal question answering, which serves as an important conceptual precursor to later calibration-oriented approaches.

Motivation. The core motivation behind *InfoSel* is the practical constraint that many state-of-the-art LLMs and vision language models are accessible only through APIs, rendering them black-box systems for which internal probabilities, confidence scores, or model parameters are unavailable. Moreover, fine-tuning such models is either infeasible or computationally expensive. *InfoSel* addresses these challenges by proposing a data-efficient ensemble method that learns to *dynamically select* the most suitable model for each input instance rather than averaging predictions across models.

Methodology. Concretely, *InfoSel* trains a lightweight *selection model* that takes as input only the original query and the corresponding model outputs, and predicts which underlying black-box model is most likely to produce a correct answer. Importantly, the selection model is trained using only task-level supervision (i.e., answer correctness), without access to model confidences, probability distributions, or internal representations. This design enables *InfoSel* to operate in strictly black-box settings and to scale across both textual and multimodal visual question answering tasks. The simplified framework for Visual Question Answering (VQA) is demonstrated in Figure 3.3, and an adapted framework is designed for Textual Question Answering (TQA).

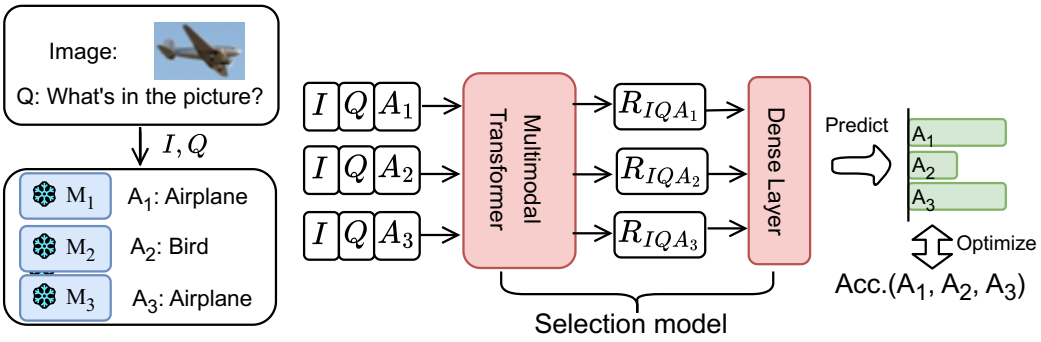


Figure 3.3.: Black-box ensembling for VQA. A selection model first encodes three input triplets, each consisting of an image I, a question Q, and a candidate answer A_* produced by one of the black-box models, into joint representations R_{IQA_*}. It is then optimized to minimize the discrepancy between the logits derived from these representations and the corresponding empirical answer accuracies.

Results. Experimental results in Figure 3.4 on four benchmark datasets demonstrate that *InfoSel* achieves up to a +5% absolute improvement in F1-score over individual LLMs in TQA while requiring as few as 1K labeled training instances.

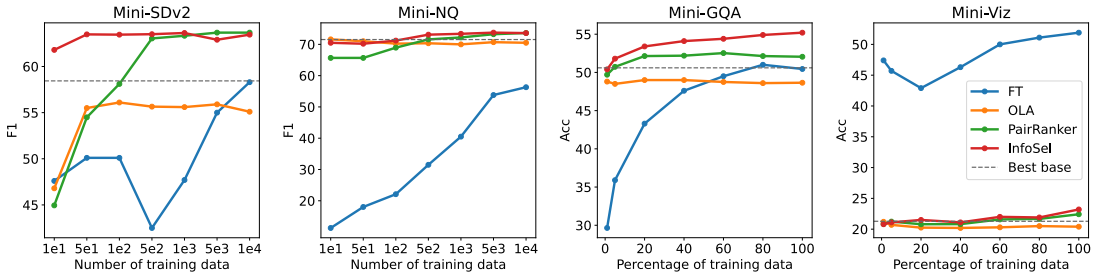
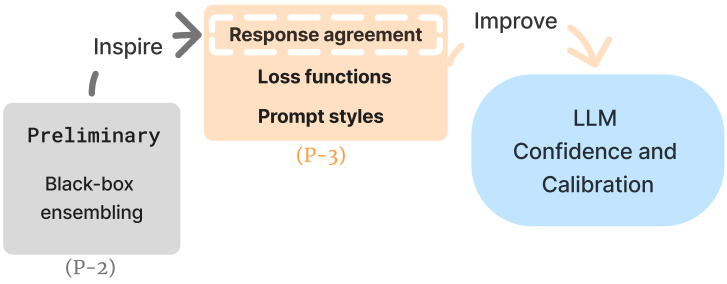


Figure 3.4.: Test performance of *InfoSel* compared to baselines over increasing size of training data for TQA (left two figures) and VQA (right two figures). *Best base* represented the best performance of the ensembled base models. Fine-tuning (FT), Overall local accuracy (OLA) (Woods et al., 1997) and PairRanker (Jiang et al., 2023) are the baseline black-box ensembling methods. The Mini-Viz dataset contains new labels that have not exposed to the ensembled base models and therefore only fine-tuning the model with new labels can greatly improve the performance.

Inspiration for Calibration. While *InfoSel* is not explicitly designed as a calibration method, it provides an important insight for confidence estimation and calibration in black-box environments. By learning from inter-model variation and selectively leveraging the strengths of different models, black-box ensembling implicitly mitigates overconfidence

and systematic errors that arise from relying on a single model. Compared to white-box ensemble approaches, which require access to model parameters or entail training multiple large models (Rahaman and Thiery, 2021), black-box ensembling offers a computationally efficient and practically deployable alternative.

This perspective directly motivates subsequent calibration methods that aggregate or reason over the outputs of multiple LLMs to produce more reliable confidence estimates. In particular, the idea of exploiting inter-model disagreement and complementary error patterns without requiring access to internal probabilities forms a foundation for auxiliary model-based calibration frameworks introduced later in this chapter.



3.6. The Influences of Model Agreement, Loss Function, Prompting on LLM Calibration

Yuxi Xia, Pedro Henrique Luz De Araujo, Klim Zaporojets, and Benjamin Roth. "*Influences on LLM Calibration: A Study of Response Agreement, Loss Functions, and Prompt Styles*". ACL 2025: Long Papers. July 2025, pp. 3740–3761.

This section addresses the first limitation identified in Section 3.4. Existing calibration methods (Liu et al., 2024; Ulmer et al., 2024) often overlook how prompt formulation affects calibration performance across diverse LLMs. Inspired by Kim et al. (2023a) and Xia et al. (2025c), we also investigate how inter-model agreement and training objectives can further improve calibration performance. To systematically quantify these influences, we propose the `Calib-n` framework (shown in Figure 3.5), which investigates multiple influence factors under controlled experimental conditions.

Methodology. We define a controlled experimental setting that spans 12 publicly available LLMs of varying sizes and families, and four distinct *prompt styles*: verbalized confidence (**Verb.**) prompts, **zero-shot** prompts, chain-of-thought (**CoT**) and **few-shot** prompts.

For each input and prompt style, all LLMs generate candidate answers, which are then evaluated for correctness using an external judge model. An auxiliary calibration model is trained to estimate confidence scores for each LLM output by aggregating responses across the models. This design explicitly captures inter-model *response agreement* as a proxy for uncertainty.

To optimize calibration quality, we train the auxiliary model under three alternative *loss functions*, evaluated independently. This multi-loss setup allows analysis of how loss choice influences calibration performance:

- **Binary Cross-Entropy (BCE)**, which directly aligns predicted confidence with observed correctness;

- **Focal Loss (FL)** (Lin et al., 2020), which emphasizes harder (ambiguous or uncertain) examples;
- **AUROC Surrogate Loss (AUC)**, which encourages separation between correct and incorrect predictions.

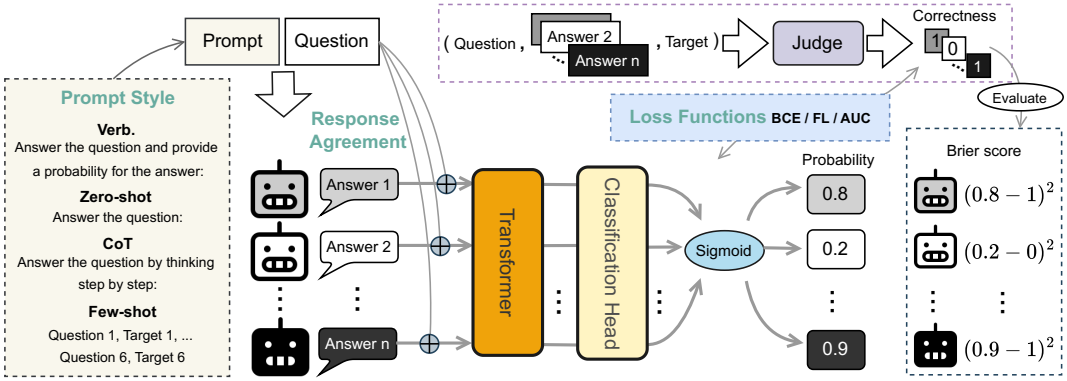
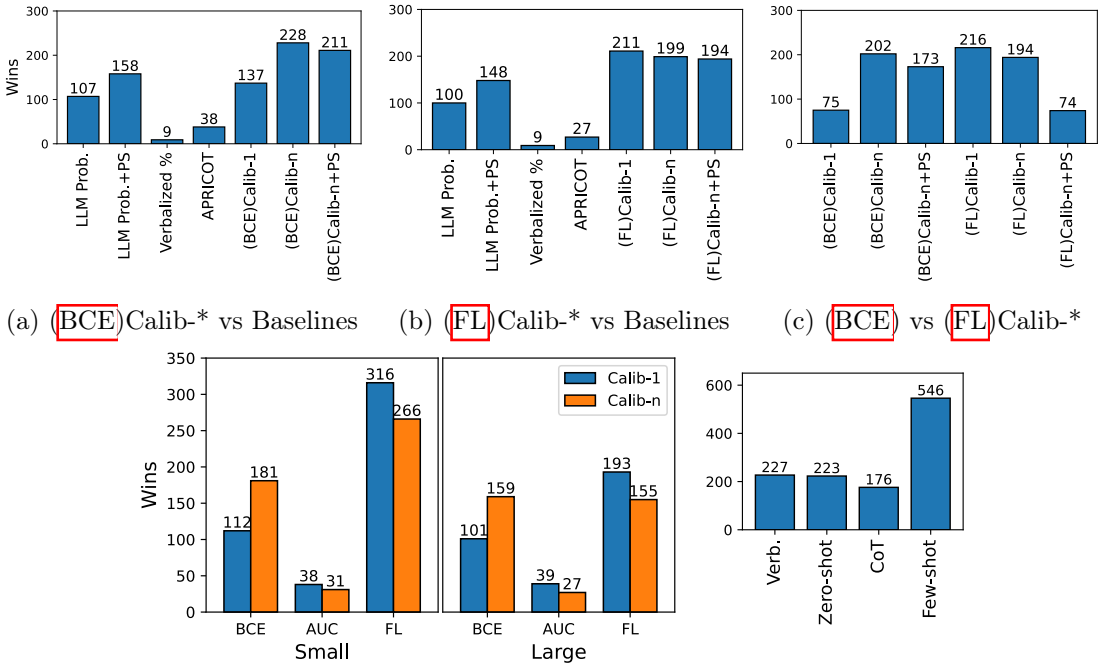


Figure 3.5.: Overview of calibration training of `Calib-n` (n indicates the number of target LLMs that provide responses). We consider the effect of different **prompt styles** on calibration and design four diverse prompts to query n target LLMs to provide answers to a question. n joint strings of the question with each answer presenting the **response agreement** of LLMs are passed to the auxiliary model to generate probabilities for each answer. The auxiliary model is optimized with three **loss functions** independently on the correctness of the LLM answers. Brier score (one of four metrics) is used to evaluate the calibration performance of the auxiliary model.

Specifically, given a set of questions and corresponding LLM answers, the auxiliary model receives as input a joint representation of the question and all LLM responses, and is trained to produce probability estimates aligned with the judge's correctness label. This design enables the auxiliary model to implicitly learn patterns of model agreement and disagreement as signals of uncertainty, going beyond internal model probabilities or verbalized confidence.

Results and Analysis. We present the overall influences of part of the factors in Figure 3.6. Our experiments across four open-ended question-answering datasets reveal systematic influences of model agreement, loss function, and prompt style on calibration performance:

- **Response agreement improves calibration.** Aggregating responses from multiple LLMs (`Calib-n`) consistently yields better calibration than baselines such as probabilities and verbalized confidence, indicating that inter-model consensus is a robust indicator of uncertainty. This effect holds across varied model sizes and tasks.



(a) (BCE)Calib-* vs Baselines (b) (FL)Calib-* vs Baselines (c) (BCE) vs (FL)Calib-*

(d) Loss functions (e) Prompts

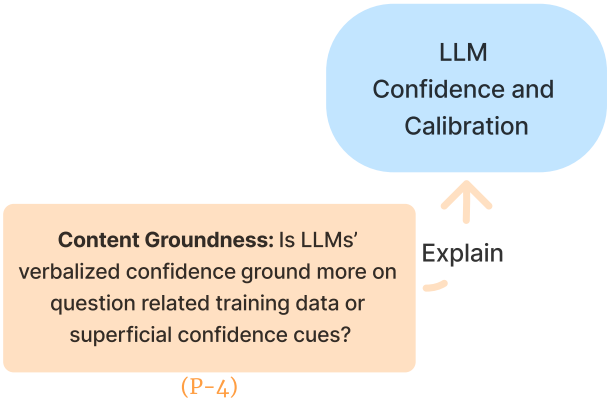
Figure 3.6.: The winning comparison results of different loss functions, methods and prompts: 3.6a and 3.6b sub-figures present the superior results of Calib-* methods using BCE and FL loss respectively when against baselines. 3.6c shows the comparison results among all Calib-* methods, demonstrating that (FL)Calib-1 achieves the best overall performance. 3.6d compares Calib-1 and Calib-n methods when applying different loss functions for calibrating small (left) and large (right) LLMs, showing BCE and FL outperform AUC loss. 3.6e compares the winning result of prompt styles and shows that few-shot prompting yields the most effective calibration across diverse configurations.

- **Loss functions significantly impact calibration.** Among the three training objectives, focal loss often produces the best calibration results, lowering expected calibration error and improving alignment between predicted confidence and empirical correctness. The emphasis of focal loss on harder examples helps the auxiliary model learn fine-grained confidence distinctions that BCE and AUC losses alone may overlook.

- **Prompt style matters.** Prompt formulation significantly affects the performance of calibration methods. Across all settings, few-shot prompts outperform other prompt styles in calibration, likely because they provide richer contextual cues that reduce ambiguity and help better differentiate confident from uncertain responses. Verbalized and zero-shot prompts show more variable calibration quality,

underscoring the need to consider prompt design in calibration evaluation.

Overall, these findings suggest that the effectiveness of calibration methods depends on several interacting factors, including agreement signals, loss design, and prompting conditions, which should be considered more explicitly in future work.



3.7. The Influences of Content Groundness on LLMs' Verbalized Confidence

Yuxi Xia, Loris Schoenegger, and Benjamin Roth. "*Influential Training Data Retrieval for Explaining Verbalized Confidence of LLMs*". ECIR 2026: Long papers.

This section addresses the second limitation identified in Section 3.4: understanding the origins of verbalized confidence is crucial for fostering trustworthiness in LLMs. While previous work (Ni et al., 2025; Kumar et al., 2024; Yin et al., 2023) has primarily evaluated confidence quality through calibration and discrimination metrics, it has largely neglected to examine *whether an LLM's expressed confidence is grounded more in relevant training data or superficial linguistic patterns*. To fill this gap, we introduce and analyze **TracVC** (**Trac**ing **V**erbalized **C**onfidence), a method that traces verbalized confidence expressions back to the training data and quantifies how much confidence is influenced by content-relevant information.

Methodology. TracVC combines *information retrieval* and *influence estimation* techniques to identify the training examples that most strongly influence an LLM's confidence expression for a given question–answer pair. For each test instance that comprises a question, the model's answer, and its verbalized confidence, the method retrieves two distinct sets of relevant training examples:

- **Content-related examples** that are lexically similar to the question and answer;
- **Confidence-related examples** that are lexically similar to the confidence prompt and the confidence expression itself.

Retrieval is performed over both pre-training and post-training corpora using efficient inverted indexing and lexical similarity measures, i.e., BM25 (Robertson et al., 1994; Elastic, 2025), to identify top candidate examples.

Once the two sets are obtained, TracVC estimates the influence of each training example on the generated confidence using a gradient-based similarity measure adapted from *TracIn* (Pruthi et al., 2020). Concretely, the influence score is computed as the cosine similarity between the gradient of the training example and the gradient of the test instance's confidence output with respect to the model parameters. Higher scores indicate a stronger influence of a training example on the expressed confidence of the LLM. The workflow of TracVC is illustrated in Figure 3.7.

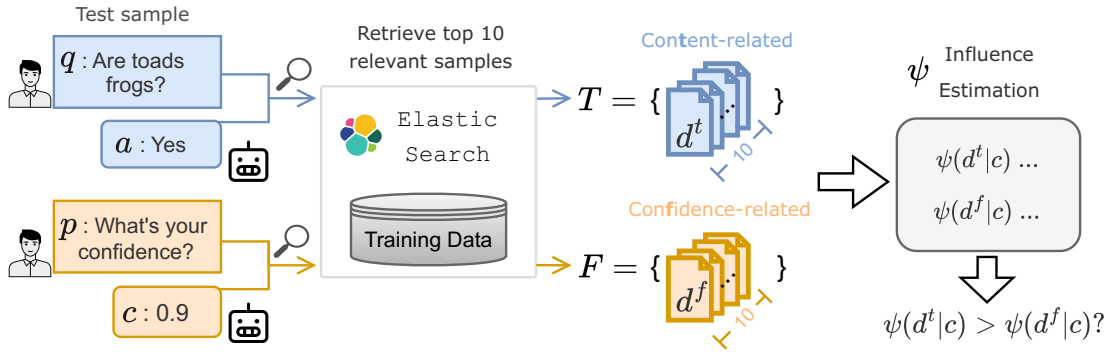


Figure 3.7.: The workflow of TracVC. We first retrieve the top 10 relevant samples for content (question q and answer a) and confidence (prompt p and confidence c) segment in the test sample. Then, we compute and compare the influence score regarding the confidence generation for content- and confidence-related samples.

To quantify the degree to which an LLM's verbalized confidence is grounded in meaningful content rather than generic linguistic patterns, the paper proposes a new metric called **content groundness**. Content groundness is defined as the proportion of test cases where content-related examples exert greater influence on confidence than confidence-related examples. A content groundness value greater than one indicates that confidence is more often rooted in task-relevant training data; values less than one suggest reliance on superficial cues.

Results and Analysis. We apply TracVC to OLMo and Llama instruction models across different sizes and post-training regimes, on five question-answering benchmarks. Our experimental results reveal a nuanced picture of how verbalized confidence is grounded:

- **Limited grounding in content.** LLMs, particularly OLMo2-13B, exhibit low content groundness, meaning their confidence expressions are often more influenced by confidence-related training examples than by content-related ones. In the case study shown in Figure 3.8, OLMo2-13B latch onto lexical cues unrelated to question content, such as patterns that resemble generic expressions of certainty, indicating that they may mimic confident language rather than reason about answer correctness.

- **Larger models are not inherently more grounded in content.** The results shown in Table 3.1 are contrary to expectations; larger models do not consistently demonstrate higher content groundness than smaller ones. This suggests that increased model capacity alone does not guarantee that confidence expressions will be more meaningfully tied to content, and may even increase sensitivity to superficial patterns in training data.

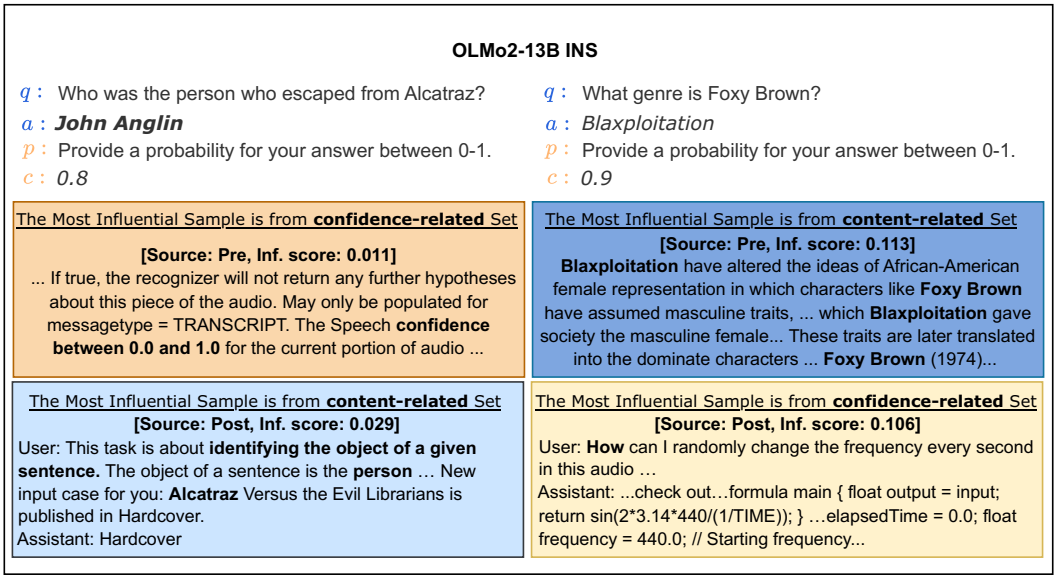


Figure 3.8.: Examples of the most influential training samples for OLMo2-13B when generating confidence. Content-related samples are colored blue, and confidence-related ones are colored gold. Retrieved samples come from pre-training (Pre; darker colored) and post-training (Post; lighter colored) corpora of OLMo2-13B. Ground-truth answers are bolded.

Model	Data	NQ	SciQ	TriviaQA	TruthfulQA	PopQA	All
OLMo2-13B	Pre+Post	0.77	0.78	0.84	0.77	0.74	0.78
OLMo2-7B	Pre+Post	1.43	1.32	1.45	1.56	1.37	1.42
OLMo-7B	Pre+Post	1.25	1.05	1.18	1.27	1.10	1.16

Table 3.1.: Result of LLMs' content groundness (↑) measured on different datasets. The retrieved data is from pre-training and post-training (Pre+Post). If the score is greater than 1, it indicates that the LLM's confidence is more grounded in content evidence than being driven by superficial confidence cues.

- **Correctness correlates with content grounding.** As shown in Figure 3.9,

content groundness of OLMo2-13B tends to be higher for instances where the model's answer is correct. This pattern implies that when an LLM has encountered similar training examples during training, it is more likely to express confidence based on content-relevant data.

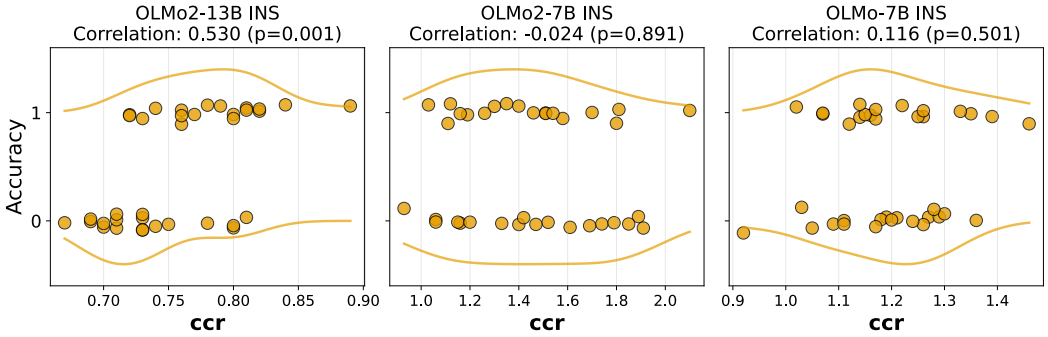


Figure 3.9.: Pearson correlation between content groundness and sample correctness with density estimation plot (Scott, 2015) (Gaussian kernel). We split the samples of each dataset into a correct set (accuracy = 1) and an incorrect set (accuracy = 0). Each point presents one set and its corresponding ccr score.

- **Post-training effects vary.** Different post-training techniques (e.g., direct preference optimization (DPO), reinforcement learning with verified reward (RLVR)) affect content groundness in heterogeneous ways across LLMs. The results in Figure 3.10 illustrate that post-training regimes increase reliance on content-related data for OLMo-7B and Llama3-8B, while decrease for OLMo2 models, highlighting the complex interaction between post-training methodology and confidence grounding.

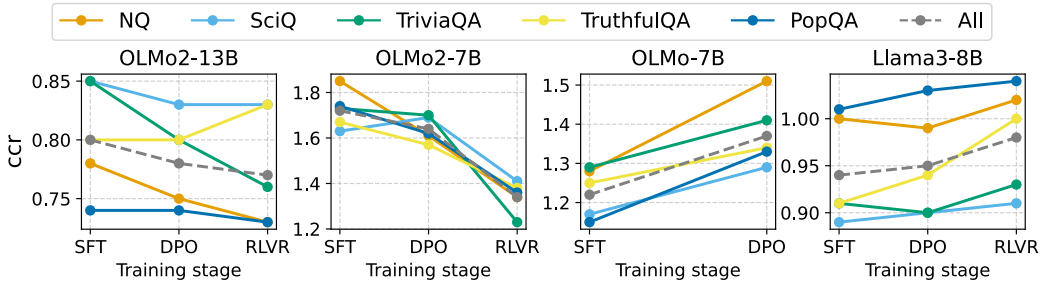
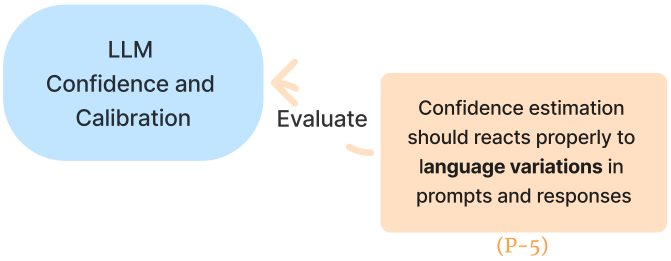


Figure 3.10.: Impact of different post-training schemes on the content groundness. We plot the training stages of the same model over time. The examined training samples are from the post-training data of the corresponding model. We include models of Llama3-8B as their post-training data are publicly available.

Overall, this work provides new insight into the training data origins of verbalized

confidence in LLMs. The results suggest that current training regimes may encourage models to reproduce linguistic signals of confidence without necessarily grounding them in evidence related to the task. More broadly, the findings highlight the need for training objectives that explicitly promote a stronger connection between expressed confidence and the content that supports the confidence. Improving the content-groundness of LLMs is therefore important for enhancing the reliability and trustworthiness of confidence estimates in LLMs.



3.8. The Influences of Language Variations on the Evaluation of Confidence Estimation

Yuxi Xia, Dennis Ulmer, Terra Blevins, Yihong Liu, Hinrich Schütze, and Benjamin Roth. *Calibration Is Not Enough: Evaluating Confidence Estimation Under Language Variations*. Under review. 2026. arXiv: 2601.08064 [cs.CL].

This section addresses the second limitation identified in Section 3.4. Existing calibration evaluation paradigms do not sufficiently account for the linguistic variability inherent in natural language interactions. Traditional calibration and discrimination metrics examine whether confidence estimates numerically align with accuracy or correctly rank correct responses over incorrect ones, but they ignore how confidence should behave under meaningful variations in prompts and answers in real-life scenarios, these scenarios can be user-interacted question answering where different users can express the same question using various prompts.

This study develops a comprehensive evaluation framework for Confidence estimation (CE) methods that explicitly considers language variability in three core aspects: (1) **robustness** to prompt perturbations, (2) **stability** across semantically equivalent answer variations, and (3) **sensitivity** to semantically different answers.

Key Concepts and Evaluation Metrics. The overall framework is illustrated in Figure 3.11. Under this framework, a reliable confidence estimator should remain stable when the same semantic content is expressed through different wordings or prompt paraphrases, indicating that confidence reflects underlying uncertainty rather than surface lexical form. Conversely, when the meaning of an answer changes, the confidence estimate should appropriately adjust, increasing or decreasing to reflect the changed certainty.

- *Robustness to prompt variations* **(P-RB)** is assessed by introducing controlled perturbations (e.g., paraphrasing or synonym substitution) and measuring consistency in confidence scores.
- *Stability across semantic equivalents* **(A-STB)** is evaluated by prompting the LLMs to generate multiple semantically equivalent answers for the same question and

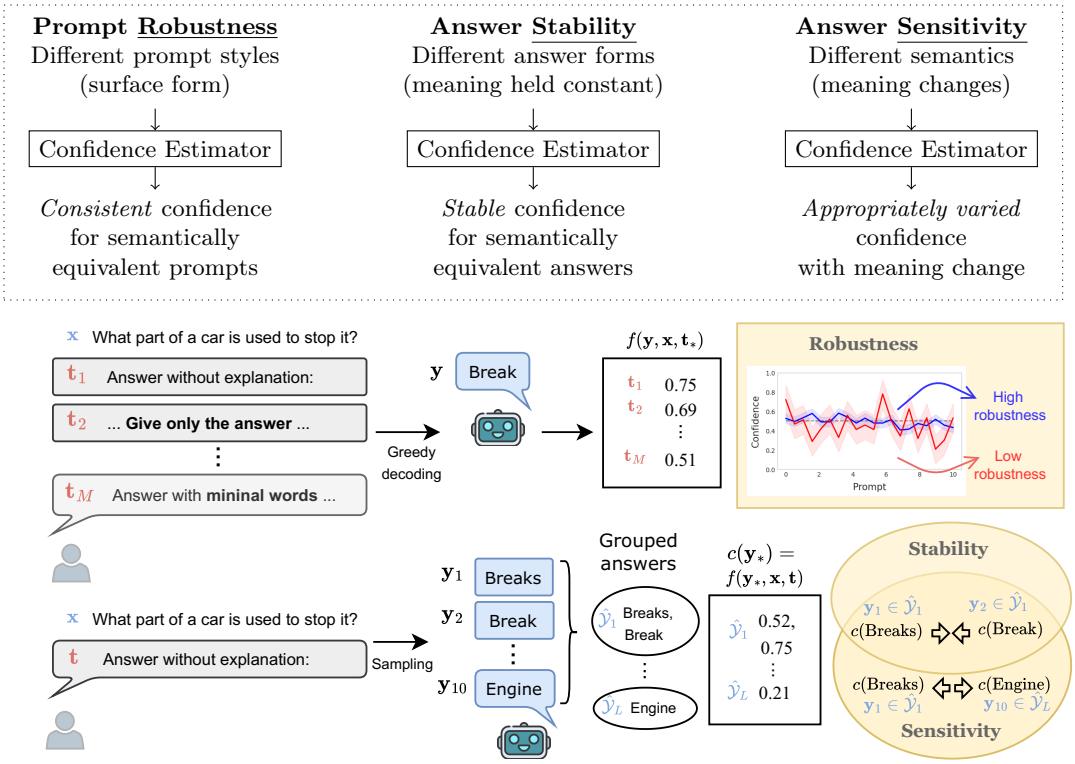


Figure 3.11.: Illustration of the three language variation criteria for confidence estimation. We assess *robustness* by measuring the variation in confidence scores elicited by semantically equivalent prompt perturbations. *Stability* measures confidence consistency across semantically equivalent answers, while *sensitivity* evaluates whether CE methods assign more distinct confidence scores to semantically different answers than to equivalent ones.

comparing estimated confidence deviations.

- *Sensitivity to semantic differences* **(A-SST)** assesses the extent to which CE methods assign larger confidence differences to semantically different answers than to semantically equivalent ones.

Accessed Methods. We benchmark five representative CE methods that span diverse settings.

1. **Probability** (**Prob.**) computes confidence by aggregating token-level probabilities of the generated answer under greedy decoding and normalizing the resulting sequence likelihood by output length.

2. **Platt Scaling** (**PS**; Platt, 1999) applies a parametric post-hoc transformation to the sequence probability, where two scaling parameters are learned by minimizing the

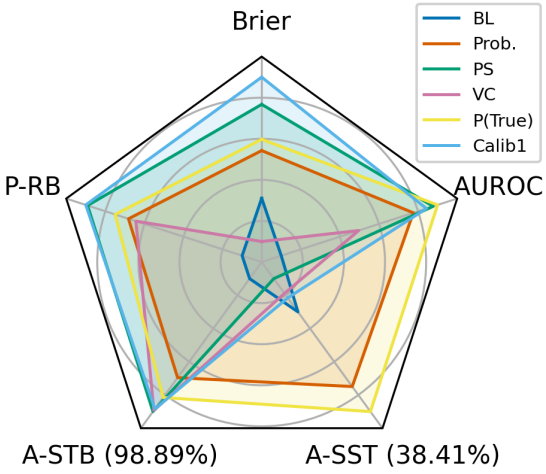


Figure 3.12.: Averaged performance (↑) among all models. The percentage indicates the amount of data that fulfills the requirement for evaluation of the metric (more than one answer have semantic same/similar meaning for A-STB, and at least two semantically different answers for A-SST).

mean squared error between predicted confidence and empirical answer correctness.

3. **Verbalized Confidence** (**VC**) directly elicits a self-reported confidence score from the LLM by prompting it to explicitly state its certainty about the generated answer, following recent work on verbalized uncertainty (Tian et al., 2023; Xiong et al., 2024).

4. **P(True)** prompts the LLM to assess whether its own response is true or false, and uses the normalized probability assigned to the *"True"* token as a confidence estimate (Kadavath et al., 2022).

5. **Calib-1-Focal** (**Calib1**) trains an auxiliary calibration classifier (Xia et al., 2025a) that takes the LLM input–response pair as features and predicts response correctness, optimized using focal loss (Mukhoti et al., 2020) to improve robustness under class imbalance.

6. As a reference point, we additionally include a **random baseline** (**BL**) that assigns confidence values uniformly at random.

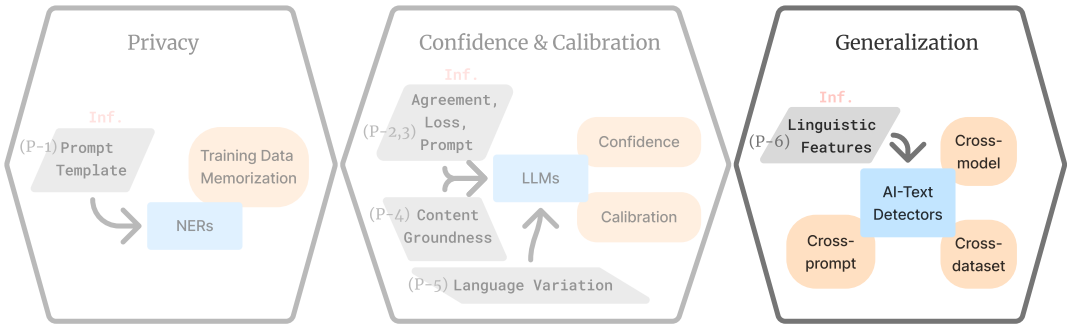
Findings. The overall results shown in Figure 3.12 demonstrate that common CE methods that are well-calibrated frequently fail on these language-variation tests. Specifically:

- CE approaches (e.g., P(True)) that achieve high AUROC (effective in discrimination) can show substantial fluctuations in confidence scores under minor prompt rephrasing, indicating sensitivity to superficial wording changes rather than genuine uncertainty.

- Several CE methods such as Prob. lack stability across semantically equivalent answers, assigning inconsistent confidence values to paraphrases that should logically warrant similar certainty.
- Many estimators (VC, PS, Calib1) exhibit poor sensitivity to semantic differences, failing to adapt confidence appropriately when the answer meaning does change, thereby undermining their usefulness for real-world decision support.

These findings reveal that traditional calibration and discrimination measures alone are insufficient for characterizing the practical quality of confidence estimation in LLMs, as they ignore how confidence reacts to meaningful language variation. The proposed evaluation framework uncovers weaknesses that would remain hidden under classic CE evaluation metrics such as Brier score and AUROC, highlighting the importance of incorporating linguistic robustness and semantic sensitivity into future CE research.

4. Influences on the Generalization of AI-Text Detectors



Generalization

Generalization in language models refers to the model's capacity to perform accurately on out-of-distribution data—that is, data drawn from a distribution different from the training distribution.

Building on the preceding chapters, which examined trustworthiness through the lenses of privacy leakage and reliable confidence estimation, this chapter turns to generalization as a third central requirement for trustworthy language model deployment. Generalization refers to a model's ability to maintain accurate performance under distributional variation, particularly when applied to data that differs from the distribution observed during training. This capability is especially important for AI-generated text detectors, which are often required to operate on text produced by newer language models and across domains not represented in their training data. As LLMs become increasingly dominant in high-stakes settings such as education, media, and content moderation (Guo et al., 2024; Hu et al., 2023), reliable detection of AI-generated text becomes correspondingly more important . Understanding when and why AI-text detectors fail to generalize is therefore essential for assessing their practical reliability.

4.1. AI-text Detection

AI-text detection models are designed to distinguish between human-written and AI-generated text. The methods to build an AI-text detector can broadly fall into two categories.

- **Statistical detectors**. These detectors exploit statistical irregularities in generated text, including token probability distributions or likelihood curvature. Representative method including GLTR (Gehrmann et al., 2019), DetectGPT (Mitchell et al., 2023), and Binoculars (Hans et al., 2024). These approaches offer interpretability and do not require supervised training, but their performance can degrade under distribution shifts or paraphrasing.

- **Classifier-based detectors**. These detectors treat the detection task as a classification task, typically fine-tune pretrained language models such as RoBERTa (Liu et al., 2019), DeBERTa (He et al., 2021), and XLM-R (Conneau et al., 2020) on labeled human- and AI-generated text. These methods consistently achieve stronger performance on large-scale AI-text detection benchmarks, including recent M4GT (Wang et al., 2024), MULTITuDE (Macko et al., 2023), and MultiSocial (Macko et al., 2025). However, it is also challenging for such detectors to generalize to out-of-distribution data, such as texts that are generated by newer models beyond the training data.

In summary, while both statistical and classifier-based detectors have demonstrated promising results, their limited generalization to unseen out-of-distribution data remains a critical vulnerability. Thus a deeper investigation into why a detector fail to generalize is needed.

4.2. Explaining Generalization in AI-Detectors

Generalization remains a core challenge for AI-text detection. Prior studies demonstrate that detectors trained on a specific domain or AI-text generator often exhibit substantial performance drops when evaluated under distribution shifts (Xu et al., 2024; Li et al., 2024; Bhattacharjee et al., 2024). Such failures highlight that many detectors rely on superficial or generation-specific cues that do not transfer across settings.

Beyond measuring generalization gaps, explaining why detection performance varies is essential for developing robust and trustworthy detectors. Several studies offer high-level explanations for generalization behavior. For instance, Xu et al. (2024) attribute performance differences primarily to prompt similarity and human–LLM alignment, showing that detectors generalize better when the prompts used during testing resemble those seen during training. In contrast, Li et al. (2024) identify generalization failures mainly as a consequence of diminishing linguistic differences between human- and AI-generated text as data sources diversify. The study demonstrates that detection performance drops substantially in out-of-distribution settings, particularly when the input text originates

from unseen domains or unseen language models. However, these analyses stop short of uncovering which specific linguistic properties correlate with the generalization failures of detectors.

The following section addresses this gap by systematically evaluating detector generalization under a wide range of distribution shifts, including variations in prompting strategies, generator model families and sizes, and content domains. It further provides detailed empirical evidence explaining detector generalization failures through a comprehensive linguistic analysis encompassing 80 syntactic, stylistic, and discourse-level features.

4.3. The Influences of Linguistic Features on the Generalization of AI-text Detectors

Yuxi Xia, Kinga Stańczak, and Benjamin Roth. "*Explaining Generalization of AI-Generated Text Detectors Through Linguistic Analysis*". EACL 2026: Long papers.

Classifier-based detectors often achieve near-perfect accuracy in in-domain settings, while their performance can degrade sharply under cross-prompt, cross-model, or cross-domain evaluation conditions. To explain these generalization gaps, it is necessary to identify the textual properties that underpin detector decision boundaries and their stability across different generalization conditions.

The paper tackles this challenge by hypothesizing that shifts in surface-level linguistic features, such as tense usage, syntactic patterns, or lexical distributions, can partially account for generalization behavior. To verify this, the study connects changes in detector accuracy to interpretable, linguistically grounded variables.

Methodology. To explore linguistic influences on generalization, we first construct a comprehensive benchmark dataset that systematically varies generation conditions. This benchmark comprises texts from seven distinct LLMs spanning five model families (e.g., Mistral (Mistral AI, 2024), Llama (Meta AI, 2024)), six prompting strategies (e.g., zero-shot, chain-of-though), and four domain datasets (e.g., abstracts, news), resulting in a large and diverse corpus of human-written and AI-generated texts.

Detectors that are fine-tuned on transformer classifiers, i.e., XLM-RoBERTa (Liu et al., 2019) and DeBERTa-V3 (He et al., 2023). The fine-tuned models are then evaluated across three generalization scenarios:

- **Cross-prompt generalization**, where detectors trained on texts generated by a specific prompt are tested on other prompt styles;
- **Cross-model generalization**, where detectors trained on outputs from a specific LLM are evaluated on outputs from unseen generators;
- **Cross-dataset generalization**, where detectors trained on one domain are tested on another.

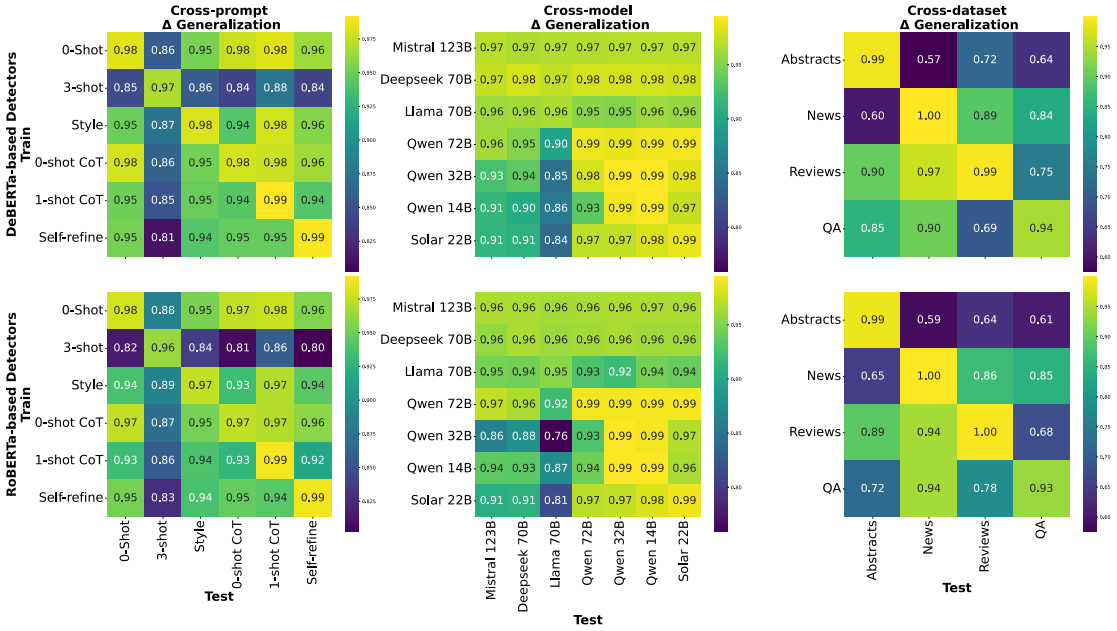
To quantify linguistic influences on detector generalization, we extract 80 linguistic features from both training and test data. These features cover multiple linguistic layers, part of the important features are:

- **Readability.** Readability measures the ease of understanding a text. We use the **Gunning Fog Index**, which estimates the years of formal education required to comprehend a passage.
- **Part-of-Speech (POS).** POS features capture the relative frequency of grammatical categories, measured as the fraction of text covered by each category. We analyze verbs, nouns, adjectives, numerals, and pronouns using StyloMetrix metrics (Okulska et al., 2023).
- **Grammatical Analysis.** Grammatical features focus on verb-related constructions. We measure the frequency of active and passive voice, as well as tense distributions (past, present, future). For example, past tense frequency includes verbs in past simple, continuous, and perfect forms, normalized by text length.
- **Lexical Analysis.** Lexical features characterize word-level stylistic patterns. We measure the frequency of personal names, comparative and superlative adjectives, and pronoun usage (e.g., personal and reflexive pronouns such as "we", "it", "our", "yourself").

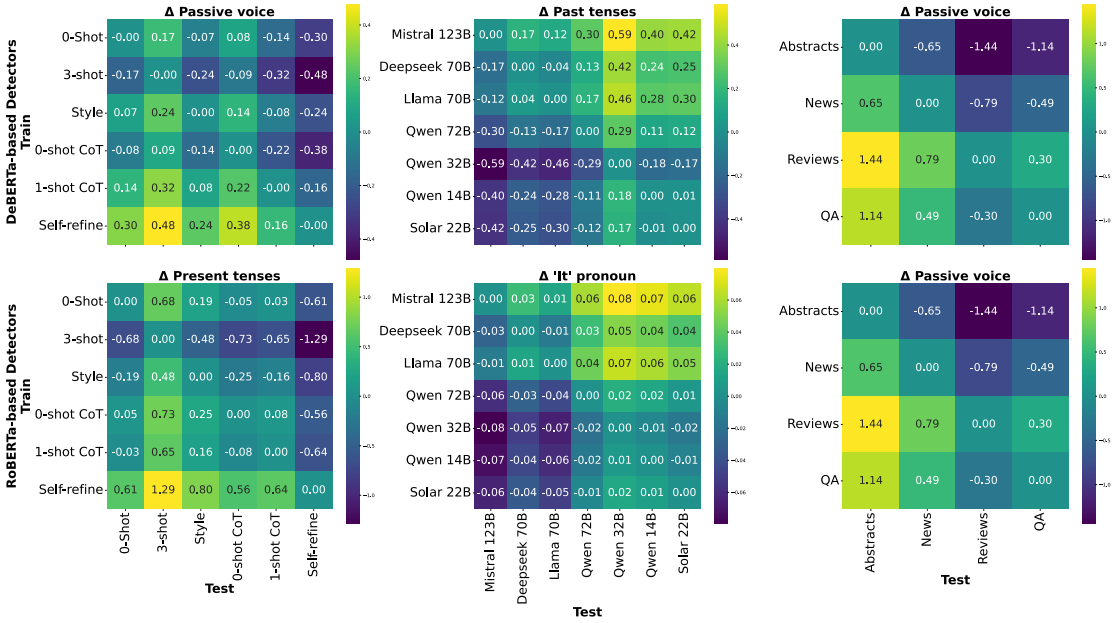
For each feature, we compute its absolute distributional shift between training and test data. These shifts are then used to compute the correlation with detector performance under cross-prompt, cross-model, and cross-domain evaluation settings to identify linguistic drivers of generalization failure.

Results and Analysis. We show the overall generalization performance and linguistic feature shift of the most correlated feature in Figure 4.1. The results reveal several important insights into how linguistic variation affects detector generalization:

- **Significant Correlations with Feature Shifts**: Many linguistic features exhibit statistically significant (p value <0.05) Pearson correlations with generalization performance. For example, shifts in the frequency of past-tense verbs correlate with cross-model generalization accuracy, indicating that certain detectors implicitly rely on tense usage as a cue for distinguishing human from AI text.
- **Setting-Specific Dependencies**: While no single feature universally explains all generalization behavior, specific linguistic signals become highly predictive in particular settings. In a notable case, detectors trained on Llama-3.3-70B generated reviews exhibited strong correlations (>0.7) between generalization accuracy and the prevalence of short sentences, suggesting detectors may latch onto stylistic patterns.



(a) The aggregated generalization performance.



(b) The aggregated feature shifts between training and test configurations.

Figure 4.1.: Comparison of aggregated generalization performance and aggregated feature shifts across all evaluation settings. Similar patterns in the two heatmaps indicate that certain feature shifts are correlated with reduced generalization accuracy.

- **Model-Dependent Feature Reliance**: Different detector architectures rely on different linguistic signals. For instance, RoBERTa-based detectors showed moderate correlation between cross-model generalization and "It" pronoun frequency, whereas DeBERTa-based detectors exhibited dependencies on the usage of past tenses, highlighting that learned decision boundaries can be architecture dependent.
- **Explanations for Generalization**: Certain generalization types pose greater challenges. The "3-shot" prompt style consistently presented the greatest difficulty in cross-prompt evaluation, while detectors trained on scientific abstracts struggled to generalize to news or reviews, likely due to stylistic features such as high passive voice frequency.

This work demonstrates that linguistic features provide a valuable lens for interpreting why detectors succeed or fail across shifts in generation conditions. By connecting performance variance to specific textual properties, the study provides interpretable explanations for detector behavior and highlights the limitations of purely performance-based evaluation. These findings deepen our understanding of detector vulnerabilities and indicate that improving robustness requires addressing the shortcut features on which detectors often rely. Reducing such reliance may enable detection systems to generalize more effectively beyond narrow training conditions.

5. Conclusion

This thesis investigated the factors that influence the trustworthiness of language models, identifying three trust-related properties: privacy, confidence reliability (estimation and calibration), and generalization. These properties are not solely determined by model architecture or scale, but are critically shaped by external factors such as prompting strategies, model agreement, and linguistic variation. Across six papers, the thesis demonstrates that these influences can substantially alter conclusions about model behavior, often by margins significant enough to impact deployment decisions. We summarize the methodology and key findings of these papers in Table 5.1.

In the Domain of Privacy. This thesis showed that memorization detection in masked language models fine-tuned for named entity recognition is highly prompt-sensitive. The same model can appear privacy-preserving or privacy-leaking depending on which prompt template is used to query entities. It further reveals that prompt engineering can increase the leakage of private training data in NER. These findings underscore that privacy evaluations are not absolute but intrinsically tied to the prompts used, requiring diverse prompt sets for robust memorization assessment for fine-tuned MLMs.

In the Domain of Confidence Estimation and Calibration. The thesis showed that model agreement, focal loss, and auxiliary calibration models can improve LLM calibration, but that verbalized confidence often reflects superficial patterns learned from training data rather than genuine grounding in the content. Our work further demonstrated that standard evaluation paradigms, which focused primarily on calibration error or discrimination, are insufficient for language models. This is because confidence estimates that appear well calibrated numerically may fluctuate under minor variations in prompts and fail to respond to meaning-changing answers. These results show that improving calibration alone is not sufficient for reliable confidence estimation, as confidence quality also depends on whether estimates remain robust to linguistic variation and grounded in evidence relevant to the content.

In the Area of Generalization. This thesis moved beyond documenting generalization failures and provided sufficient evidence to explain them. Through controlled experiments and large-scale linguistic analysis, it demonstrated that AI detector generalization is strongly correlated with shifts in syntactic, stylistic, and discourse-level features across prompts, generators, and domains. This finding suggests that many detectors may rely on linguistic shortcuts to make predictions, which limit their reliability under realistic generation conditions where new generators and different domains are encountered.

Influences	Methodology	Key Findings
	Privacy	
Prompt Template	Prompt-based memorization detection for fine-tuned masked language models (NER)	* Memorization detection is highly sensitive to prompt formulation. The same NER model can exhibit differences of more than 20 percentage points in detected memorization across prompts. * Prompt engineering can further increase the amount of memorized content that is revealed.
	Confidence and Calibration	
—	Black-box ensembling via model selection using input, outputs, and accuracy signals	* Selecting among multiple models without access to probabilities improves prediction reliability. * Model agreement serves as a strong proxy for epistemic uncertainty under black-box constraints.
Response Agreement, Loss Functions, Prompts	Auxiliary calibration models leveraging model agreement, focal loss, and prompting strategies	* Calibration quality is influenced by model agreement signals, loss function choice, and prompt style. * Agreement-aware calibration with focal loss yields more robust confidence estimates, while prompt variation can greatly alter calibration outcomes.
Content Groundness	Training data influence analysis for verbalized confidence expressions	* Verbalized confidence in LLMs is often weakly grounded in task-relevant evidence and instead reflects linguistic patterns learned during training. Models can express confidence without learning when confidence is warranted.
Language Variation	Evaluating the confidence estimation with robustness, stability and sensitivity under prompt and answer semantic variation	* Confidence estimates may fluctuate under minor prompt changes or fail to distinguish between semantically equivalent and different answers, limiting practical reliability. * Standard calibration and discrimination metrics fail to capture above aspects.
	Generalization	
Linguistic Features	Interpret the generalization behaviors across prompts, models, and domains with large-scale linguistic feature analysis	* Generalization failures in AI-text detectors can strongly correlated with shifts in syntactic and stylistic features (e.g., pronoun usage, passive voice). Many detectors may rely on linguistic cues rather than robust generative signatures.

Table 5.1.: Summary of thesis contributions across six papers, organized by investigated influence factors, methodology, and key findings.

Future Work. While this thesis provides a systematic empirical analysis of factors influencing language model trustworthiness, several important directions remain open for future investigation.

- **Broader Coverage of Training Data Influence Estimation.** The analysis of verbalized confidence presented in this thesis represents an initial step toward understanding how training data shapes confidence expression in LLMs. However, the studied models, tasks, and influence estimation techniques cover only a subset of possible settings. Extending this analysis to a wider range of model architectures, training corpora, and influence estimation methods, such as Hessian functions (Kwon et al., 2023), would enable a more comprehensive understanding of how and when confidence expressions are grounded in task-relevant evidence.

- **Grounded Confidence Training Objectives.** The finding that verbalized confidence is often weakly grounded in task-relevant content highlights a limitation of current training and alignment procedures. Future research could explore training objectives that explicitly encourage grounding confidence in evidence, such as retrieval-augmented supervision, contrastive learning between well-justified and poorly justified confidence statements, or joint optimization of answer correctness and confidence reliability.

- **From Correlation to Causality.** When explaining the generalization of AI detectors, much of the analysis relies on correlation-based methods to link observable features. While these correlations reveal consistent and interpretable associations, they do not establish causal relationships. Future work should therefore investigate causality through controlled interventions, such as counterfactual data generation, feature ablation, or prompt-level manipulations that selectively alter individual properties while holding others constant. Such approaches could enable stronger claims about which factors directly cause performance degradation.

- **Feature-Robust and Adaptive AI-Text Detection.** For AI-text detection, future work should move beyond detectors that rely on superficial stylistic cues and instead focus on models that are robust to linguistic variation while remaining sensitive to deeper generative signatures. Promising directions include feature-invariant training objectives, representation learning that emphasizes discourse-level consistency, and adaptive detectors that dynamically recalibrate under distribution shifts across prompts, domains, or generator models.

- **Toward Unified Trustworthiness Systems.** Finally, an important long-term direction is the development of unified evaluation frameworks that jointly assess privacy leakage, confidence reliability, and generalization. Understanding how these dimensions interact, why models fail along each of them, and how corresponding methods can be improved may lead to more holistic definitions of language model trustworthiness and support the design of safer and more reliable AI systems.

References

Michael Alfertshofer, Cosima C Hoch, Paul F Funk, Katharina Hollmann, Barbara Wollenberg, Samuel Knoedler, and Leonard Knoedler. 2024. Sailing the seven seas: a multinational comparison of chatgpt's performance on medical licensing examinations. *Annals of Biomedical Engineering*, 52(6):1542–1545.

Rana Salal Ali, Benjamin Zi Hao Zhao, Hassan Jameel Asghar, Tham Nguyen, Ian David Wood, and Dali Kaafar. 2022. Unintended Memorization and Timing Attacks in Named Entity Recognition Models. In *the Proceedings on Privacy Enhancing Technologies 2023(2), 329–346.*

Tariq Alqahtani, Hisham A Badreldin, Mohammed Alrashed, Abdulrahman I Alshaya, Sahar S Alghamdi, Khalid Bin Saleh, Shuroug A Alowais, Omar A Alshaya, Ishrat Rahman, Majed S Al Yami, and 1 others. 2023. The emergent role of artificial intelligence, natural learning processing, and large language models in higher education and research. *Research in social and administrative pharmacy*, 19(8):1236–1242.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, and 1 others. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862.*

Amrita Bhattacharjee, Raha Moraffah, Joshua Garland, and Huan Liu. 2024. Eagle: A domain generalization framework for ai-generated text detection. *ArXiv*, abs/2403.15690.

Sid Black, Leo Gao, Phil Wang, Connor Leahy, and Stella Biderman. 2021. GPT-Neo: Large Scale Autoregressive Language Modeling with Mesh-Tensorflow. If you use this software, please cite it using these metadata.

Andrew P. Bradley. 1997. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30(7):1145–1159.

Glenn W. Brier. 1950. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1–3.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. 2023. Discovering latent knowledge in language models without supervision. In *The Eleventh International Conference on Learning Representations.*

Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian Tramer. 2022. Membership inference attacks from first principles. In *2022 IEEE Symposium on Security and Privacy (SP)*, pages 1897–1914. IEEE.

Nicholas Carlini, Jamie Hayes, Milad Nasr, Matthew Jagielski, Vikash Sehwag, Florian Tramèr, Borja Balle, Daphne Ippolito, and Eric Wallace. 2023a. Extracting training data from diffusion models. In *Proceedings of the 32nd USENIX Conference on Security Symposium*, SEC '23, USA. USENIX Association.

Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. 2023b. Quantifying memorization across neural language models. *Preprint*, arXiv:2202.07646.

Nicholas Carlini, Chang Liu, Úlfar Erlingsson, Jernej Kos, and Dawn Song. 2019. The secret sharer: Evaluating and testing unintended memorization in neural networks. *Preprint*, arXiv:1802.08232.

Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, and 1 others. 2021. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 2633–2650.

Tuhin Chakrabarty, Vishakh Padmakumar, Faeze Brahman, and Smaranda Muresan. 2024. Creativity support in the age of large language models: An empirical study involving professional writers. In *Proceedings of the 16th Conference on Creativity & Cognition*, pages 132–155.

Bhanu Chander, Chinju John, Lekha Warrier, and Kumaravelan Gopalakrishnan. 2025. Toward trustworthy artificial intelligence (tai) in the context of explainability and robustness. *ACM Computing Surveys*, 57(6):1–49.

Banghao Chen, Zhaofeng Zhang, Nicolas Langrené, and Shengxin Zhu. 2024. Unleashing the potential of prompt engineering in large language models: a comprehensive review. *Preprint*, arXiv:2310.14735.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Harshil Darji, Jelena Mitrović, and Michael Granitzer. 2023. German bert model for legal named entity recognition. In *Proceedings of the 15th International Conference on Agents and Artificial Intelligence*, page 723–728. SCITEPRESS - Science and Technology Publications.

Anirban DasGupta. 2011. *Probability for statistics and machine learning: fundamentals and advanced topics*. Springer.

Leon Derczynski, Kalina Bontcheva, and Ian Roberts. 2016. Broad Twitter corpus: A diverse named entity recognition resource. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1169–1179, Osaka, Japan. The COLING 2016 Organizing Committee.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for*

Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Zhipeng Ding, Xu Han, Peirong Liu, and Marc Niethammer. 2021. Local temperature scaling for probability calibration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6889–6899.

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.

Elastic. 2025. Elasticsearch: Distributed, restful search and analytics engine. `https://www.elastic.co/elasticsearch/`. Version 8.17.2, released February 11, 2025.

Mingyuan Fan, Chengyu Wang, Cen Chen, Yang Liu, and Jun Huang. 2025. On the trustworthiness landscape of state-of-the-art generative models: A survey and outlook. *International Journal of Computer Vision*, 133(7):4317–4348.

Sebastian Gehrmann, Hendrik Strobelt, and Alexander Rush. 2019. GLTR: Statistical detection and visualization of generated text. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 111–116, Florence, Italy. Association for Computational Linguistics.

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. Association for Computational Linguistics.

Tobias Groot and Matias Valdenegro Toro. 2024. Overconfidence is key: Verbalized uncertainty evaluation in large language and vision-language models. In *Proceedings of the 4th Workshop on Trustworthy Natural Language Processing (TrustNLP 2024)*, pages 145–171, Mexico City, Mexico. Association for Computational Linguistics.

Xun Guo, Yongxin He, Shan Zhang, Ting Zhang, Wanquan Feng, Haibin Huang, and Chongyang Ma. 2024. Detective: Detecting ai-generated text via multi-level contrastive learning. *Advances in Neural Information Processing Systems*, 37:88320–88347.

Abhimanyu Hans, Avi Schwarzschild, Valeriia Cherepanova, Hamid Kazemi, Aniruddha Saha, Micah Goldblum, Jonas Geiping, and Tom Goldstein. 2024. Spotting llms with binoculars: zero-shot detection of machine-generated text. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2023. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. *Preprint*, arXiv:2111.09543. ICLR 2023.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*.

Xinlei He, Xinyue Shen, Zeyuan Chen, Michael Backes, and Yang Zhang. 2024. MGTBench: Benchmarking Machine-Generated Text Detection. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*. ACM.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Hongsheng Hu, Zoran Salcic, Lichao Sun, Gillian Dobbie, Philip S. Yu, and Xuyun Zhang. 2022. Membership inference attacks on machine learning: A survey. *Preprint*, arXiv:2103.07853.

Xiaomengc Hu, Pin-Yu Chen, and Tsung-Yi Ho. 2023. Radar: robust ai-text detection via adversarial learning. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23.

Yue Huang, Lichao Sun, Haoran Wang, Siyuan Wu, Qihui Zhang, Yuan Li, Chujie Gao, Yixin Huang, Wenhan Lyu, Yixuan Zhang, Xiner Li, Hanchi Sun, Zhengliang Liu, Yixin Liu, Yijue Wang, Zhikun Zhang, Bertie Vidgen, Bhavya Kailkhura, Caiming Xiong, and 52 others. 2024. Position: TrustLLM: Trustworthiness in large language models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 20166–20270. PMLR.

Shotaro Ishihara. 2023. Training data extraction from pre-trained language models: A survey. In *Proceedings of the 3rd Workshop on Trustworthy Natural Language Processing (TrustNLP 2023)*, pages 260–275, Toronto, Canada. Association for Computational Linguistics.

Bargav Jayaraman and David Evans. 2022. Are attribute inference attacks just imputation? In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, CCS '22, page 1569–1582, New York, NY, USA. Association for Computing Machinery.

Marija Jegorova, Chaitanya Kaul, Charlie Mayor, Alison Q O'Neil, Alexander Weir, Roderick Murray-Smith, and Sotirios A Tsaftaris. 2022. Survey: Leakage and privacy at inference time. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(7):9090–9108.

Marija Jegorova, Chaitanya Kaul, Charlie Mayor, Alison Q. O'Neil, Alexander Weir, Roderick Murray-Smith, and Sotirios A. Tsaftaris. 2023. Survey: Leakage and privacy

at inference time. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(7):9090–9108.

Jinyuan Jia and Neil Zhenqiang Gong. 2018. AttriGuard: A practical defense against attribute inference attacks via adversarial machine learning. In *27th USENIX Security Symposium (USENIX Security 18)*, pages 513–529, Baltimore, MD. USENIX Association.

Dongfu Jiang, Xiang Ren, and Bill Yuchen Lin. 2023. LLM-blender: Ensembling large language models with pairwise ranking and generative fusion. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14165–14178, Toronto, Canada. Association for Computational Linguistics.

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.

Zuheir N Khlaif, Allam Mousa, Muayad Kamal Hattab, Jamil Itmazi, Amjad A Hassan, Mageswaran Sanmugam, and Abedalkarim Ayyoub. 2023. The potential and concerns of using ai in scientific research: Chatgpt performance evaluation. *JMIR medical education*, 9:e47049.

Jaeyoung Kim, Dongbin Na, Sungchul Choi, and Sungbin Lim. 2023a. Bag of tricks for in-distribution calibration of pretrained transformers. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 551–563, Dubrovnik, Croatia. Association for Computational Linguistics.

Siwon Kim, Sangdoo Yun, Hwaran Lee, Martin Gubri, Sungroh Yoon, and Seong Joon Oh. 2023b. Propile: Probing privacy leakage in large language models. *Advances in Neural Information Processing Systems*, 36:20750–20762.

Sunnie SY Kim, Q Vera Liao, Mihaela Vorvoreanu, Stephanie Ballard, and Jennifer Wortman Vaughan. 2024. " i'm not sure, but...": Examining the impact of large language models' uncertainty expression on user reliance and trust. In *Proceedings of the 2024 ACM conference on fairness, accountability, and transparency*, pages 822–835.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *Proceedings of the 11th International Conference on Learning Representations (ICLR)*.

Meelis Kull, Miquel Perello Nieto, Markus Kängsepp, Telmo Silva Filho, Hao Song, and Peter Flach. 2019. Beyond temperature scaling: Obtaining well-calibrated multi-class probabilities with dirichlet calibration. *Advances in neural information processing systems*, 32.

Abhishek Kumar, Robert Morabito, Sanzhar Umbet, Jad Kabbara, and Ali Emami. 2024. Confidence under the hood: An investigation into the confidence-probability alignment in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 315–334, Bangkok, Thailand. Association for Computational Linguistics.

Aviral Kumar and Sunita Sarawagi. 2019. Calibration of encoder decoder models for neural machine translation. ICLR 2019.

Yongchan Kwon, Eric Wu, Kevin Wu, and James Zou. 2023. DataInf: Efficiently Estimating Data Influence in LoRA-tuned LLMs and Diffusion Models.

Jixuan Leng, Chengsong Huang, Banghua Zhu, and Jiaxin Huang. 2025. Taming overconfidence in LLMs: Reward calibration in RLHF. In *The Thirteenth International Conference on Learning Representations*.

Yafu Li, Qintong Li, Leyang Cui, Wei Bi, Zhilin Wang, Longyue Wang, Linyi Yang, Shuming Shi, and Yue Zhang. 2024. MAGE: Machine-generated text detection in the wild. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 36–53, Bangkok, Thailand. Association for Computational Linguistics.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. Teaching models to express their uncertainty in words. *Transactions on Machine Learning Research*.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. 2020. Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(2):318–327.

Zhen Lin, Shubhendu Trivedi, and Jimeng Sun. 2025. Generating with confidence: Uncertainty quantification for black-box large language models. In *Proceedings of the 13th International Conference on Learning Representations (ICLR)*. Accepted at ICLR 2025.

Jeremiah Zhe Liu, Shreyas Padhy, Jie Ren, Zi Lin, Yeming Wen, Ghassen Jerfel, Zachary Nado, Jasper Snoek, Dustin Tran, and Balaji Lakshminarayanan. 2023. A simple approach to improve single-model deep uncertainty via distance-awareness. *Journal of Machine Learning Research*, 24(42):1–63.

Xiaoou Liu, Tiejin Chen, Longchao Da, Chacha Chen, Zhen Lin, and Hua Wei. 2025. 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 V. 2*, pages 6107–6117.

Xin Liu, Muhammad Khalifa, and Lu Wang. 2024. Litcab: Lightweight language model calibration over short- and long-form responses. In *Proceedings of the Twelfth International Conference on Learning Representations*. ICLR.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Nils Lukas, Ahmed Salem, Robert Sim, Shruti Tople, Lukas Wutschitz, and Santiago Zanella-Béguelin. 2023. Analyzing leakage of personally identifiable information in language models. In *2023 IEEE Symposium on Security and Privacy (SP)*, pages 346–363. IEEE.

Dominik Macko, Jakub Kopál, Robert Moro, and Ivan Srba. 2025. MultiSocial: Multilingual benchmark of machine-generated text detection of social-media texts. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 727–752, Vienna, Austria. Association for Computational Linguistics.

Dominik Macko, Robert Moro, Adaku Uchendu, Jason Samuel Lucas, Michiharu Yamashita, Matúš Pikuliak, Ivan Srba, Thai Le, Dongwon Lee, Jakub Simko, and Maria Bielikova. 2023. MULTITuDE: Large-Scale Multilingual Machine-Generated Text Detection Benchmark. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9960–9987.

Inbal Magar and Roy Schwartz. 2022. Data contamination: From memorization to exploitation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 157–165, Dublin, Ireland. Association for Computational Linguistics.

Potsawee Manakul, Adian Liusie, and Mark Gales. 2023. SelfCheckGPT: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9004–9017, Singapore. Association for Computational Linguistics.

Meta AI. 2024. Introducing meta llama 3: The most capable openly available llm to date. Accessed: 2025-05-01.

Mistral AI. 2024. Mistral large instruct 2411. Accessed: 2025-05-01.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D. Manning, and Chelsea Finn. 2023. Detectgpt: zero-shot machine-generated text detection using probability curvature. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org.

Jishnu Mukhoti, Viveka Kulharia, Amartya Sanyal, Stuart Golodetz, Philip Torr, and Puneet Dokania. 2020. Calibrating deep neural networks using focal loss. *Advances in Neural Information Processing Systems*, 33:15288–15299.

Aleksandra Nastoska, Bojana Jancheska, Maryan Rizinski, and Dimitar Trajanov. 2025. Evaluating trustworthiness in ai: Risks, metrics, and applications across industries. *Electronics*, 14(13):2717.

Seth Neel and Peter Chang. 2024. Privacy issues in large language models: A survey. *Preprint*, arXiv:2312.06717.

Ngoc Dang Nguyen, Wei Tan, Lan Du, Wray Buntine, Richard Beare, and Changyou Chen. 2023. Auc maximization for low-resource named entity recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 13389–13399.

Shiyu Ni, Keping Bi, Lulu Yu, and Jiafeng Guo. 2025. Are large language models more honest in their probabilistic or verbalized confidence? In *Information Retrieval*, pages 124–135, Singapore. Springer Nature Singapore.

Inez Okulska, Daria Stetsenko, Anna Kołos, Agnieszka Karlińska, Kinga Głąbińska, and Adam Nowakowski. 2023. Stylometrix: An open-source multilingual tool for representing stylometric vectors. *Preprint*, arXiv:2309.12810.

Kazuki Osawa, Siddharth Swaroop, Mohammad Emtiyaz Khan, Anirudh Jain, Runa Eschenhagen, Richard E Turner, and Rio Yokota. 2019. Practical deep learning with bayesian principles. *Advances in neural information processing systems*, 32.

Georgios Papadopoulos, Peter J Edwards, and Alan F Murray. 2001. Confidence estimation methods for neural networks: A practical comparison. *IEEE transactions on neural networks*, 12(6):1278–1287.

John Platt. 1999. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in Large Margin Classifiers*, 10(3):61–74.

Garima Pruthi, Frederick Liu, Satyen Kale, and Mukund Sundararajan. 2020. Estimating Training Data Influence by Tracing Gradient Descent. In *Advances in Neural Information Processing Systems*, volume 33, pages 19920–19930. Curran Associates, Inc.

Rahul Rahaman and Alexandre H. Thiery. 2021. Uncertainty quantification and deep ensembles. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, NIPS '21, Red Hook, NY, USA. Curran Associates Inc.

Nitin Liladhar Rane, Abhijeet Tawde, Saurabh P Choudhary, and Jayesh Rane. 2023. Contribution and performance of chatgpt and other large language models (llm) for scientific and research advancements: a double-edged sword. *International Research Journal of Modernization in Engineering Technology and Science*, 5(10):875–899.

Jie Ren, Jiaming Luo, Yao Zhao, Kundan Krishna, Mohammad Saleh, Balaji Lakshminarayanan, and Peter J Liu. 2022. Out-of-distribution detection and selective generation for conditional language models. In *The Eleventh International Conference on Learning Representations*.

Stephen E. Robertson, Steve Walker, Susan Jones, Micheline Hancock-Beaulieu, and Mike Gatford. 1994. Okapi at trec-3. In *Text Retrieval Conference*.

Benjamin Roth, Pedro Henrique Luz de Araujo, Yuxi Xia, Saskia Kaltenbrunner, and Christoph Korab. 2024. Specification overfitting in artificial intelligence. *Artificial Intelligence Review*, 58(2):35.

Muhammad Bilal Saqib and Saba Zia. 2025. Evaluation of ai content generation tools for verification of academic integrity in higher education. *Journal of Applied Research in Higher Education*, 17(4):1430–1440.

David W Scott. 2015. *Multivariate density estimation: theory, practice, and visualization*. John Wiley & Sons.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. 2017. Membership inference attacks against machine learning models. In *2017 IEEE symposium on security and privacy (SP)*, pages 3–18. IEEE.

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. Association for Computational Linguistics.

Kushal Tirumala, Aram Markosyan, Luke Zettlemoyer, and Armen Aghajanyan. 2022. Memorization without overfitting: Analyzing the training dynamics of large language models. *Advances in Neural Information Processing Systems*, 35:38274–38290.

Dennis Ulmer, Martin Gubri, Hwaran Lee, Sangdoo Yun, and Seong Oh. 2024. Calibrating large language models using their generations only. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15440–15459, Bangkok, Thailand. Association for Computational Linguistics.

Yuxia Wang, Jonibek Mansurov, Petar Ivanov, Jinyan Su, Artem Shelmanov, Akim Tsvigun, Osama Mohammed Afzal, Tarek Mahmoud, Giovanni Puccetti, Thomas Arnold, Alham Aji, Nizar Habash, Iryna Gurevych, and Preslav Nakov. 2024. M4GT-bench: Evaluation benchmark for black-box machine-generated text detection. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3964–3992, Bangkok, Thailand. Association for Computational Linguistics.

Jiaheng Wei, Yanjun Zhang, Leo Yu Zhang, Ming Ding, Chao Chen, Kok-Leong Ong, Jun Zhang, and Yang Xiang. 2025. Memorization in deep learning: A survey. *ACM Computing Surveys*, 58(4):1–35.

Kevin Woods, W. Philip Kegelmeyer, and Kevin Bowyer. 1997. Combination of multiple classifiers using local accuracy estimates. *IEEE transactions on pattern analysis and machine intelligence*, 19(4):405–410.

Yuxi Xia, Pedro Henrique Luz De Araujo, Klim Zaporojets, and Benjamin Roth. 2025a. Influences on LLM calibration: A study of response agreement, loss functions, and prompt styles. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3740–3761, Vienna, Austria. Association for Computational Linguistics.

Yuxi Xia, Loris Schoenegger, and Benjamin Roth. 2026a. Influential training data retrieval for explaining verbalized confidence of llms. In *Proceedings of the 48th European Conference on Information Retrieval (ECIR 2026): Long papers*.

Yuxi Xia, Anastasiia Sedova, Pedro Henrique Luz de Araujo, Vasiliki Kougia, Lisa Nußbaumer, and Benjamin Roth. 2025b. Exploring prompts to elicit memorization in masked language model-based named entity recognition. *PLoS One*, 20(9):e0330877.

Yuxi Xia, Kinga Stańczak, and Benjamin Roth. 2026b. Explaining generalization of ai-generated text detectors through linguistic analysis. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2026): Long papers*.

Yuxi Xia, Dennis Ulmer, Terra Blevins, Yihong Liu, Hinrich Schütze, and Benjamin Roth. 2026c. Calibration is not enough: Evaluating confidence estimation under language variations. *Preprint*, arXiv:2601.08064. Under review.

Yuxi Xia, Klim Zaporojets, and Benjamin Roth. 2025c. Learn to pick the winner: Black-box ensembling for textual and visual question answering. In *Proceedings of the 21st Conference on Natural Language Processing (KONVENS 2025): Long and Short Papers*, pages 12–26, Hannover, Germany. HsH Applied Academics.

Dongfu Xiao, Chen Gao, Zhengquan Luo, Chi Liu, and Sheng Shen. 2025. Can llms assist computer education? an empirical case study of deepseek. *Preprint*, arXiv:2504.00421.

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 *Proceedings of the 12th International Conference on Learning Representations (ICLR)*.

Han Xu, Jie Ren, Pengfei He, Shenglai Zeng, Yingqian Cui, Amy Liu, Hui Liu, and Jiliang Tang. 2024. On the generalization of training-based ChatGPT detection methods. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 7223–7243, Miami, Florida, USA. Association for Computational Linguistics.

Zhangyue Yin, Qiushi Sun, Qipeng Guo, Jiawen Wu, Xipeng Qiu, and Xuanjing Huang. 2023. Do large language models know what they don't know? In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8653–8665, Toronto, Canada. Association for Computational Linguistics.

Ruide Zhang, Ning Zhang, Assad Moini, Wenjing Lou, and Y Thomas Hou. 2020. Privacyscope: Automatic analysis of private data leakage in tee-protected applications. In *2020 IEEE 40th International Conference on Distributed Computing Systems (ICDCS)*, pages 34–44. IEEE.

Jieqiong Zhao, Yixuan Wang, Michelle V Mancenido, Erin K Chiou, and Ross Maciejewski. 2023. Evaluating the impact of uncertainty visualization on model reliance. *IEEE Transactions on Visualization and Computer Graphics*, 30(7):4093–4107.

Kaitlyn Zhou, Jena Hwang, Xiang Ren, and Maarten Sap. 2024. Relying on the unreliable: The impact of language models' reluctance to express uncertainty. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3623–3643, Bangkok, Thailand. Association for Computational Linguistics.

Yutao Zhu, Huaying Yuan, Shuting Wang, Jiongnan Liu, Wenhan Liu, Chenlong Deng, Haonan Chen, Zheng Liu, Zhicheng Dou, and Ji-Rong Wen. 2025. Large language models for information retrieval: A survey. *ACM Transactions on Information Systems*, 44(1):1–54.

A. Exploring Prompts to Elicit Memorization in Masked Language Model-based Named Entity Recognition

Authors: Yuxi Xia, Anastasiia Sedova, Pedro Henrique Luz de Araujo, Vasiliki Kougia, Lisa Nußbaumer, Benjamin Roth

Status: Published in PLoS ONE

DOI: https://doi.org/10.1371/journal.pone.0330877

Reference: Xia et al. (2025b)

Work Division

Yuxi Xia: conceptualization, data curation, formal analysis, investigation, methodology, software, validation, visualization, original draft preparation, review and editing.
Anastasiia Sedova: formal analysis, investigation, visualization, original draft preparation
Pedro Henrique Luz de Araujo: formal analysis, investigation, visualization, original draft preparation
Vasiliki Kougia: formal analysis, investigation, visualization, original draft preparation
Lisa Nußbaumer: conceptualization, data curation
Benjamin Roth: conceptualization, funding acquisition, methodology, project administration, resources, supervision, review and editing

PLOS ONE

RESEARCH ARTICLE

Exploring prompts to elicit memorization in masked language model-based named entity recognition

Yuxi Xia[1,2]*, **Anastasiia Sedova**[1,2], **Pedro Henrique Luz de Araujo**[1,2], **Vasiliki Kougia**[1,2], **Lisa Nußbaumer**[3], **Benjamin Roth**[1,3]

1 Faculty of Computer Science, University of Vienna, Vienna, Austria, **2** UniVie Doctoral School Computer Science, Vienna, Austria, **3** Faculty of Philological and Cultural Studies, University of Vienna, Vienna, Austria

* yuxi.xia@univie.ac.at

OPEN ACCESS

Citation: Xia Y, Sedova A, Luz de Araujo PH, Kougia V, Nußbaumer L, et al. (2025) Exploring prompts to elicit memorization in masked language model-based named entity recognition. PLoS One 20(9): e0330877. https://doi.org/10.1371/journal.pone.0330877

Editor: Thiago P. Fernandes, Federal University of Paraiba, BRAZIL

Received: January 29, 2025

Accepted: August 6, 2025

Published: September 15, 2025

Peer Review History: PLOS recognizes the benefits of transparency in the peer review process; therefore, we enable the publication of all of the content of peer review and author responses alongside final, published articles. The editorial history of this article is available here: https://doi.org/10.1371/journal.pone.0330877

Data availability statement: All data is held in a public repository named "NER-memorization-detection" (https://github.com/Yuuxii/NER-memorization-detection). And the data are

Abstract

The possibility of identifying specific information about the training data a language model memorized poses a privacy risk. In this study, we analyze the ability of prompts to detect training data memorization in six masked language models, fine-tuned for named entity recognition. Specifically, we employ a diverse set of 1,200 automatically generated prompts for three entity types and a detection dataset that contains entity names present in the training set (in-sample names) and names not present (out-of-sample names). Here, prompts constitute patterns that can be instantiated with candidate entity names, and the prediction confidence of a corresponding entity name serves as an indicator of memorization strength. The prompt performance of detecting memorization is measured by comparing the confidences of in-sample and out-of-sample names. We show that the performance of different prompts varies by as much as 24.5 percentage points on the same model, and prompt engineering further increases the gap. Moreover, our experiments demonstrate that prompt performance is model-dependent but does generalize across different name sets. We comprehensively analyze how prompt performance is influenced by prompt properties (e.g., length) and contained tokens.

1 Introduction

Recent studies have highlighted the memorization of training data in language models [1], especially in auto-regressive ones (e.g., GPT models) [2,3]. Such studies are particularly important in exploring models' privacy risk [1], as attacks like membership inference (MI) [4, 5] can leverage data memorization to extract sensitive training data from the model. While training data can be extracted through direct prompt querying and text generation in GPT models, that method does not apply to Masked Language Models (MLMs).

Despite the efficiency and state-of-the-art performance achieved by MLM-based Named Entity Recognition (NER) models across various domains compared to GPT models [6], there is a lack of work studying memorization in NER models. During fine-tuning for NER, the model's objective shifts entirely to sequence labeling, and the token reconstruction head

all contained within the Supporting information files.

Funding: This study has been funded by the Vienna Science and Technology Fund (WWTF)[10.47379/VRG19008] "Knowledge-infused Deep Learning for Natural Language Processing" (https://www.wwtf.at) in the form of a grant, received by YX, BR, PHLdA, and VK. This study was also financially supported by Open Access funding provided by University of Vienna (https://openaccess.univie.ac.at/) in the form of a grant, received by YX.

Competing interests: The authors have declared that no competing interests exist.

used during pre-training is replaced with a task-specific classification head. This new output layer predicts only NER tags and no longer retains the capability to generate raw text [7–9]. Hence, NER models cannot be directly assessed for memorization via text generation, such as prompting them to reproduce verbatim in-sample entities given a prefix, a strategy commonly applied to auto-regressive models [3].

This paper explores prompts' impact on detecting the memorization of three types of entity names in six NER models, where prompts refer to linguistic context templates into which named entities are inserted [10–12]. Unlike previous work [13], who used only five hand-written prompts to detect memorization in a self-fine-tuned NER model on private, non-public data, which overlooks the impact of prompts, leading to limitations in robustness. We assess model sensitivity to prompt variations [14]. Specifically, we employ a set of 1,200 diverse prompts generated by GPT-3.5-turbo [15] for *PER* (person), *LOC* (location), and *ORG* (organization) entities. Our prompt set spans four sentence types (declarative, exclamatory, imperative, and interrogative), 15 prompt lengths, and 11 token positions for the target name. To ensure transparency and reproducibility, we evaluate publicly accessible state-of-the-art NER models fine-tuned on the widely-used CoNLL2003 dataset [16].

To quantify model memorization, we create a detection dataset by sampling 5,000 in-sample entity names from the CoNLL-2003 training data and 5,000 out-of-sample names from external sources (Wikidata [17], WikiANN [18]). Fig 1 illustrates our full memorization detection pipeline. Each prompt from the prompt set is completed with all entity names in the detection dataset, and the resulting sentences are input into the NER model to generate confidence scores for each name. These scores are ranked to compute the model memorization (*M-MEM*) score, which corresponds to the AUC score used in membership inference attacks (MIA) [4], thus quantifying the extent of memorization in the model. The *M-MEM* score detected by a prompt is referred to as the performance of this prompt.

Our experimental results demonstrate that memorization detection is sensitive to prompt variations. The performance gap using different prompts on the same model can be as large as 24.5 percentage points, addressing the limitation of using a single hand-written prompt for memorization detection. To provide solid verification of this finding, we perform extensive experiments to answer three research questions:

- **Are all models sensitive to prompt variations?** The average performance gaps over 3 entity types in the prompt set of six studied NERs range from 9.93 to 20.07, demonstrating that all models are sensitive to prompt variations regardless of pertaining schemes and model sizes;

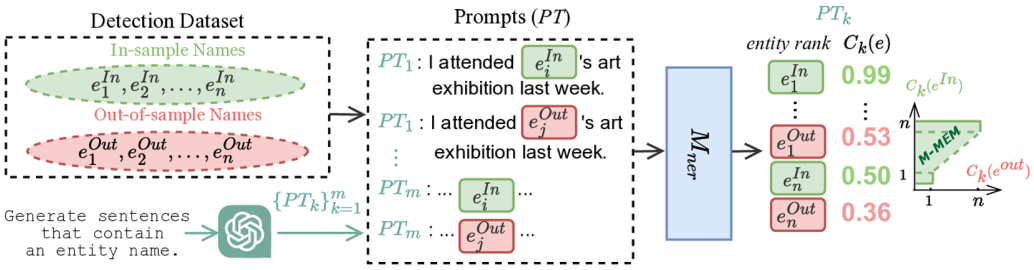


Fig 1. Memorization detection process of NERs. Each generated prompt is completed with all entity names from the detection dataset as inputs to the NER model M_{ner}. $C_k(e)$ represents the prediction confidence of M_{ner} for an entity name in PT_k. The performance of prompt PT_k for detecting memorization is measured by *M-MEM* score.

https://doi.org/10.1371/journal.pone.0330877.g001

- **Are prompt performances generalizable?** By computing Kendall's τ coefficient of prompt performance between different models and name sets, we find that prompt performance is model-dependent but generalizes across mutually exclusive name sets;
- **Can we further increase the prompt performance gap?** Experiments using prompt engineering and different ensembling techniques show that prompt engineering can increase the performance gap up to 51.3 percentage points.

For better guidance in generating higher/lower performing prompts, we perform two different levels of analysis of how various prompt properties impact the performance: (1) Sentence-level analysis of prompt types, length and token positions of the name; (2) Token-level analysis of individual token importance of a prompt. Furthermore, we use self-attention weights to interpret how different prompts change the model's attention weights which leads to performance differences.

Overall, our study quantifies memorization of NER models in a structured prediction task, without relying on models' generative capacity. We are the first to analyze the prompts' influence on the memorization detection of MLM-based NERs, using a diverse set of 1,200 prompts and six publicly accessible NER models for three entity types. Our experimental results strongly prove the importance of prompt choice in the memorization detection of NERs. We recommend that future research utilize diverse prompts (or generate target prompts by following our paper) to detect more robust memorization scores for privacy studies. The data and code used for the paper are provided in S2 Dataset and S3 Code.

2 Related work

Memorization in auto-regressive language models. This is first studied by Carlini et al. [19], who show that LSTMs memorize a significant fraction of their training data. Later, Carlini et al. [20] and Carlini et al. [3] use extractability to measure the memorization of GPT-2 and GPT-Neo models. Wei et al. [21] show that prefixed prompts can extract sensitive training data. However, measuring memorization with extractability only applies to auto-regressive language models, but not to fine-tuned MLMs. This is because auto-regressive language models can be directly probed for memorization by prompting them to generate text given a prefix. A model that reproduces long sequences in-sample during training, especially verbatim, indicates a strong signal of memorization. Thus, memorization in auto-regressive models is typically studied through generation-based evaluations [3,20,21]. Wang et al. [22] demonstrate the possibility to generate low-quality text from pre-trained BERT based on Gibbs sampling. However, this can not be applied to fine-tuned MLMs, because the token reconstruction head used during pre-training is replaced with a task-specific classification head which is restricted to the target space, e.g., named entity labels [7–9]. This makes standard generation-based memorization probing inapplicable to NERs.

Memorization in MLMs. Tirumala et al. [23] and Magar et al. [24] study memorization in MLMs by detecting specific data items. However, they focus more broadly by measuring model generalization using metrics related to the accuracy difference between training and test sets. Ali et al. [13] examine **memorization in fine-tuned MLMs** (NER models) and average the confidence ranks of in-sample entities in an entity set to measure model memorization. However, Ali et al. [13] self-fine-tune the NER model and study the impact of the training data duplication and complexity on the memorization dynamics. Differently, we explore 6 publicly accessible NER models fine-tuned by previous work and focus on the impact of different prompts on memorization detection.

Hand-written vs. Generated prompts. Language model performance has been shown to be sensitive to prompt variations in previous work [14,25]. Thus, randomly choosing one of the five hand-written prompts for each studied entity to detect memorization as in Ali et al. [13] may not robustly represent the model memorization. Instead, we employed 1,200 generated diverse prompts and studied the memorization, showing a large variability of memorization values associated with different prompts.

3 Methodology

In this section, we first describe the processes for creating the detection dataset and prompt set. Then, we introduce how we use them to detect memorization in NER models.

3.1 Detection dataset creation

We create a detection dataset D_{dt} to quantify the memorization of NER models fine-tuned on the CoNLL-2003 dataset [16]. Specifically, we first generate an in-sample name set D_{In} that contains all in-sample names of *PER*, *LOC* and *ORG* from the CoNLL-2003 dataset. Then, we create an out-of-sample name set D_{Out} by gathering names from publicly available data sources–Wikidata [17] and WikiANN [18]. Finally, the detection dataset D_{dt} contains equal number of in-sample and out-of-sample names ($\{e_i^{In}, e_i^{Out}\}_{i=1}^{n}$) sampled from D_{In} and D_{Out} for each entity type. As there is no training process in our setting, we exclusively split in-sample and out-of-sample names in D_{dt} equally into development (dev) and test. The details of the dataset statistics are shown in Table 1. The full dataset is provided in S1 Dataset.

In-sample Name Set (D_{in}). CoNLL-2003 is a commonly used NER dataset for research with a large number of examples of named entities. The training dataset of the CoNLL-2003 corpus contains 6,600 *PER* entity names, i.e., entity names with ground-truth label *B-PER* and/or *I-PER* which denote the beginning and the rest of a person's name respectively. After the post-processing of removing the duplicates and single-token *PER*, D_{in} contains 2,645 unique multi-token *PER* entity names. Similarly, we collect the *LOC* entity names by extracting the entity names with ground-truth label *B-LOC* and/or *I-LOC*, and *ORG* entity names with ground-truth label *B-ORG* and/or *I-ORG*. Different from the post-processing of *PER* entity, we delete the duplicates but keep the single-token names for *LOC* and *ORG*, resulting in 1,295 names of *LOC* and 2,297 names of *ORG*.

Out-of-sample Name Set (D_{out}). Wikidata [17] as a source for *PER* entity names in D_{out} enables the compilation of a list of real-world person names. Specifically, a SPARQL query is formulated to retrieve *PER* entity names. Similar to D_{in}, duplicates and single-token *PER* names are filtered out, resulting in 7,617,797 multi-token *PER* names. Additionally, we also removed the *PER* names presented in D_{in}, (7,617,797 - 1,651) *PER* names remained in the dataset D_{out} in the end. To simplify the process, the in-sample names of *LOC* and *ORG* are collected from WikiANN [18]. WikiANN is a NER dataset consisting of Wikipedia articles that have been annotated with *LOC*, *PER*, and *ORG* tags. After deleting the duplicates and names presented in D_{in}, D_{out} contains 7,266 *LOC* and 10,479 *ORG* names.

Table 1. Detection dataset statistics. One name is randomly assigned to dev/test if the total number of names is odd.

D_{dt}	***PER***		***LOC***		***ORG***	
	#In-sample	**#Out-of-sample**	**#In-sample**	**#Out-of-sample**	**#In-sample**	**#Out-of-sample**
dev	826	826	647	647	1,148	1,148
test	825	825	648	648	1,149	1,149
total	1,651	1,651	1,295	1,295	2,297	2,297

https://doi.org/10.1371/journal.pone.0330877.t001

Detection Dataset (D_{dt}). For in-sample *PER* names in D_{dt}, we select the ones also presented in Wikidata to ensure a better quality of name selection. This reduces the number of in-sample *PER* names from D_{in} to 1,651. We select all the names of *LOC* and *ORG* in D_{in}. Out-of-sample names are randomly sampled from D_{out} for each entity type.

3.2 Prompt set generation

To collect a large and diverse set of prompts for each entity type, we use the generative large language model GPT-3.5-turbo-1106 (ChatGPT [15]), to automatically generate m diverse prompts $\{PT_k\}_{k=1}^{m}$, $m = 400$, for each entity type.

To increase the diversity of the prompts, we consider four types of sentences: (1) declarative; (2) exclamatory; (3) imperative; and (4) interrogative. Next, we prompted ChatGPT to generate 100 sentences of each sentence type. The prompt template we used is: *"Generate 100 different [sentence type] sentences that must contain [entity type], and replace the [entity type] with 'MASK' "*. Details about the prompts and generated sentence examples are shown in Table 2. The full prompt set is provided in S1 Dataset.

The *"MASK"* string in a prompt PT_k is completed with e_i^{In} and e_i^{Out} resulting in two separate sentences, which are then fed to NER models to obtain the confidence scores for e_i^{In} and e_i^{Out}.

3.3 Memorization detection

Let M_{ner} denotes a fine-tuned MLM model for NER task on the CoNLL-2003 dataset. Such models can achieve high classification accuracy in predicting entity labels. When evaluating each token in a target entity, the NER model outputs probabilities over all possible tags (e.g., B-PER and I-PER, which represent the beginning and the inside tags of a person entity).

Table 2. Examples of different types of prompts for different entities.

Entity	ChatGPT Prompt	Sentence Types	Auto-Generated Prompts for NER
PER	Generate 100 different [Sentence Type] sentences that must contain a person's name, and replace the person's name with "MASK".	Declarative	MASK is a talented artist.
			I admire MASK's creativity.
		Exclamatory	Oh, MASK, you're such a genius!
			Whoa, MASK, that was mind-blowing!
		Imperative	MASK, please pass the salt.
			MASK, don't forget to lock the door.
		Interrogative	Can MASK pass the message to him?
			Can you tell me where I can find MASK?
LOC	Generate 100 different [Sentence Type] sentences that must contain a location, and replace the location with "MASK".	Declarative	The concert will be held in MASK.
			We spent our summer vacation in MASK.
		Exclamatory	Wow, I can't believe how beautiful MASK is!
			The sunsets in MASK are absolutely stunning!
		Imperative	Drive to MASK now.
			Send the package to MASK.
		Interrogative	Where can I find the best coffee in MASK?
			What is the population of MASK?
ORG	Generate 100 different [Sentence Type] sentences that must contain an organization name, and replace the organization name with "MASK".	Declarative	MASK offers excellent employee benefits.
			The stock price of MASK surged last week.
		Exclamatory	I can't believe MASK won the award!
			MASK is revolutionizing the industry!
		Imperative	Attend the MASK meeting next Monday.
			Donate to MASK to support their cause.
		Interrogative	How do I apply for a job at MASK?
			What is the mission statement of MASK?

https://doi.org/10.1371/journal.pone.0330877.t002

We extract the maximum probability between the corresponding beginning and inside tags (e.g., B-PER and I-PER) for each token as we insert the target entities and are aware of their correct type (e.g., PER or ORG). This ensures we capture the model's confidence that the token belongs to the correct entity type, regardless of whether it was predicted as a beginning or continuation token. Then, we average these per-token maximum probabilities across all tokens in the entity (T_e), yielding a single score that reflects the model's overall confidence $C(e)$ in labeling the full entity with the correct type. We formulate the entity confidence $C(e)$ as:

$$C(e) = \sum_{t \in T_e} \frac{\max(P(B\text{-}e|t), P(I\text{-}e|t))}{|T_e|} \tag{1}$$

Where *B-e* and *I-e* denote the beginning and inside tags of an entity. We define the confidence of an entity name in PT_k as $C_k(e)$.

Definition 3.1. **NER Memorization:** We define NER memorization as the extent to which a NER model exposes training data through its predictions. Specifically, it reflects the likelihood that an entity name from the training set can be identified via a membership inference attack. We quantify memorization as the percentage of instances in which a model assigns higher confidence to an in-sample entity name compared to an out-of-sample counterpart when both are presented in the same prompt.

Assume an entity type in dataset D_{dt} contains n in-sample names $\{e_i^{In}\}_{i=1}^n$ and n out-of-sample names $\{e_j^{Out}\}_{i=1}^n$, we formulate the performance of a prompt PT_k on detecting memorization of M_{ner} for this entity type as *M-MEM*$_k$ score:

$$M\text{-}MEM_k = \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{|\{C_k(e_i^{In}) > C_k(e_j^{out})\}|}{n^2} \tag{2}$$

Note that this formulation is equivalent to the AUC metric used for membership inference attack in [4]. It can be interpreted as the average rank of an in-sample name in the out-of-sample name set, where 100% means the in-sample name ranked the top among all out-of-sample names, 0% means the opposite, and 50% is considered as no significant memorization detected.

Sensitivity and robustness in memorization detection. To measure the model's sensitivity to prompt variations, we calculate the performance gap (Δ) between the highest and lowest performance across different prompts. The model's robustness is assessed by computing the standard deviation (σ) of the prompts' performances across the entire prompt set:

$$\sigma_{M_{ner}} = \sqrt{\frac{1}{m} \sum_{k=1}^{m} (M\text{-}MEM_k - \mu)^2} \tag{3}$$

Where μ is the average value of prompt performances. A lower σ value indicates greater robustness to prompt variation.

4 Experimental setup

We performed our model memorization analysis on publicly accessible MLM-based NER models fine-tuned by other researchers on the CoNLL-2003 dataset [16]. To provide a more comprehensive and diverse analysis, we selected six models built on MLMs across

three different pretraining schemes (ALBERT-v2 [26], BERT [27], and RoBERTa [28]) and 2 model sizes (base and large sizes, *-B* and *-L* notations are used correspondingly) for each scheme. More details about the models and accessibility information can be found in S1 Appendix.

4.1 Strategies of using the prompt set

In addition to the baseline, we investigate three strategies for exploiting the prompt set.

Baselines (BSL). **∅-PT** uses the entity name without any additional text as the input to query corresponding confidences for memorization detection. One-PT and Mix-PT use the same strategies as in [13]: **One-PT** employs one hand-written prompt (e.g., *"My name is XX."* for *PER* entity), corresponding to the *known*-setting in [13], while **Mix-PT** randomly chooses for each name one of the 5 hand-written prompts, like the *unknown*-setting in [13]. More prompt details are shown in Table 3.

Original Prompt (OPT) selects the best prompt (**B-PT**) and worst prompt (**W-PT**) from the prompt set that achieved the highest and lowest *M-MEM* scores on the dev set of D_{dt} for each entity type.

Prompt Engineering (PTE) is inspired by [29] who apply a token-removal operation to investigate how tokens of the input influence the model prediction. Here, the goal is to maximize the *M-MEM* score gap of the original best and worst prompts. Specifically, the most important token (which contributes the most to score improvement) and the least important token are removed from the worst and best prompts, respectively, resulting in two modified prompts. This process is iteratively repeated for the modified prompts until only one token (excluding the entity name) remains. The two modified prompts that achieved the highest and lowest *M-MEM* scores on the dev set among all the modified prompts are named **BM-PT** (best-modified prompt) and **WM-PT** (worst-modified prompt). It is important to note that such prompts may be ungrammatical. More details are presented in Sect 5.5.

Ensembling of Prompts (EPT) employs several ensemble techniques: (1) Majority Voting (**MV**) decides the rank between an in-sample and an out-of-sample names by majority agreement of the prompt set. (2) Average Confidence (**AVG-C**) uses the average entity confidence of all prompts $avg(\{C_k(e)\}_{k=1}^{k=m})$ as the final confidence score of the entity; (3) Weighted Confidence (**WED-C**) weights the entity confidence of each prompt by its *M-MEM* score; (4) Maximum Confidence (**MAX-C**) denotes the maximum confidence of the entity over all prompts $(max(\{C_k(e)\}_{k=1}^{k=m}))$ as the final confidence; (5) Minimum Confidence score (**MIN-C**) is the contrast to MAX-C, which uses the minimum value $min(\{C_k(e)\}_{k=1}^{k=m})$ as the final confidence.

Table 3. Hand-written prompts for different entities as baselines.

Baseline	Prompt		
	PER	***LOC***	***ORG***
∅-PT	-	-	-
One-PT	My name is MASK.	I am at MASK.	I work for MASK.
Mix-PT	My name is MASK.	I am at MASK.	I work for MASK.
	I am MASK.	I like MASK.	I like MASK.
	I am named MASK.	MASK is a good place.	MASK is a good organization.
	Here is my name: MASK.	Meet at MASK.	See you in MASK.
	Call me MASK.	Do you live in MASK?	Do you know MASK?

https://doi.org/10.1371/journal.pone.0330877.t003

4.2 Interpretability method

Self-attention weights [30] of a model can show where the model attends in the sequence and provide insights into what affected the predictions. Specifically, we first average the extracted attention weights of the tokens that correspond to the entity names for all completed sentences of a prompt (e.g., the best prompt completed with all in-sample entity names). Then we create a heatmap using the averaged values over the attention heads and the layers.

5 Results and analysis

Results showed in Table 4 demonstrate that the *M-MEM* scores vary considerably depending on the prompt: the maximum performance difference between the best and the worst prompt for a given model is as large as 24.5 percentage points for ALBERT-B on the test set for *LOC* entity.

We validate the statistical significance of the results through Cochran's Q test and found that the differences are significant for all models ($p < 0.001$). For an in-depth investigation, the following sections answer three research questions and perform sentence- and token-level analysis of how different factors impact the prompt performance. Furthermore, we use self-attention weights to interpret how different prompts change the model's attention weights, which leads to the performance differences. Note that we emphasize the *PER* results and analysis because this is the only entity type for which privacy has a special legal status reflected in regulation. E.g., the GDPR institutes a "right to be forgotten" that only applies to persons [31].

5.1 Are all models sensitive to prompt variations?

The last row of Table 4 reports the average performance difference across three entity types for the 6 studied NER models. Notably, RoBERTa-L shows the highest robustness, with a standard deviation (σ) of 1.51, while BERT-L is the least robust, with a σ of 2.47. The scores of different models range from 9.93 to 20.07, indicating that **all models are sensitive to prompt variations**, regardless of their pretraining strategies or model sizes.

5.2 Are prompt performances generalizable?

To examine how prompt performance generalizes across models and entities, we compute the Kendall's τ coefficient among the *M-MEM* scores of all the prompts for different models and data splits and show the results of *PER* entity in Fig 2. Results of other entity types are shown in S1 Appendix.

Generalizability across models: We observe that *M-MEM* scores on the dev set for BERT-B and BERT-L are negatively correlated, demonstrating a weak correlation between prompts' *M-MEM* scores for different models even when comparing different variants of the same

Table 4. Sensitivity (Δ) and robustness (σ) of models. The results are measured with the original prompt set on dev set. Bold/italic highlights the highest/lowest average Δ and σ values.

	ALBERT-B		ALBERT-L		BERT-B		BERT-L		RoBERTa-B		RoBERTa-L	
	Δ	σ	Δ	σ	Δ	σ	Δ	σ	Δ	σ	Δ	σ
PER	15.92	1.63	8.52	1.66	8.85	1.52	3.78	0.55	13.76	3.15	4.66	0.79
LOC	24.46	3.10	14.98	2.76	7.46	1.35	15.07	3.53	12.66	1.16	9.22	0.48
ORG	19.83	2.62	18.66	1.94	14.49	2.87	13.20	3.32	20.26	2.62	15.91	3.26
Avg.	**20.07**	2.45	14.05	2.12	10.27	1.91	10.68	**2.47**	15.56	2.31	*9.93*	*1.51*

https://doi.org/10.1371/journal.pone.0330877.t004

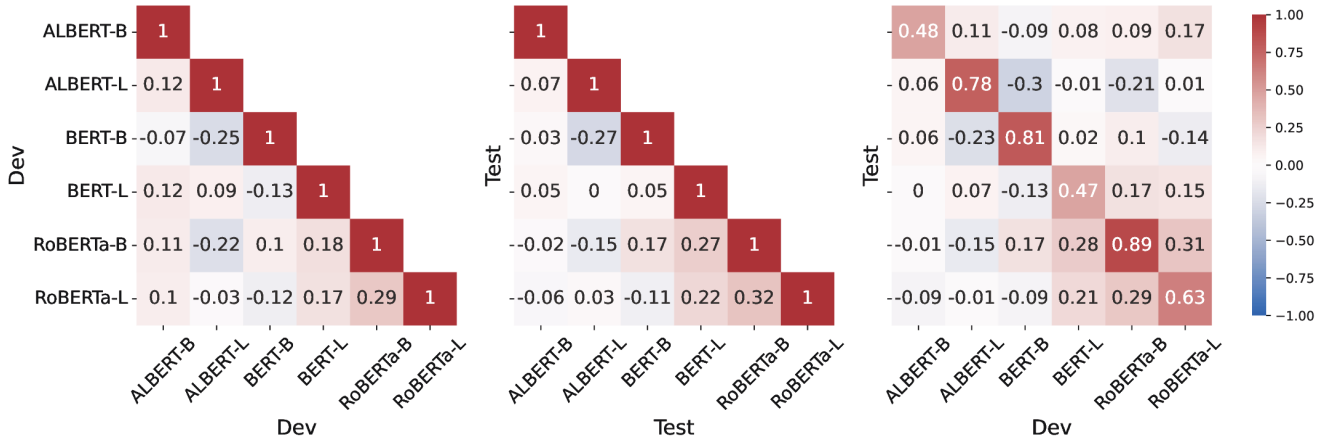


Fig 2. Correlations of the *M-MEM* scores of prompts across models on *PER* entity. Left: correlations (Kendall's τ) for the dev set scores. Middle: correlations for the test set scores. Right: correlations between dev and test set scores.

https://doi.org/10.1371/journal.pone.0330877.g002

model on the same data split. From this, we conclude that the *M-MEM* scores of prompts do not generalize across models—the best prompts for a given model may not be the best for other models.

Generalizability across name sets: Conversely, we found higher correlations between prompts' *M-MEM* scores across data splits when using the same model (diagonal of the right plot in Fig 2). We conclude that the *M-MEM* scores of prompts generalize to some extent across splits—for a given model, the best prompts for a given set of *PER* are likely to work well for a different set.

Table 5 illustrates this by comparing each model's best and worst prompts, showing that the (model-dependent) best and worst prompts for the dev set are still among the best and worst for the test set. These results show that while prompt quality depends on the model, it still generalizes for different names.

Table 5. Generalization of prompt ranks. The *PER* entity *M-MEM* scores and corresponding ranks of the best (Rank=1) and worst (Rank=-1) prompts from the dev set on the test set.

Model	Prompts	dev		test	
		Rank	*M-MEM*	Rank	*M-MEM*
ALBERT-B	Bravo, MASK, what an impressive performance!	1	74.84	2	75.10
	Are you going to MASK's art gallery opening tonight?	-1	58.92	-1	58.74
ALBERT-L	Oh, MASK, you're a true gem in our team.	1	73.12	3	71.38
	MASK, practice forgiveness towards yourself and others.	-1	64.60	-3	64.70
BERT-B	Are you going to MASK's art gallery opening tonight?	1	73.96	1	71.84
	Did MASK give you any advice on starting something new?	-1	65.11	-2	63.30
BERT-L	What project is MASK working on?	1	71.56	2	69.30
	I had a chance to meet MASK's family.	-1	67.78	-1	64.42
RoBERTa-B	MASK, can you recommend a good restaurant in town?	1	70.76	2	70.78
	I had a great conversation with MASK at the party.	-1	57.00	-2	59.63
RoBERTa-L	MASK, invest in meaningful relationships.	1	75.27	1	72.79
	MASK, practice playing the guitar.	-1	70.61	-96	70.05

https://doi.org/10.1371/journal.pone.0330877.t005

5.3 Can we further increase the prompt performance gap?

The prompt set outperforms baselines: In Table 6, all the *M-MEM* scores achieved by best prompts (B-PT) of the prompt set are higher than the three baselines in the dev set (only a few exceptions in the test set), showing the necessity to robustly detect the model memorization with diverse prompts.

Prompt engineering outperforms the original prompt set: We observe that the best/worst-modified prompts (BM-PT and WM-PT) using prompt engineering increase the score gap between the B-PT and the W-PT. For example, the BM-PT of RoBERTa-B improves by approximately 2 percentage points up on the B-PT on the dev set, and the WM-PT decreases the *M-MEM* score of BERT-L by 19 percentage points for the *PER* entity.

Ensembling techniques do not bring consistent improvement: The ensembling results of *PER* entity in Table 7 demonstrate that the ensemble techniques (EPT) do not improve/decrease the *M-MEM* scores from the B-PT/W-PT scores. However, we observed a few exceptions for the results of *LOC* and *ORG* entities.

5.4 Sentence-level analysis of prompt properties

This section investigates different prompt properties that may influence the *M-MEM* scores for *PER* entity. Analysis of other entity types is shown in S1 Appendix. **Prompt type:** Fig 3 presents prompts' *M-MEM* scores grouped by prompt type. RoBERTa-B has very distinct M-MEM scores for different types: imperative prompts generally yield higher scores than declarative prompts. Conversely, BERT-L has similar *M-MEM* score profiles for different prompt types. Overall, we observe that some models show similar *M-MEM* scores across sentence types, others are more sensitive to particular types.

Table 6. Results of using different prompt strategies. One-PT and Mix-PT are baselines adapted from [13]. Bold/italic highlights the highest/lowest values. Δ measures the gap between the highest and lowest values.

	Prompt Strategy		ALBERT-B		ALBERT-L		BERT-B		BERT-L		RoBERTa-B		RoBERTa-L	
			dev	test	dev	test	dev	test	dev	test	dev	test	dev	test
PER	BSL	∅-PT	71.11	72.44	70.81	68.91	66.78	66.10	70.16	68.11	66.39	65.46	74.40	70.29
		One-PT	70.42	71.18	69.02	67.57	72.15	70.31	69.91	66.49	68.57	70.72	72.68	70.22
		Mix-PT	68.63	70.02	67.58	66.98	70.59	69.27	69.29	66.56	66.47	67.29	72.80	70.42
	OPT	B-PT	74.84	**75.10**	73.12	71.38	73.96	71.84	71.56	69.30	70.76	70.78	75.27	72.79
		W-PT	58.92	58.74	64.60	64.70	65.11	63.30	67.78	64.42	57.00	59.63	70.61	70.05
	PTE	BM-PT	75.14	74.96	73.14	71.44	75.14	73.31	73.03	72.18	72.68	71.59	76.28	73.33
		WM-PT	41.92	44.53	63.76	63.52	59.15	62.04	48.68	49.98	55.29	58.21	69.15	68.52
	Δ		33.22	30.57	9.38	7.92	15.99	11.27	24.35	22.20	17.39	13.38	7.13	4.81
LOC	BSL	∅-PT	70.06	69.15	72.32	72.83	65.70	66.62	63.31	64.31	70.34	68.53	73.18	70.69
		One-PT	63.35	61.35	79.02	79.70	67.55	68.06	68.76	68.67	71.55	73.62	74.18	74.81
		Mix-PT	69.78	68.81	76.45	76.39	68.99	68.89	71.86	70.51	73.18	75.22	73.16	72.52
	OPT	B-PT	81.19	80.24	82.38	**81.60**	**73.68**	**72.70**	78.39	77.60	78.24	80.37	79.06	80.28
		W-PT	56.73	55.74	67.40	69.73	66.22	66.61	63.32	63.65	65.58	67.65	*69.84*	*68.36*
	PTE	BM-PT	**81.39**	**81.22**	**82.40**	81.38	73.01	72.51	**78.11**	**78.41**	**78.29**	**80.66**	**80.06**	**81.01**
		WM-PT	*54.96*	*54.86*	*31.10*	*33.67*	*61.30*	*62.28*	*61.66*	*62.97*	*50.96*	*50.99*	70.38	70.19
	Δ		26.43	26.36	51.30	47.93	12.38	10.42	16.45	15.44	27.33	29.67	10.22	12.65
ORG	BSL	∅-PT	74.38	75.88	74.06	77.54	75.90	78.58	77.58	80.47	74.77	75.07	73.46	76.28
		One-PT	72.57	74.49	72.00	76.13	69.88	73.53	72.92	76.00	74.19	73.99	66.72	69.09
		Mix-PT	68.15	68.91	69.33	72.04	69.76	72.44	69.49	71.83	68.94	68.03	66.38	68.27
	OPT	B-PT	76.22	76.14	75.41	77.00	78.46	79.63	79.10	79.94	77.08	76.51	74.38	75.52
		W-PT	*56.39*	*58.66*	*56.75*	*60.97*	63.97	67.34	65.90	68.14	*56.82*	*57.59*	*58.47*	*60.47*
	PTE	BM-PT	**77.21**	**76.87**	**77.31**	**78.19**	**78.92**	**80.05**	**79.55**	**81.63**	**78.17**	**77.46**	**75.50**	**76.49**
		WM-PT	57.59	60.01	58.25	62.33	*63.86*	*66.70*	*65.63*	*67.77*	56.85	57.73	58.82	60.95
	Δ		20.82	18.21	20.56	17.22	15.06	13.35	13.92	13.86	21.35	19.87	17.03	16.02

https://doi.org/10.1371/journal.pone.0330877.t006

Table 7. Results of using different prompt ensemble strategies. Bold/italic highlights the scores that are higher/lower than the prompt set (B-PT to W-PT) respectively.

	Prompt Strategy	ALBERT-B		ALBERT-L		BERT-B		BERT-L		RoBERTa-B		RoBERTa-L	
		dev	test	dev	test	dev	test	dev	test	dev	test	dev	test
PER	MV	73.24	74.30	70.96	69.71	70.75	69.34	69.91	67.98	64.87	66.30	73.73	71.05
	AVG-C	72.81	73.22	68.20	67.43	69.66	67.95	69.91	67.95	67.76	67.67	73.97	71.16
	WED-C	72.82	73.22	68.24	67.45	69.69	68.01	69.92	67.95	67.76	67.66	73.97	71.16
	MAX-C	71.53	73.71	70.65	68.99	73.52	71.40	69.97	67.31	61.47	63.22	73.69	71.25
	MIN-C	60.12	59.77	64.74	65.39	68.79	66.56	68.62	67.94	68.35	68.08	73.08	71.58
LOC	MV	80.55	80.40	81.44	81.46	70.90	70.29	73.51	72.53	74.57	77.35	77.18	77.97
	AVG-C	78.70	77.84	81.01	81.08	69.62	69.61	70.42	69.69	75.06	75.33	76.71	75.02
	WED-C	78.90	78.12	81.03	81.09	69.68	69.65	70.60	69.85	75.12	75.38	76.75	75.06
	MAX-C	79.21	**81.02**	81.87	81.51	**73.98**	**73.80**	76.61	76.24	76.92	79.11	78.96	79.81
	MIN-C	64.53	63.08	71.57	70.03	*65.16*	*64.52*	*60.70*	*60.73*	71.06	69.08	*69.44*	*68.33*
ORG	MV	73.58	75.65	70.58	74.89	72.12	75.20	74.65	77.63	71.20	72.37	66.17	68.94
	AVG-C	72.95	75.75	71.39	75.63	72.56	75.45	74.29	76.71	69.15	70.94	70.25	72.70
	WED-C	73.02	75.83	71.53	75.74	72.62	75.49	74.34	76.76	69.30	71.08	70.33	72.78
	MAX-C	**76.44**	75.91	65.23	67.31	71.80	74.07	71.51	74.11	74.73	74.83	*57.69*	58.60
	MIN-C	64.17	67.60	67.97	70.77	68.37	71.61	66.89	70.05	60.22	60.91	65.22	67.49

https://doi.org/10.1371/journal.pone.0330877.t007

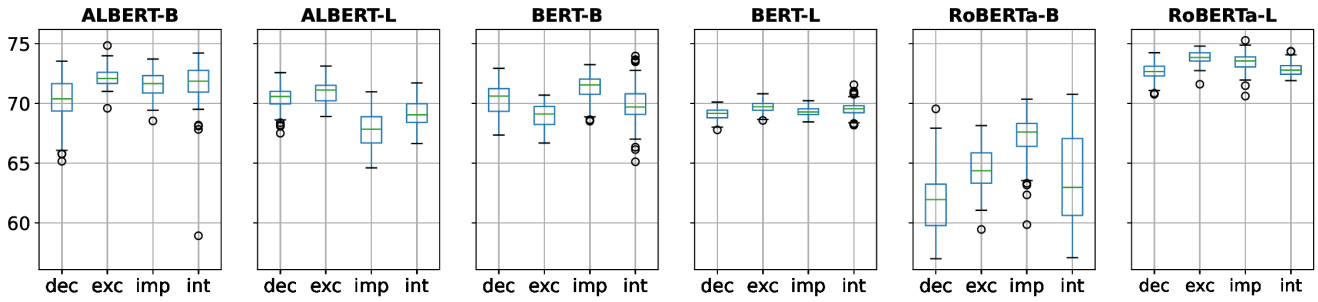


Fig 3. *M-MEM* scores grouped by prompt type. The four prompt types are **dec**larative, **exc**lamatory, **imp**erative, and **int**errogative.

https://doi.org/10.1371/journal.pone.0330877.g003

Person's name token position: Fig 4 shows prompts' *M-MEM* scores grouped by the token position of the name. The closer the name is to the beginning of the prompt, the higher the M-MEM score tends to be for RoBERTa-B and BERT-B.

Prompt token length: Fig 5 contrasts prompts' *M-MEM* scores and length (in number of tokens). We find that BERT-B stands out as an outlier, showing that a moderate negative correlation existed between *M-MEM* scores and prompt token lengths. While a weak correlation

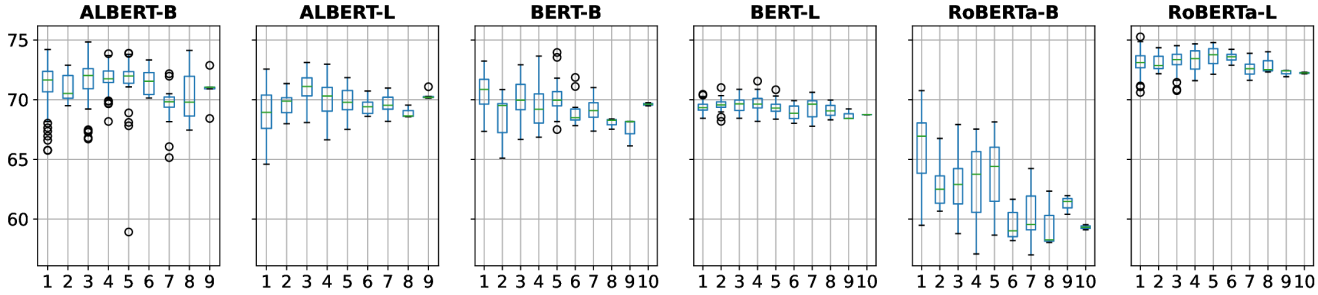


Fig 4. *M-MEM* scores grouped by person's name token position.

https://doi.org/10.1371/journal.pone.0330877.g004

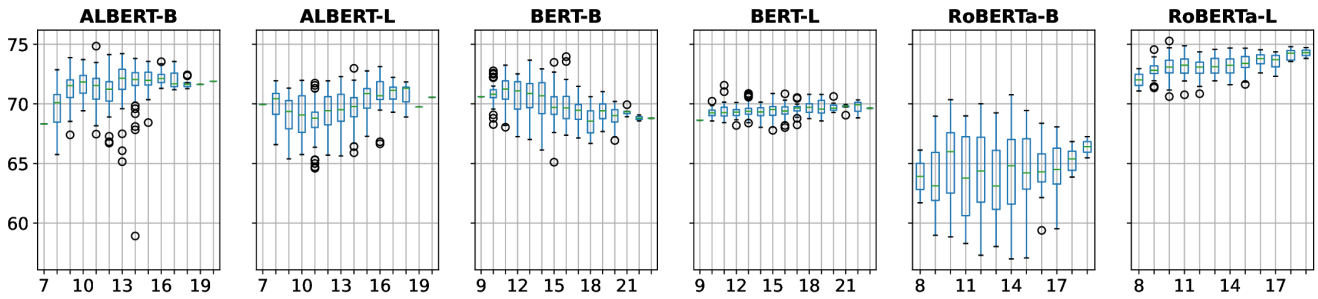


Fig 5. *M-MEM* scores grouped by prompt token length.

https://doi.org/10.1371/journal.pone.0330877.g005

between *M-MEM* scores and prompt length is observed for most models, meaning that longer prompts are generally more effective at identifying memorization.

5.5 Token-level analysis of token importance

Fig 6 shows a token-level analysis of how tokens within a prompt impact the *M-MEM* scores on the example of the BERT-B model for *PER* entity names. Please refer to S1 Appendix for the analysis of other models considered in our experiments.

Individual token importance for a prompt is calculated as the difference between the *M-MEM* scores of the prompt with and without this token on the dev set. These scores are then normalized using a softmax function.

Our results demonstrate that **removing the least important tokens from the prompt increases the *M-MEM* score, while removing the most important tokens has the opposite**

Prompt	M-MEM	Prompt	M-MEM
Are you going to MASK 's art gallery opening tonight ? 0.1 0.08 0.18 0.04 0.16 0.12 0.06 0.12 0.09 0.06	73.96	Did MASK give you any advice on starting something new ? 0.01 0.78 0.01 0.03 0.04 0.03 0.03 0.03 0.03 0.01	65.11
Are you going MASK 's art gallery opening tonight ? 0.08 0.07 0.2 0.45 0.05 0.04 0.03 0.04 0.05	74.65	Did MASK you any advice on starting something new ? 0.05 0.0 0.0 0.28 0.18 0.18 0.14 0.14 0.03	61.89
Are you going MASK 's art gallery tonight ? 0.08 0.06 0.24 0.23 0.06 0.25 0.03 0.05	74.75	Did MASK you any on starting something new ? 0.02 0.0 0.07 0.07 0.47 0.12 0.21 0.05	60.76
Are you going MASK 's art gallery ? 0.08 0.1 0.22 0.15 0.07 0.35 0.04	74.94	Did MASK you any on something new ? 0.03 0.0 0.22 0.12 0.2 0.37 0.06	60.34
Are you going MASK 's art gallery 0.03 0.07 0.13 0.37 0.04 0.36	75.03	Did MASK you any on something ? 0.15 0.0 0.28 0.27 0.16 0.14	60.29
you going MASK 's art gallery 0.08 0.16 0.4 0.07 0.28	75.14	Did MASK you on something ? 0.18 0.0 0.64 0.08 0.1	61.44
you going MASK 's gallery 0.06 0.09 0.24 0.6	74.70	Did MASK you something ? 0.92 0.0 0.05 0.03	60.71
going MASK 's gallery 0.23 0.5 0.27	74.33	MASK you something ? 0.01 0.09 0.9	59.15
MASK 's gallery 0.71 0.29	73.72	MASK you something 0.87 0.13	61.70
MASK 's 1.0	72.49	MASK something 1.0	67.69
MASK	66.78	MASK	66.78

Fig 6. Analysis of token importance in prompts. The figure shows the prompt analysis for the *PER* entity with the *M-MEM* scores on the BERT-B model. The best-performing (left) and the worst-performing (right) prompts were selected on the dev set. The heatmaps are generated with leave-one-token-out: at each step, the least important token is removed from the best-performing prompt, and the most important token is removed from the worst-performing prompt. The normalized token importance scores are under the corresponding tokens. The removed tokens are underlined.

https://doi.org/10.1371/journal.pone.0330877.g006

effect. For example, iteratively removing the least important tokens from the best-performing prompt for *PER* names (*"Are you going to MASK's art gallery opening tonight?"*) enables an increase in the *M-MEM* score from 73.96 to 75.14 for the prompt *"you going MASK's art gallery"*. Continuing to remove tokens results in a slight decrease in prompt performance, which leads to noun-phrase prompts performing worse than the initial prompt. In the case of the worst-performing prompt, removing the most important token *give* from the initial prompt results in a major decrease in the *M-MEM* score by approximately 3.5 percentage points. Further removal of important tokens slightly reduces the score to a minimum of 59.15 for the prompt *"MASK you something?"*.

The tokens that function as verbs and the tokens that are located near the *PER* name tokens tend to have a higher importance score. Tokens like *"going"*, *"give"*, *"practice"*, *"working"*, *"recommend"*, etc. (see Fig 6 and figures in S1 Appendix) have a higher influence on the *M-MEM* score in each considered prompt. A similar trend can be seen for the tokens *"'s"*, *"any"*, *"Oh"*, *"Bravo"*, *"What"*, and *","* located nearer to the *MASK* token (substituted by the *PER* names in the experiments) than tokens located further away.

5.6 Self-attention interpretation analysis

Fig 7 shows the attention heatmaps of BERT-B for the best and the worst prompts with each prompt completed with in-sample and out-of-sample *PER* names separately.

Generally, we notice that attention tends to be distributed similarly given a prompt, focusing on the first and the entity name tokens. This observation applies to all analyzed models (heatmaps for the other five models are shown in S1 Appendix).

Name level analysis: We observe a slightly higher focus on in-sample entity names than out-of-sample entity names when comparing the averaged attention weights on both the best and worst prompts.

Prompt level analysis: When comparing the heatmaps in the prompt level, we notice that models tend to focus more (or equally for BERT-B shown in Fig 7) on the entity name tokens in the best prompt than the worst, except for when the entity name tokens in the worst prompt are positioned in the beginning the prompt.

6 Discussion

Through our comprehensive analysis, we acknowledge the following questions for discussion.

How much memorization poses an unacceptable risk? While our study quantifies memorization based on a model's differential confidence between in-sample and out-of-sample

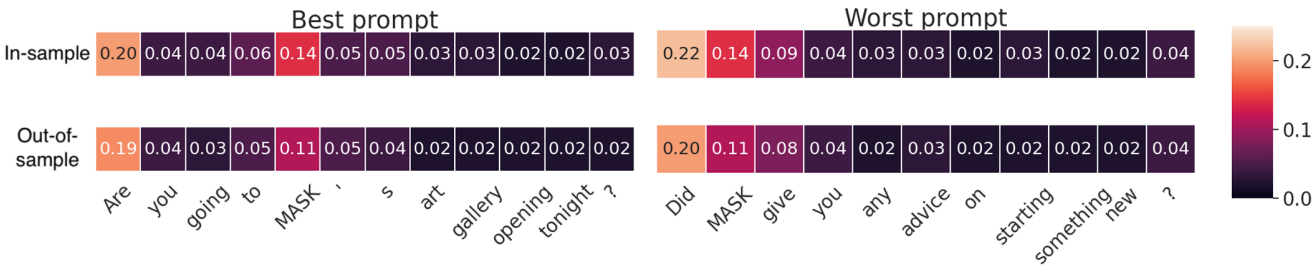


Fig 7. Attention heatmaps of the best and the worst prompts. The figure shows the result of BERT-B for *PER* entity names on the dev set averaged over all attention heads and layers. The attention weights corresponding to each prompt's "MASK" token are used and averaged over the in-sample and out-of-sample *PER* entity names separately.

https://doi.org/10.1371/journal.pone.0330877.g007

entity names, we do not define a threshold for what constitutes an "unacceptable" level of memorization. This is because the determination of the level is highly application-dependent. For example, even minimal memorization may be intolerable in domains involving sensitive personal information, such as healthcare or finance (e.g., HIPAA compliance or banking privacy) [20,32], or in contexts involving intellectual property and copyrighted material [33,34]. However, more leniency may be acceptable in tasks involving public or non-sensitive data [35]. Defining acceptable risk levels in practice typically requires considering domain-specific factors, regulatory standards, and ethical judgments. We acknowledge this as an important area for future interdisciplinary research.

How do we weigh risk versus performance in determining acceptable risk in various contexts? In this work, we propose a method to measure memorization in NER models by comparing prediction confidences for in-sample and out-of-sample entity names. While this quantification is a necessary step toward understanding privacy risks, determining what constitutes an "acceptable" level of memorization remains an open and context-dependent question. The trade-off between performance and privacy varies across applications and domains. Thus, we do not attempt to define this threshold here; instead, we highlight that our measurement framework provides the basis for such judgments. A crucial next step is to develop and evaluate mitigation strategies, such as regularization techniques, differential privacy, or selective data obfuscation, that can reduce memorization without substantially degrading model performance [36].

How to interpret M-MEM scores in context? The M-MEM score offers a way to quantify and compare memorization-related risk in NER models by measuring how consistently a model favors in-sample entity names over out-of-sample alternatives. While this metric provides valuable insights into model behavior, interpreting it as a measure of privacy risk depends on the application context. For example:

- High M-MEM + sensitive data (e.g., patient names): likely an unacceptable privacy risk.
- High M-MEM + public data (e.g., Wikipedia): possibly tolerable depending on usage.
- Low M-MEM: indicates weaker memorization, generally favorable from a privacy standpoint.

M-MEM does not directly estimate the probability of a successful attack or legal threshold violations; rather, it serves as a comparative indicator of how memorized a model is with respect to its training entities. Future work can refine this score into a more interpretable privacy metric or combine it with attack success rates to estimate risk more holistically [37].

7 Limitations

We address the limitations of our work as follows:

- This research focused on MLM-based models. Therefore, the analysis and findings in this paper may not be generalizable to other types of models, such as auto-regressive models, in the same way as previous studies of memorization in auto-regressive models are not generalizable to MLMs. Our work fills a research gap of examining memorization in fine-tuned MLMs.
- The models that we investigated had been trained by other researchers and the snapshots at different training stages (as well as the complete training settings) are unavailable to us. Thus, we can not provide dynamic memorization analysis across training procedures.
- We opted to study widely-used state-of-the-art models, in a setting where all components (base models, training data, etc) are widely available rather than non-SOTA models, or

models trained for specific domains. Unfortunately, those models are overwhelmingly fine-tuned on the CoNLL-2003 dataset, the predominant benchmark for NER. However, the CoNLL-2003 dataset is sampled from the Reuters Corpus (a corpus consisting of Reuters news stories), which contains rich and diverse topics such as sports, business, and politics. Nevertheless, our experiments across three distinct entity types enhance the generalizability of our findings.

8 Conclusion

We studied memorization in fine-tuned MLM-based NER models. Unlike auto-regressive language models, where memorization can be assessed by prompting the model to generate verbatim segments of its training data [3,20], the output of NER models consists of structured label predictions (e.g., B-PER, I-ORG), and thus does not support such evaluation. Our study quantifies memorization of NER models without relying on generative capacity. In this setting, we compared a fixed set of five hand-written prompts to 400 automatically generated prompts for each entity type, and we measured how well those prompts could distinguish entity names that were presented in the fine-tuning data from those that were not, using a large set of candidate entity names.

A detailed analysis revealed that a prompt's ability to detect memorization varies between models but remains stable across different entity sets. We find a large variability in the performance of generated prompts, and that many generated prompts significantly outperform hand-written ones. The effectiveness of prompts can be increased by removing the least important tokens even if the prompts become ungrammatical after the removal. Overall, we demonstrate the importance of using diverse prompts through comprehensive experiments and analysis. We provide all the prompts in supplemental material to detect memorization of NERs for future privacy studies.

Supporting information

S1 Appendix.
(PDF)

S2 Dataset.
(TXT)

S3 Code. (**GitHub repository:** https://github.com/Yuuxii/NER-memorization-detection/releases/tag/v1.0).

Acknowledgments

We thank Vasisht Duddu and N. Asokan who dedicated time and expertise to provide valuable feedback on this paper.

Author contributions

Conceptualization: Yuxi Xia, Lisa Nußbaumer, Benjamin Roth.

Data curation: Yuxi Xia, Lisa Nußbaumer.

Formal analysis: Yuxi Xia, Anastasiia Sedova, Pedro Henrique Luz de Araujo, Vasiliki Kougia, Lisa Nußbaumer.

Funding acquisition: Benjamin Roth.

Investigation: Yuxi Xia, Anastasiia Sedova, Pedro Henrique Luz de Araujo, Vasiliki Kougia, Lisa Nußbaumer.

Methodology: Yuxi Xia, Benjamin Roth.

Project administration: Yuxi Xia.

Resources: Benjamin Roth.

Software: Yuxi Xia.

Supervision: Benjamin Roth.

Validation: Yuxi Xia.

Visualization: Yuxi Xia, Anastasiia Sedova, Pedro Henrique Luz de Araujo, Vasiliki Kougia.

Writing – original draft: Yuxi Xia, Anastasiia Sedova, Pedro Henrique Luz de Araujo, Vasiliki Kougia.

Writing – review & editing: Yuxi Xia, Anastasiia Sedova, Pedro Henrique Luz de Araujo, Vasiliki Kougia, Benjamin Roth.

References

1. Neel S, Chang P. Privacy issues in large language models: a survey. arXiv preprint 2023. https://arxiv.org/abs/2312.06717
2. Mireshghallah F, Uniyal A, Wang T, Evans D, Berg-Kirkpatrick T. An empirical analysis of memorization in fine-tuned autoregressive language models. In: Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing. 2022. p. 1816–26. https://aclanthology.org/2022.emnlp-main.119
3. Carlini N, Ippolito D, Jagielski M, Lee K, Tramèr F, Zhang C. Quantifying memorization across neural language models. In: The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1–5, 2023. OpenReview.net; 2023. https://openreview.net/pdf?id=TatRHT_1cK
4. Salem A, Zhang Y, Humbert M, Berrang P, Fritz M, Backes M. ML-Leaks: model and data independent membership inference attacks and defenses on machine learning models. In: Proceedings of the 26th Annual Network and Distributed System Security Symposium (NDSS). 2019. https://arxiv.org/abs/1806.01246
5. Carlini N, Chien S, Nasr M, Song S, Terzis A, Tramer F. Membership inference attacks from first principles. In: 2022 IEEE Symposium on Security and Privacy (SP). 2022. p. 1897–914. https://arxiv.org/abs/2112.03570
6. Zaratiana U, Tomeh N, Holat P, Charnois T. GLiNER: generalist model for named entity recognition using bidirectional transformer. In: Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), Mexico City, Mexico, 2024. p. 5364–76. https://aclanthology.org/2024.naacl-long.300
7. Jin D, Jin Z, Zhou JT, Szolovits P. Is BERT really robust? A strong baseline for natural language attack on text classification and entailment. In: The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, 2020. p. 8018–25. https://aaai.org/ojs/index.php/AAAI/article/view/6311
8. Yang Z, Ding M, Guo Y, Lv Q, Tang J. Parameter-efficient tuning makes a good classification head. In: Goldberg Y, Kozareva Z, Zhang Y, editors. Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics; 2022. p. 7576–86. https://aclanthology.org/2022.emnlp-main.514
9. Keraghel I, Morbieu S, Nadif M. Recent advances in named entity recognition: a comprehensive survey and comparative study. arXiv preprint 2024. https://arxiv.org/abs/2401.10825
10. Xu Z, Peng K, Ding L, Tao D, Lu X. Take care of your prompt bias! investigating and mitigating prompt bias in factual knowledge extraction. In: Proceedings of the 2024 Joint International

Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024). 2024. p. 15552–65. https://aclanthology.org/2024.lrec-main.1352

11. Liu P, Yuan W, Fu J, Jiang Z, Hayashi H, Neubig G. Pre-train, prompt, and predict: a systematic survey of prompting methods in natural language processing. ACM Comput Surv. 2023;55(9):1–35. https://doi.org/10.1145/3560815
12. Jiang Z, Xu FF, Araki J, Neubig G. How can we know what language models know?. Transactions of the Association for Computational Linguistics. 2020;8:423–38. https://doi.org/10.1162/tacl_a_00324
13. Ali RS, Zhao BZH, Asghar HJ, Nguyen T, Wood ID, Kaafar D. Unintended memorization and timing attacks in named entity recognition models. In: Proceedings on Privacy Enhancing Technologies. 2023. http://arxiv.org/abs/2211.02245
14. Feng Z, Zhou H, Zhu Z, Qian J, Mao K. Unveiling and manipulating prompt influence in large language models. In:The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7–11, 2024. OpenReview.net; 2024. https://openreview.net/forum?id=ap1ByuwQrX
15. OpenAI. ChatGPT Conversation. https://chatgpt.com/
16. Tjong Kim Sang EF, De Meulder F. Introduction to the CoNLL-2003 shared task: language-independent named entity recognition. In: Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003. 2003. p. 142–7. https://www.aclweb.org/anthology/W03-0419
17. Vrandečić D, Krötzsch M. Wikidata: a free collaborative knowledgebase. Communications of the ACM. 2014;57(10):78–85.
18. Pan X, Zhang B, May J, Nothman J, Knight K, Ji H. Cross-lingual name tagging and linking for 282 languages. In: Barzilay R, Kan MY, editors. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Vancouver, Canada: Association for Computational Linguistics; 2017. p. 1946–58. https://aclanthology.org/P17-1178
19. Carlini N, Liu C, Erlingsson U, Kos J, Song D. In: 28th USENIX Security Symposium (USENIX Security 19), Santa Clara, CA, 2019. p. 267–84.
20. Carlini N, Tramer F, Wallace E, Jagielski M, Herbert-Voss A, Lee K, et al. Extracting training data from large language models. In: 30th USENIX Security Symposium (USENIX Security 21); 2021. p. 2633–50.
21. Wei C, Wang Y-C, Wang B, Kuo C-CJ. An overview of language models: recent developments and outlook. SIP. 2024;13(2). https://doi.org/10.1561/116.00000010
22. Wang A, Cho K. BERT has a mouth, and it must speak: BERT as a markov random field language model. In: Bosselut A, Celikyilmaz A, Ghazvininejad M, Iyer S, Khandelwal U, Rashkin H, et al., editors. Proceedings of the Workshop on Methods for Optimizing and Evaluating Neural Language Generation. Minneapolis, Minnesota: Association for Computational Linguistics; 2019. p. 30–6. https://aclanthology.org/W19-2304
23. Tirumala K, Markosyan AH, Zettlemoyer L, Aghajanyan A. Memorization without overfitting: analyzing the training dynamics of large language models. In: Advances in Neural Information Processing Systems 35, New Orleans, LA, USA, 2022. http://papers.nips.cc/paper_files/paper/2022/hash/fa0509f4dab6807e2cb465715bf2d249-Abstract-Conference.html
24. Magar I, Schwartz R. Data contamination: from memorization to exploitation. In: Muresan S, Nakov P, Villavicencio A, editors. Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). Dublin, Ireland: Association for Computational Linguistics; 2022. p. 157–65. https://aclanthology.org/2022.acl-short.18
25. Sclar M, Choi Y, Tsvetkov Y, Suhr A. Quantifying language models' sensitivity to spurious features in prompt design or: how i learned to start worrying about prompt formatting. In: The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7–11, 2024. OpenReview.net; 2024. https://openreview.net/forum?id=RIu5lyNXjT
26. Lan Z, Chen M, Goodman S, Gimpel K, Sharma P, Soricut R. ALBERT: a lite BERT for self-supervised learning of language representations. In: Addis Ababa, Ethiopia. 2020. https://openreview.net/forum?id=H1eA7AEtvS
27. Devlin J, Chang MW, Lee K, Toutanova K. BERT: pre-training of deep bidirectional transformers for language understanding. In: Burstein J, Doran C, Solorio T, editors. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). Minneapolis, Minnesota: Association for Computational Linguistics; 2019. p. 4171–86. https://aclanthology.org/N19-1423
28. Liu Y, Ott M, Goyal N, Du J, Joshi M, Chen D. RoBERTa: a robustly optimized BERT pretraining approach. arXiv preprint 2019. https://doi.org/abs/1907.11692

29. Feng S, Wallace E, Grissom II A, Iyyer M, Rodriguez P, Boyd-Graber J. Pathologies of neural models make interpretations difficult. In: Riloff E, Chiang D, Hockenmaier J, Tsujii J, editors. Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. Brussels, Belgium: Association for Computational Linguistics; 2018. p. 3719–28. https://aclanthology.org/D18-1407

30. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, et al. Attention is All You Need. In: Advances in Neural Information Processing Systems, Long Beach, CA, USA, 2017. p. 5998–6008. https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html

31. Mondschein CF, Monda C. The EU's General Data Protection Regulation (GDPR) in a Research Context. In: Kubben P, Dumontier M, Dekker A, editors. Fundamentals of Clinical Data Science. Cham (CH): Springer; 2018. p. 55–71. https://doi.org/10.1007/978-3-319-99713-1_5

32. Shokri R, Stronati M, Song C, Shmatikov V. Membership inference attacks against machine learning models. In: Proceedings of the 2017 IEEE Symposium on Security and Privacy (SP). IEEE; 2017. p. 3–18.

33. U S D of H& HS. Summary of the HIPAA Privacy Rule. 2003. https://www.hhs.gov/hipaa/for-professionals/privacy/laws-regulations/index.html

34. European Parliament. General Data Protection Regulation (GDPR). 2016. https://eur-lex.europa.eu/eli/reg/2016/679/oj

35. Radford A, Wu J, Child R, Luan D, Amodei D, Sutskever I. Language models are unsupervised multitask learners. 2019. https://api.semanticscholar.org/CorpusID:160025533

36. Abadi M, Chu A, Goodfellow I, McMahan HB, Mironov I, Talwar K. Deep learning with differential privacy. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. 2016. p. 308–18.

37. Yeom S, Giacomelli I, Fredrikson M, Jha S. Privacy risk in machine learning: Analyzing the connection to overfitting. In: 2018 IEEE 31st Computer Security Foundations Symposium (CSF). IEEE; 2018. p. 268–82.

B. Learn to pick the winner: Black-box ensembling for textual and visual question answering

Authors: Yuxi Xia, Klim Zaporojets, Benjamin Roth

Status: Published in the Proceedings of the 21st Conference on Natural Language Processing (KONVENS 2025): Long and Short Papers

URL: https://aclanthology.org/2025.konvens-1.2/

Reference: Xia et al. (2025c)

Work Division

Yuxi Xia: conceptualization, data curation, formal analysis, investigation, methodology, software, validation, visualization, original draft preparation, review and editing.
Klim Zaporojets: validation, investigation, review and editing
Benjamin Roth: funding acquisition, methodology, project administration, resources, supervision, review and editing.

Learn to pick the winner: Black-box ensembling for textual and visual question answering

Yuxi Xia [1,2*], Klim Zaporojets[3], Benjamin Roth[1,4],

[1]Faculty of Computer Science, University of Vienna,
[2] UniVie Doctoral School Computer Science, University of Vienna,
[3]Department of Computer Science, Aarhus University,
[4]Faculty of Philological and Cultural Studies, University of Vienna,
***Correspondence:** yuxi.xia@univie.ac.at

Abstract

A diverse range of large language models (LLMs), e.g., ChatGPT, and visual question answering (VQA) models, e.g., BLIP, have been developed for solving textual and visual question answering tasks. However, fine-tuning these models is either difficult, as it requires access via APIs, rendering them as black-boxes, or costly due to the need to tune a large number of parameters. To address this, we introduce ***InfoSel***, a data-efficient ensemble method that learns to dynamically pick the winner from existing black-box models for predictions on both textual and multimodal visual question answering tasks. Unlike traditional ensemble models, ***InfoSel*** does not rely on prediction probabilities or confidences, which typically are not available in black-box models. Experimental results on four datasets demonstrate that our approach achieves an absolute increase of up to +5% in the F1-score compared to standalone LLMs using only 1K training instances.

1 Introduction

Large language models (LLMs) have demonstrated remarkable proficiency across a wide range of tasks (Laskar et al., 2023; Yuan et al., 2024). For example, ChatGPT finds extensive utilization in daily textual question answering (TQA) tasks, rendering substantial convenience to a myriad of users (OpenAI, 2024). Furthermore, for visual question answering (VQA) tasks, VQA models have exhibited exceptional versatility, primarily due to their capability to comprehend both visual and textual context (Gong et al., 2023).

However, recent work (Mao et al., 2024; Laskar et al., 2023) indicates that LLMs, such as ChatGPT, fall short of state-of-the-art performance on task-specific datasets such as question answering. Similarly, VQA models (Li et al., 2022, 2021b; Bao et al., 2022) face challenges when applied to specialized datasets due to the idiosyncrasies in the content, format or structure of these datasets (Arora et al., 2018). Unfortunately, fine-tuning LLMs (e.g., LLaMA 70b model (Touvron et al., 2023)) on task-specific data requires a large number of GPU hours. Alternatively, training smaller, task-specific models from scratch requires a large amount of labeled data to achieve comparable performance (Tajbakhsh et al., 2016). Furthermore, fine-tuning LLMs through proprietary APIs with self-uploaded labeled training data not only requires LLM experts' knowledge but is also expensive.[1] These fine-tuned models further remain black-box, with restricted access to details regarding architectural intricacies, model weights, training data, and even prediction confidences.

In order to address these computational and accessibility challenges associated with fine-tuning, we introduce a scalable ensemble method called ***InfoSel*** (*Informed Selection*), which trains a small-sized selection model with just a few labeled task-specific samples. Unlike current ensemble methods (e.g., MetaQA (Puerto et al., 2023)), which depend on the confidence scores and thus can not be applied to black-box models like GPT3.5 text-davinci, *InfoSel* does not rely on such information and offers black-box ensembling. While traditional ensemble methods such as OLA (Woods et al., 1997) and PageRank (Brin and Page, 2012) are not adapted to task-specific particularities (e.g., different features) of different datasets. *InfoSel* incorporates *task-specific* optimization by considering variations of both the inputs and predicted answers from the ensembled LLMs/VQA models (*base models*), which allows it to be easily adapted to different datasets. Finally, our method efficiently deals with *multimodal* inputs. We showcase the adaptability and effectiveness of the method by testing it on textual and visual question answering tasks. Concretely,

[1] https://platform.openai.com/docs/guides/fine-tuning/

Proceedings of the 21st Conference on Natural Language Processing (KONVENS), Volume 1: Long and Short Papers, pages 12–26, 2025

our results exhibit superior performance on multimodal VQA inputs compared to the state-of-the-art PairRanker (Jiang et al., 2023) ensemble method that is designed to work exclusively with text. Table 1 compares our method with alternatives.

	FT	Pair-Ranker	OLA/ PageRank	***InfoSel***
Task-specific	✓	✓	✗	✓
Data-efficient	✗	✗	✓	✓
Black-box	✗	✓	✓	✓
Multimodal	✗	✗	✓	✓
Ensemble	✗	✓	✓	✓

Table 1: Our method (*InfoSel*) aims to optimize ***task-specific*** ensembling of ***black-box*** models, where confidences and parameters can not be accessed. We use only a small portion of training data (***data-efficient***). We optimize the performance of the ***ensemble*** instead of standalone fine-tuned (FT) models. Finally, our method is ***multimodal***, and applicable to VQA task.

At its core, *InfoSel* (see Figure 1) trains a small-size ensemble model to dynamically identify the most accurate base model (i.e., LLM or VQA model) for a given input, which we refer to as the *winner*. This is achieved by designing a meta-level classification task considering all the base models as labels for every input. We designed and implemented two ensemble architectures for textual and visual QA tasks. Our first proposed architecture, *InfoSel*-TT, uses a textual transformer (TT, 110M parameters) (Devlin et al., 2019) as the backbone to generate a textual representation of the question with the predicted answers by base models. Although *InfoSel*-TT is straightforward and effective, it cannot handle multimodal data. To address this, we propose a second architecture named *InfoSel*-MT, where we incorporate a multimodal transformer (MT, 115M parameters) (Li et al., 2020) to generate fused contextual representations of a multimodal input (image, question, and the predicted answers). These fused representations are used to train a dense layer for selecting the winning model. The challenge with this approach is the lack of exposure of the base models to new (unseen) labels appearing in the task-specific datasets. To address this, we fine-tune TT and MT models (FT-TT and FT-MT) separately to learn these new labels. The predictions of these fine-tuned models are fused with the output from *InfoSel* using a second, separately trained ***InfoSel**** ensemble model. We experiment with and without *InfoSel**, as this component is considered optional when *InfoSel* is already performing well.

To demonstrate the effectiveness of our method with limited resources. We selects three less costly LLMs (GPT-3.5-turbo (OpenAI, 2023), LLaMA-2-70b-chat (Touvron et al., 2023) and GPT3.5 text-davinci-003) and three local VQA models (ALBEF (Li et al., 2021a), BLIP (Li et al., 2022) and VLMo (Bao et al., 2022)). These models are used as base models to provide answers for textual and visual QA ensemble tasks respectively. For showing the *data efficiency* of the proposed architectures, we train them on a subsample of training data from public benchmark datasets and test on the corresponding full test data. Experimental results showcase improvements in the performance up to +5% on TQA tasks and +31% on VQA tasks when compared to the ensembled base models.

In summary, our contributions are: (1) *InfoSel*, a novel data-efficient approach to ensemble black-box models without relying on access to model architecture, weights or prediction confidences for optimizing on task-specific datasets; (2) Assessment of the performance on textual and multimodal visual QA tasks, demonstrating gains of up to +5% with *InfoSel* and up to +31% with *InfoSel**compared to ensembled base models on four benchmark datasets; (3) A detailed analysis of data efficiency, demonstrating that *InfoSel* surpasses the performance of the leading base models with as few as 10 training samples. [2]

2 Related Work

Domain Adaptation. These methods aim to improve the performance of a model on a task-specific domain by leveraging knowledge from other domains (Zhou et al., 2022). Methods such as fine-tuning (Yosinski et al., 2014), feature adaptation (Long et al., 2015) and data augmentation (Choi et al., 2019) aim to improve the performance of standalone models and thus typically require large amounts of labeled training data or access to the model architecture and weights.

Ensemble Learning. Ensemble methods generate and combine multiple learners (ML models) to address a particular ML task (Sagi and Rokach, 2018). Classical ensembling approaches like boosting (Schapire, 2013) and bagging (Breiman, 1996) are designed to train and combine a large number of individual models and are thus computationally

[2]Code and data are available at: https://github.com/Yuuxii/Black-box-QA-Ensemling.

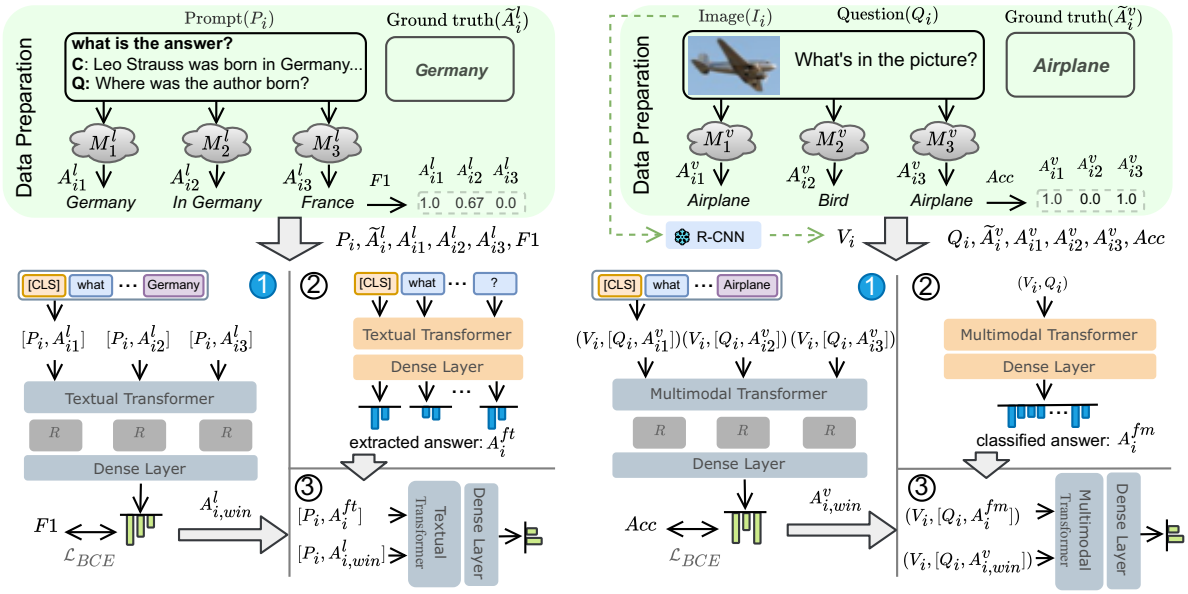


Figure 1: Architecture of our ***InfoSel*** (step ①) , FT (step ②) and ***InfoSel**** (step ③) models. M_*^l and M_*^v refer to black-box LLMs and VQA base models respectively, which are not trainable. The number of these base models is flexible and is not restricted to 3 as in the figure. The models on the left are trained for the TQA tasks, while the models on the right are trained for the VQA tasks. All our models are trained independently. Note that FT and *InfoSel** (step ② and ③) are optional and are best suited for datasets that contain a high percentage of labels that the base models have not been exposed to.

expensive. Stacking (Wolpert, 1992) uses a meta-learner to integrate the probabilities of the predictions from base models for the final output. Recent methods proposed by Jitkrittum et al. (2024); Puerto et al. (2023) require either the knowledge of confidence score or the base model's training data. Other methods train their base models to avoid dataset biases (Han et al., 2021), while Xu et al. (2020) aim to learn joint feature embeddings across different domains. However, these methods require at least one piece of knowledge that the black-box models can not provide, including base models' confidence scores or training data (not available for ChatGPT).

Multimodal Black-box Models Ensembling. Dynamic classifier selection methods, most notably OLA (Woods et al., 1997), can be applied to black-box models by ranking the best-performed local classifier dynamically in the nearest region of the input. Alternatively, majority voting (Chan and van der Schaar, 2022) and PageRank (Brin and Page, 2012) weight the predictions by their internal agreements. Yet, these methods are not designed for task-specific optimization. Pair-Ranker (Jiang et al., 2023) is task-specific but not designed for handling multimodal inputs. To address this, *InfoSel* proposes a transformer-based setup that utilizes multimodal information in the black-box setting to enhance task-specific performance.

3 InfoSel Ensemble Training

Figure 1 illustrates the proposed *InfoSel* and *InfoSel** frameworks to ensemble LLMs for TQA tasks (left), and VQA models for VQA tasks (right). We differentiate TQA components using LLMs and VQA components using VQA models by denoting them with superscripts l and v respectively. Similarly, to distinguish models used in TQA and VQA tasks, we add suffixes "-TT" and "-MT" respectively. For example, the *InfoSel*-MT model, refers to the *InfoSel* for the VQA task.

3.1 InfoSel Training for TQA

Before training *InfoSel*, we first perform the *data preparation* (top of Figure 1) for both training and testing. Next, we train *InfoSel* (step ①). Training FT models (step ②) and *InfoSel** (step ③) is optional, but can significantly improve performance for datasets that contain a high percentage of labels that the base models have not been exposed to.

Data Preparation. First, we randomly sample N content-question pairs $\{(C_i, Q_i)\}_{i=1}^N$ and the corresponding ground truth answers $\{\tilde{A}_i^l\}_{i=1}^N$ from various benchmark datasets (refer to Section

4.1). Next, we build prompts P_i following specific prompt rules $P_i = R(C_i, Q_i)$ (refer to Section 4.1). Using these prompts instead of plain (C_i, Q_i) text improves the LLMs' answer quality (Bach et al., 2022). We select K (K=3) black-box LLMs $\{M_j^l\}_{j=1}^K$ to generate answers on the N prompts. The answer generated by M_j^l on P_i is denoted as A_{ij}^l ($A_{ij}^l = M_j^l(P_i)$). Thereby, K LLMs provide $N * K$ candidate answers for N prompts. We calculate the word-level $F1$-scores (Rajpurkar et al., 2018) of all the candidate answers $\{A_{ij}^l\}_{j=1}^K$ respectively for P_i. These $F1$-scores serve as target Y_i^l to optimize the ensemble model:

$$Y_i^l = \{F1(A_{ij}^l, \widetilde{A}_i^l)\}_{j=1}^K, Y_i^l \in \mathbb{R}^K.$$

The input for the ensemble training consists of K texts. Each text is formed by concatenating P_i with each individual answer predicted by a base model j, A_{ij}^l. More formally, the input X_i^l is:

$$X_i^l = \{[P_i, A_{ij}^l]\}_{j=1}^K, |X_i^l| = K.$$

The inputs $\{X_i^l\}_{i=1}^N$ and the corresponding target labels $\{Y_i^l\}_{i=1}^N$ are used for ensemble training.

***InfoSel*-TT.** We use a textual BERT-base (Devlin et al., 2019) transformer f_θ^t, (θ denote trainable model parameters) as the backbone of *InfoSel*-TT. To achieve faster convergence, we load the pre-trained weights of *bert-base-uncased* model. The input vector X_i^l is passed to f_θ^t to generate K sentence representations for each value in X_i^l respectively. Thus, the sentence representation R_{ij}^t of $[P_i, A_{ij}^l]$ from f_θ^t is:

$$R_{ij}^t = f_\theta^t([P_i, A_{ij}^l]), R_{ij}^t \in \mathbb{R}^{768}.$$

A dense layer (f_θ^d) is followed to classify $\{R_{ij}^t\}_{j=1}^K$, and is trained to match the target label Y_i^l using binary cross entropy loss $\mathcal{L}_{BCE}$. More formally, the training objective of *InfoSel*-TT is:

$$\min_\theta \sum_{i=1}^{N} \mathcal{L}_{BCE}(f_\theta^d([f_\theta^t([P_i, A_{ij}^l])]_{j=1}^K), Y_i^l).$$

Finally, the trained *InfoSel*-TT model (M^{it}) selects the winner model $M_{i,win}^l$ from $\{M_j^l\}_{j=1}^K$ for the input P_i with the highest probability score based on the selection logits produced by f_θ^d. $A_{i,win}^l$ denotes the answer provided by $M_{i,win}^l$.

FT-TT. Using only the *InfoSel*-TT model may limit performance due to the base models' lack of exposure to new (unseen) labels in the task-specific datasets. To address this, we fine-tune a separate lightweight TT model directly on the TQA datasets to learn these new labels. Specifically, the training objective is to locate the start and end token position of the answer from the context C_i. We provide the token positions of $\widetilde{A}_i^l$ as the target label, such that the model is optimized to classify each token in two classes (start/end token). This fine-tuned textual transformer model is referred to as FT-TT (M^{ft}).[3] We denote the answer predicted by M^{ft} on P_i as A_i^{ft}.

***InfoSel**-TT.** This model performs a further ensemble training of FT-TT and *InfoSel*-TT models with the same training scheme and labeled training data as *InfoSel*-TT. We anticipate that the thus trained *InfoSel**-TT model on the output of *InfoSel*-TT and the label finetuned FT-TT, will improve the ability to handle labels unseen by base models. As a result, we expect an improvement in the overall task-specific performance. The winner model selected by *InfoSel**-TT belong to $\{M^{it}, M^{ft}\}$.

3.2 InfoSel Training for VQA

Data Preparation. Given N image-question pairs $\{(I_i, Q_i)\}_{i=1}^N$ from dev data of VQA benchmark datasets, we use K (K=3) pre-trained VQA models to predict answers A_{ij}^v as follows: $\{M_j^v((I_i, Q_i)) \rightarrow A_{ij}^v\}_{j=1}^K$. We denote the ground truth answer for image-question pair (I_i, Q_i) as $\widetilde{A}_i^v$. Target labels Y_i^v for ensemble training are given by the accuracy scores of the K candidate answers evaluated on $\widetilde{A}_i^v$:

$$Y_i^v = \{Acc(A_{ij}^v, \widetilde{A}_i^v)\}_{j=1}^K, Y_i^v \in \mathbb{R}^K.$$

The concatenation of question (Q_i) with each of the candidate answers (A_{ij}^v) obtained from the base models and the corresponding image (I_i) serves the input to our ensemble model *InfoSel*-MT:

$$X_i^v = \{(I_i, [Q_i, A_{ij}^v])\}_{j=1}^K, |X_i^v| = K.$$

***InfoSel*-MT.** A Multimodal Transformer (MT, f_θ^m) (Li et al., 2021b) is employed as the backbone for *InfoSel*-MT. Specifically, we first generate visual features V_i of I_i using a pre-trained R-CNN model (M_r) (Anderson et al., 2018). V_i is composed of a vector of the image region features v_i

[3]The training scheme is adapted from https://huggingface.co/learn/nlp-course/chapter7/7?fw=pt with the additional option to allow the model to return empty answers for unanswerable questions.

and the detected *tags* (i.e., object labels of the image) (Li et al., 2021b). The concatenated question-answer pair $[Q_i, A^v_{ij}]$ and V_i is then passed together to MT (f^m_θ) to generate a fused contextual representation R^m_{ij}:

$$V_i = (v_i, tags) = M^r(I_i),$$

$$R^m_{ij} = f^m_\theta(V_i, [Q_i, A^v_{ij}]), R^m_{ij} \in \mathbb{R}^{768}.$$

Finally, we use an additional dense layer (f^d_θ) to map R^m_{ij} to the target label Y^v_i. The training is optimized using binary cross-entropy loss:

$$\min_\theta \sum_{i=1}^{N} \mathcal{L}_{BCE}(f^d_\theta([f^m_\theta(V_i, [Q_i, A^v_{ij}])]^K_{j=1}), Y^v_i).$$

We denote M^{im} to the trained *InfoSel*-MT model, M^{im} selects the winner model $M^v_{i,win}$ from $\{M^v_j\}^K_{j=1}$ to predict answer $A^v_{i,win}$ for the image-question pair (I_i, Q_i) based on the selection logits produced by f^d_θ.

FT-MT. Similar to FT-TT, FT-MT composed of a trainable MT and a Multilayer Perceptron (MLP) is fine-tuned with the same training data as *InfoSel*-MT. Differently, FT-MT solves a multi-label classification task by classifying the fused contextual representation of Q_i (instead of $[Q_i, A^v_{ij}]$ like *InfoSel*-MT) and V_i to a predefined answer list (labels). This list contains frequent answers from the training data. As a result, a trained FT-MT model (M^{fm}) can learn to predict the unseen (new) labels (answers) contained in the task-specific datasets, but not in the pre-training data of the base models. A^{fm}_i denotes the answer predicted by M^{fm} over (I_i, Q_i). The training scheme is adapted from (Li et al., 2021b).

***InfoSel**-MT.** Similar to *InfoSel**-TT, *InfoSel**-MT model ensembles the FT-MT and *InfoSel*-MT models using the same training scheme as in *InfoSel*-MT. The winner model selected by *InfoSel**-MT belong to $\{M^{im}, M^{fm}\}$.

4 Experiments and Analysis

We first introduce the setup of our experiments, followed by detailed results and analysis.

4.1 Datasets

To demonstrate the *data efficiency* of our approach, we subsampled four publicly available benchmark datasets. This resulted in four *Mini* datasets, the train set amounts to ∼1% of the TQA datasets' and ∼10% of the VQA datasets' original full size. Table 2 presents the details of these datasets.

***Mini* Dataset**	**Source Dataset**	**Num.**	**%**
Mini-SDv2 train	SQuAD-V2 train	800	0.56
Mini-SDv2 val	SQuAD-V2 train	200	0.14
Mini-SDv2 test	SQuAD-V2 dev	11,873	8.39
Mini-NQ train	NQ-Open train	800	0.87
Mini-NQ val	NQ-Open train	200	0.22
Mini-NQ test	NQ-Open dev	3,499	3.83
Mini-GQA train	GQA dev	105,640	9.80
Mini-GQA val	GQA dev	26,422	2.45
Mini-GQA test	GQA test	12,578	1.17
Mini-Viz train	VizWiz dev	3,456	10.5
Mini-Viz val	VizWiz dev	863	2.63
Mini-Viz test	VizWiz test	8,000	24.39

Table 2: Details of the *Mini* datasets used for *InfoSel* ensemble training and testing. % stands for the percentage of the original full dataset.

TQA datasets. We generated two *Mini* datasets, Mini-SDv2 and Mini-NQ, consisting of 1,000 randomly sampled instances from SQuAD-V2 (Rajpurkar et al., 2018) and NQ-Open (Kwiatkowski et al., 2019) train splits, respectively. Details of the datasets are shown in Appendix A.1.

VQA datasets. Our results (Figure 2) reveal that VQA tasks demand a greater quantity of training samples compared to TQA tasks. Therefore, we constructed Mini-GQA and Mini-Viz datasets using a larger fraction (the dev data) of GQA (Hudson and Manning, 2019) and VizWiz (Gurari et al., 2018) datasets compared to TQA datasets.

4.2 Base Models

We experiment with ensembling GPT-3.5-turbo-0613 (ChatGPT), LLaMA-2-70b-chat (hereinafter referred to as "LLaMA") (Touvron et al., 2023) and GPT-3.5 text-davinci-003 (hereinafter referred to as "Davinci") to generate answers for TQA tasks.[4] To tackle VQA tasks, we employ three local VQA models (VLMo (Bao et al., 2022), ALBEF (Li et al., 2021a) and BLIP (Li et al., 2022)), which are pre-trained on VQA v2 dataset (Antol et al., 2015). Note that we use less costly and publicly accessible VQA models to save experimental costs.

Evaluation Metric. LLMs tend to generate contextual answers that lead to lower scores in the exact match (EM). Therefore, we mainly use the (per-answer) token-level F1-score from the official evaluation guidance of the datasets as the main evaluation metric for TQA performance. Our re-

[4]GPT3.5 text-davinci-003 is deprecated after our experiments, but this does not influence the effectiveness of our method.

Textual Question Answering					Visual Question Answering		
Model	**Mini-SDv2**		**Mini-NQ**		**Model**	**Mini-GQA**	**Mini-Viz**
	EM	**F1**	**EM**	**F1**		**Acc**	**Acc**
LLaMA-2-70b-chat	0.24	11.34	28.07	46.47	ALBEF	50.60	21.28
text-davinci-003	52.37	58.44	52.24	69.44	BLIP	48.08	20.80
ChatGPT	30.89	44.95	57.53	71.54	VLMo	48.21	19.77
Oracle	58.61	66.20	64.02	79.21	**Oracle**	65.03	-
MV	26.95	37.75	46.07	62.43	MV	51.05	21.47
WV	52.37	58.44	57.53	71.54	WV	52.10	19.43
PageRank	25.39	37.31	51.76	68.53	PageRank	51.47	21.66
OLA	47.90	55.59	54.70	70.05	OLA	48.65	20.32
PairRanker	57.28	63.33	57.96	72.21	PairRanker	52.05	22.42
(0-shot) LLM-Blender	4.90	21.20	1.03	25.06	(0-shot) LLM-Blender	0.0	0.0
FT-TT	46.80	47.68	36.52	40.60	FT-MT	50.48	51.76
***InfoSel*-TT (ours)**	**57.74**	**63.63**	**58.45**	**73.37**	***InfoSel*-MT (ours)**	**55.16**	23.16
InfoSel-TT (ours)*	49.09	49.85	48.16	53.70	***InfoSel***-MT (ours)*	52.54	**52.91**

Table 3: Test performance comparison on textual and visual QA datasets. The overall best results are highlighted in bold, and the best results of base models are underlined. The test data of Mini-Viz is not publicly accessible and thus the Oracle cannot be reported.

sults differ from the ones reported in (Laskar et al., 2023; Kocoń et al., 2023) because we do not apply any post-processing, human evaluation or output constraints for the generated answers.

Finally, we use **Oracle** to represent the maximum capability of a combination of base models. Specifically, for each input, the Oracle selects the answer with the highest agreement to the ground truth among all the answers predicted by the base models. Thus, the Oracle score represents the performance of an ideal ensemble model.

4.3 Baselines

We use Majority Voting (MV), Weighted Voting (WV) (Schick and Schütze, 2021), PageRank (Brin and Page, 2012), Overall Local Accuracy (OLA) (Woods et al., 1997), PairRanker and LLM-Blender (Jiang et al., 2023) as baselines.

Majority Voting (MV). MV makes a collective decision by considering the predicted answers as a group of individuals voting on a particular input. The answer that receives the most votes is the winner; otherwise, ties are broken randomly.

Weighted Voting (WV). We adopt a strategy similar to Schick and Schütze (2021), where the model accuracy of the train data before training is used as the weight for average weighting. In our case, we use the corresponding accuracy of the base models as the weight for voting.

PageRank (Brin and Page, 2012). We adapt PageRank as a baseline to determine the most suitable answer in a graph where all the answers to one question are connected by their BLEURT (Sellam et al., 2020) similarities.

Overall Local Accuracy (OLA) (Woods et al., 1997). Following (Cruz et al., 2018), we use the k-nearest neighbors algorithm to divide the input space (representations of prompts for TQA, representations of images and questions for VQA) of training data into 7 regions. The overall local accuracy of each base model in different regions is computed as its region competence. The model presenting the highest competence level is selected to predict the label of inputs that fall in the same region.

PairRanker and LLM-Blender (Jiang et al., 2023). The PairRanker model is trained to rank a pair of candidate predictions from two LLMs using multiple optimizing objectives (i.e., EM, F1). The logits of the LLMs produced by the trained model are sorted to get the top k (we use k=2) predictions among multiple pairwise comparison results for a GENFUSER model (Flan-T5-XL (Chung et al., 2022), 3B parameters) to generate the final fused prediction. LLM-Blender is a composite of the PairRanker and the GENFUSER model. We train PairRanker using the same textual transformer backbone as *InfoSel* for comparison, and we use 0-shot LLM-Blender models which have been trained over massive data (105k), including TQA data to test on our data.

4.4 Performance Comparison

In this section, we analyze the performance of our method, taking into account its distinctive characteristics as described in Table 1. Concretely, we focus on comparing our models in terms of *task-specific* performance, *data efficiency*, and *multi-*

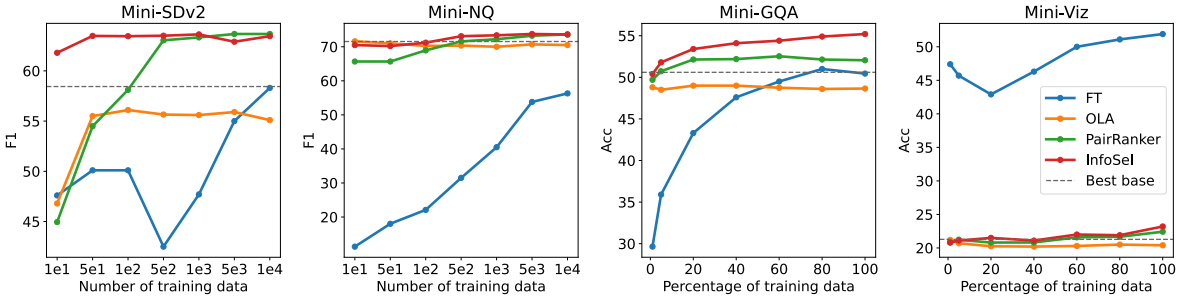


Figure 2: Test performance of *InfoSel* compared to baselines over increasing size of training data for TQA (left two figures) and VQA (right two figures). *Best base* represented the best performance of the base models.

modal capabilities.

4.4.1 Task-specific Performance

Table 3 demonstrates the *task-specific* performance of *InfoSel*, base models and ensemble baselines on textual and visual QA datasets. For TQA, we observe that LLaMA underperforms other base models. Upon closer examination, we found that LLaMA generates longer explanation text which, although often accurate, decreases the EM and F1-score values. Conversely, a more consistent performance of base models is observed for VQA. Mini-Viz contains 28% of unanswerable questions, and the label "unanswerable" has never been seen by base models. Consequently, this lack of exposure leads to significantly lower performance scores.

Baseline ensemble methods such as WV, PageRank and OLA achieve only marginal improvements compared to base models ($\leq$+1.5%) on VQA datasets. These results highlight the limitations of these methods when applied to *task-specific* datasets (see also Table 1). PairRanker underperformed *InfoSel* even though it has been trained with the same data. 0-shot LLM-Blender tends to generate longer answers compared to PairRanker which leads to a low score especially when evaluated in the exact-match settings (EM, ACC).

For TQA tasks, *InfoSel*-TT achieves 5.19% (63.63-58.44) improvement compared to individual base models, and reaches 96.12% (63.63/66.20) of the Oracle on Mini-SDv2. Similarly, the corresponding improvement in Mini-NQ performance is 1.83%, reaching 93.06% of the performance achieved by the Oracle. In contrast, FT-TT, despite its superior performance over two base models on Mini-SDv2, underperforms *InfoSel*-TT by more than 15% due to the small-size training data (refer to Figure 2). This low performance of FT-TT models negatively impacts the performance of *InfoSel**. As a result, we conclude that while *InfoSel** can exhibit superior performance (see further for VQA tasks), it also requires more training data.

For VQA tasks, the results in Table 3 showcase an improvement of 4.56% in the accuracy score of *InfoSel*-MT compared to the base models (55.16-50.60) on Mini-GQA. Furthermore, FT-MT improves 30.48% (51.76-21.28) accuracy on Mini-Viz due to a high percentage of unseen labels (e.g., "unanswerable") introduced during fine-tuning. Finally, the superior performance of *InfoSel**-MT model on Mini-Viz dataset demonstrates the effectiveness of the proposed blending approach, which improves 31.63% (52.91-21.28) accuracy upon the *InfoSel*-MT model.

4.4.2 Data Efficiency

The experimental results shown in Figure 2 demonstrate the *data efficiency* of our method by evaluating the model's performance across varying training data sizes. We observe that *InfoSel*-TT achieves a higher F1-score compared to the best base model when trained on as few as 10 samples from Mini-SDv2. This result has been further verified with the mean F1-score of 10 test results using different seed variations for sampling the training data (shown in Figure 5 in Appendix). Conversely, the number of training samples needed to surpass the performance of base models is higher for VQA datasets: 5% (6,603 samples) for Mini-GQA and 20% (864 samples) for Mini-Viz. We hypothesize that this is due to the inherent complexity of the VQA task.

PairRanker is less data-efficient than *InfoSel* as it only achieves close performance when the training samples are more than 500 on both Mini-SDv2 and Mini-NQ. Additionally, we find that a larger training data size benefits FT-TT more than *InfoSel*-TT and OLA. For example, the F1-score of FT-TT increases $\sim$200% and $\sim$500% from 10 to 10,000 training samples on Mini-SDv2 and Mini-

NQ respectively, while *InfoSel*-TT only increases by ∼3% and ∼4%. However, FT-TT still underperforms the best base model, which suggests that fine-tuning a small-sized model requires larger training data for getting a comparable performance with LLMs or *InfoSel*. Finally, we observe that a larger training data size does not necessarily lead to improved performance for the fine-tuned FT-TT model (e.g., when increasing from 80% to 100% the training data size on Mini-GQA). In contrast, OLA does not benefit as much as *InfoSel* and FT from a larger size of training data, only outperforming *InfoSel*-TT on Mini-NQ when 10 and 20 training samples are used.

4.4.3 Multimodal Data

InfoSel is able to utilize multimodal data (image and text) for VQA tasks, and thus outperform the latest text-exclusive LLM ensemble methods (PairRanker and LLM-Blender) as evidenced in Table 3. In contrast, PairRanker and LLM-Blender cannot process image features, thereby lacking crucial information in the multimodal setting, leading to a low accuracy on VQA datasets.[5] Further insights into the significance of modality information are elaborated in Section A.8 and Table 4.

4.4.4 Lightweight Model

Table 7 (Appendix) reports the parameter size of the base models and their ensemble models (*InfoSel*-TT and *InfoSel*-MT). *InfoSel* provides an efficient method for ensembling large LLMs such as ChatGPT (175B parameters) using only 110M trainable parameters. Even though only 37% ((182M-115M)/182M) trainable parameters are saved for the VQA task, we still demonstrate that *InfoSel* can effectively enhance the task-specific performance of small-size black-box VQA models, offering a lightweight solution.

4.5 Ablation Studies

Is *InfoSel* robust to the base models' individual performances? We carry out this study to assess whether *InfoSel* can effectively utilize the predictions obtained from various base models, regardless of their individual performance levels. In Figure 3, we observe a minor F1-score difference (0.15%) on the Mini-SDv2 dataset between the *InfoSel* model ensembled with and without the lowest performing

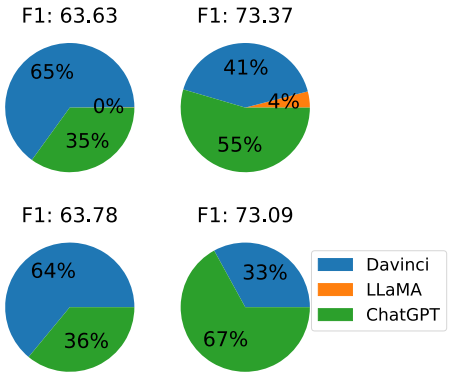


Figure 3: The portions of answers selected from different base models by *InfoSel* models on Mini-SDv2 (right column) and Mini-NQ (left column). The upper row represents the results of the *InfoSel* model ensembled with all three LLMs, and the model in the lower row excluded the worst base model (LLaMA).

base model (LLaMA). This finding suggests that *InfoSel* is robust, and not significantly affected by the individual model's performance. In a more detailed analysis, we observe that *InfoSel* selects 4% of answers from LLaMA, resulting in an overall gain of +0.28% of the F1-score. This observation highlights the effectiveness of *InfoSel*, as it can leverage the knowledge contained in the answers provided even by the lowest performing base model to some extent.

Which modality information helps the most for ensembling? In Table 4, we compare the effect of providing different modality information to *InfoSel*-MT during ensemble training. Notice that even with just the question and answer (Q+A) information, our model surpasses the performance of the PairRanker. The setting that yields the lowest accuracy solely utilizes the image (V) as the signal. This can be explained by the fact that a single image often corresponds to multiple questions in VQA datasets, making it challenging for the model to acquire discriminative features. Furthermore, we conclude that the superior performance of our model when utilizing image, question, and answer (V+Q+A) data demonstrates the effectiveness of our model in *multimodal* setting.

4.6 Case Study

The first case of Table 5 indicates that *InfoSel* captures the only correct answer ("toaster") from base models. The second case demonstrates the ability of *InfoSel** to recognize a task-specific label (i.e., "unanswerable") introduced by FT-MT. *InfoSel*-MT

[5]The most frequent answer of LLM-Blender on VQA datasets is *"I'm sorry, I don't have enough context to answer that question."*

Model	Mini-GQA	Mini-Viz
PairRanker(Q+A)	52.05	22.42
InfoSel-MT(V)	50.56	20.79
InfoSel-MT(Q)	51.11	21.21
InfoSel-MT(V+Q)	50.83	20.06
InfoSel-MT(V+A)	52.38	22.66
InfoSel-MT(Q+A)	54.76	22.89
***InfoSel*-MT(V+Q+A)**	**55.16**	**23.16**

Table 4: Accuracy of ***InfoSel***-MT models when using different input information for training compared to baseline models. V, Q, and A represent visual, question, and answer information respectively.

	Mini-GQA	Mini-Viz
Image:		
Question:	What appliance is it?	What is this product?
ALBEF	blender	refrigerator
BLIP	**toaster**	toilet
VLMo	microwave	door
FT-MT	coffee maker	**unanswerable**
***InfoSel*-MT**	**toaster**	toilet
***InfoSel**-MT*	coffee maker	**unanswerable**

Table 5: Case study of our models on Mini-GQA test and Mini-Viz validation data. Ground truth answers are bolded.

struggles with such labels as they are unseen to the base models. This showcases the benefits of training *InfoSel** models on datasets containing a high percentage of task-specific unseen labels. The corresponding case study of TQA is shown in Table 9 (Appendix).

5 Conclusion

In this paper, we propose *InfoSel*, a novel lightweight and task-specific ensemble method designed to learn the dynamic selection of the optimal model from a range of distinct black-box base LLMs. We find that using only 110M trainable parameters, our method is able to substantially increase the performance upon the best performing base LLM. Additionally, our analysis reveals that *InfoSel* remains robust regardless the incorrect predictions of the lowest performing LLM. Our findings also show that our solution is highly data-efficient. Concretely, it requires only a fraction of instances (as few as 10) from the training set to outperform base LLMs. Finally, our experimental results reveal the ability of *InfoSel* to be adapted to multimodal setting, showing a substantial increase in performance compared to state-of-the-art alternatives.

6 Limitations

InfoSel offers an effective approach to enhancing out-domain black-box model performance and addressing answer selection. However, it is important to acknowledge certain limitations that come with its application:

Dependency on Annotated Data: *InfoSel*, like many machine learning techniques, relies on a small amount of annotated training and development data specific to the new domain. While this requirement is relatively modest, and *InfoSel*'s strength is it's data efficiency (as demonstrated in the experiments), this may still pose a limitation in scenarios where obtaining such data is challenging or costly.

Limited Applicability to Open-Ended Text Generation: *InfoSel*'s primary strength lies in its ability to select the best answer from a set of base models, making it particularly valuable in question-answering scenarios. However, for more open-ended text-generation tasks, where it may be beneficial to combine multiple answers, *InfoSel*'s single-answer selection mechanism may not be the ideal choice, and future research directions may include approaches for combining several long-form answers.

API Fine-Tuning Availability: At the time of this study, *InfoSel* operates based on the assumption that many APIs do not offer the ability to fine-tune models, which is a constraint driven by the current landscape of AI services. However, since the field of AI is rapidly evolving, API providers may potentially introduce fine-tuning as a standard feature in the future. However, our experiments show that selection may still help even when one (and potentially more) of the answer models are fine-tuned.

Transparency and Explainability: *InfoSel*, like other machine learning models, which selects answers from black-box models may itself operate as a "black box". This means its decision-making process might not be readily interpretable or explainable to end-users. Pairing *InfoSel* with explainability techniques may give users a clearer understanding of how the model makes its selections.

References

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6077–6086.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. 2015. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433.

Sanjeev Arora, Rong Ge, Behnam Neyshabur, and Yi Zhang. 2018. Stronger generalization bounds for deep nets via a compression approach. In *International Conference on Machine Learning*, pages 254–263. PMLR.

Stephen Bach, Victor Sanh, Zheng Xin Yong, Albert Webson, Colin Raffel, Nihal V. Nayak, Abheesht Sharma, Taewoon Kim, M Saiful Bari, Thibault Fevry, Zaid Alyafeai, Manan Dey, Andrea Santilli, Zhiqing Sun, Srulik Ben-david, Canwen Xu, Gunjan Chhablani, Han Wang, Jason Fries, Maged Alshaibani, Shanya Sharma, Urmish Thakker, Khalid Almubarak, Xiangru Tang, Dragomir Radev, Mike Tian-jian Jiang, and Alexander Rush. 2022. PromptSource: An integrated development environment and repository for natural language prompts. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 93–104, Dublin, Ireland. Association for Computational Linguistics.

Hangbo Bao, Wenhui Wang, Li Dong, Qiang Liu, Owais Khan Mohammed, Kriti Aggarwal, Subhojit Som, Songhao Piao, and Furu Wei. 2022. Vlmo: Unified vision-language pre-training with mixture-of-modality-experts. *Advances in Neural Information Processing Systems*, 35:32897–32912.

Leo Breiman. 1996. Bagging predictors. *Machine learning*, 24:123–140.

Sergey Brin and Lawrence Page. 2012. Reprint of: The anatomy of a large-scale hypertextual web search engine. *Comput. Networks*, 56:3825–3833.

Alex Chan and Mihaela van der Schaar. 2022. Synthetic model combination: an instance-wise approach to unsupervised ensemble learning. *Advances in Neural Information Processing Systems*, 35:27797–27809.

Jaehoon Choi, Taekyung Kim, and Changick Kim. 2019. Self-ensembling with gan-based data augmentation for domain adaptation in semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6830–6840.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models. *Journal of Machine Learning Research 25 (2024) 1-53*.

Rafael MO Cruz, Robert Sabourin, and George DC Cavalcanti. 2018. Dynamic classifier selection: Recent advances and perspectives. *Information Fusion*, 41:195–216.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*.

Adam Fisch, Alon Talmor, Robin Jia, Minjoon Seo, Eunsol Choi, and Danqi Chen. 2019. MRQA 2019 shared task: Evaluating generalization in reading comprehension. In *Proceedings of 2nd Machine Reading for Reading Comprehension (MRQA) Workshop at EMNLP*.

Tao Gong, Chengqi Lyu, Shilong Zhang, Yudong Wang, Miao Zheng, Qian Zhao, Kuikun Liu, Wenwei Zhang, Ping Luo, and Kai Chen. 2023. Multimodal-gpt: A vision and language model for dialogue with humans. *arXiv preprint arXiv:2305.04790*.

Danna Gurari, Qing Li, Abigale J Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P Bigham. 2018. Vizwiz grand challenge: Answering visual questions from blind people. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3608–3617.

Xinzhe Han, Shuhui Wang, Chi Su, Qingming Huang, and Qi Tian. 2021. Greedy gradient ensemble for robust visual question answering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1584–1593.

Drew A Hudson and Christopher D Manning. 2019. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6700–6709.

Dongfu Jiang, Xiang Ren, and Bill Yuchen Lin. 2023. LLM-blender: Ensembling large language models with pairwise ranking and generative fusion. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14165–14178, Toronto, Canada. Association for Computational Linguistics.

Wittawat Jitkrittum, Neha Gupta, Aditya Krishna Menon, Harikrishna Narasimhan, Ankit Singh Rawat, and Sanjiv Kumar. 2024. When does confidence-based cascade deferral suffice? In *Advances in Neural Information Processing Systems 36 (NeurIPS 2023)*.

Jan Kocoń, Igor Cichecki, Oliwier Kaszyca, Mateusz Kochanek, Dominika Szydło, Joanna Baran, Julita Bielaniewicz, Marcin Gruza, Arkadiusz Janz, Kamil Kanclerz, Anna Kocoń, Bartłomiej Koptyra, Wiktoria Mieleszczenko-Kowszewicz, Piotr Miłkowski, Marcin Oleksy, Maciej Piasecki, Łukasz Radliński, Konrad Wojtasik, Stanisław Woźniak, and Przemysław Kazienko. 2023. ChatGPT: Jack of all trades, master of none. *Information Fusion*, 99:101861.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Md Tahmid Rahman Laskar, M Saiful Bari, Mizanur Rahman, Md Amran Hossen Bhuiyan, Shafiq Joty, and Jimmy Huang. 2023. A systematic study and comprehensive evaluation of ChatGPT on benchmark datasets. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 431–469, Toronto, Canada. Association for Computational Linguistics.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. 2022. BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 12888–12900. PMLR.

Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. 2021a. Align before fuse: Vision and language representation learning with momentum distillation. In *Advances in Neural Information Processing Systems*, volume 34, pages 9694–9705. Curran Associates, Inc.

Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. 2020. What does BERT with vision look at? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5265–5275, Online. Association for Computational Linguistics.

Liunian Harold Li, Haoxuan You, Zhecan Wang, Alireza Zareian, Shih-Fu Chang, and Kai-Wei Chang. 2021b. Unsupervised vision-and-language pre-training without parallel images and captions. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5339–5350, Online. Association for Computational Linguistics.

Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. 2015. Learning transferable features with deep adaptation networks. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 97–105, Lille, France. PMLR.

Rui Mao, Guanyi Chen, Xulang Zhang, Frank Guerin, and Erik Cambria. 2024. GPTEval: A survey on assessments of ChatGPT and GPT-4. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 7844–7866, Torino, Italia. ELRA and ICCL.

OpenAI. 2023. Chatgpt. `https://chat.openai.com/`. Mar 14 version.

OpenAI. 2024. Hello gpt-4o. Accessed: 2025-05-01.

Haritz Puerto, Gözde Şahin, and Iryna Gurevych. 2023. MetaQA: Combining expert agents for multi-skill question answering. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 3566–3580, Dubrovnik, Croatia. Association for Computational Linguistics.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Omer Sagi and Lior Rokach. 2018. Ensemble learning: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8(4):e1249.

Robert E Schapire. 2013. Explaining adaboost. In *Empirical Inference: Festschrift in Honor of Vladimir N. Vapnik*, pages 37–52. Springer.

Timo Schick and Hinrich Schütze. 2021. Exploiting cloze-questions for few-shot text classification and natural language inference. In *Proceedings of the*

16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, pages 255–269, Online. Association for Computational Linguistics.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.

Nima Tajbakhsh, Jae Y Shin, Suryakanth R Gurudu, R Todd Hurst, Christopher B Kendall, Michael B Gotway, and Jianming Liang. 2016. Convolutional neural networks for medical image analysis: Full training or fine tuning? *IEEE transactions on medical imaging*, 35(5):1299–1312.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *Preprint*, arXiv:2307.09288.

David H Wolpert. 1992. Stacked generalization. *Neural networks*, 5(2):241–259.

Kevin Woods, W. Philip Kegelmeyer, and Kevin Bowyer. 1997. Combination of multiple classifiers using local accuracy estimates. *IEEE transactions on pattern analysis and machine intelligence*, 19(4):405–410.

Yiming Xu, Lin Chen, Zhongwei Cheng, Lixin Duan, and Jiebo Luo. 2020. Open-ended visual question answering by multi-modal domain adaptation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 367–376, Online. Association for Computational Linguistics.

Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. 2014. How transferable are features in deep neural networks? *Advances in neural information processing systems*, 27.

M. Yuan, P. Bao, J. Yuan, Y. Shen, Z. Chen, Y. Xie, J. Zhao, Q. Li, Y. Chen, L. Zhang, L. Shen, and B. Dong. 2024. Large language models illuminate a progressive pathway to artificial intelligent healthcare assistant. *Medicine Plus*, 1(2):100030.

Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. 2022. Domain generalization: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

A Appendix

A.1 TQA Datasets

SQuAD-V2 (Rajpurkar et al., 2018) stands for Stanford Question Answering Dataset 2.0, a dataset designed for the task of question answering. It is an extension of the original SQuAD dataset by including over 50,000 unanswerable questions written adversarially by crowdworkers. The dataset is widely used in natural language understanding research.

NQ-Open (Kwiatkowski et al., 2019) is derived from Natural Questions and serves as an open-domain question-answering evaluation. The entirety of the questions can be addressed using the information found in the English Wikipedia. It was created for research purposes.

For Mini-NQ, we followed (Fisch et al., 2019) to use long answers as the context, and short answers as the ground truth answers. The 1,000 samples are divided into train and validation data using an 8:2 ratio, while the trained models are tested on the dev data of the original datasets due to the unavailability of original test data. We use the setup proposed in (Laskar et al., 2023) to generate the answers from LLMs.

Note that the answers of LLMs can be greatly influenced by some factors such as the use of different prompts or temperatures. However, our study does not focus on prompt engineering but rather on selecting the optimal base model to generate an answer. We will publicly release our prompts as well as the answers from LLMs for reproducibility.

A.2 VQA Datasets

GQA (Hudson and Manning, 2019) is a large-scale dataset for visual reasoning and compositional question answering research. The dataset contains over 113k images collected from a diverse set of sources and over 22 million questions. Only one ground-truth answer is provided for each image-question pair.

VizWiz (Gurari et al., 2018) is a benchmark dataset for visual question answering. It includes

Dataset	Sample Prompts
Mini-SDv2	What is the answer? Context:[context]; Question:[question]; If you can't find the answer, please respond "unanswerable". Answer:
	Answer the question depending on the context. Context: [context]; Question: [question]; If you can't find the answer, please respond "unanswerable". Answer:
Mini-NQ	Answer the question depending on the context without explanation. Context: [context]; Question: [question]; Answer:

Table 6: Our sample prompts in QA datasets. SQuAD-V2 were available in PromptSource (Bach et al., 2022) for prompt generation, we selected the prompt from PromptSource for Mini-SDv2, which contains two forms of prompts.

31K images, 250K questions, and answers collected through a mobile app for visually impaired users. 10 ground-truth answers are provided for each image-question pair.

Additionally, we compare the label differences of the pre-trained dataset (VQA v2 (Antol et al., 2015)) of VQA base models with task-specific datasets (GQA, VizWiz) for ensemble training. Figure 4 shows the top 7 most frequent answers and their percentages of GQA, VQA v2 and VizWiz. Four answers in GQA do not appear in the top list of VQA v2 and three for VizWiz (including the top frequent answer "unanswerable"). We also sample 3k most frequent answers from each dataset and calculate their percentage of overlapping, which is reported on the intersection in the figure. GQA and VizWiz have 32.9 % and 21.6% of overlap with VQA v2 respectively, showcasing significant differences between the pre-trained dataset and task-specific datasets.

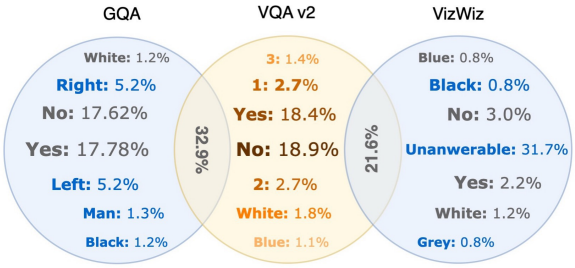


Figure 4: Top 7 most frequent answers of VQA v2 (pre-trained dataset of VQA base models), GQA and VizWiz (task-specific datasets for ensemble training). This explains why *InfoSel* performs poorly in Mini-Viz, as the new label "unanswerable" is the top frequent answer in VizWiz, but this new label has been exposed to base models that are pre-trained on VQA v2.

A.3 Base Models

LLMs		VQA Models	
Model	**#Param**	**Model**	**#Param**
LLaMA-2-70b-chat	70B	ALBEF	290M
text-davinci-003	175B	BLIP	361M
ChatGPT	175B	VLMo	182M
PairRanker	110M	PairRanker	110M
InfoSel-TT	110M	*InfoSel*-MT	115M
InfoSel*-TT	110M	*InfoSel**-MT	115M

Table 7: Parameter size of the models used in our experiments.

ChatGPT is a large language model with 175B parameters, it allows you to have human-like conversations and much more with the chatbot. ChatGPT can generate context-based responses to user prompts (questions). However, this model is currently only accessible by cloud APIs.

GPT 3.5 text-davinci-003 is similar to ChatGPT but designed specifically for instruction-following tasks which enables it to respond concisely and more accurately. This model is deprecated after our experiments.

LLaMA-2-70b-chat (Touvron et al., 2023) is a large language model with 70B parameters. It is fine-tuned on LLaMA 2 with publicly available instruction datasets and over 1 million human annotations, while Llama 2 models are trained on 2 trillion tokens from publicly available online data sources. LLaMA 2 models are currently publicly accessible.

ALBEF (Li et al., 2021a)[6] first encodes the image and text with an image encoder (visual trans-

[6] https://github.com/salesforce/ALBEF

former (Dosovitskiy et al., 2021)) and a text encoder respectively. Then a multimodal encoder is used to fuse the image features with the text features through cross-modal attention. The V&L representation is trained with objectives of image-text contrastive learning, masked language modeling and image-text matching. Different from U-VisualBERT, ALBEF uses a 6-layer transformer decoder to generate answers for VQA tasks.

BLIP (Li et al., 2022)[7] uses a visual transformer as the image encoder, and a multi-task model (multimodal mixture of encoder-decoder) as a unified model with both understanding and generation capabilities. The model is jointly pre-trained with three vision-language objectives: image-text contrastive learning, image-text matching, and image-conditioned language modeling. Similarly to ALBEF, the VQA task is considered an answer generation task in this method.

VLMo (Bao et al., 2022)[8] is a unified vision-language pre-training method with Mixture-of-Modality-Experts. VLMO leverages large-scale image and text data to learn joint representations of vision and language. It employs a mixture model to capture diverse interactions between visual and textual information, achieving state-of-the-art performance on various vision-language tasks.

A.4 Experiment Setup

We fixed the batch size to the upper limit of the server capacity, while the learning rates and epochs are selected after a grid search on a set of values (learning rates: {e3, 5e4, e4, 5e5, e5, 5e6, e6}, epochs: {3, 5, 10, 15, 20}). Models for TQA are trained for 5 epochs using a learning rate of 5×10^{-5} and batch size of 4. Models for VQA use the same learning rate but a batch size of 16 for 20 epochs. We spent ∼74 and ∼290 seconds training 1 epoch on 1,000 samples for TQA and 4,319 samples for VQA respectively. The training was performed on 1 GPU with 16GB memory of a DGX1 server ((Pascal) Tesla P100).

A.5 Multi-modal Information Concatenation or Fusion?

We studied the impact of concatenating and fusing multi-modal input information for the VQA task. *InfoSel*-MLP is an alternative model type for *InfoSel* which processes all the input information separately with a simple Multilayer perceptron (MLP) instead of MT. A pre-trained Sentence-BERT (Reimers and Gurevych, 2019) [9] M_{qa} is used for generating question embedding R^q and answer embeddings R^a.

$$R_i^q = M^{qa}(Q_i), R^q \in \mathbb{R}^{768}$$
$$R_{ij}^a = M^{qa}(A_{ij}^v), R_{ij}^a \in \mathbb{R}^{768}$$

MLP takes the concatenated representation of question, answer, and visual embeddings R_i^v as input and maps it to the label space. The objective function of *InfoSel*-MLP is formalized as:

$$\min_{\theta} \sum_{i=1}^{N} \mathcal{L}_{BCE}(MLP_\theta(\{R_i^v, [R_i^q, R_{ij}^a]\}_{j=1}^K), Y_i^v) \quad (1)$$

The input layer of the MLP maps the concatenated representations to a hidden layer with a size equal to 300, followed by a ReLU activation layer and then an output layer with an output size equal to the number of models.

Model	Mini-GQA	Mini-Viz
	ACC	
InfoSel-MLP	52.35	21.12
***InfoSel*-MT**	**55.16**	**23.16**

Table 8: Comparison of using different architecture for processing input information differently. Input concatenation result is demonstrated by *InfoSel*-MLP and the fusion result is shown by *InfoSel*-MT.

Table 8 demonstrates the performance of the input concatenation result (*InfoSel*-MLP) and fusion result (*InfoSel*-MT). We observe that *InfoSel*-MT achieves ∼3% and ∼2% higher accuracy than *InfoSel*-MLP in Mini-GQA and Mini-Viz respectively, which proves that a fused contextual representation of inputs provides more discriminative information than a concatenation of input embeddings.

A.6 Case Study for TQA

Table 9 illustrates two insightful cases from the predictions of different models on textual Mini-SDv2 and Mini-NQ QA datasets. The first case showcases the ability of *InfoSel*-TT to select the right model (Davinci) when the rest of the models is incorrect. However, *InfoSel**-TT selects the wrong answers from the FT-TT model and underperforms

[7] https://github.com/salesforce/BLIP

[8] https://github.com/microsoft/unilm/tree/master/vlmo

[9] https://huggingface.co/sentence-transformers/multi-qa-mpnet-base-dot-v1

	Mini-SDv2	Mini-NQ
Context:	...Derrick Norman Lehmer's list of primes up to 10,006,721...	...in 2005 and the release of her eponymous debut album the following year...
Question:	How many primes were included in Derrick Norman Lehmer's list of prime numbers?	When did Taylor Swift 's first album release?
LLaMA	unanswerable	2006
Davinci	**10,006,721**	2006
ChatGPT	unanswerable	2006
FT-TT	unanswerable	**2005**
***InfoSel*-TT**	**10,006,721**	2006
InfoSel-TT**	unanswerable	**2005**

Table 9: Case study of our models on Mini-SDv2 and Mini-NQ test data. Answers of LLMs are shortened to keywords for better demonstration. Ground truth answers are bolded, and one incorrect ground truth answer is colored red.

InfoSel-TT. The second case illustrates the ability of LLMs to generate correct answer ("2006") despite the ground truth annotation error ("2005"). This demonstrates the advantage of ensembling highly expressive LLMs instead of relying only on fine-tuning small-size models such as FT-TT.

A.7 Robustness Analysis.

We analyze the robustness of the data efficiency property of *InfoSel* by training *InfoSel*-TT with 10 different sets of randomly sampled train data. Figure 5 demonstrates the mean and standard deviation of these 10 sets of results on Mini-SDv2 and Mini-NQ. We observe that the deviation decreases as the number of training data increases, and the deviation is less than 0.5 when using 1,000 training samples (0.29 for Mini-SDv2, 0.40 for Mini-NQ). The mean value (61.12%) when using 10 samples from Mini-SDv2 is still better than the F1-score of the best base model, while the mean (71.60%) when using 100 samples from Mini-NQ is already higher than the F1-score of the best base model which is fewer samples than the result reported in the main paper (500 samples).

A.8 Ablation Studies

Figure 6 shows the results when removing ChatGPT and Davinci models respectively. It is not recommended to use *InfoSel* to ensemble only two base models where one of the base models performs significantly worse than the other one.

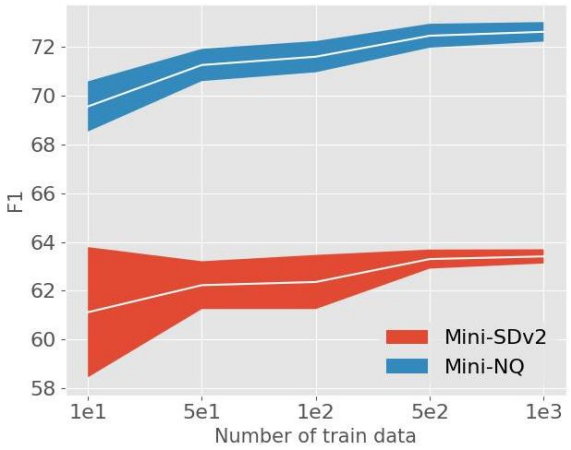


Figure 5: Mean and standard deviation of results on Mini-SDv2 and Mini-NQ when training *InfoSel*-TT with 10 sets of randomly sampled training data.

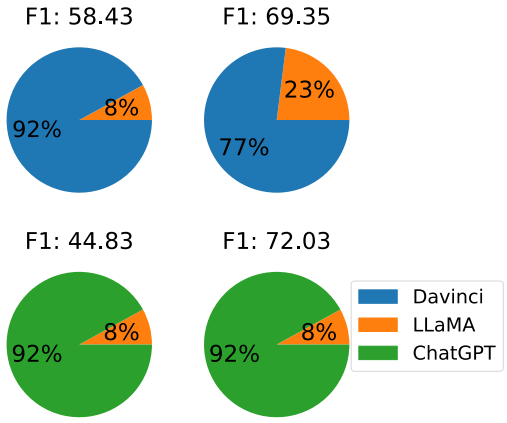


Figure 6: The portions of answers selected from different base models by *InfoSel* models on Mini-SDv2 and Mini-NQ test data. The upper row represents the results of the *InfoSel* model ensembled with Davinci and LLaMA, and the model in the lower row ensembled with ChatGPT and LLaMA.

C. Influences on LLM Calibration: A Study of Response Agreement, Loss Functions, and Prompt Styles

Authors: Yuxi Xia, Pedro Henrique Luz De Araujo, Klim Zaporojets, and Benjamin Roth

Status: Published in the Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)

DOI: 10.18653/v1/2025.acl-long.188

Reference: Xia et al. (2025a)

Work Division

Yuxi Xia: conceptualization, data curation, formal analysis, investigation, methodology, software, validation, visualization, original draft preparation, review and editing
Pedro Henrique Luz De Araujo: formal analysis, software, original draft preparation, review and editing
Klim Zaporojets: conceptualization, investigation, review and editing
Benjamin Roth: conceptualization, funding acquisition, methodology, project administration, resources, supervision, review and editing

Influences on LLM Calibration: A Study of Response Agreement, Loss Functions, and Prompt Styles

Yuxi Xia[1,2*], **Pedro Henrique Luz de Araujo**[1,2], **Klim Zaporojets**[3]
Benjamin Roth[1,4]
[1]Faculty of Computer Science, University of Vienna, Vienna, Austria
[2]UniVie Doctoral School Computer Science, Vienna, Austria
[3]Department of Computer Science, Aarhus University, Aarhus, Denmark
[4]Faculty of Philological and Cultural Studies, University of Vienna, Vienna, Austria
[*]yuxi.xia@univie.ac.at

Abstract

Calibration, the alignment between model confidence and prediction accuracy, is critical for the reliable deployment of large language models (LLMs). Existing works neglect to measure the generalization of their methods to other prompt styles and different sizes of LLMs. To address this, we define a controlled experimental setting covering 12 LLMs and four prompt styles. We additionally investigate if incorporating the response agreement of multiple LLMs and an appropriate loss function can improve calibration performance. Concretely, we build Calib-n, a novel framework that trains an auxiliary model for confidence estimation that aggregates responses from multiple LLMs to capture inter-model agreement. To optimize calibration, we integrate focal and AUC surrogate losses alongside binary cross-entropy. Experiments across four datasets demonstrate that both response agreement and focal loss improve calibration from baselines. We find that few-shot prompts are the most effective for auxiliary model-based methods, and auxiliary models demonstrate robust calibration performance across accuracy variations, outperforming LLMs' internal probabilities and verbalized confidences. [1]

1 Introduction

Improving the calibration of Large Language Models (LLMs), i.e., aligning the model's confidence with the accuracy of its predictions, can maintain their reliability, usability, and ethical deployment in domains like medicine, law, and education (Guo et al., 2017a; Jiang et al., 2021; Geng et al., 2024). As LLMs are increasingly integrated into decision-making processes, poor calibration can amplify risks of misinformation, propagate biases, and foster over-reliance among users (Raji et al., 2020).

Recent study (Ni et al., 2024) indicates that LLMs struggle to accurately express their internal confidence in natural language, particularly in the form of verbalized confidence (Tian et al., 2023). Liu et al. (2024) addresses this issue by training a linear layer to adjust the hidden states of the LLM's final layer for confidence prediction, but this approach only applies to LLMs with accessible weights. In contrast, Ulmer et al. (2024) introduces an auxiliary model for confidence estimation using only the generations of the target LLM but is constrained by its evaluation on just two LLMs and two prompt styles. These narrow experimental setups limit the generalization of their findings.

Our study tackles this limitation by comprehensively examining the generalization of different methods over 12 LLMs and four prompt styles. Additionally, we investigate new influence factors on the calibration of LLMs, which includes response agreements among LLMs, loss functions, and prompt styles. Concretely, we introduce **Calib-n** (illustrated in Fig. 1), a novel framework aggregating responses from multiple LLMs (n indicates the number of LLMs) to train a single auxiliary model for confidence estimation. Calib-n captures inter-model agreement, mitigating overfitting and reducing overconfidence associated with individual LLMs for calibration (Kim et al., 2023). Except for standard binary cross-entropy (BCE) loss, we incorporate focal (FL) (Lin et al., 2017) and AUC surrogate (AUC) (Yuan et al., 2021) loss functions that are effective for improving the calibration of non-transformer type of neural networks (Mukhoti et al., 2020; Moon et al., 2020). Finally, we systematically study the effects of prompt styles, testing four diverse prompt types: Verbalized (Tian et al., 2023), Chain-of-Thought (CoT) (Wei et al., 2022), Zero-shot, and Few-shot prompts.

Our experiments span four open-ended question answering datasets and 12 LLMs, including seven small models (2–9B parameters) and five

[1]Code and data are available at https://github.com/Yuuxii/Influences-on-LLM-Calibration.git.

Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 3740–3761
July 27 - August 1, 2025

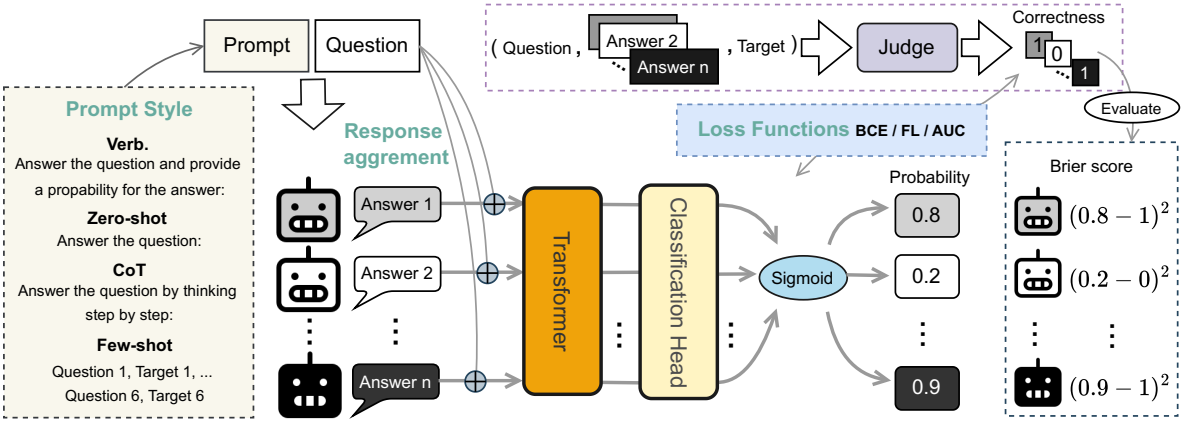


Figure 1: Overview of calibration training of Calib-n (n indicates the number of target LLMs that provide responses). We consider the effect of different **prompt styles** on calibration and design four diverse prompts to query n target LLMs to provide answers to a question. n joint strings of the question with each answer presenting the **response agreement** of LLMs are passed to the auxiliary model to generate probabilities for each answer. The auxiliary model is optimized with three **loss functions** respectively on the correctness of the LLM answers. Brier score (one of four metrics) is used to evaluate the calibration performance of the auxiliary model.

large models (27–72B parameters) from five distinct model families. The results indicate that no single method consistently outperforms all others across the various models, datasets, and prompt configurations. Yet, after aggregating the results and counting the number of overall wins for each method, we uncover new insights regarding the calibration factors of the analyzed settings:

- **Response Agreement:** By leveraging inter-model response agreement, Calib-n outperforms the state-of-the-art baselines.
- **Loss Function:** FL loss improves calibration compared to BCE and AUC losses, demonstrating its effectiveness for both Calib-1 (i.e., using responses from one LLM) and Calib-n. Calib-1 trained with FL yields the best results.
- **Prompt Style:** We find that the effectiveness of methods is highly influenced by the prompt styles, with few-shot prompts proving to be the most beneficial in improving calibration.
- **Accuracy-Calibration Correlation:** With accuracy variances caused by different dataset complexities, prompt styles and models, we find that auxiliary models maintain robust calibration performance across accuracy changes, in contrast to the fluctuating calibration of LLMs that rely on internal probabilities and verbalized confidences.

Our findings underscore the importance of reexamining calibration strategies for LLMs. Specifically, we suggest using response agreement (Calib-n) to prevent overconfidence and reduce computational costs by training a single auxiliary model for calibrating multiple target LLMs. In the case of one target LLM, we recommend incorporating focal loss over BCE (Calib-1 with FL).

2 Related Work

Calibration The concept of calibration of neural networks was introduced by Guo et al. (2017b). Lin et al. (2022) show that GPT-3 can learn to express uncertainty about its own answers without the use of logits. Later, Tian et al. (2023) demonstrate that verbalized confidence is generally better calibrated than the conditional probabilities w.r.t the consistency of the LLMs. However, Zhang et al. (2024) show that LLM probabilities and verbalized confidence tend to overly concentrate within a fixed range. Similarly, Ni et al. (2024) analyze and compare probabilistic and verbalized perceptions of the knowledge boundaries of LLM, highlighting their challenges in confidence estimation. To address these issues, Liu et al. (2024) train a linear layer to adjust the last layer's hidden states of LLMs for confidence generation. Ulmer et al. (2024) propose a method to estimate LLM confidence based only on textual input and output. However, none of these works perform a comprehensive analysis of **different influence factors in LLM calibration**. These factors can be loss functions, response agreement and prompt styles. Mukhoti et al. (2020) demonstrate that focal loss (Lin et al., 2017) can

improve the calibration of neural networks. AUC surrogate loss (Yuan et al., 2021) and correctness ranking loss (Moon et al., 2020) calibrate models by adjusting the ranking of logits to ensure positive samples are ranked higher than negative samples. Kim et al. (2023) propose that ensemble methods promoting prediction diversity can enhance calibration performance, particularly in scenarios with limited data. Min et al. (2022); Wang et al. (2023); Chen et al. (2023) showcase that LLMs output is highly dependent on the prompt and different prompt styles can significantly influence the performance of LLMs. However, these works either do not perform on LLMs or only study individual influence factors. Additionally, the impact of prompts and model agreement has never been explored in calibration. Our work comprehensively studies the influence of response agreement achieved with the ensemble method from Xia et al. (2024), three loss functions, and four prompt styles with 12 LLMs.

3 Methodology

Fig. 1 demonstrates our framework, which includes four prompt styles, assessment of the correctness of LLM generation, and calibration training of auxiliary models for confidence estimation and calibration evaluation for these models.

3.1 Prompt Styles

Prompt styles significantly impact LLMs' performance (Chen et al., 2023). Liu et al. (2024) use Few-shot prompts to query LLM answers for calibration experiments. Tian et al. (2023) employ verbalized prompts to elicit probability estimates from LLMs regarding their responses. Ulmer et al. (2024) evaluate their methods on both CoT and verbalized prompts. However, these studies either evaluate their methods against baselines using inconsistent prompt styles or fail to provide a comprehensive comparison across diverse prompt styles. For example, Liu et al. (2024) use verbalized prompts to generate verbalized probabilities as a baseline, but apply few-shot prompts to their method, which ignores the potential influence of prompts in calibration evaluation. Our work comprehensively studies the impact of prompt styles in LLM calibration and employs the most commonly used prompts: Verbalized (Verb.), Zero-shot, CoT, and Few-shot prompts. The detailed prompts are shown in Appendix A.1. Given a question q with one of the prompts, we process the answers generated by n target LLMs with regular expression-based text processing.

3.2 Correctness of LLM Generation

We employ a Judge model $\mathcal{J}$, Prometheus-8x7b-v2.0 (Kim et al., 2024), to assess the correctness of generated answers w.r.t target answers. We selected this model because it is open-source and its judgments have been shown to strongly correlate with those of human evaluators and large proprietary models (Kim et al., 2024). Given an input question q, a target answer y, and a generated answer a_i by the i-th target LLM $\mathcal{M}_i$, $\mathcal{J}$ is prompted to provide a binary correctness score c_i that reflects the semantical equivalence between a_i and y. The specific prompt is shown in Appendix A.2.

$$c_i = \mathcal{J}(a_i \overset{\text{semantic}}{=} y|q, y, a_i),\ c_i \in \{0, 1\} \quad (1)$$

3.3 Response Agreements of Multiple LLMs

Different from previous work (Liu et al., 2024; Ulmer et al., 2024; Tian et al., 2023) that estimate confidence using the information from a single target LLM, Calib-n leverages the responses from n target LLMs to train an auxiliary model for jointly estimating the confidence for each LLM. This setting is inspired by Kim et al. (2023), which suggests that combining predictions can mitigate overfitting and reduce overconfidence inherent to individual models for confidence estimation. By having access to the responses of n LLMs, the auxiliary model can infer cases of low consensus among models, signaling increased uncertainty. We verify the effectiveness of this setting by comprehensively comparing the results of using the generations produced by one LLM (**Calib-1**) to using the generations from multiple LLMs (**Calib-n**).

The auxiliary model, $f(\cdot)$, is composed of a transformer backbone (bert-base-uncased (Devlin et al., 2019)), a classification head ($n * 768 \rightarrow n$) and a sigmoid activation function, which outputs probabilities P for LLM answers $\{a_i\}_{i=1}^n$. The input to $f(\cdot)$ consists of n concatenated question–answer pairs of the form q[SEP]a_i (e.g., "What is the capital of France?[SEP]Paris"), denoted as $\{q + a_i\}_{i=1}^n$.

$$P = f(\{q + a_i\}_{i=1}^n) = \{p_i\}_{i=1}^n,\ p_i \in [0, 1] \quad (2)$$

Specifically, when the auxiliary model processes the input $\{q + a_1, q + a_2, \ldots, q + a_n\}$, each pair of question and LLM response $(q + a_i)$ is treated

as an independent input sequence. This enables the model to solve a binary classification task—predicting whether a LLM's response is correct w.r.t the target answer. The auxiliary model outputs logits for each LLM response, which are then transformed as corresponding confidence scores $\{p_1, p_2, \ldots, p_n\}$. This approach allows the auxiliary model to evaluate each LLM's response individually while still capturing inter-model agreement through the aggregation of results.

Training objective. The goal is to optimize the predicted probability to align with the correctness of the input answer. Given k questions, we minimize the average **BCE loss** of each LLM answer:

$$\mathcal{L}_{BCE} = -\frac{1}{k \cdot n} \sum_{j=1}^{k} \sum_{i=1}^{n} \Big(c_i^{(j)} \log(p_i^{(j)}) + (1 - c_i^{(j)}) \log(1 - p_i^{(j)}) \Big) \quad (3)$$

Evaluation. A target LLM $\mathcal{M}_i$ achieves a low calibration Brier score if p_i accurately reflects the reliability of a_i, which is the correctness c_i. Specifically, for all k generated answers of $\mathcal{M}_i$ regarding k questions, the Brier score (Brier, 1950) of $\mathcal{M}_i$ average squared error between all predicted probabilities and the correctness of these answers:

$$Brier(\mathcal{M}_i) = \frac{1}{k} \sum_{j=1}^{k} (p_i^{(j)} - c_i^{(j)})^2 \quad (4)$$

Following Tian et al. (2023), we also evaluate all methods with three other metrics (details in Section 4.3).

3.4 Loss Functions

To further improve the calibration, we experiment with focal and AUC losses in addition to BCE loss.

Focal loss (FL) (Lin et al., 2017; Mukhoti et al., 2020) focuses on hard-to-classify examples, reducing the weight of correctly classified samples and encouraging the model to focus on predictions with high BCE loss. This loss is commonly used in imbalanced datasets but can benefit calibration since it emphasizes predictions with a large discrepancy between confidence and correctness. The FL is defined as follows:

$$\mathcal{L}_{\text{FL}} = -\frac{1}{k} \sum_{j=1}^{k} \Big(\alpha (1 - e^{-\mathcal{L}_{BCE_j}})^{\gamma} \cdot \mathcal{L}_{BCE_j} \Big) \quad (5)$$

Where $\mathcal{L}_{BCE_j}$ is the BCE loss of the data sample j. We use the default parameters from Lin et al. (2017) to set $\alpha = 0.25$ and $\gamma = 2.0$.

AUC Surrogate loss (AUC) (Yuan et al., 2021) uses a logistic loss to maximize the differences between true and false answers' scores (logits x generated by the classification head). The equation is:

$$\mathcal{L}_{\text{AUC}} = \frac{1}{|T| \cdot |F|} \sum_{t \in T} \sum_{f \in F} \sigma (x_f - x_t) \quad (6)$$

Where T and F are the indexes set for all true and false answers respectively. σ stands for sigmoid function.

In the end, we propose the following methods to integrate the techniques mentioned above and validate their effectiveness with comprehensive experiments across different models, datasets and prompts.

(BCE)/(FL)/(AUC)Calib-1: calibration training using the generations from one target LLM and optimizing with BCE/FL/AUC loss function.

(BCE)/(FL)/(AUC)Calib-n: calibration training using the generations from n target LLM and optimizing with BCE/FL/AUC loss function.

(BCE)/(FL)/(AUC)Calib-n+PS: Platt Scaling (PS) (Platt, 1999) (explained in Section 4.4) rescales the probabilities of test data by learning on the probabilities of validation data. Those probabilities are generated by corresponding Calib-n models.

4 Experiments

To comprehensively compare confidence estimation methods, we include diverse datasets, LLMs, and state-of-the-art baselines in our experimental setting.

4.1 Datasets

We cover four open-ended quenstion-answering datasets: TriviaQA (Joshi et al., 2017), Sciq (Welbl et al., 2017), WikiQA (Yang et al., 2015), and NQ (Kwiatkowski et al., 2019). We adopt the setting from Liu et al. (2024), where 2k/1k samples are selected as training/test data for TriviaQA, Sciq, and NQ, and 1040/293 for WikiQA.

4.2 Models

We include 12 models from five families: Llama (Grattafiori et al., 2024; Touvron et al., 2023), Phi (Abdin et al., 2024), Gemma (Team et al., 2024), Qwen (Yang et al., 2024), and Mixtral (Jiang et al., 2024).[2] In our analyses, we cluster

[2]All models are available at https://huggingface.co/models.

the models based on their number of parameters:

Small models (2-9B parameters): Llama-2-7b-chat-hf (referred as Llama2-7b), Llama-3.1-8B-Instruct (Llama3.1-8b), Llama-3-8B-Instruct (Llama3-8b), Phi-3-small-128k-instruct (Phi3-7b), Phi-3.5-mini-instruct (Phi3-4b), gemma-2-2b-it (Gemma2-2b), gemma-2-9b-it (Gemma2-9b).

Large models (27-72B parameters): Qwen2-72B-Instruct (Qwen2-72b), Llama-3-70B-Instruct (Llama3-72b), Llama-3.1-70B-Instruct (Llama3.1-70b), Mixtral-8x7B-Instruct-v0.1 (Mixtral-8x7b), gemma-2-27b-it (Gemma2-27b).

Calib-n models are trained with all LLMs inside the same group and provide confidence scores for each model in that group.

4.3 Evaluation Metrics

We report four metrics for calibration evaluation. **ECE:** The expected calibration error (Guo et al., 2017b) is computed by partitioning the predictions into 10 bins based on their confidence and then taking the weighted (by the number of samples in a bin) average of the squared difference between bin average accuracy and confidence. **ECE-t:** The temperature-scaled expected calibration error (Tian et al., 2023) finds a single temperature scaling parameter β that minimizes the negative log-likelihood between model confidences and answer correctness. Then, β is used to scale the confidences before the ECE is computed. **Brier:** The Brier Score (Brier, 1950) is the average squared error between predictions' confidence and correctness (see Eq. 4). **AUC:** The area under the receiver operating characteristic curve used in Ulmer et al. (2024).

Aggregate analysis. We aggregate the performance of different confidence estimation methods by counting their number of **wins**: given each metric above, we count the number of times a method outperforms the others in all possible combinations of prompt style, dataset, and model.

4.4 Baseline Methods

We compare Calib-* with standard baselines and the state-of-the-art confidence estimation methods:

LLM Probabilities (LLM Prob.): The conditional sequence probability $P_\theta(y|x)$ of an answer y given an input x, according to the model parametrized by θ.

LLM Prob. + Platt scaling (PS): This method applies Platt scaling (Platt, 1999) to the previous baseline. That is, two scalars $a, b \in \mathcal{R}$ are used to scale the original LLM probability p: $p_{\text{ps}} = \sigma(ap + b)$, where σ is the sigmoid function. We obtain parameters a and b by minimizing the mean-squared error between model confidences and answer correctness.

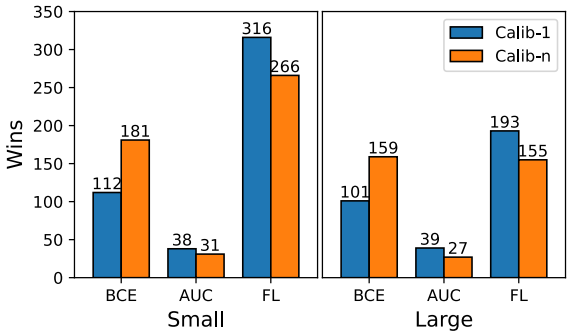


Figure 2: Comparison of Calib-1 and Calib-n methods based on the number of wins across different loss functions for calibrating small (left) and large (right) LLMs.

Verbalized confidences (Verbalized %) (Tian et al., 2023): The probability of correctness expressed in models' (text) responses given the Verb. prompts.

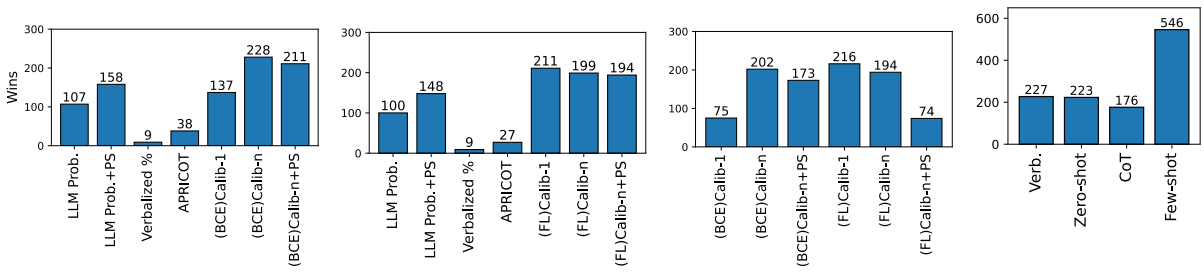
APRICOT (Ulmer et al., 2024): A recent method for calibrating LLMs. It consists of clustering related questions and measuring the per-cluster accuracies given answers from a target LLM. Then the cluster accuracies are used as the references to train an auxiliary transformer model that outputs confidence values for the target LLM.

5 Results and Analysis

The detailed results (Table 1-9) indicate that no single method consistently outperforms all others across various models, datasets, and prompt configurations. Therefore, we first present the aggregated results, which summarize all the results, followed by a more detailed discussion.

5.1 Aggregated Result

What is the best loss function for improving calibration? In Fig. 2, we compare the performance of Calib-1 and Calib-n when optimizing with different loss functions. We observe that **FL loss wins in most settings**, followed by BCE loss. Additionally, we notice that BCE outperforms FL loss when applied in the Calib-n method for large models. This is because focal loss is designed for imbalanced datasets, while Calib-n which utilizes model responses from multiple models improves the bal-



(a) (BCE)Calib-* vs Baselines (b) (FL)Calib-* vs Baselines (c) (BCE) vs (FL)Calib-* (d) Prompts

Figure 3: The winning comparison results of different methods and prompts: 3a and 3b sub-figures present the superior results of Calib-* methods using BCE and FL loss respectively when against baselines. 3c shows the comparison results among all Calib-* methods, demonstrating that (FL)Calib-1 achieves the best overall performance. 3d compares the winning result of prompt styles and shows that few-shot prompting yields the most effective calibration across diverse configurations.

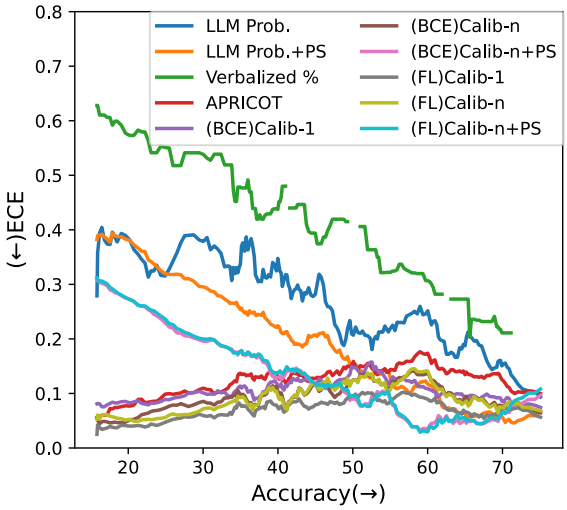


Figure 4: The correlation between accuracies achieved by different configurations (i.e., prompts, models, datasets) and corresponding ECE scores evaluated on different methods. The line of Verbalized % is not continuous because it can only be obtained using Verb. prompts and thus has fewer accuracy points than other methods. The result indicates that Calib-* and APRICOT are robust to accuracy variations. Different methods achieve the lowest ECE scores in different accuracy ranges.

ance of the training data (moderate the overall accuracy in this dataset) for the auxiliary model.

What is the best overall method? Fig. 3a and 3b present the winning comparison results of (BCE)Calib-* and (FL)Calib-* methods against baseline methods respectively. The results demonstrate that Calib-* methods gain more wins than baselines in both sub-figures. Verbalized confidences get the lowest number of wins. Applying Platt Scaling can further improve the calibration performance of LLM probabilities but this technique is not generalizable to enhance the calibration of Calib-n. Fig. 3a shows that (BCE)Calib-n gains the highest wins against the baseline methods. While (FL)Calib-1 accrues more wins in Fig. 3b. To identify the best method, we present Fig. 3c to compare the performance among our Calib-* methods, demonstrating that **(FL)Calib-1 exhibits the best overall performance**.

Which prompt style is most effective? The differences in prompt styles are well-known for their impact on the performance of LLMs. Fig. 3d showcases that prompt styles can also impact calibration performance. The results indicate that **few-shot is the most beneficial prompt** contributing to the highest wins among all other prompt styles.

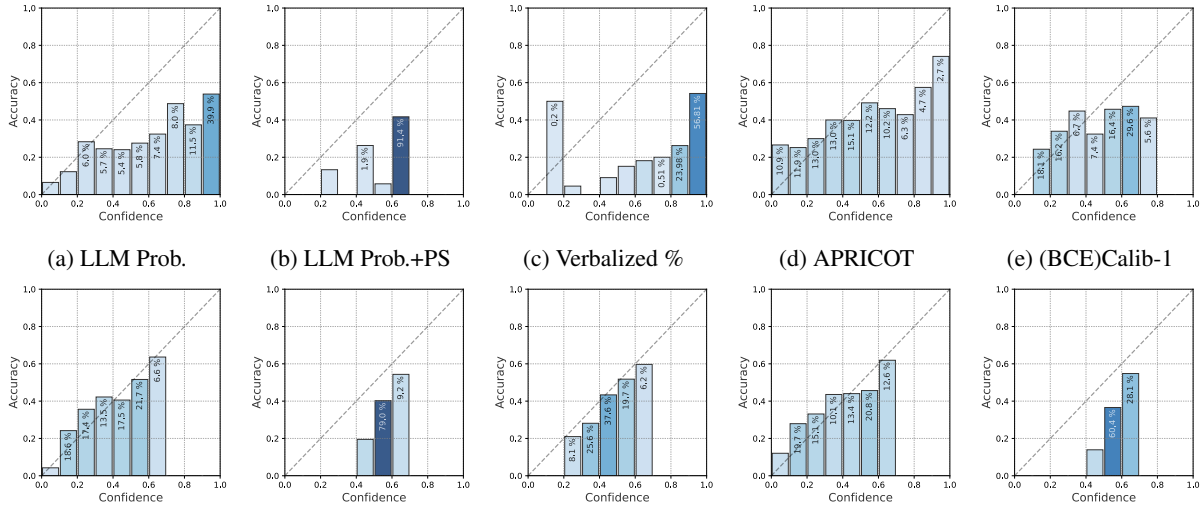
Which calibration methods maintain robustness to accuracy variations? Previous work (Zhang et al., 2024) reveals that confidence estimation methods like LLM probabilities and Verbalized confidence excessively concentrate on a fixed range (tend to be overconfident) and remain unchanged regardless of the dataset's complexity. To analyze this issue, we test all the confidence estimation methods with different datasets, prompts, and models. Each setting combination (e.g., using Zero-shot prompts to test the TriviaQA dataset on Gemma2-27b model) can result in one single accuracy value and multiple ECE scores—one for each confidence estimation method. We sort the accuracy values from all setting combinations and analyze the correlation between the accuracies and ECE scores. The results are presented in Fig. 4. We observe that ECE scores of LLM probabilities, LLM Prob.+PS, Verbalized confidences and (BCE)Calib-n+PS are highly correlated with accuracies, i.e., the ECE scores decrease when the

	Method	Verb. ECE↓	Verb. ECE-t↓	Verb. Brier ↓	Verb. AUC↑	Zero-shot ECE ↓	Zero-shot ECE-t↓	Zero-shot Brier ↓	Zero-shot AUC↑	CoT ECE ↓	CoT ECE-t↓	CoT Brier ↓	CoT AUC↑	Few-shot ECE ↓	Few-shot ECE-t↓	Few-shot Brier ↓	Few-shot AUC↑
Gemma2-27b	LLM Prob.	0.445	0.249	0.417	0.704	0.447	0.255	0.419	0.713	0.395	0.268	0.392	0.671	0.235	0.184	0.308	0.584
	LLM Prob.+PS	0.282	0.052	0.293	0.690	0.276	0.064	0.288	0.713	0.226	0.059	0.283	0.672	0.133	0.011	0.264	0.584
	Verbalized %	0.512	0.187	0.471	0.685	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.111	0.094	0.228	0.607	0.110	0.115	0.232	0.627	0.087	0.077	0.226	0.678	0.092	0.090	0.235	0.678
	(BCE)Calib-1	0.066	0.068	0.220	0.596	0.064	0.066	0.223	0.590	0.046	0.045	0.224	0.666	0.146	0.147	0.232	0.717
	(BCE)Calib-n	0.069	0.066	0.212	0.648	0.056	0.041	0.216	0.651	0.095	0.081	0.226	0.695	0.088	0.067	0.220	0.713
	(BCE)Calib-n+PS	0.202	0.062	0.255	0.637	0.198	0.046	0.257	0.651	0.170	0.076	0.255	0.695	0.103	0.122	0.241	0.713
	(FL)Calib-1	0.038	0.038	0.215	0.597	0.056	0.058	0.219	0.607	0.051	0.049	0.228	0.650	0.123	0.123	0.227	0.717
	(FL)Calib-n	0.083	0.083	0.217	0.642	0.070	0.057	0.219	0.645	0.084	0.083	0.226	0.692	0.085	0.080	0.220	0.710
	(FL)Calib-n+PS	0.203	0.057	0.254	0.642	0.191	0.056	0.255	0.645	0.167	0.079	0.254	0.692	0.101	0.120	0.241	0.710
Llama3-70b	LLM Prob.	0.445	0.302	0.430	0.712	0.458	0.259	0.426	0.747	0.412	0.251	0.416	0.612	0.336	0.117	0.364	0.532
	LLM Prob.+PS	0.355	0.113	0.350	0.703	0.266	0.132	0.291	0.747	0.224	0.023	0.289	0.612	0.224	0.009	0.298	0.532
	Verbalized %	0.406	0.084	0.380	0.698	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.137	0.130	0.255	0.596	0.111	0.116	0.239	0.635	0.114	0.117	0.244	0.656	0.106	0.068	0.236	0.683
	(BCE)Calib-1	0.075	0.071	0.231	0.610	0.066	0.067	0.225	0.641	0.066	0.045	0.232	0.662	0.119	0.121	0.232	0.709
	(BCE)Calib-n	0.058	0.062	0.223	0.644	0.040	0.025	0.215	0.670	0.096	0.088	0.229	0.690	0.061	0.062	0.213	0.730
	(BCE)Calib-n+PS	0.171	0.038	0.255	0.632	0.179	0.050	0.256	0.670	0.163	0.081	0.256	0.690	0.134	0.107	0.238	0.730
	(FL)Calib-1	0.079	0.065	0.228	0.625	0.080	0.080	0.229	0.630	0.057	0.037	0.234	0.647	0.105	0.111	0.225	0.713
	(FL)Calib-n	0.080	0.093	0.228	0.638	0.040	0.022	0.215	0.675	0.096	0.095	0.231	0.686	0.066	0.058	0.216	0.725
	(FL)Calib-n+PS	0.182	0.046	0.259	0.638	0.176	0.060	0.254	0.675	0.168	0.070	0.258	0.686	0.134	0.110	0.240	0.725
Llama3.1-70b	LLM Prob.	0.301	0.231	0.326	0.704	0.308	0.237	0.326	0.722	0.274	0.221	0.330	0.619	0.172	0.098	0.266	0.640
	LLM Prob.+PS	0.254	0.116	0.293	0.690	0.236	0.066	0.279	0.722	0.195	0.023	0.275	0.619	0.135	0.064	0.258	0.640
	Verbalized %	0.435	0.122	0.405	0.711	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.120	0.115	0.248	0.621	0.080	0.078	0.239	0.635	0.083	0.080	0.230	0.688	0.097	0.075	0.234	0.687
	(BCE)Calib-1	0.130	0.123	0.249	0.602	0.089	0.088	0.244	0.580	0.073	0.058	0.211	0.733	0.105	0.102	0.221	0.726
	(BCE)Calib-n	0.062	0.055	0.224	0.656	0.058	0.047	0.230	0.637	0.049	0.054	0.211	0.730	0.081	0.080	0.217	0.719
	(BCE)Calib-n+PS	0.157	0.015	0.258	0.606	0.143	0.078	0.254	0.637	0.138	0.121	0.246	0.730	0.084	0.099	0.237	0.719
	(FL)Calib-1	0.041	0.034	0.223	0.660	0.047	0.046	0.236	0.596	0.060	0.050	0.206	0.742	0.074	0.075	0.216	0.724
	(FL)Calib-n	0.077	0.079	0.227	0.657	0.055	0.049	0.229	0.643	0.055	0.053	0.210	0.732	0.068	0.072	0.217	0.714
	(FL)Calib-n+PS	0.169	0.051	0.257	0.657	0.147	0.060	0.255	0.643	0.142	0.117	0.248	0.732	0.092	0.104	0.238	0.714
Qwen2-72b	LLM Prob.	0.470	0.321	0.455	0.692	0.483	0.280	0.450	0.734	0.380	0.276	0.393	0.620	0.220	0.092	0.280	0.645
	LLM Prob.+PS	0.259	0.120	0.289	0.745	0.270	0.057	0.292	0.734	0.202	0.021	0.282	0.620	0.203	0.085	0.281	0.645
	Verbalized %	0.433	0.065	0.404	0.734	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.114	0.113	0.241	0.614	0.096	0.085	0.227	0.647	0.113	0.112	0.247	0.641	0.086	0.078	0.230	0.694
	(BCE)Calib-1	0.038	0.039	0.230	0.586	0.027	0.029	0.217	0.638	0.092	0.092	0.234	0.668	0.115	0.117	0.229	0.714
	(BCE)Calib-n	0.063	0.064	0.225	0.635	0.046	0.020	0.215	0.664	0.083	0.084	0.231	0.682	0.076	0.078	0.222	0.712
	(BCE)Calib-n+PS	0.166	0.039	0.255	0.625	0.185	0.047	0.256	0.664	0.137	0.084	0.252	0.682	0.117	0.091	0.244	0.712
	(FL)Calib-1	0.051	0.050	0.231	0.595	0.031	0.020	0.219	0.627	0.077	0.084	0.230	0.671	0.089	0.087	0.223	0.712
	(FL)Calib-n	0.095	0.095	0.233	0.622	0.051	0.040	0.217	0.664	0.083	0.086	0.230	0.683	0.071	0.074	0.221	0.712
	(FL)Calib-n+PS	0.183	0.032	0.261	0.622	0.188	0.048	0.257	0.664	0.135	0.083	0.251	0.683	0.116	0.090	0.244	0.712
Mixtral-8x7b	LLM Prob.	0.513	0.269	0.479	0.703	0.412	0.155	0.401	0.637	0.452	0.270	0.447	0.627	0.369	0.075	0.380	0.574
	LLM Prob.+PS	0.264	0.015	0.292	0.653	0.233	0.066	0.290	0.637	0.245	0.013	0.305	0.627	0.141	0.001	0.267	0.574
	Verbalized %	0.472	0.167	0.447	0.655	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.124	0.120	0.247	0.602	0.092	0.088	0.239	0.646	0.115	0.113	0.252	0.620	0.161	0.070	0.263	0.640
	(BCE)Calib-1	0.040	0.031	0.224	0.605	0.037	0.037	0.225	0.658	0.082	0.083	0.249	0.607	0.145	0.097	0.269	0.618
	(BCE)Calib-n	0.082	0.061	0.227	0.619	0.032	0.021	0.229	0.634	0.064	0.068	0.241	0.631	0.085	0.085	0.230	0.686
	(BCE)Calib-n+PS	0.188	0.022	0.260	0.610	0.143	0.070	0.255	0.634	0.116	0.060	0.254	0.631	0.101	0.078	0.244	0.686
	(FL)Calib-1	0.031	0.025	0.226	0.580	0.033	0.035	0.224	0.662	0.077	0.046	0.247	0.598	0.121	0.098	0.262	0.619
	(FL)Calib-n	0.093	0.083	0.231	0.618	0.024	0.025	0.227	0.643	0.078	0.076	0.242	0.631	0.072	0.077	0.230	0.682
	(FL)Calib-n+PS	0.183	0.034	0.256	0.618	0.147	0.074	0.255	0.643	0.129	0.054	0.257	0.631	0.104	0.068	0.244	0.682

Table 1: Test performance of our methods (Calib-*) and baseline methods on NQ dataset using four different prompts (Verb., Zero-shot, CoT, Few-shot). Calib-n is trained with the responses of all LLMs in the table. Only the Verb. prompt requests the LLM to provide a probability for a given answer and thus has the results for Verbalized % performance. We color the text with a scale normalized by the values gap in each column, with darker shades indicating better performance. The results of the other three datasets and seven models are shown in Appendix A.8.

accuracies get higher which verified the findings of Zhang et al. (2024). **APRICOT, Calib-1 and Calib-n are robust to the accuracy changes**, the ECE scores of these methods remain relatively stable across different accuracies. Verbalized confidence consistently shows the lowest performance across all accuracy levels.

What is the best method for different accuracy levels? The optimal goal of current state-of-the-art confidence estimation methods should not only focus on achieving a low calibration error at a narrow accuracy range but also analyze the performance of the method across different accuracy levels. We perform this analysis in Fig. 4. We observe that (FL)Calib-1 performs the best in low accuracy ranges up to 50%, and Calib-n+PS achieves the lowest ECE scores for accuracies between 50% and 70%. LLM Prob.+PS works the best for high accuracy (>70%) settings mainly because LLM probabilities are usually overconfident.



(a) LLM Prob. (b) LLM Prob.+PS (c) Verbalized % (d) APRICOT (e) (BCE)Calib-1

(f) (BCE)Calib-n (g) (BCE)Calib-n+PS (h) (FL)Calib-1 (i) (FL)Calib-n (j) (FL)Calib-n+PS

Figure 5: Reliability diagrams for our different methods using 10 bins each for Llama3.1-70b on NQ. The color and the percentage number within each bar present the proportion of total data samples contained in each bin. More figures of other models and datasets are shown in Appendix A.6.

5.2 Detailed Fraction Results

Table 1 presents the test results of our methods and baselines on the NQ dataset when evaluated on five large-size LLMs. We analyze the performance of all methods across prompts, models, and evaluation metrics. The results reveal that the effectiveness of methods is highly influenced by the prompt styles and models. However, our proposed methods (Calib-*) achieve better calibration than baselines on most prompts and models.

Across Methods: LLM probabilities and Verbalized confidence show poor performance, exhibiting high ECE, ECE-t, and Brier scores across all prompts. Adding Platt Scaling (LLM Prob.+PS) improves calibration but is still outperformed by our proposed Calib-* methods. Among baselines, APRICOT performs better than Verbalized confidence, although it does not achieve the best results in most settings. Across all evaluation metrics, our Calib-* methods outperform others. For example, (BCE)Calib-n achieves the lowest ECE and Brier scores, particularly with the CoT and few-shot prompts, e.g., ECE of 0.049 on Llama3.1-70b. PS only helps decrease the ECE-t scores but can not enhance overall performance, this is because PS is similar to ECE-t which is a posthoc method for scaling probabilities. **Across Prompts**: Calibration quality improves with few-shot and CoT prompts compared to the verb. and zero-shot prompts. For instance, (BCE)Calib-1 achieves an ECE-t of 0.045 with the CoT prompts compared to 0.068 with the verb. prompts on Gemma2-27b. **Across Models**: Larger models (e.g., Qwen2-72b and Llama3.1-70b) generally exhibit better calibration when using Calib-* methods, reflecting their better alignment with calibration techniques. For instance, (FL)Calib-n+PS achieves an ECE of 0.071 on Qwen2-72b compared to 0.095 on smaller models like Mixtral-8x7b.

Reliability diagrams analysis. Fig. 5 shows that PS produces a narrow range of confidence scores, indicating limited diversity in the emitted confidence levels. LLM probabilities and Verbalized confidence often exhibit overconfidence, even when applied to datasets with low accuracy (<40%, reported in Fig. 8 in Appendix). In contrast, our Calib-* methods show a more conservative approach, aligning their confidence levels more closely with the true accuracy of the model, reflecting improved calibration and reliability.

6 Conclusion

Previous studies (Ulmer et al., 2024; Tian et al., 2023) have assessed calibration methods within limited settings, overlooking their generalization across diverse model sizes and prompt styles. This study addresses this limitation by conducting experiments on 12 LLMs with parameters ranging from 2B-72B and four prompt styles. We also comprehensively analyze the influence of response agreement and loss functions in LLM calibration. Experimental results show that both response agree-

ment and FL loss enhance calibration from baselines, with (FL)Calib-1 achieving the best performance. We also find that few-shot prompts improve LLM accuracy and calibration, and auxiliary-based methods show robust performance across diverse settings—maintaining stable calibration regardless of accuracy levels. These findings highlight the effectiveness of various calibration strategies and encourage future methods to re-evaluate the importance of our explored factors for achieving reliable confidence estimation.

Limitations

Although our work covers a wide range of factors, there are potentially more factors worth exploring. For example, we analyze a wide range of prompt types, including Few-shot and Chain-of-Thought, the influence of fine-grained prompt variations or automatically generated prompts remains unexplored. The interplay between prompt engineering and calibration could warrant deeper investigation.

Our study focuses on calibration performance metrics like ECE, ECE-t, and Brier scores. While these metrics are widely used, they may not fully capture all aspects of calibration quality, such as user-perceived confidence or task-specific utility.

Existing works use various ways of determining the accuracy of LLM generations. For instance, some works (Tian et al., 2023; Liu et al., 2024) use LLMs as a Judge, other works (Ulmer et al., 2024; Xiong et al., 2024) use certain metrics such as extract match or ROUGE score. While the optimal solution is underexplored, we choose the more commonly used and cost-efficient method.

Future work can address these limitations by testing broader LLMs and tasks, automating prompt optimization, and developing hybrid approaches that adapt to varying accuracy levels and application constraints.

Acknowledgments

This research has been funded by the Vienna Science and Technology Fund (WWTF)[10.47379/VRG19008] "Knowledge infused Deep Learning for Natural Language Processing", and co-funded by the European Union.

References

Marah Abdin et al. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *ArXiv preprint*, abs/2404.14219.

Glenn W. Brier. 1950. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1 – 3.

Banghao Chen, Zhaofeng Zhang, Nicolas Langrené, and Shengxin Zhu. 2023. Unleashing the potential of prompt engineering in large language models: a comprehensive review.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

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. Association for Computational Linguistics.

Aaron Grattafiori et al. 2024. The llama 3 herd of models. *ArXiv preprint*, abs/2407.21783.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017a. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 1321–1330. PMLR.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017b. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 1321–1330. PMLR.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2024. Mixtral of experts.

Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. 2021. How can we know when language models know? on the calibration of language models for question answering. *Transactions of the Association for Computational Linguistics*, 9:962–977.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada. Association for Computational Linguistics.

Jaeyoung Kim, Dongbin Na, Sungchul Choi, and Sungbin Lim. 2023. Bag of tricks for in-distribution calibration of pretrained transformers. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 551–563, Dubrovnik, Croatia. Association for Computational Linguistics.

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.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. Teaching models to express their uncertainty in words. *Transactions on Machine Learning Research*.

Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. 2017. Focal loss for dense object detection. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 2999–3007. IEEE Computer Society.

Xin Liu, Muhammad Khalifa, and Lu Wang. 2024. Litcab: Lightweight language model calibration over short- and long-form responses. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Rethinking the role of demonstrations: What makes in-context learning work? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11048–11064, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jooyoung Moon, Jihyo Kim, Younghak Shin, and Sangheum Hwang. 2020. Confidence-aware learning for deep neural networks. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 7034–7044. PMLR.

Jishnu Mukhoti, Viveka Kulharia, Amartya Sanyal, Stuart Golodetz, Philip H. S. Torr, and Puneet K. Dokania. 2020. Calibrating deep neural networks using focal loss. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Shiyu Ni, Keping Bi, Lulu Yu, and Jiafeng Guo. 2024. Are large language models more honest in their probabilistic or verbalized confidence?

John Platt. 1999. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in Large Margin Classifiers*, 10(3):61–74.

Inioluwa Deborah Raji, Andrew Smart, Rebecca N White, Margaret Mitchell, Timnit Gebru, Ben Hutchinson, Jamila Smith-Loud, Daniel Theron, and Parker Barnes. 2020. Closing the ai accountability gap: Defining an end-to-end framework for internal algorithmic auditing. In *Proceedings of the 2020 conference on fairness, accountability, and transparency*, pages 33–44.

Gemma Team et al. 2024. Gemma 2: Improving open language models at a practical size. *ArXiv preprint*, abs/2408.00118.

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. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu,

Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.

Dennis Ulmer, Martin Gubri, Hwaran Lee, Sangdoo Yun, and Seong Oh. 2024. Calibrating large language models using their generations only. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15440–15459, Bangkok, Thailand. Association for Computational Linguistics.

Lean Wang, Lei Li, Damai Dai, Deli Chen, Hao Zhou, Fandong Meng, Jie Zhou, and Xu Sun. 2023. Label words are anchors: An information flow perspective for understanding in-context learning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9840–9855, Singapore. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Johannes Welbl, Nelson F. Liu, and Matt Gardner. 2017. Crowdsourcing multiple choice science questions. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 94–106, Copenhagen, Denmark. Association for Computational Linguistics.

Yuxi Xia, Kilm Zaporojets, and Benjamin Roth. 2024. Black-box model ensembling for textual and visual question answering via information fusion.

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, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jianxin Yang, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Xuejing Liu, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, Zhifang Guo, and Zhihao Fan. 2024. Qwen2 technical report.

Yi Yang, Wen-tau Yih, and Christopher Meek. 2015. WikiQA: A challenge dataset for open-domain question answering. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2013–2018, Lisbon, Portugal. Association for Computational Linguistics.

Zhuoning Yuan, Yan Yan, Milan Sonka, and Tianbao Yang. 2021. Large-scale robust deep AUC maximization: A new surrogate loss and empirical studies on medical image classification. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 3020–3029. IEEE.

Min Zhang, Jianfeng He, Taoran Ji, and Chang-Tien Lu. 2024. Don't go to extremes: Revealing the excessive sensitivity and calibration limitations of LLMs in implicit hate speech detection. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12073–12086, Bangkok, Thailand. Association for Computational Linguistics.

A Appendix

A.1 Prompts for target LLM

A.1.1 Verb. prompt

> Provide your best guess and the probability that it is correct (0.0 to 1.0) for the following question. Give ONLY the guess and probability, no other words or explanation. For example: \n\n Guess: <most likely guess, as short as possible; not a complete sentence, just the guess!>\n "Probability: <the probability between 0.0 and 1.0 that your guess is correct, without any extra commentary whatsoever; just the probability!>\n\n The question is: [Question]

A.1.2 Zero-shot prompt

> Provide your best guess for the following question. Give ONLY the guess, no other words or explanations, as short as possible; not a complete sentence, just the guess! \n\n The question is: [Question]

A.1.3 CoT prompt

> Briefly answer the following question by thinking step by step. Give the final answer (start with 'Answer: ') with minimal words at the end. \n\n The question is: [Question]

A.1.4 Few-shot prompt

> user: [Question 1] assistant: [Target 1]
> user: [Question 2] assistant: [Target 2]
> ...
> user: [Question 6] assistant: [Target 6]
> user: [Question] assistant:

A.2 Prompt for Judge

> Task Description: \n An instruction (might include an Input inside it), a response to evaluate, a reference answer that gets a score of 1, and a score rubric representing a evaluation criteria are given.
> 1. Write detailed feedback that assesses the quality of the response strictly based on the given score rubric, not evaluating in general.
> 2. After writing feedback, write a score that is an integer between 0 and 1. You should refer to the score rubric.
> 3. The output format should look as follows: "Feedback: (write a feedback for criteria) [RESULT] (an integer number between 0 and 1)"
> 4. Please do not generate any other opening, closing, and explanations.
> The instruction to evaluate:[Question]
> Response to evaluate: [LLM Answer]
> Reference Answer (Score 1): [Target]
> Score Rubrics:\n Score 0: the response and reference answer to the instruction are not semantically equivalent.\n Score 1: the response and reference answer to the instruction are semantically equivalent.
> Feedback:

A.3 Technical Details

After a grid search of hyperparameters, we trained our auxiliary models (BERT-base, 110M parameters) using a learning rate of 1e-5 and a batch size of 16 for five epochs. All experiments, including LLM inferences, are performed on a maximum of 2 NVIDIA H100 GPUs. The training time for one epoch of 2k samples is around 200 seconds on one GPU, this time can be different depends on the dataset size and number of joint LLMs in training.

A.4 Aggregated Results Across Different Configurations

Fig. 6 shows the calibration performances of different methods across different configurations such as prompt styles, model sizes and datasets. We observe that the best method is highly dependent on these factors.

Prompt specific performance. The first row of Fig. 6 presents the prompt specific performance for different methods. We observe that Calib-n always

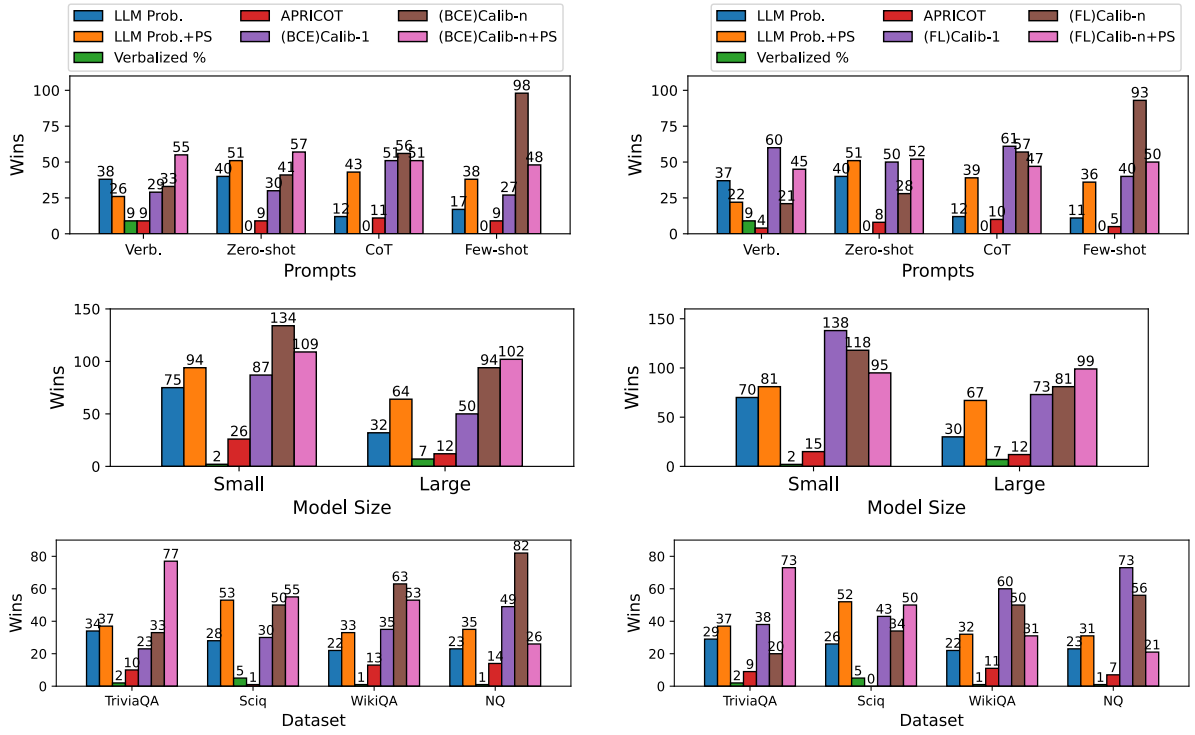


Figure 6: Performance Comparison results of different methods against baselines across three setting configurations (prompt styles, model sizes and datasets).

outperforms Calib-1 across different prompt styles when both are optimized with BCE loss. When FL loss is applied, Calib-n only outperforms Calib-1 in few-shot prompts. We hypothesize that the longer LLM answers generated with few-shot prompts usually lead to higher response agreement and thus enhance the performance of Calib-n.

Model size specific performance. The second row of Fig. 6 presents the model-size specific performance for different methods. We find that Calib-n outperforms Calib-1 on large-size models. However, (FL)Calib-1 performs better for small-size models. We also observe that the best method is model size dependent.

Dataset specific performance. The second row of Fig. 6 presents the dataset specific performance for different methods. (BCE)Calib-n consistently achieves better performances than (BCE)Calib-1 over all datasets. In contrast, (FL)Calib-1 always outperform (FL)Calib-n.

A.5 Accuracy Statistics of LLMs

We present Fig. 7 and 8 to demonstrate the accuracy performance of LLMs across different prompts and datasets. Large size models typically yield better performance than small size models. Few shot prompts improve the performance more than other prompt styles. Most LLMs achieve their highest accuracy on the Sciq dataset, while the NQ dataset proves to be the most challenging.

A.6 Reliability Diagrams

Similar to Fig. 5, Fig. 9-11 show the reliability diagrams of other three datasets for Llama3.1-70 with Verb. prompts. We find that for high accuracy (>50%) datasets (TriviaQA and Sciq), Calib-* using BCE and FL loss is less conservative and more likely to predict high confidence.

A.7 Out-domain Generalization

Table 2 presents the generalization results for the best two settings in our paper ((FL)Calib-1 and (BCE)Calib-n). We observe performance drops in the out-of-distribution (OOD) setting, but no catastrophic degradation. In some settings, the OOD auxiliary models perform on par with or better than the in-domain models.

A.8 Detailed Results of LLM Calibration

Similar to Table 1, Table 3 -9 show the rest of the detailed calibration results of different methods.

Method	Train	Test	ECE↓	ECE-t↓	Brier↓	AUC↑
(FL)Calib-1	TriviaQA	TriviaQA	0.042	0.053	0.181	0.708
(FL)Calib-1	WikiQA+NQ+Sciq	TriviaQA	0.097	0.090	**0.171**	**0.733**
(BCE)Calib-n	TriviaQA	TriviaQA	0.068	0.065	0.178	0.715
(BCE)Calib-n	WikiQA+NQ+Sciq	TriviaQA	0.089	0.074	**0.173**	0.707
(FL)Calib-1	WikiQA	WikiQA	0.073	0.068	0.193	0.574
(FL)Calib-1	TriviaQA+NQ+Sciq	WikiQA	0.250	0.100	0.280	**0.628**
(BCE)Calib-n	WikiQA	WikiQA	0.085	0.033	0.194	0.576
(BCE)Calib-n	TriviaQA+NQ+Sciq	WikiQA	0.311	0.084	0.317	**0.616**
(FL)Calib-1	NQ	NQ	0.074	0.075	0.216	0.724
(FL)Calib-1	TriviaQA+Sciq+WikiQA	NQ	0.149	0.118	0.249	0.682
(BCE)Calib-n	NQ	NQ	0.081	0.080	0.217	0.719
(BCE)Calib-n	TriviaQA+Sciq+WikiQA	NQ	0.119	0.101	0.239	0.686
(FL)Calib-1	Sciq	Sciq	0.023	0.017	0.144	0.738
(FL)Calib-1	TriviaQA+WikiQA+NQ	Sciq	0.057	0.053	0.152	0.707
(BCE)Calib-n	Sciq	Sciq	0.038	0.042	0.146	0.745
(BCE)Calib-n	TriviaQA+WikiQA+NQ	Sciq	**0.018**	**0.021**	0.147	0.696

Table 2: Generalization results test on Llama3.1-70b model using few-shot prompt. We bold the results when the out-domain outperforms in-domain performance.

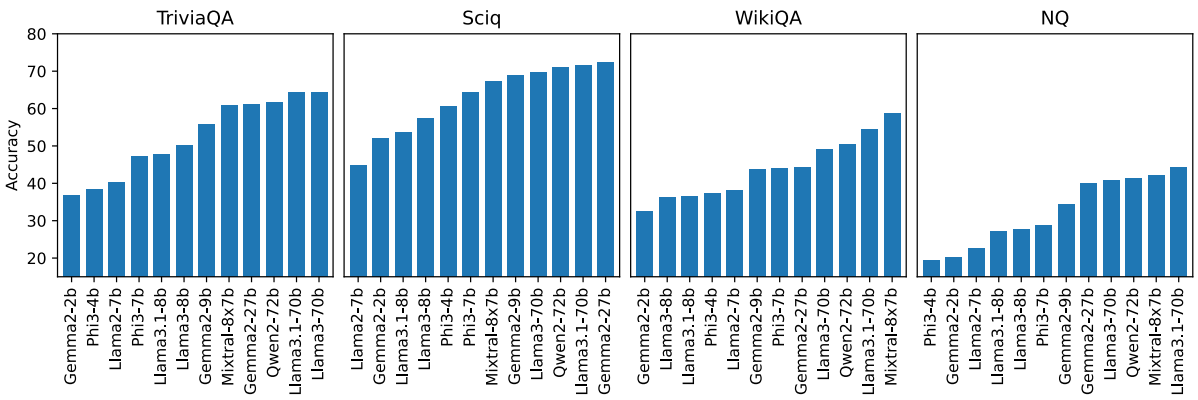


Figure 7: Model performance ranking across different datasets, performance is averaged over four prompt styles.

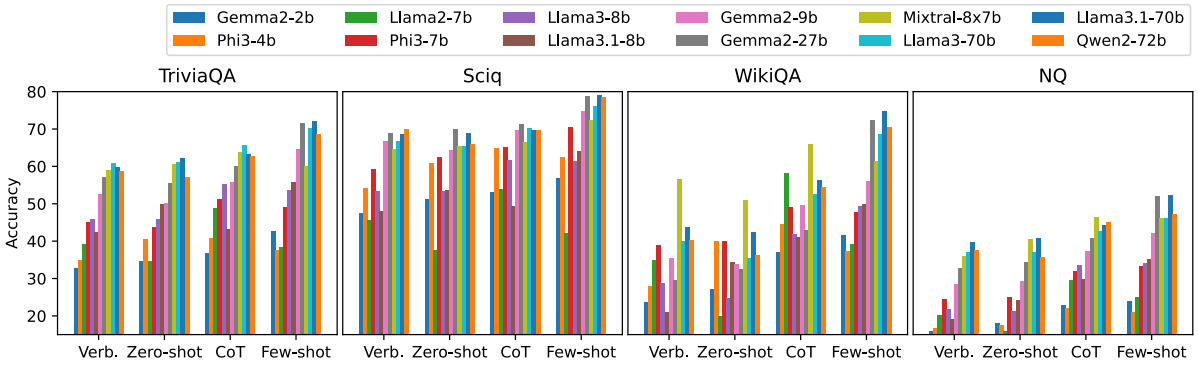
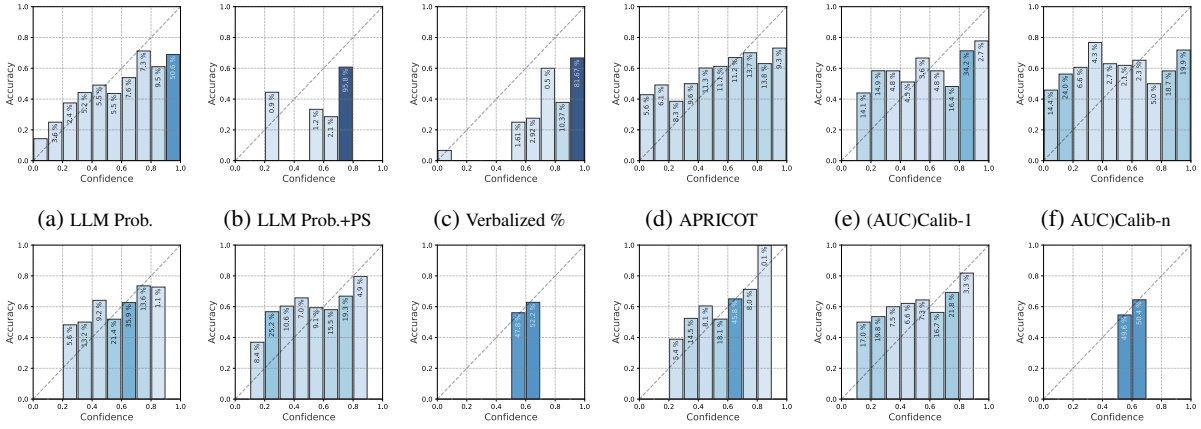


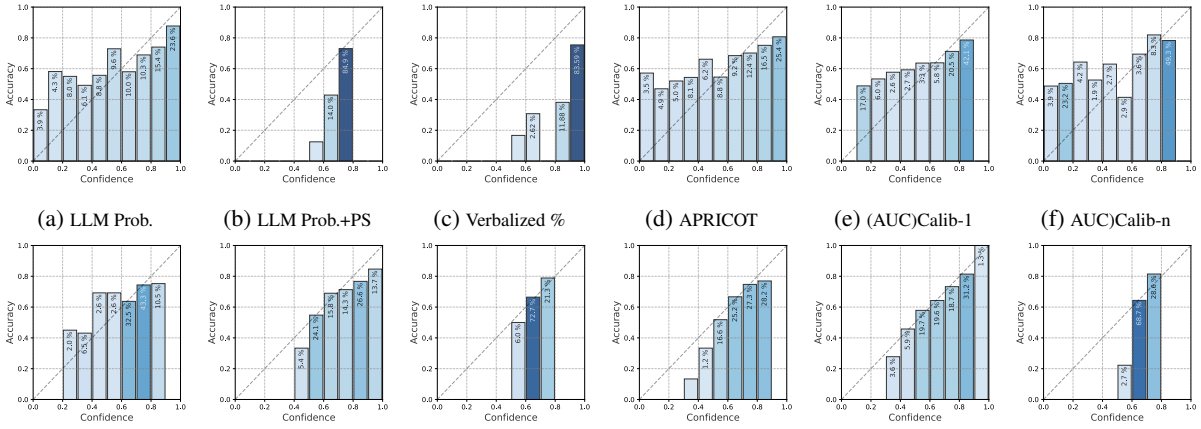
Figure 8: Model performance across different prompts and datasets.



(a) LLM Prob. (b) LLM Prob.+PS (c) Verbalized % (d) APRICOT (e) (AUC)Calib-1 (f) AUC)Calib-n

(g) (BCE)Calib-1 (h) (BCE)Calib-n (i) (BCE)Calib-n+PS (j) (FL)Calib-1 (k) (FL)Calib-n (l) (FL)Calib-n+PS

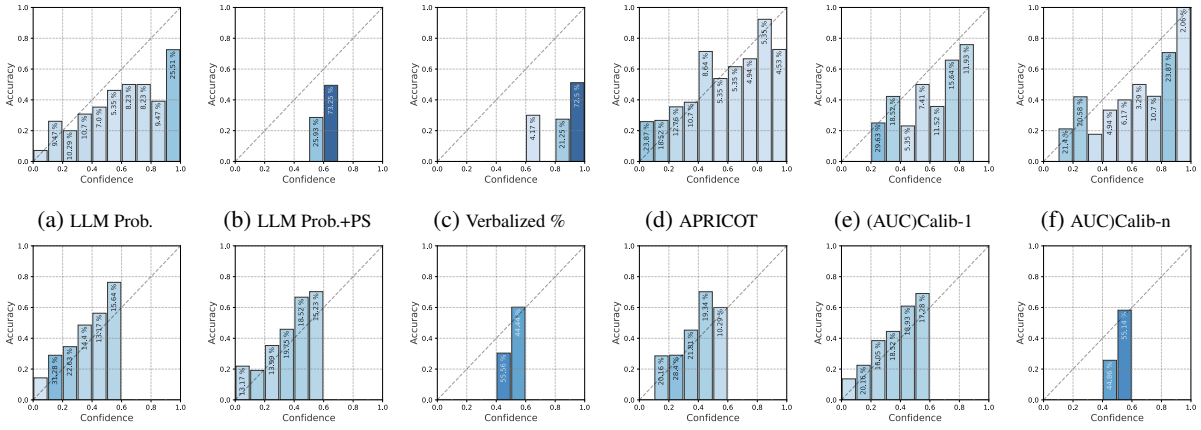
Figure 9: Reliability diagrams for Llama3.1-70b on TriviaQA with Verb. prompts.



(a) LLM Prob. (b) LLM Prob.+PS (c) Verbalized % (d) APRICOT (e) (AUC)Calib-1 (f) AUC)Calib-n

(g) (BCE)Calib-1 (h) (BCE)Calib-n (i) (BCE)Calib-n+PS (j) (FL)Calib-1 (k) (FL)Calib-n (l) (FL)Calib-n+PS

Figure 10: Reliability diagrams for Llama3.1-70b on Sciq with Verb. prompts.



(a) LLM Prob. (b) LLM Prob.+PS (c) Verbalized % (d) APRICOT (e) (AUC)Calib-1 (f) AUC)Calib-n

(g) (BCE)Calib-1 (h) (BCE)Calib-n (i) (BCE)Calib-n+PS (j) (FL)Calib-1 (k) (FL)Calib-n (l) (FL)Calib-n+PS

Figure 11: Reliability diagrams for Llama3.1-70b on WikiQA with Verb. prompts.

	Method	Verb. ECE↓	Verb. ECE-t↓	Verb. Brier ↓	Verb. AUC↑	Zero-shot ECE ↓	Zero-shot ECE-t↓	Zero-shot Brier ↓	Zero-shot AUC↑	CoT ECE ↓	CoT ECE-t↓	CoT Brier ↓	CoT AUC↑	Few-shot ECE ↓	Few-shot ECE-t↓	Few-shot Brier ↓	Few-shot AUC↑
Llama2-7b	LLM Prob.	0.437	0.239	0.401	0.737	0.296	0.170	0.291	0.746	0.314	0.180	0.324	0.693	0.419	0.042	0.400	0.643
	LLM Prob.+PS	0.256	0.065	0.299	0.613	0.246	0.092	0.266	0.746	0.165	0.113	0.261	0.708	0.249	0.054	0.295	0.643
	Verblized %	0.395	0.049	0.384	0.618	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.140	0.087	0.236	0.685	0.146	0.109	0.232	0.658	0.141	0.122	0.253	0.673	0.118	0.073	0.232	0.667
	(AUC)Calib-1	0.179	0.180	0.267	0.636	0.163	0.165	0.257	0.620	0.239	0.238	0.302	0.599	0.328	0.201	0.326	0.693
	(AUC)Calib-n	0.259	0.250	0.308	0.610	0.256	0.256	0.298	0.635	0.226	0.222	0.298	0.634	0.251	0.202	0.286	0.717
	(BCE)Calib-1	0.092	0.036	0.227	0.664	0.117	0.045	0.234	0.612	0.089	0.080	0.252	0.617	0.116	0.093	0.228	0.679
	(BCE)Calib-n	0.124	0.091	0.240	0.637	0.102	0.085	0.225	0.632	0.098	0.076	0.245	0.654	0.093	0.081	0.216	0.705
	(BCE)Calib-n+PS	0.134	0.032	0.246	0.629	0.178	0.037	0.249	0.632	0.099	0.056	0.246	0.654	0.162	0.065	0.242	0.705
	(FL)Calib-1	0.073	0.041	0.227	0.652	0.084	0.051	0.223	0.606	0.091	0.086	0.254	0.602	0.086	0.078	0.221	0.683
	(FL)Calib-n	0.121	0.093	0.243	0.617	0.098	0.079	0.226	0.620	0.093	0.065	0.241	0.662	0.098	0.080	0.216	0.702
	(FL)Calib-n+PS	0.138	0.034	0.249	0.617	0.156	0.030	0.243	0.620	0.105	0.068	0.245	0.662	0.164	0.067	0.241	0.702
Llama3-8b	LLM Prob.	0.274	0.165	0.277	0.751	0.336	0.222	0.325	0.739	0.246	0.181	0.296	0.638	0.251	0.048	0.286	0.691
	LLM Prob.+PS	0.192	0.037	0.279	0.623	0.188	0.129	0.264	0.739	0.130	0.058	0.254	0.638	0.160	0.073	0.264	0.691
	Verblized %	0.385	0.089	0.381	0.642	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.172	0.089	0.259	0.663	0.152	0.140	0.257	0.661	0.169	0.127	0.263	0.657	0.131	0.073	0.240	0.701
	(AUC)Calib-1	0.248	0.235	0.302	0.636	0.277	0.254	0.316	0.609	0.250	0.229	0.308	0.591	0.198	0.153	0.251	0.727
	(AUC)Calib-n	0.239	0.176	0.288	0.665	0.248	0.206	0.304	0.629	0.220	0.223	0.289	0.635	0.257	0.208	0.275	0.757
	(BCE)Calib-1	0.123	0.098	0.252	0.637	0.124	0.086	0.256	0.626	0.127	0.083	0.264	0.606	0.095	0.068	0.217	0.738
	(BCE)Calib-n	0.168	0.070	0.256	0.676	0.148	0.073	0.261	0.649	0.140	0.079	0.260	0.647	0.060	0.049	0.209	0.741
	(BCE)Calib-n+PS	0.100	0.051	0.243	0.635	0.088	0.052	0.244	0.649	0.068	0.048	0.237	0.647	0.100	0.111	0.233	0.741
	(FL)Calib-1	0.090	0.076	0.238	0.660	0.112	0.098	0.252	0.626	0.102	0.081	0.258	0.603	0.075	0.047	0.212	0.740
	(FL)Calib-n	0.165	0.070	0.268	0.618	0.149	0.075	0.260	0.640	0.135	0.058	0.255	0.655	0.052	0.049	0.208	0.743
	(FL)Calib-n+PS	0.094	0.034	0.245	0.618	0.080	0.048	0.243	0.640	0.071	0.054	0.237	0.655	0.108	0.109	0.232	0.743
Llama3.1-8b	LLM Prob.	0.255	0.129	0.256	0.786	0.167	0.127	0.234	0.750	0.298	0.154	0.315	0.685	0.107	0.037	0.204	0.774
	LLM Prob.+PS	0.223	0.050	0.286	0.610	0.169	0.140	0.237	0.810	0.209	0.053	0.277	0.685	0.158	0.117	0.238	0.775
	Verblized %	0.440	0.175	0.429	0.604	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.154	0.122	0.245	0.661	0.126	0.066	0.226	0.725	0.115	0.061	0.207	0.771	0.119	0.089	0.232	0.703
	(AUC)Calib-1	0.251	0.251	0.292	0.687	0.256	0.260	0.291	0.699	0.236	0.166	0.245	0.789	0.126	0.137	0.201	0.778
	(AUC)Calib-n	0.203	0.189	0.258	0.705	0.215	0.186	0.268	0.706	0.190	0.165	0.240	0.733	0.255	0.203	0.255	0.780
	(BCE)Calib-1	0.107	0.083	0.220	0.692	0.121	0.134	0.237	0.703	0.083	0.057	0.178	0.806	0.062	0.067	0.187	0.777
	(BCE)Calib-n	0.126	0.069	0.226	0.704	0.127	0.063	0.232	0.723	0.074	0.042	0.197	0.779	0.041	0.038	0.185	0.783
	(BCE)Calib-n+PS	0.146	0.060	0.244	0.643	0.119	0.122	0.238	0.723	0.176	0.140	0.234	0.779	0.135	0.155	0.224	0.783
	(FL)Calib-1	0.096	0.089	0.217	0.686	0.088	0.112	0.225	0.708	0.065	0.047	0.180	0.796	0.055	0.052	0.189	0.772
	(FL)Calib-n	0.117	0.070	0.238	0.640	0.122	0.077	0.231	0.720	0.084	0.053	0.193	0.790	0.042	0.032	0.186	0.780
	(FL)Calib-n+PS	0.148	0.057	0.244	0.640	0.116	0.117	0.240	0.720	0.178	0.157	0.234	0.790	0.137	0.152	0.224	0.780
Gemma2-2b	LLM Prob.	0.185	0.114	0.209	0.801	0.195	0.111	0.211	0.822	0.142	0.107	0.222	0.751	0.209	0.225	0.303	0.620
	LLM Prob.+PS	0.272	0.025	0.286	0.635	0.244	0.140	0.253	0.822	0.203	0.099	0.250	0.751	0.155	0.025	0.260	0.620
	Verblized %	0.526	0.099	0.493	0.667	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.150	0.073	0.221	0.691	0.143	0.129	0.236	0.658	0.148	0.097	0.232	0.685	0.145	0.093	0.247	0.671
	(AUC)Calib-1	0.189	0.203	0.268	0.637	0.208	0.208	0.273	0.633	0.206	0.216	0.280	0.654	0.291	0.205	0.317	0.677
	(AUC)Calib-n	0.240	0.251	0.296	0.644	0.269	0.270	0.311	0.637	0.251	0.234	0.303	0.672	0.394	0.303	0.393	0.690
	(BCE)Calib-1	0.120	0.081	0.221	0.635	0.103	0.070	0.226	0.639	0.114	0.046	0.226	0.671	0.092	0.101	0.233	0.665
	(BCE)Calib-n	0.094	0.099	0.213	0.653	0.104	0.094	0.227	0.629	0.073	0.074	0.213	0.687	0.128	0.106	0.240	0.685
	(BCE)Calib-n+PS	0.184	0.052	0.242	0.651	0.185	0.037	0.253	0.629	0.176	0.081	0.249	0.687	0.163	0.066	0.257	0.685
	(FL)Calib-1	0.049	0.027	0.203	0.658	0.068	0.049	0.219	0.637	0.067	0.035	0.216	0.680	0.084	0.090	0.238	0.645
	(FL)Calib-n	0.088	0.072	0.211	0.644	0.098	0.096	0.227	0.621	0.065	0.062	0.209	0.697	0.129	0.105	0.240	0.683
	(FL)Calib-n+PS	0.195	0.055	0.246	0.644	0.178	0.033	0.248	0.621	0.160	0.082	0.244	0.697	0.162	0.067	0.257	0.683
Gemma2-9b	LLM Prob.	0.244	0.180	0.265	0.767	0.276	0.207	0.289	0.762	0.217	0.165	0.269	0.688	0.176	0.177	0.260	0.589
	LLM Prob.+PS	0.126	0.042	0.260	0.679	0.177	0.119	0.258	0.762	0.120	0.072	0.245	0.689	0.042	0.039	0.225	0.589
	Verblized %	0.406	0.118	0.395	0.697	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.168	0.106	0.264	0.675	0.183	0.144	0.279	0.633	0.179	0.142	0.271	0.655	0.164	0.111	0.246	0.661
	(AUC)Calib-1	0.248	0.210	0.310	0.590	0.247	0.228	0.304	0.603	0.238	0.239	0.298	0.604	0.160	0.164	0.240	0.670
	(AUC)Calib-n	0.284	0.193	0.327	0.604	0.266	0.205	0.315	0.617	0.218	0.225	0.289	0.635	0.266	0.229	0.283	0.685
	(BCE)Calib-1	0.146	0.106	0.268	0.606	0.101	0.097	0.258	0.598	0.090	0.053	0.249	0.613	0.099	0.087	0.225	0.652
	(BCE)Calib-n	0.230	0.078	0.304	0.609	0.176	0.088	0.279	0.618	0.145	0.080	0.260	0.648	0.068	0.078	0.218	0.679
	(BCE)Calib-n+PS	0.048	0.039	0.241	0.625	0.069	0.038	0.243	0.618	0.056	0.054	0.237	0.648	0.035	0.037	0.218	0.679
	(FL)Calib-1	0.126	0.102	0.264	0.600	0.102	0.103	0.258	0.602	0.090	0.074	0.251	0.621	0.098	0.073	0.224	0.657
	(FL)Calib-n	0.224	0.061	0.299	0.609	0.175	0.082	0.279	0.610	0.141	0.061	0.256	0.653	0.070	0.079	0.218	0.678
	(FL)Calib-n+PS	0.039	0.039	0.243	0.609	0.082	0.043	0.244	0.610	0.067	0.048	0.238	0.653	0.034	0.034	0.218	0.678
Phi3-4b	LLM Prob.	0.331	0.140	0.287	0.832	0.321	0.137	0.277	0.832	0.301	0.155	0.289	0.776	0.135	0.041	0.233	0.674
	LLM Prob.+PS	0.273	0.051	0.266	0.801	0.232	0.177	0.266	0.832	0.223	0.135	0.267	0.776	0.212	0.040	0.272	0.674
	Verblized %	0.472	0.066	0.437	0.753	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.139	0.068	0.205	0.728	0.125	0.114	0.226	0.712	0.164	0.116	0.249	0.674	0.161	0.070	0.251	0.636
	(AUC)Calib-1	0.183	0.176	0.236	0.711	0.215	0.231	0.290	0.655	0.218	0.220	0.285	0.636	0.277	0.188	0.295	0.717
	(AUC)Calib-n	0.219	0.222	0.269	0.696	0.277	0.287	0.319	0.637	0.238	0.217	0.296	0.667	0.276	0.195	0.289	0.731
	(BCE)Calib-1	0.133	0.081	0.210	0.712	0.103	0.049	0.229	0.682	0.122	0.059	0.243	0.648	0.156	0.046	0.243	0.673
	(BCE)Calib-n	0.098	0.097	0.199	0.714	0.110	0.108	0.238	0.641	0.084	0.065	0.224	0.677	0.053	0.049	0.203	0.728
	(BCE)Calib-n+PS	0.199	0.060	0.239	0.708	0.151	0.041	0.249	0.641	0.139	0.071	0.246	0.677	0.160	0.086	0.240	0.728
	(FL)Calib-1	0.107	0.089	0.203	0.706	0.084	0.040	0.222	0.693	0.103	0.057	0.236	0.648	0.124	0.052	0.232	0.666
	(FL)Calib-n	0.101	0.093	0.201	0.699	0.104	0.099	0.239	0.631	0.070	0.057	0.222	0.678	0.052	0.046	0.203	0.726
	(FL)Calib-n+PS	0.205	0.064	0.241	0.699	0.167	0.025	0.261	0.631	0.148	0.082	0.248	0.678	0.161	0.087	0.240	0.726
Phi3-7b	LLM Prob.	0.234	0.176	0.261	0.783	0.276	0.186	0.272	0.798	0.182	0.171	0.269	0.681	0.085	0.057	0.242	0.636
	LLM Prob.+PS	0.227	0.053	0.278	0.727	0.196	0.128	0.256	0.798	0.131	0.067	0.250	0.681	0.123	0.043	0.256	0.636
	Verblized %	0.369	0.150	113.60(	0.724	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.126	0.121	0.247	0.672	0.120	0.097	0.239	0.683	0.174	0.138	0.268	0.660	0.196	0.073	0.274	0.650
	(AUC)Calib-1	0.249	0.238	0.302	0.612	0.292	0.281	0.330	0.568	0.249	0.250	0.303	0.621	0.185	0.206	0.273	0.681
	(AUC)Calib-n	0.260	0.230	0.314	0.602	0.283	0.274	0.325	0.609	0.191	0.198	0.276	0.653	0.263	0.223	0.297	0.699
	(BCE)Calib-1	0.104	0.084	0.249	0.623	0.111	0.098	0.252	0.593	0.082	0.072	0.245	0.641	0.106	0.090	0.247	0.655
	(BCE)Calib-n	0.148	0.083	0.262	0.636	0.137	0.104	0.256	0.623	0.137	0.059	0.252	0.665	0.091	0.075	0.230	0.690
	(BCE)Calib-n+PS	0.092	0.061	0.244	0.638	0.114	0.047	0.248	0.623	0.091	0.065	0.240	0.665	0.094	0.062	0.241	0.690
	(FL)Calib-1	0.090	0.092	0.247	0.615	0.062	0.052	0.242	0.602	0.077	0.065	0.245	0.639	0.089	0.054	0.234	0.674
	(FL)Calib-n	0.148	0.082	0.262	0.624	0.130	0.109	0.258	0.609	0.139	0.047	0.250	0.667	0.088	0.068	0.230	0.689
	(FL)Calib-n+PS	0.099	0.038	0.246	0.624	0.103	0.041	0.248	0.609	0.095	0.080	0.240	0.667	0.090	0.069	0.241	0.689

Table 3: Test results of small-size models on TriviaQA dataset.

	Method	Verb.				Zero-shot				CoT				Few-shot			
		ECE↓	ECE-t↓	Brier ↓	AUC↑	ECE ↓	ECE-t↓	Brier ↓	AUC↑	ECE ↓	ECE-t↓	Brier ↓	AUC↑	ECE ↓	ECE-t↓	Brier ↓	AUC↑
Gemma2-27b	LLM Prob.	0.294	0.251	0.308	0.700	0.315	0.227	0.312	0.729	0.257	0.254	0.307	0.593	0.136	0.173	0.226	0.563
	LLM Prob.+PS	0.100	0.016	0.249	0.679	0.144	0.091	0.250	0.729	0.087	0.018	0.242	0.594	0.013	0.020	0.200	0.563
	Verblized %	0.347	0.122	0.345	0.689	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.195	0.168	0.288	0.605	0.183	0.152	0.277	0.622	0.175	0.133	0.268	0.639	0.096	0.095	0.209	0.637
	(AUC)Calib-1	0.243	0.209	0.302	0.596	0.217	0.204	0.295	0.588	0.242	0.203	0.298	0.587	0.171	0.152	0.241	0.630
	(AUC)Calib-n	0.304	0.219	0.337	0.589	0.284	0.220	0.321	0.604	0.261	0.213	0.310	0.585	0.217	0.211	0.241	0.626
	(BCE)Calib-1	0.135	0.104	0.262	0.625	0.107	0.109	0.259	0.594	0.097	0.081	0.252	0.591	0.098	0.085	0.207	0.621
	(BCE)Calib-n	0.163	0.114	0.270	0.598	0.120	0.098	0.257	0.603	0.102	0.097	0.250	0.591	0.107	0.101	0.212	0.605
	(BCE)Calib-n+PS	0.018	0.029	0.240	0.594	0.055	0.033	0.244	0.603	0.022	0.007	0.236	0.591	0.048	0.012	0.201	0.605
	(FL)Calib-1	0.118	0.059	0.254	0.621	0.111	0.093	0.258	0.593	0.080	0.049	0.240	0.589	0.095	0.053	0.206	0.614
	(FL)Calib-n	0.162	0.122	0.275	0.593	0.105	0.077	0.250	0.623	0.109	0.097	0.251	0.594	0.121	0.109	0.216	0.601
	(FL)Calib-n+PS	0.031	0.015	0.240	0.593	0.042	0.042	0.242	0.623	0.013	0.012	0.235	0.594	0.051	0.016	0.201	0.601
Llama3-70b	LLM Prob.	0.308	0.278	0.324	0.659	0.308	0.295	0.327	0.649	0.254	0.230	0.289	0.569	0.149	0.095	0.227	0.601
	LLM Prob.+PS	0.109	0.091	0.239	0.665	0.098	0.040	0.241	0.649	0.048	0.002	0.225	0.569	0.028	0.029	0.207	0.601
	Verblized %	0.282	0.069	0.298	0.673	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.178	0.150	0.278	0.589	0.176	0.134	0.279	0.593	0.184	0.124	0.267	0.610	0.118	0.094	0.213	0.677
	(AUC)Calib-1	0.244	0.219	0.306	0.575	0.209	0.172	0.285	0.611	0.210	0.179	0.286	0.577	0.145	0.154	0.210	0.680
	(AUC)Calib-n	0.314	0.210	0.344	0.578	0.283	0.190	0.318	0.611	0.282	0.227	0.311	0.570	0.147	0.149	0.204	0.706
	(BCE)Calib-1	0.113	0.084	0.252	0.595	0.114	0.101	0.252	0.614	0.137	0.098	0.254	0.559	0.081	0.062	0.194	0.697
	(BCE)Calib-n	0.176	0.124	0.277	0.590	0.148	0.087	0.262	0.608	0.121	0.107	0.247	0.578	0.123	0.072	0.209	0.693
	(BCE)Calib-n+PS	0.021	0.001	0.234	0.596	0.030	0.029	0.233	0.608	0.023	0.036	0.223	0.578	0.101	0.077	0.202	0.693
	(FL)Calib-1	0.098	0.104	0.257	0.571	0.123	0.088	0.254	0.609	0.124	0.099	0.255	0.573	0.095	0.062	0.199	0.687
	(FL)Calib-n	0.180	0.126	0.282	0.584	0.138	0.061	0.253	0.625	0.136	0.092	0.251	0.578	0.139	0.063	0.214	0.692
	(FL)Calib-n+PS	0.012	0.010	0.234	0.584	0.039	0.042	0.232	0.625	0.028	0.031	0.223	0.578	0.108	0.073	0.203	0.692
Llama3.1-70b	LLM Prob.	0.203	0.215	0.271	0.659	0.212	0.234	0.267	0.670	0.168	0.150	0.254	0.619	0.065	0.064	0.172	0.716
	LLM Prob.+PS	0.163	0.096	0.262	0.657	0.073	0.046	0.225	0.670	0.039	0.048	0.224	0.619	0.105	0.087	0.188	0.716
	Verblized %	0.306	0.131	0.311	0.661	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.149	0.123	0.261	0.611	0.164	0.134	0.266	0.605	0.161	0.109	0.244	0.653	0.087	0.082	0.183	0.692
	(AUC)Calib-1	0.206	0.195	0.278	0.609	0.216	0.201	0.283	0.587	0.219	0.188	0.272	0.596	0.113	0.120	0.177	0.714
	(AUC)Calib-n	0.310	0.225	0.340	0.585	0.294	0.223	0.327	0.586	0.269	0.229	0.307	0.572	0.157	0.162	0.195	0.725
	(BCE)Calib-1	0.066	0.067	0.244	0.583	0.070	0.053	0.236	0.592	0.047	0.027	0.225	0.620	0.072	0.072	0.179	0.710
	(BCE)Calib-n	0.172	0.129	0.274	0.590	0.147	0.106	0.266	0.583	0.100	0.094	0.245	0.591	0.068	0.065	0.178	0.715
	(BCE)Calib-n+PS	0.010	0.011	0.237	0.584	0.031	0.001	0.233	0.583	0.004	0.004	0.228	0.591	0.115	0.096	0.191	0.715
	(FL)Calib-1	0.055	0.064	0.241	0.602	0.092	0.090	0.248	0.580	0.048	0.033	0.225	0.624	0.042	0.053	0.181	0.708
	(FL)Calib-n	0.177	0.125	0.278	0.586	0.135	0.092	0.261	0.592	0.110	0.083	0.246	0.603	0.067	0.063	0.181	0.714
	(FL)Calib-n+PS	0.001	0.001	0.236	0.586	0.024	0.006	0.232	0.592	0.022	0.014	0.227	0.603	0.126	0.098	0.192	0.714
Qwen2-72b	LLM Prob.	0.305	0.253	0.316	0.708	0.341	0.261	0.340	0.697	0.228	0.222	0.282	0.603	0.074	0.062	0.207	0.661
	LLM Prob.+PS	0.116	0.053	0.242	0.682	0.137	0.056	0.253	0.697	0.068	0.046	0.231	0.603	0.067	0.053	0.209	0.661
	Verblized %	0.314	0.077	0.317	0.684	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.194	0.156	0.281	0.602	0.176	0.136	0.279	0.615	0.145	0.134	0.251	0.632	0.095	0.090	0.218	0.632
	(AUC)Calib-1	0.224	0.203	0.292	0.591	0.172	0.170	0.271	0.579	0.168	0.174	0.260	0.605	0.146	0.175	0.222	0.672
	(AUC)Calib-n	0.293	0.234	0.334	0.584	0.274	0.220	0.316	0.605	0.260	0.231	0.303	0.586	0.154	0.168	0.216	0.691
	(BCE)Calib-1	0.099	0.080	0.249	0.594	0.062	0.015	0.238	0.625	0.045	0.025	0.225	0.627	0.094	0.077	0.208	0.674
	(BCE)Calib-n	0.150	0.130	0.269	0.592	0.112	0.096	0.257	0.607	0.091	0.087	0.242	0.597	0.120	0.060	0.216	0.680
	(BCE)Calib-n+PS	0.031	0.014	0.238	0.595	0.034	0.034	0.240	0.607	0.014	0.013	0.229	0.597	0.088	0.059	0.208	0.680
	(FL)Calib-1	0.120	0.092	0.259	0.584	0.069	0.015	0.239	0.619	0.060	0.020	0.228	0.625	0.101	0.070	0.210	0.676
	(FL)Calib-n	0.158	0.130	0.273	0.591	0.096	0.079	0.249	0.622	0.104	0.095	0.248	0.588	0.136	0.083	0.222	0.678
	(FL)Calib-n+PS	0.011	0.007	0.238	0.591	0.048	0.056	0.239	0.622	0.013	0.016	0.230	0.588	0.080	0.059	0.209	0.678
Mixtral-8x7b	LLM Prob.	0.342	0.273	0.342	0.683	0.280	0.212	0.303	0.654	0.296	0.246	0.315	0.585	0.194	0.085	0.276	0.535
	LLM Prob.+PS	0.088	0.015	0.244	0.607	0.142	0.075	0.252	0.655	0.075	0.015	0.234	0.585	0.131	0.004	0.257	0.535
	Verblized %	0.318	0.052	0.324	0.624	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.215	0.166	0.300	0.584	0.153	0.153	0.261	0.603	0.149	0.132	0.262	0.606	0.242	0.141	0.319	0.531
	(AUC)Calib-1	0.215	0.210	0.290	0.581	0.157	0.144	0.253	0.647	0.142	0.146	0.255	0.583	0.105	0.117	0.257	0.560
	(AUC)Calib-n	0.305	0.263	0.341	0.565	0.291	0.292	0.327	0.552	0.192	0.196	0.274	0.573	0.184	0.192	0.271	0.597
	(BCE)Calib-1	0.093	0.079	0.253	0.582	0.061	0.057	0.226	0.658	0.057	0.051	0.232	0.588	0.172	0.052	0.271	0.567
	(BCE)Calib-n	0.154	0.142	0.270	0.580	0.121	0.134	0.260	0.561	0.085	0.055	0.239	0.576	0.143	0.064	0.256	0.594
	(BCE)Calib-n+PS	0.018	0.011	0.239	0.587	0.037	0.042	0.238	0.561	0.026	0.021	0.228	0.576	0.015	0.020	0.236	0.594
	(FL)Calib-1	0.071	0.044	0.243	0.580	0.068	0.062	0.227	0.660	0.054	0.024	0.230	0.593	0.165	0.054	0.271	0.555
	(FL)Calib-n	0.169	0.160	0.279	0.568	0.106	0.110	0.254	0.572	0.099	0.043	0.239	0.603	0.151	0.070	0.260	0.589
	(FL)Calib-n+PS	0.025	0.014	0.240	0.568	0.007	0.010	0.236	0.572	0.045	0.020	0.228	0.603	0.023	0.018	0.237	0.589

Table 4: Test results of large-size models on TriviaQA dataset.

	Method	Verb.				Zero-shot				CoT				Few-shot			
		ECE↓	ECE-t↓	Brier ↓	AUC↑	ECE ↓	ECE-t↓	Brier ↓	AUC↑	ECE ↓	ECE-t↓	Brier ↓	AUC↑	ECE ↓	ECE-t↓	Brier ↓	AUC↑
Llama2-7b	LLM Prob.	0.323	0.196	0.329	0.712	0.238	0.152	0.270	0.735	0.240	0.147	0.286	0.661	0.398	0.045	0.394	0.604
	LLM Prob.+PS	0.313	0.012	0.341	0.599	0.234	0.093	0.268	0.736	0.164	0.087	0.253	0.722	0.203	0.022	0.282	0.606
	Verblized %	0.369	0.026	0.375	0.616	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.130	0.118	0.259	0.630	0.118	0.108	0.231	0.666	0.120	0.125	0.244	0.675	0.121	0.088	0.237	0.682
	(AUC)Calib-1	0.216	0.216	0.284	0.619	0.255	0.255	0.297	0.652	0.222	0.162	0.275	0.654	0.207	0.207	0.259	0.728
	(AUC)Calib-n	0.304	0.308	0.340	0.654	0.297	0.294	0.324	0.674	0.228	0.224	0.283	0.679	0.215	0.205	0.256	0.735
	(BCE)Calib-1	0.065	0.063	0.242	0.624	0.081	0.080	0.222	0.673	0.088	0.085	0.233	0.659	0.073	0.071	0.216	0.718
	(BCE)Calib-n	0.170	0.132	0.259	0.661	0.189	0.139	0.260	0.689	0.136	0.104	0.238	0.694	0.106	0.104	0.214	0.735
	(BCE)Calib-n+PS	0.126	0.032	0.256	0.615	0.194	0.066	0.257	0.689	0.140	0.089	0.253	0.694	0.153	0.092	0.242	0.735
	(FL)Calib-1	0.040	0.012	0.236	0.636	0.069	0.067	0.222	0.661	0.057	0.080	0.229	0.660	0.086	0.048	0.218	0.710
	(FL)Calib-n	0.198	0.173	0.278	0.617	0.184	0.131	0.257	0.689	0.126	0.107	0.235	0.689	0.094	0.092	0.211	0.734
	(FL)Calib-n+PS	0.136	0.039	0.259	0.617	0.203	0.065	0.261	0.689	0.123	0.084	0.251	0.689	0.165	0.097	0.240	0.734
Llama3-8b	LLM Prob.	0.162	0.108	0.229	0.752	0.232	0.172	0.265	0.737	0.201	0.182	0.277	0.603	0.165	0.045	0.255	0.612
	LLM Prob.+PS	0.177	0.057	0.277	0.621	0.136	0.104	0.246	0.737	0.074	0.028	0.236	0.603	0.095	0.042	0.243	0.612
	Verblized %	0.321	0.070	0.340	0.626	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.185	0.187	0.286	0.587	0.146	0.146	0.259	0.636	0.163	0.147	0.254	0.639	0.115	0.111	0.225	0.691
	(AUC)Calib-1	0.233	0.229	0.288	0.646	0.224	0.223	0.290	0.615	0.201	0.201	0.269	0.634	0.212	0.211	0.245	0.730
	(AUC)Calib-n	0.257	0.256	0.290	0.697	0.280	0.276	0.315	0.654	0.224	0.218	0.270	0.650	0.202	0.208	0.238	0.753
	(BCE)Calib-1	0.118	0.120	0.239	0.673	0.111	0.094	0.252	0.616	0.094	0.097	0.233	0.638	0.106	0.103	0.211	0.733
	(BCE)Calib-n	0.122	0.126	0.233	0.704	0.147	0.147	0.255	0.660	0.112	0.098	0.229	0.669	0.088	0.088	0.201	0.752
	(BCE)Calib-n+PS	0.110	0.022	0.256	0.600	0.091	0.042	0.246	0.660	0.081	0.067	0.227	0.669	0.103	0.115	0.214	0.752
	(FL)Calib-1	0.112	0.105	0.237	0.676	0.094	0.088	0.249	0.610	0.075	0.079	0.230	0.644	0.089	0.084	0.207	0.731
	(FL)Calib-n	0.203	0.213	0.289	0.598	0.136	0.135	0.251	0.661	0.091	0.087	0.226	0.658	0.075	0.071	0.199	0.750
	(FL)Calib-n+PS	0.094	0.029	0.253	0.598	0.087	0.050	0.245	0.661	0.045	0.061	0.227	0.658	0.104	0.117	0.215	0.750
Llama3.1-8b	LLM Prob.	0.145	0.103	0.223	0.762	0.086	0.072	0.235	0.682	0.263	0.157	0.302	0.674	0.070	0.031	0.212	0.689
	LLM Prob.+PS	0.252	0.098	0.308	0.588	0.180	0.147	0.229	0.803	0.186	0.085	0.269	0.674	0.100	0.049	0.222	0.689
	Verblized %	0.415	0.184	0.413	0.587	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.182	0.179	0.273	0.590	0.108	0.103	0.216	0.734	0.092	0.084	0.224	0.721	0.101	0.082	0.202	0.738
	(AUC)Calib-1	0.251	0.211	0.276	0.729	0.240	0.221	0.268	0.753	0.233	0.197	0.253	0.761	0.180	0.178	0.214	0.740
	(AUC)Calib-n	0.223	0.224	0.258	0.755	0.200	0.193	0.243	0.782	0.223	0.182	0.251	0.763	0.192	0.193	0.224	0.758
	(BCE)Calib-1	0.084	0.106	0.199	0.770	0.118	0.133	0.207	0.771	0.089	0.091	0.196	0.781	0.082	0.090	0.192	0.744
	(BCE)Calib-n	0.118	0.116	0.206	0.758	0.117	0.104	0.198	0.788	0.106	0.073	0.192	0.793	0.081	0.082	0.192	0.756
	(BCE)Calib-n+PS	0.173	0.008	0.263	0.587	0.166	0.121	0.243	0.788	0.126	0.158	0.236	0.793	0.115	0.127	0.206	0.756
	(FL)Calib-1	0.068	0.091	0.198	0.757	0.093	0.101	0.202	0.759	0.074	0.077	0.192	0.774	0.061	0.069	0.190	0.741
	(FL)Calib-n	0.240	0.245	0.295	0.584	0.119	0.103	0.196	0.787	0.110	0.074	0.197	0.780	0.075	0.068	0.191	0.755
	(FL)Calib-n+PS	0.166	0.017	0.261	0.584	0.155	0.126	0.240	0.787	0.151	0.150	0.235	0.780	0.110	0.117	0.207	0.755
Gemma2-2b	LLM Prob.	0.060	0.065	0.196	0.774	0.069	0.067	0.192	0.791	0.104	0.097	0.236	0.691	0.141	0.145	0.260	0.633
	LLM Prob.+PS	0.179	0.001	0.279	0.586	0.156	0.151	0.229	0.791	0.083	0.080	0.237	0.691	0.060	0.033	0.238	0.633
	Verblized %	0.424	0.108	0.420	0.629	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.158	0.172	0.277	0.602	0.164	0.159	0.266	0.632	0.150	0.153	0.261	0.641	0.110	0.105	0.228	0.704
	(AUC)Calib-1	0.183	0.184	0.270	0.631	0.221	0.218	0.292	0.604	0.161	0.161	0.259	0.654	0.225	0.225	0.266	0.708
	(AUC)Calib-n	0.277	0.282	0.319	0.662	0.300	0.294	0.331	0.630	0.236	0.229	0.293	0.674	0.258	0.258	0.282	0.716
	(BCE)Calib-1	0.061	0.070	0.240	0.635	0.109	0.110	0.255	0.604	0.076	0.073	0.235	0.659	0.129	0.134	0.225	0.716
	(BCE)Calib-n	0.142	0.124	0.247	0.680	0.167	0.167	0.265	0.640	0.119	0.100	0.240	0.687	0.122	0.109	0.224	0.723
	(BCE)Calib-n+PS	0.143	0.041	0.261	0.630	0.131	0.021	0.257	0.640	0.098	0.073	0.245	0.687	0.111	0.099	0.233	0.723
	(FL)Calib-1	0.024	0.036	0.235	0.637	0.113	0.109	0.256	0.613	0.066	0.066	0.236	0.653	0.092	0.088	0.215	0.716
	(FL)Calib-n	0.177	0.175	0.271	0.627	0.173	0.169	0.265	0.640	0.103	0.074	0.235	0.682	0.111	0.096	0.221	0.720
	(FL)Calib-n+PS	0.137	0.034	0.260	0.627	0.099	0.013	0.251	0.640	0.094	0.081	0.245	0.682	0.102	0.107	0.232	0.720
Gemma2-9b	LLM Prob.	0.164	0.165	0.242	0.655	0.163	0.146	0.227	0.722	0.138	0.120	0.215	0.669	0.097	0.093	0.195	0.614
	LLM Prob.+PS	0.056	0.023	0.220	0.675	0.098	0.075	0.214	0.722	0.064	0.059	0.202	0.669	0.053	0.013	0.187	0.614
	Verblized %	0.269	0.101	0.276	0.698	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.153	0.149	0.250	0.590	0.141	0.135	0.248	0.607	0.145	0.144	0.231	0.642	0.096	0.087	0.179	0.693
	(AUC)Calib-1	0.119	0.132	0.241	0.596	0.119	0.123	0.244	0.620	0.200	0.175	0.245	0.648	0.186	0.167	0.214	0.690
	(AUC)Calib-n	0.262	0.229	0.296	0.620	0.282	0.248	0.312	0.618	0.176	0.172	0.226	0.672	0.170	0.175	0.191	0.726
	(BCE)Calib-1	0.078	0.044	0.223	0.590	0.083	0.063	0.233	0.598	0.057	0.052	0.200	0.659	0.091	0.081	0.173	0.700
	(BCE)Calib-n	0.116	0.115	0.244	0.637	0.142	0.137	0.247	0.631	0.060	0.062	0.193	0.688	0.059	0.061	0.165	0.732
	(BCE)Calib-n+PS	0.041	0.036	0.218	0.615	0.016	0.020	0.222	0.631	0.082	0.077	0.200	0.688	0.108	0.086	0.177	0.732
	(FL)Calib-1	0.037	0.032	0.218	0.597	0.060	0.044	0.224	0.617	0.039	0.033	0.200	0.651	0.068	0.058	0.170	0.698
	(FL)Calib-n	0.142	0.131	0.258	0.611	0.126	0.137	0.245	0.634	0.055	0.050	0.195	0.678	0.056	0.052	0.165	0.728
	(FL)Calib-n+PS	0.030	0.023	0.217	0.611	0.029	0.030	0.221	0.634	0.091	0.077	0.203	0.678	0.113	0.080	0.178	0.728
Phi3-4b	LLM Prob.	0.187	0.121	0.226	0.796	0.186	0.121	0.225	0.785	0.142	0.119	0.220	0.720	0.039	0.020	0.225	0.624
	LLM Prob.+PS	0.154	0.092	0.201	0.835	0.150	0.138	0.226	0.785	0.104	0.054	0.216	0.720	0.022	0.051	0.230	0.624
	Verblized %	0.322	0.034	0.324	0.703	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.064	0.059	0.204	0.755	0.143	0.132	0.247	0.672	0.127	0.133	0.238	0.643	0.125	0.120	0.231	0.685
	(AUC)Calib-1	0.205	0.147	0.228	0.810	0.161	0.151	0.240	0.709	0.140	0.121	0.239	0.711	0.237	0.174	0.262	0.732
	(AUC)Calib-n	0.178	0.182	0.216	0.814	0.238	0.182	0.278	0.686	0.182	0.181	0.244	0.690	0.203	0.184	0.240	0.740
	(BCE)Calib-1	0.072	0.089	0.177	0.828	0.086	0.083	0.217	0.711	0.068	0.065	0.210	0.697	0.132	0.116	0.219	0.723
	(BCE)Calib-n	0.079	0.077	0.174	0.826	0.110	0.103	0.232	0.691	0.085	0.065	0.212	0.705	0.092	0.075	0.205	0.746
	(BCE)Calib-n+PS	0.169	0.163	0.228	0.823	0.072	0.082	0.224	0.691	0.075	0.079	0.213	0.705	0.109	0.112	0.212	0.746
	(FL)Calib-1	0.040	0.045	0.168	0.827	0.062	0.063	0.215	0.703	0.041	0.042	0.202	0.707	0.100	0.097	0.212	0.720
	(FL)Calib-n	0.078	0.086	0.178	0.821	0.096	0.082	0.229	0.695	0.060	0.041	0.208	0.703	0.082	0.067	0.203	0.746
	(FL)Calib-n+PS	0.173	0.163	0.227	0.821	0.082	0.088	0.223	0.695	0.090	0.069	0.215	0.703	0.115	0.113	0.212	0.746
Phi3-7b	LLM Prob.	0.133	0.123	0.219	0.737	0.118	0.128	0.217	0.730	0.122	0.125	0.236	0.625	0.092	0.021	0.205	0.638
	LLM Prob.+PS	0.086	0.052	0.234	0.691	0.108	0.091	0.216	0.729	0.022	0.034	0.219	0.625	0.081	0.050	0.207	0.638
	Verblized %	0.314	0.055	0.324	0.670	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.113	0.109	0.236	0.689	0.129	0.137	0.243	0.654	0.130	0.137	0.227	0.654	0.101	0.099	0.209	0.654
	(AUC)Calib-1	0.152	0.159	0.236	0.712	0.190	0.201	0.260	0.666	0.217	0.195	0.255	0.676	0.229	0.202	0.256	0.676
	(AUC)Calib-n	0.235	0.206	0.275	0.697	0.268	0.227	0.296	0.664	0.174	0.172	0.233	0.697	0.194	0.203	0.222	0.706
	(BCE)Calib-1	0.071	0.071	0.216	0.704	0.088	0.081	0.224	0.674	0.094	0.065	0.210	0.692	0.119	0.128	0.207	0.666
	(BCE)Calib-n	0.099	0.092	0.228	0.708	0.100	0.095	0.232	0.682	0.064	0.067	0.205	0.707	0.090	0.088	0.192	0.711
	(BCE)Calib-n+PS	0.084	0.084	0.226	0.706	0.050	0.058	0.222	0.682	0.072	0.077	0.212	0.707	0.094	0.079	0.193	0.711
	(FL)Calib-1	0.054	0.062	0.215	0.701	0.076	0.085	0.223	0.664	0.063	0.042	0.205	0.691	0.125	0.124	0.208	0.662
	(FL)Calib-n	0.127	0.100	0.237	0.700	0.107	0.097	0.233	0.681	0.067	0.058	0.205	0.700	0.073	0.077	0.190	0.711
	(FL)Calib-n+PS	0.082	0.076	0.227	0.700	0.053	0.057	0.222	0.681	0.100	0.089	0.214	0.700	0.093	0.080	0.193	0.711

Table 5: Test results of small-size models on Sciq dataset.

	Method	Verb. ECE↓	Verb. ECE-t↓	Verb. Brier ↓	Verb. AUC↑	Zero-shot ECE ↓	Zero-shot ECE-t↓	Zero-shot Brier ↓	Zero-shot AUC↑	CoT ECE ↓	CoT ECE-t↓	CoT Brier ↓	CoT AUC↑	Few-shot ECE ↓	Few-shot ECE-t↓	Few-shot Brier ↓	Few-shot AUC↑
Gemma2-27b	LLM Prob.	0.193	0.181	0.236	0.705	0.204	0.179	0.229	0.713	0.161	0.158	0.227	0.630	0.080	0.088	0.172	0.616
	LLM Prob.+PS	0.077	0.038	0.216	0.658	0.068	0.081	0.199	0.713	0.028	0.035	0.199	0.630	0.080	0.026	0.170	0.616
	Verblized Prob.	0.224	0.071	0.247	0.685	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.140	0.139	0.239	0.594	0.165	0.163	0.236	0.595	0.111	0.111	0.215	0.618	0.092	0.097	0.171	0.621
	(AUC)Calib-1	0.127	0.137	0.245	0.606	0.118	0.152	0.254	0.589	0.122	0.122	0.212	0.624	0.147	0.140	0.181	0.683
	(AUC)Calib-n	0.201	0.188	0.275	0.598	0.231	0.207	0.289	0.583	0.217	0.207	0.250	0.637	0.168	0.150	0.193	0.684
	(BCE)Calib-1	0.101	0.099	0.223	0.598	0.098	0.098	0.221	0.571	0.059	0.059	0.200	0.621	0.065	0.045	0.157	0.675
	(BCE)Calib-n	0.095	0.103	0.222	0.585	0.113	0.114	0.224	0.577	0.065	0.060	0.197	0.643	0.052	0.037	0.153	0.673
	(BCE)Calib-n+PS	0.021	0.024	0.213	0.578	0.026	0.023	0.208	0.577	0.054	0.061	0.199	0.643	0.092	0.060	0.162	0.673
	(FL)Calib-1	0.081	0.077	0.219	0.599	0.066	0.075	0.214	0.594	0.058	0.059	0.199	0.625	0.038	0.037	0.155	0.662
	(FL)Calib-n	0.064	0.073	0.216	0.597	0.094	0.098	0.221	0.576	0.061	0.063	0.195	0.649	0.040	0.041	0.155	0.664
	(FL)Calib-n+PS	0.041	0.042	0.211	0.597	0.020	0.017	0.208	0.576	0.058	0.063	0.199	0.649	0.092	0.060	0.163	0.664
Llama3-70b	LLM Prob.	0.172	0.177	0.251	0.626	0.212	0.223	0.261	0.660	0.182	0.168	0.233	0.664	0.062	0.042	0.184	0.571
	LLM Prob.+PS	0.104	0.018	0.219	0.680	0.044	0.020	0.219	0.661	0.033	0.050	0.204	0.664	0.048	0.000	0.182	0.571
	Verblized Prob.	0.203	0.021	0.244	0.685	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.155	0.156	0.250	0.580	0.139	0.142	0.244	0.614	0.108	0.112	0.215	0.630	0.105	0.084	0.181	0.715
	(AUC)Calib-1	0.156	0.168	0.254	0.575	0.144	0.151	0.254	0.599	0.170	0.177	0.223	0.655	0.172	0.092	0.214	0.738
	(AUC)Calib-n	0.203	0.188	0.271	0.610	0.243	0.227	0.295	0.602	0.237	0.228	0.253	0.645	0.209	0.100	0.227	0.766
	(BCE)Calib-1	0.094	0.087	0.232	0.536	0.122	0.120	0.243	0.563	0.073	0.066	0.201	0.654	0.057	0.064	0.163	0.738
	(BCE)Calib-n	0.081	0.091	0.224	0.612	0.123	0.122	0.236	0.601	0.074	0.071	0.199	0.660	0.038	0.031	0.158	0.744
	(BCE)Calib-n+PS	0.036	0.016	0.219	0.593	0.041	0.017	0.222	0.601	0.068	0.077	0.202	0.660	0.113	0.081	0.173	0.744
	(FL)Calib-1	0.052	0.051	0.225	0.557	0.079	0.071	0.226	0.615	0.055	0.053	0.198	0.655	0.049	0.021	0.159	0.739
	(FL)Calib-n	0.067	0.073	0.220	0.613	0.110	0.115	0.233	0.598	0.072	0.067	0.197	0.665	0.062	0.021	0.162	0.741
	(FL)Calib-n+PS	0.047	0.022	0.218	0.613	0.037	0.014	0.223	0.598	0.060	0.083	0.201	0.665	0.132	0.081	0.176	0.741
Llama3.1-70b	LLM Prob.	0.133	0.119	0.220	0.688	0.120	0.101	0.205	0.704	0.125	0.119	0.213	0.671	0.062	0.047	0.154	0.692
	LLM Prob.+PS	0.054	0.009	0.212	0.698	0.078	0.063	0.199	0.705	0.073	0.070	0.202	0.671	0.105	0.058	0.165	0.692
	Verblized Prob.	0.231	0.066	0.246	0.709	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.136	0.129	0.236	0.636	0.146	0.147	0.239	0.574	0.134	0.132	0.215	0.644	0.076	0.070	0.160	0.700
	(AUC)Calib-1	0.113	0.118	0.231	0.650	0.167	0.093	0.253	0.622	0.157	0.112	0.246	0.658	0.138	0.117	0.178	0.732
	(AUC)Calib-n	0.179	0.144	0.250	0.664	0.232	0.189	0.289	0.605	0.209	0.203	0.245	0.681	0.188	0.131	0.206	0.740
	(BCE)Calib-1	0.037	0.039	0.212	0.602	0.082	0.081	0.214	0.617	0.098	0.095	0.199	0.674	0.041	0.040	0.145	0.751
	(BCE)Calib-n	0.054	0.061	0.204	0.666	0.089	0.089	0.221	0.597	0.056	0.057	0.193	0.691	0.038	0.042	0.146	0.745
	(BCE)Calib-n+PS	0.023	0.038	0.213	0.619	0.033	0.036	0.211	0.597	0.079	0.087	0.202	0.691	0.129	0.103	0.161	0.745
	(FL)Calib-1	0.036	0.035	0.206	0.636	0.057	0.064	0.213	0.624	0.079	0.092	0.198	0.666	0.023	0.017	0.144	0.738
	(FL)Calib-n	0.027	0.040	0.200	0.669	0.079	0.082	0.220	0.592	0.065	0.053	0.193	0.690	0.036	0.038	0.147	0.742
	(FL)Calib-n+PS	0.051	0.070	0.210	0.669	0.040	0.029	0.212	0.592	0.078	0.085	0.202	0.690	0.133	0.099	0.163	0.742
Qwen2-72b	LLM Prob.	0.177	0.167	0.231	0.666	0.231	0.185	0.254	0.743	0.191	0.172	0.242	0.623	0.093	0.021	0.166	0.687
	LLM Prob.+PS	0.065	0.117	0.205	0.697	0.056	0.056	0.216	0.743	0.037	0.052	0.207	0.623	0.120	0.045	0.176	0.687
	Verblized Prob.	0.211	0.051	0.232	0.715	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.145	0.146	0.227	0.623	0.149	0.150	0.241	0.607	0.125	0.121	0.222	0.613	0.094	0.102	0.184	0.653
	(AUC)Calib-1	0.189	0.163	0.255	0.616	0.156	0.153	0.244	0.601	0.139	0.163	0.216	0.665	0.210	0.131	0.237	0.721
	(AUC)Calib-n	0.187	0.172	0.267	0.638	0.230	0.220	0.290	0.608	0.223	0.199	0.246	0.688	0.209	0.119	0.234	0.750
	(BCE)Calib-1	0.068	0.079	0.208	0.620	0.101	0.100	0.233	0.596	0.088	0.097	0.200	0.680	0.077	0.060	0.160	0.717
	(BCE)Calib-n	0.043	0.057	0.203	0.638	0.114	0.102	0.232	0.610	0.063	0.072	0.194	0.687	0.052	0.035	0.151	0.750
	(BCE)Calib-n+PS	0.035	0.035	0.207	0.604	0.058	0.011	0.221	0.610	0.079	0.076	0.202	0.687	0.114	0.092	0.165	0.750
	(FL)Calib-1	0.054	0.055	0.205	0.625	0.065	0.061	0.226	0.584	0.074	0.072	0.197	0.679	0.062	0.045	0.162	0.715
	(FL)Calib-n	0.051	0.053	0.205	0.623	0.099	0.093	0.229	0.611	0.053	0.062	0.194	0.683	0.061	0.025	0.152	0.746
	(FL)Calib-n+PS	0.046	0.048	0.206	0.623	0.050	0.012	0.221	0.611	0.070	0.088	0.202	0.683	0.126	0.095	0.167	0.746
Mixtral-8x7b	LLM Prob.	0.261	0.216	0.286	0.675	0.204	0.163	0.256	0.660	0.258	0.214	0.289	0.582	0.097	0.071	0.209	0.528
	LLM Prob.+PS	0.058	0.024	0.228	0.603	0.052	0.024	0.223	0.660	0.066	0.005	0.225	0.582	0.011	0.001	0.199	0.528
	Verblized Prob.	0.273	0.059	0.288	0.611	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.157	0.152	0.254	0.609	0.120	0.117	0.230	0.640	0.098	0.101	0.218	0.661	0.101	0.095	0.211	0.605
	(AUC)Calib-1	0.206	0.231	0.263	0.631	0.183	0.188	0.237	0.687	0.175	0.156	0.240	0.677	0.343	0.097	0.339	0.636
	(AUC)Calib-n	0.199	0.182	0.277	0.634	0.177	0.157	0.266	0.654	0.276	0.098	0.297	0.636	0.273	0.136	0.295	0.698
	(BCE)Calib-1	0.071	0.078	0.224	0.633	0.079	0.087	0.211	0.678	0.079	0.085	0.213	0.671	0.087	0.083	0.196	0.654
	(BCE)Calib-n	0.053	0.059	0.220	0.647	0.060	0.041	0.219	0.632	0.014	0.016	0.207	0.660	0.053	0.053	0.184	0.701
	(BCE)Calib-n+PS	0.050	0.008	0.226	0.609	0.036	0.014	0.221	0.632	0.061	0.053	0.215	0.660	0.094	0.070	0.192	0.701
	(FL)Calib-1	0.072	0.091	0.224	0.642	0.043	0.043	0.205	0.684	0.051	0.067	0.210	0.669	0.075	0.056	0.194	0.676
	(FL)Calib-n	0.042	0.042	0.221	0.631	0.038	0.029	0.217	0.631	0.032	0.020	0.208	0.657	0.049	0.044	0.187	0.696
	(FL)Calib-n+PS	0.051	0.004	0.224	0.631	0.033	0.012	0.221	0.631	0.052	0.060	0.215	0.657	0.095	0.066	0.194	0.696

Table 6: Test results of large-size models on Sciq dataset.

	Method	Verb.				Zero-shot				CoT				Few-shot			
		ECE↓	ECE-t↓	Brier ↓	AUC↑	ECE ↓	ECE-t↓	Brier ↓	AUC↑	ECE ↓	ECE-t↓	Brier ↓	AUC↑	ECE ↓	ECE-t↓	Brier ↓	AUC↑
Llama2-7b	LLM Prob.	0.426	0.207	0.392	0.692	0.369	0.140	0.301	0.751	0.248	0.153	0.291	0.620	0.409	0.117	0.402	0.578
	LLM Prob.+PS	0.269	0.039	0.274	0.720	0.344	0.069	0.264	0.752	0.155	0.090	0.248	0.716	0.265	0.044	0.306	0.581
	Verblized Prob.	0.432	0.047	0.398	0.666	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.180	0.117	0.256	0.590	0.139	0.070	0.169	0.693	0.153	0.117	0.262	0.634	0.145	0.075	0.231	0.688
	(AUC)Calib-1	0.216	0.227	0.283	0.534	0.178	0.145	0.212	0.641	0.233	0.190	0.274	0.651	0.273	0.121	0.297	0.663
	(AUC)Calib-n	0.233	0.243	0.298	0.615	0.245	0.204	0.251	0.653	0.186	0.171	0.266	0.651	0.226	0.233	0.267	0.730
	(BCE)Calib-1	0.158	0.100	0.252	0.565	0.086	0.029	0.157	0.677	0.147	0.061	0.249	0.648	0.146	0.057	0.242	0.646
	(BCE)Calib-n	0.112	0.089	0.229	0.616	0.034	0.022	0.146	0.697	0.128	0.058	0.233	0.684	0.068	0.050	0.216	0.693
	(BCE)Calib-n+PS	0.133	0.015	0.239	0.588	0.246	0.068	0.214	0.697	0.049	0.120	0.242	0.684	0.126	0.107	0.242	0.693
	(FL)Calib-1	0.113	0.072	0.237	0.574	0.058	0.025	0.152	0.674	0.133	0.081	0.236	0.649	0.073	0.024	0.230	0.648
	(FL)Calib-n	0.109	0.089	0.232	0.589	0.040	0.038	0.148	0.686	0.141	0.090	0.229	0.696	0.086	0.054	0.213	0.704
	(FL)Calib-n+PS	0.143	0.022	0.242	0.589	0.256	0.049	0.219	0.686	0.046	0.151	0.241	0.696	0.121	0.125	0.241	0.704
Llama3-8b	LLM Prob.	0.344	0.125	0.290	0.795	0.422	0.196	0.359	0.756	0.357	0.182	0.355	0.667	0.271	0.121	0.326	0.529
	LLM Prob.+PS	0.305	0.134	0.288	0.664	0.316	0.083	0.271	0.756	0.191	0.053	0.271	0.667	0.184	0.009	0.283	0.529
	Verblized Prob.	0.494	0.089	0.436	0.694	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.150	0.089	0.210	0.639	0.106	0.112	0.185	0.649	0.156	0.151	0.258	0.598	0.148	0.060	0.229	0.739
	(AUC)Calib-1	0.133	0.136	0.210	0.693	0.142	0.123	0.210	0.634	0.138	0.123	0.257	0.602	0.238	0.060	0.261	0.756
	(AUC)Calib-n	0.104	0.106	0.194	0.721	0.161	0.178	0.235	0.654	0.252	0.253	0.305	0.570	0.177	0.166	0.215	0.809
	(BCE)Calib-1	0.126	0.053	0.190	0.692	0.097	0.050	0.179	0.675	0.028	0.022	0.240	0.586	0.149	0.085	0.228	0.748
	(BCE)Calib-n	0.101	0.035	0.182	0.710	0.082	0.059	0.177	0.641	0.044	0.064	0.231	0.641	0.108	0.076	0.214	0.762
	(BCE)Calib-n+PS	0.195	0.065	0.221	0.627	0.226	0.018	0.225	0.641	0.124	0.060	0.253	0.641	0.144	0.148	0.237	0.762
	(FL)Calib-1	0.074	0.063	0.184	0.691	0.068	0.054	0.176	0.622	0.019	0.011	0.240	0.578	0.132	0.090	0.241	0.667
	(FL)Calib-n	0.085	0.050	0.185	0.661	0.038	0.055	0.171	0.665	0.032	0.038	0.227	0.656	0.110	0.067	0.213	0.766
	(FL)Calib-n+PS	0.214	0.072	0.225	0.661	0.231	0.031	0.233	0.665	0.134	0.075	0.254	0.656	0.144	0.165	0.237	0.766
Llama3.1-8b	LLM Prob.	0.306	0.119	0.244	0.788	0.176	0.148	0.246	0.704	0.276	0.159	0.319	0.617	0.183	0.096	0.273	0.594
	LLM Prob.+PS	0.344	0.029	0.276	0.638	0.234	0.079	0.253	0.733	0.189	0.019	0.271	0.617	0.166	0.010	0.269	0.594
	Verblized Prob.	0.629	0.143	0.560	0.676	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.098	0.053	0.148	0.756	0.148	0.061	0.218	0.691	0.123	0.089	0.234	0.683	0.151	0.088	0.259	0.652
	(AUC)Calib-1	0.209	0.129	0.206	0.727	0.212	0.159	0.254	0.679	0.298	0.127	0.315	0.672	0.027	0.062	0.244	0.588
	(AUC)Calib-n	0.182	0.142	0.191	0.760	0.170	0.186	0.255	0.685	0.294	0.287	0.329	0.546	0.262	0.255	0.282	0.700
	(BCE)Calib-1	0.047	0.037	0.135	0.760	0.106	0.063	0.210	0.681	0.076	0.031	0.217	0.710	0.135	0.054	0.265	0.584
	(BCE)Calib-n	0.054	0.050	0.133	0.769	0.115	0.041	0.215	0.660	0.075	0.034	0.222	0.685	0.104	0.047	0.244	0.654
	(BCE)Calib-n+PS	0.255	0.052	0.212	0.676	0.167	0.039	0.233	0.660	0.102	0.090	0.244	0.685	0.052	0.091	0.242	0.654
	(FL)Calib-1	0.030	0.040	0.134	0.766	0.062	0.034	0.203	0.681	0.064	0.026	0.218	0.702	0.096	0.032	0.252	0.593
	(FL)Calib-n	0.046	0.049	0.143	0.674	0.086	0.050	0.211	0.645	0.119	0.061	0.223	0.672	0.112	0.052	0.241	0.664
	(FL)Calib-n+PS	0.264	0.051	0.218	0.674	0.163	0.024	0.236	0.645	0.134	0.078	0.252	0.672	0.087	0.088	0.243	0.664
Gemma2-2b	LLM Prob.	0.204	0.089	0.203	0.769	0.210	0.092	0.196	0.825	0.184	0.141	0.261	0.667	0.107	0.090	0.225	0.665
	LLM Prob.+PS	0.309	0.025	0.272	0.625	0.277	0.125	0.249	0.825	0.208	0.047	0.264	0.667	0.189	0.056	0.252	0.665
	Verblized Prob.	0.623	0.155	0.571	0.622	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.163	0.073	0.193	0.665	0.124	0.107	0.201	0.663	0.191	0.141	0.262	0.552	0.133	0.092	0.240	0.676
	(AUC)Calib-1	0.309	0.122	0.281	0.603	0.333	0.050	0.290	0.698	0.192	0.093	0.273	0.573	0.280	0.158	0.289	0.718
	(AUC)Calib-n	0.248	0.243	0.285	0.598	0.218	0.197	0.267	0.647	0.274	0.249	0.326	0.555	0.234	0.261	0.288	0.711
	(BCE)Calib-1	0.094	0.033	0.184	0.613	0.108	0.059	0.192	0.685	0.096	0.041	0.234	0.560	0.118	0.043	0.236	0.678
	(BCE)Calib-n	0.082	0.071	0.175	0.643	0.072	0.048	0.187	0.680	0.064	0.077	0.230	0.581	0.049	0.055	0.219	0.691
	(BCE)Calib-n+PS	0.217	0.004	0.223	0.639	0.209	0.074	0.228	0.680	0.178	0.008	0.262	0.581	0.152	0.115	0.252	0.691
	(FL)Calib-1	0.051	0.031	0.175	0.623	0.074	0.047	0.188	0.683	0.050	0.048	0.229	0.573	0.104	0.048	0.233	0.679
	(FL)Calib-n	0.063	0.036	0.174	0.656	0.043	0.049	0.184	0.669	0.054	0.060	0.228	0.595	0.062	0.068	0.219	0.688
	(FL)Calib-n+PS	0.227	0.011	0.227	0.656	0.207	0.067	0.230	0.669	0.184	0.003	0.264	0.595	0.129	0.113	0.249	0.688
Gemma2-9b	LLM Prob.	0.316	0.203	0.308	0.753	0.338	0.162	0.303	0.799	0.275	0.182	0.303	0.674	0.177	0.166	0.268	0.618
	LLM Prob.+PS	0.221	0.022	0.271	0.739	0.238	0.113	0.258	0.799	0.121	0.083	0.254	0.676	0.121	0.056	0.249	0.618
	Verblized Prob.	0.531	0.147	0.490	0.732	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.186	0.085	0.243	0.672	0.114	0.054	0.209	0.703	0.190	0.122	0.282	0.593	0.145	0.119	0.249	0.654
	(AUC)Calib-1	0.131	0.120	0.246	0.638	0.152	0.119	0.234	0.701	0.124	0.118	0.262	0.549	0.199	0.145	0.265	0.684
	(AUC)Calib-n	0.106	0.110	0.220	0.691	0.179	0.187	0.247	0.696	0.245	0.249	0.317	0.509	0.236	0.247	0.271	0.703
	(BCE)Calib-1	0.159	0.055	0.231	0.674	0.123	0.032	0.216	0.693	0.116	0.024	0.262	0.563	0.079	0.075	0.224	0.701
	(BCE)Calib-n	0.176	0.042	0.230	0.714	0.145	0.024	0.224	0.690	0.128	0.059	0.264	0.565	0.132	0.087	0.235	0.689
	(BCE)Calib-n+PS	0.150	0.081	0.229	0.691	0.118	0.004	0.230	0.690	0.033	0.018	0.249	0.565	0.072	0.092	0.234	0.689
	(FL)Calib-1	0.122	0.066	0.222	0.677	0.092	0.048	0.214	0.676	0.096	0.025	0.255	0.575	0.094	0.055	0.220	0.702
	(FL)Calib-n	0.161	0.057	0.224	0.712	0.114	0.032	0.216	0.690	0.119	0.042	0.260	0.573	0.137	0.077	0.234	0.693
	(FL)Calib-n+PS	0.170	0.065	0.230	0.712	0.140	0.022	0.233	0.690	0.038	0.028	0.249	0.573	0.090	0.088	0.235	0.693
Phi3-4b	LLM Prob.	0.374	0.187	0.332	0.742	0.245	0.123	0.253	0.760	0.209	0.100	0.249	0.739	0.181	0.084	0.243	0.665
	LLM Prob.+PS	0.310	0.054	0.279	0.768	0.192	0.117	0.257	0.760	0.170	0.096	0.257	0.739	0.223	0.086	0.277	0.665
	Verblized Prob.	0.546	0.088	0.489	0.749	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.135	0.095	0.195	0.631	0.141	0.076	0.238	0.675	0.133	0.135	0.258	0.609	0.103	0.048	0.228	0.661
	(AUC)Calib-1	0.200	0.182	0.255	0.572	0.196	0.059	0.273	0.601	0.127	0.120	0.240	0.639	0.314	0.123	0.313	0.678
	(AUC)Calib-n	0.186	0.202	0.256	0.616	0.313	0.311	0.345	0.529	0.205	0.204	0.284	0.596	0.256	0.267	0.297	0.694
	(BCE)Calib-1	0.064	0.054	0.191	0.614	0.124	0.004	0.251	0.584	0.056	0.043	0.242	0.607	0.083	0.077	0.226	0.652
	(BCE)Calib-n	0.100	0.071	0.189	0.645	0.163	0.063	0.262	0.575	0.061	0.052	0.236	0.638	0.054	0.065	0.220	0.665
	(BCE)Calib-n+PS	0.205	0.058	0.230	0.653	0.085	0.044	0.245	0.575	0.098	0.016	0.251	0.638	0.151	0.054	0.248	0.665
	(FL)Calib-1	0.052	0.029	0.189	0.584	0.090	0.037	0.239	0.625	0.039	0.011	0.242	0.596	0.082	0.055	0.221	0.662
	(FL)Calib-n	0.071	0.070	0.185	0.652	0.125	0.056	0.251	0.590	0.053	0.046	0.233	0.647	0.048	0.081	0.216	0.676
	(FL)Calib-n+PS	0.208	0.052	0.231	0.652	0.101	0.055	0.247	0.590	0.122	0.013	0.253	0.647	0.154	0.058	0.248	0.676
Phi3-7b	LLM Prob.	0.165	0.138	0.230	0.751	0.194	0.139	0.235	0.744	0.206	0.153	0.287	0.640	0.077	0.082	0.247	0.603
	LLM Prob.+PS	0.193	0.139	0.257	0.720	0.182	0.113	0.251	0.744	0.136	0.043	0.258	0.640	0.124	0.031	0.260	0.603
	Verblized Prob.	0.444	0.097	0.415	0.702	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.143	0.091	0.227	0.717	0.131	0.077	0.242	0.642	0.094	0.132	0.247	0.641	0.132	0.122	0.254	0.628
	(AUC)Calib-1	0.164	0.168	0.263	0.587	0.205	0.197	0.285	0.521	0.174	0.175	0.277	0.564	0.107	0.075	0.260	0.585
	(AUC)Calib-n	0.178	0.180	0.257	0.680	0.252	0.259	0.310	0.583	0.209	0.204	0.293	0.574	0.288	0.271	0.316	0.695
	(BCE)Calib-1	0.125	0.041	0.237	0.649	0.124	0.092	0.260	0.478	0.114	0.068	0.254	0.570	0.086	0.050	0.248	0.613
	(BCE)Calib-n	0.174	0.056	0.242	0.682	0.166	0.076	0.265	0.576	0.113	0.070	0.252	0.605	0.078	0.046	0.239	0.666
	(BCE)Calib-n+PS	0.156	0.103	0.234	0.692	0.086	0.003	0.244	0.576	0.098	0.010	0.250	0.605	0.070	0.084	0.246	0.666
	(FL)Calib-1	0.096	0.029	0.232	0.646	0.081	0.091	0.253	0.490	0.079	0.082	0.248	0.606	0.072	0.045	0.246	0.615
	(FL)Calib-n	0.162	0.050	0.238	0.687	0.132	0.100	0.258	0.565	0.107	0.044	0.248	0.607	0.067	0.035	0.236	0.676
	(FL)Calib-n+PS	0.150	0.105	0.236	0.687	0.096	0.011	0.247	0.565	0.092	0.016	0.250	0.607	0.071	0.090	0.246	0.676

Table 7: Test results of small-size models on WikiQA dataset.

	Method	Verb.				Zero-shot				CoT				Few-shot			
		ECE↓	ECE-t↓	Brier ↓	AUC↑	ECE ↓	ECE-t↓	Brier ↓	AUC↑	ECE ↓	ECE-t↓	Brier ↓	AUC↑	ECE ↓	ECE-t↓	Brier ↓	AUC↑
Gemma2-27b	LLM Prob.	0.423	0.258	0.382	0.775	0.448	0.259	0.410	0.730	0.359	0.222	0.356	0.654	0.158	0.164	0.226	0.501
	LLM Prob.+PS	0.285	0.101	0.284	0.712	0.234	0.078	0.261	0.730	0.160	0.095	0.261	0.656	0.057	0.065	0.204	0.501
	Verblized Prob.	0.578	0.132	0.519	0.752	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.114	0.070	0.186	0.749	0.127	0.071	0.201	0.734	0.152	0.095	0.244	0.679	0.109	0.103	0.216	0.619
	(AUC)Calib-1	0.171	0.157	0.240	0.650	0.104	0.112	0.217	0.709	0.098	0.108	0.236	0.692	0.075	0.063	0.204	0.621
	(AUC)Calib-n	0.170	0.157	0.230	0.705	0.085	0.086	0.192	0.752	0.200	0.169	0.261	0.640	0.093	0.124	0.205	0.615
	(BCE)Calib-1	0.099	0.039	0.191	0.725	0.129	0.038	0.204	0.735	0.132	0.059	0.230	0.719	0.052	0.052	0.186	0.660
	(BCE)Calib-n	0.042	0.052	0.182	0.718	0.102	0.052	0.187	0.778	0.084	0.086	0.230	0.650	0.092	0.026	0.198	0.618
	(BCE)Calib-n+PS	0.174	0.131	0.228	0.721	0.193	0.120	0.226	0.778	0.109	0.090	0.250	0.650	0.066	0.050	0.201	0.618
	(FL)Calib-1	0.070	0.041	0.192	0.689	0.103	0.074	0.197	0.745	0.105	0.067	0.227	0.715	0.040	0.028	0.185	0.669
	(FL)Calib-n	0.064	0.049	0.183	0.715	0.095	0.069	0.186	0.780	0.114	0.098	0.224	0.674	0.102	0.043	0.203	0.612
	(FL)Calib-n+PS	0.184	0.104	0.231	0.715	0.189	0.133	0.226	0.780	0.094	0.112	0.246	0.674	0.122	0.052	0.206	0.612
Llama3-70b	LLM Prob.	0.359	0.245	0.353	0.727	0.424	0.234	0.385	0.752	0.319	0.239	0.351	0.582	0.125	0.111	0.230	0.512
	LLM Prob.+PS	0.207	0.136	0.278	0.673	0.219	0.071	0.263	0.752	0.091	0.008	0.254	0.582	0.024	0.043	0.214	0.512
	Verblized Prob.	0.439	0.098	0.415	0.686	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.181	0.077	0.234	0.700	0.151	0.081	0.207	0.744	0.174	0.099	0.274	0.614	0.125	0.114	0.239	0.620
	(AUC)Calib-1	0.127	0.133	0.245	0.650	0.111	0.114	0.228	0.690	0.121	0.119	0.251	0.628	0.087	0.057	0.199	0.699
	(AUC)Calib-n	0.156	0.158	0.242	0.679	0.122	0.124	0.216	0.719	0.181	0.115	0.264	0.638	0.123	0.120	0.212	0.669
	(BCE)Calib-1	0.165	0.069	0.244	0.669	0.131	0.041	0.213	0.729	0.147	0.095	0.261	0.626	0.042	0.030	0.201	0.666
	(BCE)Calib-n	0.123	0.095	0.221	0.717	0.108	0.024	0.201	0.746	0.125	0.055	0.244	0.668	0.072	0.036	0.200	0.670
	(BCE)Calib-n+PS	0.140	0.125	0.237	0.672	0.179	0.098	0.229	0.746	0.029	0.048	0.241	0.668	0.039	0.075	0.211	0.670
	(FL)Calib-1	0.147	0.069	0.241	0.667	0.095	0.054	0.206	0.729	0.132	0.072	0.255	0.632	0.045	0.019	0.202	0.662
	(FL)Calib-n	0.109	0.049	0.219	0.715	0.097	0.034	0.202	0.737	0.111	0.049	0.240	0.675	0.081	0.016	0.208	0.631
	(FL)Calib-n+PS	0.155	0.135	0.236	0.715	0.176	0.076	0.230	0.737	0.019	0.068	0.241	0.675	0.032	0.077	0.211	0.631
Llama3.1-70b	LLM Prob.	0.174	0.179	0.250	0.729	0.229	0.182	0.273	0.706	0.156	0.177	0.260	0.634	0.085	0.074	0.194	0.570
	LLM Prob.+PS	0.168	0.023	0.269	0.687	0.152	0.073	0.250	0.706	0.037	0.063	0.237	0.634	0.078	0.004	0.191	0.570
	Verblized Prob.	0.453	0.130	0.432	0.708	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.135	0.103	0.228	0.718	0.195	0.131	0.277	0.619	0.162	0.108	0.264	0.644	0.142	0.120	0.204	0.589
	(AUC)Calib-1	0.093	0.092	0.227	0.695	0.098	0.098	0.242	0.622	0.209	0.203	0.270	0.573	0.148	0.152	0.223	0.566
	(AUC)Calib-n	0.150	0.151	0.234	0.704	0.154	0.151	0.247	0.657	0.278	0.228	0.316	0.532	0.111	0.126	0.197	0.598
	(BCE)Calib-1	0.140	0.054	0.236	0.694	0.159	0.041	0.253	0.644	0.150	0.077	0.266	0.582	0.064	0.062	0.193	0.565
	(BCE)Calib-n	0.130	0.033	0.228	0.716	0.155	0.049	0.241	0.690	0.144	0.092	0.270	0.584	0.085	0.033	0.194	0.576
	(BCE)Calib-n+PS	0.122	0.072	0.239	0.685	0.132	0.089	0.237	0.690	0.027	0.059	0.244	0.584	0.087	0.000	0.194	0.576
	(FL)Calib-1	0.117	0.070	0.234	0.689	0.123	0.009	0.249	0.612	0.142	0.037	0.261	0.587	0.073	0.068	0.193	0.574
	(FL)Calib-n	0.120	0.050	0.228	0.712	0.141	0.057	0.243	0.668	0.145	0.120	0.265	0.598	0.090	0.051	0.198	0.575
	(FL)Calib-n+PS	0.123	0.105	0.239	0.712	0.110	0.067	0.240	0.668	0.015	0.057	0.243	0.598	0.091	0.000	0.194	0.575
Qwen2-72b	LLM Prob.	0.403	0.249	0.389	0.693	0.500	0.235	0.459	0.750	0.301	0.246	0.345	0.542	0.127	0.133	0.225	0.551
	LLM Prob.+PS	0.196	0.100	0.272	0.716	0.227	0.032	0.273	0.749	0.079	0.002	0.253	0.542	0.049	0.037	0.209	0.551
	Verblized Prob.	0.480	0.042	0.449	0.745	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.159	0.120	0.251	0.645	0.118	0.096	0.226	0.665	0.169	0.135	0.267	0.628	0.070	0.070	0.215	0.581
	(AUC)Calib-1	0.185	0.169	0.283	0.563	0.139	0.145	0.248	0.615	0.102	0.110	0.250	0.612	0.066	0.101	0.208	0.608
	(AUC)Calib-n	0.270	0.279	0.318	0.557	0.194	0.201	0.264	0.638	0.172	0.152	0.271	0.629	0.126	0.160	0.218	0.635
	(BCE)Calib-1	0.108	0.044	0.247	0.564	0.118	0.029	0.235	0.631	0.103	0.081	0.248	0.627	0.065	0.063	0.207	0.592
	(BCE)Calib-n	0.106	0.092	0.249	0.593	0.095	0.061	0.226	0.648	0.118	0.056	0.243	0.662	0.062	0.025	0.199	0.635
	(BCE)Calib-n+PS	0.098	0.023	0.246	0.586	0.119	0.078	0.237	0.648	0.047	0.046	0.240	0.662	0.042	0.082	0.206	0.635
	(FL)Calib-1	0.068	0.056	0.244	0.553	0.072	0.042	0.226	0.631	0.095	0.043	0.248	0.621	0.069	0.064	0.211	0.581
	(FL)Calib-n	0.105	0.091	0.246	0.593	0.082	0.074	0.228	0.636	0.107	0.088	0.238	0.669	0.069	0.020	0.200	0.628
	(FL)Calib-n+PS	0.119	0.027	0.251	0.593	0.124	0.061	0.239	0.636	0.063	0.067	0.240	0.669	0.040	0.069	0.205	0.628
Mixtral-8x7b	LLM Prob.	0.306	0.264	0.340	0.590	0.297	0.137	0.317	0.664	0.268	0.195	0.295	0.614	0.222	0.093	0.286	0.529
	LLM Prob.+PS	0.099	0.036	0.254	0.574	0.113	0.091	0.255	0.664	0.061	0.018	0.227	0.614	0.078	0.039	0.243	0.529
	Verblized Prob.	0.327	0.151	0.336	0.615	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.123	0.107	0.258	0.617	0.165	0.126	0.274	0.584	0.080	0.085	0.228	0.612	0.106	0.084	0.242	0.617
	(AUC)Calib-1	0.187	0.187	0.284	0.522	0.098	0.113	0.258	0.588	0.049	0.030	0.220	0.609	0.030	0.028	0.220	0.665
	(AUC)Calib-n	0.252	0.252	0.301	0.546	0.254	0.228	0.304	0.561	0.173	0.191	0.274	0.484	0.221	0.165	0.269	0.639
	(BCE)Calib-1	0.119	0.032	0.263	0.520	0.099	0.038	0.257	0.576	0.046	0.037	0.214	0.638	0.054	0.047	0.221	0.666
	(BCE)Calib-n	0.170	0.076	0.278	0.567	0.106	0.101	0.267	0.552	0.121	0.104	0.253	0.533	0.068	0.017	0.229	0.636
	(BCE)Calib-n+PS	0.015	0.022	0.247	0.516	0.038	0.002	0.249	0.552	0.048	0.014	0.229	0.533	0.060	0.052	0.233	0.636
	(FL)Calib-1	0.110	0.014	0.257	0.551	0.088	0.042	0.257	0.561	0.013	0.014	0.217	0.613	0.064	0.055	0.219	0.683
	(FL)Calib-n	0.163	0.078	0.275	0.566	0.096	0.087	0.264	0.552	0.115	0.150	0.248	0.554	0.061	0.055	0.226	0.645
	(FL)Calib-n+PS	0.024	0.025	0.244	0.566	0.045	0.007	0.249	0.552	0.040	0.006	0.228	0.554	0.030	0.077	0.232	0.645

Table 8: Test results of large-size models on WikiQA dataset.

	Method	Verb.				Zero-shot				CoT				Few-shot			
		ECE↓	ECE-t↓	Brier ↓	AUC↑	ECE ↓	ECE-t↓	Brier ↓	AUC↑	ECE ↓	ECE-t↓	Brier ↓	AUC↑	ECE ↓	ECE-t↓	Brier ↓	AUC↑
Llama2-7b	LLM Prob.	0.554	0.248	0.480	0.700	0.436	0.182	0.356	0.719	0.488	0.182	0.445	0.647	0.543	0.047	0.477	0.628
	LLM Prob.+PS	0.513	0.020	0.420	0.640	0.397	0.032	0.283	0.721	0.328	0.050	0.297	0.680	0.333	0.002	0.297	0.628
	Verblized Prob.	0.549	0.056	0.467	0.638	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.066	0.066	0.155	0.655	0.050	0.055	0.123	0.712	0.086	0.076	0.200	0.646	0.071	0.044	0.180	0.681
	(AUC)Calib-1	0.212	0.221	0.258	0.643	0.176	0.178	0.208	0.649	0.223	0.211	0.271	0.654	0.193	0.193	0.240	0.686
	(AUC)Calib-n	0.266	0.249	0.284	0.676	0.248	0.228	0.257	0.680	0.290	0.235	0.319	0.651	0.200	0.197	0.231	0.722
	(BCE)Calib-1	0.043	0.050	0.156	0.651	0.047	0.049	0.128	0.670	0.052	0.060	0.191	0.657	0.093	0.060	0.181	0.687
	(BCE)Calib-n	0.052	0.052	0.156	0.675	0.048	0.041	0.131	0.714	0.076	0.072	0.197	0.657	0.060	0.057	0.172	0.727
	(BCE)Calib-n+PS	0.278	0.048	0.232	0.667	0.305	0.013	0.221	0.714	0.233	0.037	0.246	0.657	0.229	0.062	0.227	0.727
	(FL)Calib-1	0.026	0.039	0.153	0.653	0.060	0.038	0.130	0.660	0.050	0.049	0.190	0.655	0.061	0.043	0.176	0.691
	(FL)Calib-n	0.051	0.049	0.156	0.663	0.068	0.037	0.133	0.710	0.048	0.048	0.193	0.644	0.060	0.054	0.170	0.730
	(FL)Calib-n+PS	0.270	0.050	0.228	0.663	0.307	0.022	0.223	0.710	0.227	0.021	0.245	0.644	0.228	0.052	0.228	0.730
Llama3-8b	LLM Prob.	0.398	0.150	0.328	0.755	0.435	0.199	0.366	0.737	0.415	0.211	0.392	0.653	0.408	0.046	0.380	0.639
	LLM Prob.+PS	0.353	0.042	0.280	0.712	0.378	0.059	0.298	0.737	0.290	0.051	0.298	0.653	0.318	0.011	0.322	0.639
	Verblized Prob.	0.517	0.084	0.437	0.686	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.065	0.055	0.156	0.689	0.075	0.072	0.169	0.639	0.073	0.081	0.220	0.641	0.089	0.059	0.213	0.682
	(AUC)Calib-1	0.191	0.199	0.236	0.656	0.145	0.146	0.201	0.619	0.306	0.120	0.304	0.679	0.213	0.220	0.274	0.706
	(AUC)Calib-n	0.222	0.232	0.253	0.676	0.218	0.227	0.264	0.654	0.280	0.273	0.323	0.671	0.233	0.233	0.267	0.734
	(BCE)Calib-1	0.044	0.043	0.158	0.642	0.076	0.073	0.169	0.638	0.014	0.013	0.211	0.647	0.093	0.084	0.206	0.702
	(BCE)Calib-n	0.055	0.056	0.156	0.680	0.056	0.055	0.164	0.668	0.067	0.064	0.209	0.683	0.081	0.074	0.197	0.733
	(BCE)Calib-n+PS	0.260	0.055	0.224	0.683	0.257	0.063	0.228	0.668	0.196	0.076	0.250	0.683	0.184	0.110	0.240	0.733
	(FL)Calib-1	0.049	0.035	0.159	0.628	0.073	0.068	0.173	0.605	0.029	0.022	0.213	0.639	0.064	0.065	0.202	0.702
	(FL)Calib-n	0.039	0.037	0.153	0.682	0.046	0.046	0.163	0.664	0.053	0.049	0.210	0.666	0.061	0.061	0.194	0.738
	(FL)Calib-n+PS	0.262	0.059	0.226	0.682	0.269	0.054	0.234	0.664	0.193	0.075	0.252	0.666	0.183	0.117	0.240	0.738
Llama3.1-8b	LLM Prob.	0.335	0.104	0.255	0.808	0.303	0.146	0.272	0.737	0.428	0.182	0.395	0.630	0.278	0.029	0.277	0.716
	LLM Prob.+PS	0.363	0.038	0.279	0.621	0.340	0.068	0.280	0.747	0.320	0.050	0.306	0.630	0.283	0.059	0.296	0.716
	Verblized Prob.	0.597	0.132	0.518	0.684	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.068	0.067	0.143	0.646	0.074	0.078	0.175	0.674	0.071	0.055	0.194	0.699	0.071	0.026	0.200	0.732
	(AUC)Calib-1	0.168	0.189	0.221	0.681	0.215	0.226	0.266	0.607	0.401	0.218	0.355	0.712	0.175	0.166	0.224	0.760
	(AUC)Calib-n	0.228	0.224	0.247	0.692	0.242	0.248	0.282	0.641	0.276	0.259	0.306	0.675	0.212	0.207	0.242	0.775
	(BCE)Calib-1	0.054	0.056	0.137	0.695	0.060	0.050	0.174	0.640	0.075	0.071	0.190	0.719	0.080	0.056	0.196	0.751
	(BCE)Calib-n	0.043	0.040	0.136	0.704	0.047	0.044	0.168	0.683	0.056	0.051	0.186	0.729	0.064	0.055	0.184	0.777
	(BCE)Calib-n+PS	0.282	0.019	0.218	0.636	0.238	0.043	0.229	0.683	0.221	0.086	0.244	0.729	0.193	0.138	0.238	0.777
	(FL)Calib-1	0.044	0.034	0.137	0.692	0.040	0.042	0.174	0.629	0.058	0.057	0.187	0.715	0.055	0.029	0.190	0.757
	(FL)Calib-n	0.063	0.063	0.143	0.632	0.043	0.042	0.167	0.684	0.038	0.034	0.183	0.728	0.053	0.050	0.184	0.769
	(FL)Calib-n+PS	0.286	0.030	0.221	0.632	0.256	0.044	0.238	0.684	0.210	0.106	0.241	0.728	0.196	0.133	0.236	0.769
Gemma2-2b	LLM Prob.	0.279	0.124	0.229	0.758	0.290	0.130	0.244	0.755	0.261	0.152	0.260	0.701	0.240	0.134	0.256	0.646
	LLM Prob.+PS	0.382	0.013	0.273	0.705	0.367	0.049	0.270	0.756	0.318	0.052	0.266	0.701	0.301	0.026	0.265	0.646
	Verblized Prob.	0.628	0.099	0.551	0.700	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.056	0.039	0.127	0.694	0.059	0.048	0.142	0.693	0.086	0.042	0.171	0.687	0.071	0.021	0.156	0.755
	(AUC)Calib-1	0.215	0.156	0.208	0.716	0.185	0.174	0.220	0.666	0.225	0.199	0.263	0.642	0.167	0.183	0.219	0.763
	(AUC)Calib-n	0.269	0.211	0.269	0.714	0.254	0.224	0.269	0.692	0.289	0.259	0.312	0.663	0.267	0.241	0.267	0.785
	(BCE)Calib-1	0.043	0.032	0.127	0.683	0.065	0.060	0.146	0.650	0.054	0.032	0.171	0.643	0.071	0.030	0.156	0.765
	(BCE)Calib-n	0.054	0.041	0.130	0.688	0.050	0.043	0.144	0.705	0.072	0.068	0.174	0.688	0.096	0.069	0.166	0.782
	(BCE)Calib-n+PS	0.309	0.015	0.222	0.679	0.293	0.041	0.227	0.705	0.259	0.065	0.235	0.688	0.265	0.094	0.234	0.782
	(FL)Calib-1	0.025	0.034	0.126	0.665	0.040	0.036	0.145	0.645	0.026	0.027	0.169	0.639	0.032	0.033	0.154	0.761
	(FL)Calib-n	0.053	0.037	0.130	0.688	0.064	0.033	0.143	0.708	0.044	0.045	0.169	0.682	0.089	0.056	0.163	0.776
	(FL)Calib-n+PS	0.312	0.019	0.223	0.688	0.294	0.052	0.228	0.708	0.272	0.068	0.244	0.682	0.273	0.102	0.240	0.776
Gemma2-9b	LLM Prob.	0.390	0.230	0.368	0.716	0.396	0.201	0.355	0.746	0.374	0.232	0.377	0.641	0.298	0.207	0.341	0.594
	LLM Prob.+PS	0.306	0.095	0.289	0.696	0.309	0.100	0.285	0.746	0.243	0.046	0.284	0.641	0.213	0.029	0.284	0.594
	Verblized Prob.	0.547	0.196	0.496	0.695	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.096	0.099	0.210	0.626	0.084	0.073	0.206	0.646	0.097	0.088	0.234	0.638	0.077	0.076	0.226	0.689
	(AUC)Calib-1	0.210	0.218	0.279	0.594	0.192	0.203	0.265	0.604	0.227	0.223	0.299	0.621	0.240	0.236	0.287	0.727
	(AUC)Calib-n	0.256	0.257	0.287	0.642	0.219	0.230	0.275	0.649	0.270	0.268	0.311	0.662	0.273	0.272	0.296	0.740
	(BCE)Calib-1	0.098	0.087	0.206	0.597	0.048	0.040	0.203	0.616	0.053	0.061	0.223	0.637	0.122	0.120	0.224	0.722
	(BCE)Calib-n	0.108	0.075	0.209	0.629	0.093	0.040	0.201	0.674	0.092	0.064	0.224	0.667	0.067	0.072	0.208	0.742
	(BCE)Calib-n+PS	0.178	0.035	0.230	0.631	0.180	0.064	0.231	0.674	0.155	0.052	0.247	0.667	0.148	0.107	0.246	0.742
	(FL)Calib-1	0.057	0.064	0.203	0.604	0.046	0.042	0.204	0.599	0.024	0.024	0.223	0.634	0.108	0.097	0.221	0.718
	(FL)Calib-n	0.091	0.063	0.206	0.621	0.066	0.051	0.196	0.677	0.083	0.044	0.224	0.655	0.067	0.082	0.210	0.736
	(FL)Calib-n+PS	0.182	0.030	0.232	0.621	0.199	0.071	0.239	0.677	0.157	0.048	0.249	0.655	0.152	0.096	0.248	0.736
Phi3-4b	LLM Prob.	0.467	0.145	0.367	0.788	0.436	0.143	0.337	0.767	0.385	0.143	0.319	0.734	0.306	0.026	0.244	0.714
	LLM Prob.+PS	0.389	0.017	0.279	0.768	0.400	0.061	0.292	0.767	0.368	0.059	0.296	0.734	0.378	0.077	0.303	0.714
	Verblized Prob.	0.593	0.077	0.505	0.751	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.068	0.038	0.131	0.722	0.066	0.040	0.133	0.758	0.061	0.059	0.162	0.718	0.070	0.028	0.147	0.761
	(AUC)Calib-1	0.134	0.148	0.185	0.705	0.129	0.128	0.175	0.732	0.147	0.161	0.204	0.709	0.158	0.153	0.187	0.753
	(AUC)Calib-n	0.230	0.215	0.252	0.712	0.273	0.228	0.278	0.699	0.275	0.209	0.282	0.703	0.148	0.146	0.172	0.803
	(BCE)Calib-1	0.078	0.049	0.131	0.717	0.050	0.031	0.130	0.756	0.055	0.043	0.161	0.705	0.078	0.050	0.150	0.760
	(BCE)Calib-n	0.030	0.032	0.126	0.729	0.062	0.038	0.138	0.731	0.072	0.065	0.171	0.694	0.038	0.038	0.134	0.806
	(BCE)Calib-n+PS	0.297	0.030	0.216	0.715	0.297	0.045	0.224	0.731	0.278	0.060	0.242	0.694	0.283	0.113	0.226	0.806
	(FL)Calib-1	0.027	0.029	0.124	0.725	0.024	0.016	0.128	0.750	0.020	0.012	0.158	0.704	0.037	0.035	0.144	0.760
	(FL)Calib-n	0.040	0.036	0.129	0.712	0.079	0.034	0.139	0.727	0.057	0.050	0.166	0.687	0.041	0.035	0.133	0.812
	(FL)Calib-n+PS	0.297	0.022	0.216	0.712	0.309	0.054	0.232	0.727	0.261	0.079	0.235	0.687	0.252	0.107	0.214	0.812
Phi3-7b	LLM Prob.	0.294	0.166	0.272	0.747	0.303	0.148	0.272	0.758	0.312	0.159	0.310	0.693	0.165	0.036	0.228	0.679
	LLM Prob.+PS	0.358	0.023	0.303	0.713	0.333	0.081	0.280	0.758	0.286	0.065	0.286	0.693	0.258	0.074	0.280	0.679
	Verblized Prob.	0.484	0.198	0.441	0.710	-	-	-	-	-	-	-	-	-	-	-	-
	APRICOT	0.078	0.073	0.176	0.696	0.079	0.066	0.178	0.702	0.084	0.089	0.212	0.679	0.093	0.067	0.208	0.690
	(AUC)Calib-1	0.207	0.209	0.256	0.650	0.224	0.233	0.272	0.656	0.207	0.204	0.274	0.659	0.180	0.171	0.242	0.692
	(AUC)Calib-n	0.256	0.253	0.280	0.657	0.263	0.265	0.297	0.639	0.253	0.248	0.295	0.671	0.256	0.256	0.286	0.717
	(BCE)Calib-1	0.051	0.057	0.177	0.660	0.030	0.018	0.173	0.688	0.063	0.059	0.209	0.663	0.096	0.078	0.207	0.700
	(BCE)Calib-n	0.069	0.058	0.176	0.670	0.052	0.046	0.176	0.682	0.084	0.063	0.205	0.676	0.105	0.103	0.207	0.711
	(BCE)Calib-n+PS	0.223	0.021	0.227	0.666	0.227	0.035	0.231	0.682	0.194	0.081	0.243	0.676	0.200	0.076	0.247	0.711
	(FL)Calib-1	0.044	0.039	0.177	0.647	0.022	0.007	0.173	0.682	0.035	0.034	0.206	0.655	0.074	0.063	0.205	0.697
	(FL)Calib-n	0.059	0.056	0.177	0.658	0.047	0.052	0.176	0.676	0.074	0.052	0.208	0.655	0.090	0.089	0.203	0.714
	(FL)Calib-n+PS	0.224	0.021	0.228	0.658	0.241	0.032	0.238	0.676	0.191	0.081	0.247	0.655	0.192	0.075	0.244	0.714

Table 9: Test results of small-size models on NQ dataset.

D. Influential Training Data Retrieval for Explaining Verbalized Confidence of LLMs

Authors: Yuxi Xia, Loris Schoenegger, Benjamin Roth

Status: Published in the Proceedings of the 48th European Conference on Information Retrieval (ECIR 2026) (Long Papers)

URL: https://arxiv.org/pdf/2601.10645

Reference: Xia et al. (2026a)

Work Division

Yuxi Xia: conceptualization, data curation, formal analysis, investigation, methodology, software, validation, visualization, original draft preparation, review and editing
Loris Schoenegger: conceptualization, formal analysis, software, original draft preparation, review and editing
Benjamin Roth: funding acquisition, methodology, project administration, resources, supervision, review and editing.

Influential Training Data Retrieval for Explaining Verbalized Confidence of LLMs

Yuxi Xia[1,2], Loris Schoenegger[1,2], and Benjamin Roth[1,3]

[1] Faculty of Computer Science, University of Vienna, Vienna, Austria
[2] UniVie Doctoral School Computer Science, Vienna, Austria
[3] Faculty of Philological and Cultural Studies, University of Vienna, Vienna, Austria
{yuxi.xia,loris.schoenegger,benjamin.roth}@univie.ac.at

Abstract. Large language models (LLMs) can increase users' perceived trust by verbalizing confidence in their outputs. However, prior work has shown that LLMs are often overconfident, making their stated confidence unreliable since it does not consistently align with factual accuracy. To better understand the sources of this verbalized confidence, we introduce TracVC (**Trac**ing **V**erbalized **C**onfidence), a method that builds on information retrieval and influence estimation to trace generated confidence expressions back to the training data. We evaluate TracVC on OLMo and Llama models in a question answering setting, proposing a new metric, content groundness, which measures the extent to which an LLM grounds its confidence in content-related training examples (relevant to the question and answer) versus in generic examples of confidence verbalization. Our analysis reveals that OLMo2-13B is frequently influenced by confidence-related data that is unrelated to the query, suggesting that it may mimic superficial linguistic expressions of certainty rather than rely on genuine content grounding. These findings reveal a fundamental limitation in current training regimes: LLMs may learn how to sound confident without learning when confidence is justified.

Keywords: Training Data Retrieval · Influence Functions · Explainability · Uncertainty Estimation · LLMs.

1 Introduction

Verbalized confidence (VC) in large language models (LLMs) is increasingly used to estimate the certainty of their generated outputs to improve transparency and user trust [5,23,45]. However, recent studies have shown that LLMs frequently exhibit overconfidence, which is not aligned with factual accuracy, resulting in the poor reliability of their expressed confidence [36,48,50,55]. This finding leads to a foundational question: what drives confidence in LLMs, and do LLMs understand the intended meaning of expressing confidence?

In this paper, we investigate the origins of verbalized confidence in LLMs by examining the role of training data. Specifically, we ask: **Do LLMs ground their confidence in lexically relevant, content-related training examples, or are they more influenced by superficial, confidence-related cues?**

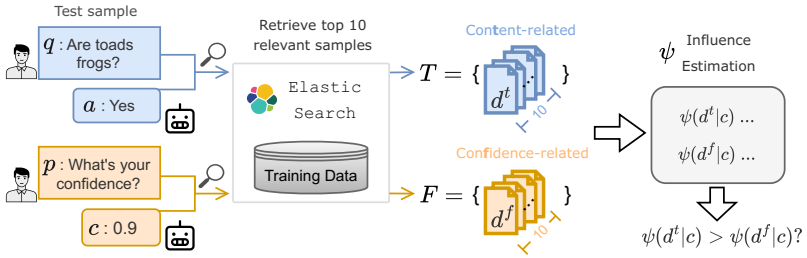


Fig. 1: The workflow of TracVC. We first retrieve the top 10 relevant samples for content (question q and answer a) and confidence (prompt p and confidence c) segment in the test sample. Then, we compute and compare the influence score regarding the confidence generation for content- and confidence-related samples.

To address this question, we propose **TracVC** (shown in Fig. 1), a method for tracing verbalized confidence back to its influential training data. For each test instance, which consists of a content segment (question and LLM answer) and a verbalized confidence segment, TracVC retrieves two sets of top 10 relevant training examples: one that is lexically similar to the content and another that is similar to the confidence expression. Then, TracVC estimates the influence of the retrieved examples on the model's confidence generation by adapting the gradient-based attribution method *TracIn* [39]. After comparing the influence of these two sets, TracVC reveals whether content-related or confidence-related training data plays a greater role in shaping the model's verbalized confidence.

We apply our method to 11 open-source LLMs with publicly available training corpora, including OLMo [38], Llama [14] instruction models, and their corresponding checkpoints trained with different post-training techniques (e.g., direct preference optimization [40]). Those models are evaluated on five question answering benchmarks. To quantify the extent to which content-related training data outweighs confidence-related data in shaping verbalized confidence, we introduce **content groundness** as a new metric, defined as the proportion of cases where content-related examples dominate over confidence-related ones. Our results demonstrate that:

1. LLMs, especially the OLMo2-13B model, can be influenced by confidence-related training examples that are unrelated to the question, often latching onto keywords like "confidence" regardless of context. This suggests that they do not fully ground their confidence in content-related information, but instead learn to mimic linguistic markers of certainty.
2. Larger LLMs do not demonstrate higher content groundness than smaller ones. We hypothesize that larger models, due to their higher capacity, may be more sensitive to stylistic or superficial patterns in training data.
3. Content groundness is higher when tested on samples that are all correctly answered by the evaluated LLM. We hypothesize that LLMs are likely to

ground more on content-related examples if they have seen those examples during training.
4. Post-training techniques can impact content groundness in the opposite direction on different LLMs.

These findings highlight a fundamental limitation in current training regimes: **LLMs may learn to sound confident without understanding when confidence is warranted.** Our work introduces a data-driven perspective of model confidence, offering insights that can guide future training approaches toward improving the trustworthiness of model confidence. [4]

2 Related Work

Trustworthiness of Verbalized Confidence in LLMs. Verbalized confidence offers a convenient way for users to gauge the trustworthiness of LLM outputs [13,45,50]. Empirical evidence suggests that moderate confidence expressions can foster higher user trust, satisfaction, and task performance than either high or low confidence levels [51]. However, recent studies show that LLMs often exhibit overconfidence in their answers [17,48,49,50,55], a mismatch between expressed certainty and factual correctness that can undermine user trust. Explaining verbalized confidence is thus crucial.

Retrieving Influential Training Data for Explanation. Gradient-based influence estimation methods [2,3,7,21,26,31] provide an alternative to leave-one-out retraining for studying the effect of individual training examples on model behavior. Those methods can broadly be grouped into two categories: (1) First-order approaches, such as *TracIn* [39], which measure gradient similarity between training examples and test predictions; and (2) Hessian-based approaches, such as DataInf [29], are potentially more precise, however, while several adaptations make such methods more scalable [4,18,42], they are still computationally prohibitive for large-scale LLMs. To address scalability, information retrieval-based influence estimation methods [12,33], such as OLMoTRACE [33], retrieve lexically similar training examples under the assumption that they are likely to influence predictions. While efficient, such methods rely on surface-level token overlap and do not leverage internal model states as gradient-based influence estimation methods. Other approaches [8,24] train models explicitly to report their influential data, but these require additional training and are unsuitable for our setting. Our proposed method, TracVC, utilizes both information retrieval techniques [12] and the gradient-similarity-based method *TracIn* [39] to analyze how training data influences LLMs' confidence expression, which combines the strengths of both methods and thus is efficient to apply while considering internal model states.

Explaining Verbalized Confidence of LLMs. Existing work has primarily examined verbalized confidence through the lens of its alignment with internal model

[4] Source code is available at https://github.com/Yuuxii/training_data_confidence/.

probabilities [27,36] or with human confidence [44]; others [50,52,54] investigate prompting strategies or calibration techniques aimed at improving the reliability of the expressed confidence. While these approaches provide useful insights into the surface behavior of LLMs, they do not address a deeper question: what role does the training data play in shaping verbalized confidence? Our work fills this gap by tracing verbalized confidence back to influential training samples, thereby offering a data-centric perspective on the sources of LLM confidence.

3 Methodology

The workflow of our proposed TracVC is shown in Fig. 1. TracVC identifies which types of training data, i.e., content-related or confidence-related data, are more influential for verbalized confidence in LLMs. The following sections explain our prompt design, the information retrieval mechanism to retrieve related data for content and confidence from the training corpus, and finally, the method we use to estimate the influence score of the retrieved data.

3.1 Prompt Design

We use a two-stage prompt inspired by Tian et.al. [45] for obtaining verbalized confidence. The first-stage prompt contains the question q for an LLM $\mathcal{M}$ to generate ($\sim$) the answer a ($a \sim \mathcal{M}(q)$). The specific prompt is *"Answer the question, give ONLY the answer, no other words or explanation: $< q >$"*. The second-stage prompt consists of a confidence prompt p to require $\mathcal{M}$ to provide a probability for its generated answer. The specific prompt p is *"Provide the probability that your answer is correct. Give ONLY the probability between 0.0 and 1.0, no other words or explanation"*. Finally, $\mathcal{M}$ generates the verbalized confidence c regarding content a and q, formulated as: $c \sim \mathcal{M}(q, a, p)$.

3.2 Relevant Training Data Retrieval

We want to verify whether *content*-related training data that contains information of question and answer is more influential than *confidence*-related training data that mostly contains superficial confidence cues. For a given test instance, we retrieve two sets of training examples: a set of con**t**ent-related samples $T = \{d_1^t, ..., d_{10}^t\}$, based on lexical similarity to the question q and answer a, and a set of con**f**idence-related samples $F = \{d_1^f, ..., d_{10}^f\}$ based on similarity to the confidence prompt p and verbalized confidence c (samples in Fig. 2).

Specifically, we analyze pre-training data (**Pre**) and post-training data (**Post**) separately, as they differ in format and serve distinct training objectives. For the pre-training corpus, we retrieved examples using the inverted index over model pre-training data provided by Elazar et al. [12]. For the post-training corpora, we constructed our own inverted index with an Elasticsearch cluster, following the setup of Elazar et al. [12]. To identify relevant examples, we issued multi-match queries in *best fields mode* across all indexed fields. Ranking was performed with

Elasticsearch's default scoring function, BM25 [11,41], using default parameters ($k_1 = 1.2$, $b = 0.75$). For each query, we retrieved the top 10 training examples ranked by relevance score.

3.3 Training Data Influence Estimation

Having obtained the content-related and confidence-related sets of examples, we now introduce the method to test which set more strongly influences a language model's confidence generation. Following TracIn [39], our method performs efficient point-wise comparisons of loss gradients with respect to the model parameters ($\nabla\ell(w, \cdot)$), enabling our analysis to scale to large pre-training corpora. Specifically, we estimate the influence of a training sample d on the model's generated confidence c, denoted as $\psi(d|c)$, by computing the cosine similarity between loss gradients. One gradient is obtained by evaluating the loss on the training sample d, while the other is computed by evaluating the loss on the completion (q, a, p, c). The formula to get $\psi(d|c)$ is:

$$\psi(d|c_i) = \frac{\nabla\ell(w,d)\cdot\nabla\ell(w,(q_i,a_i,p,c_i))}{\|\nabla\ell(w,d)\|\cdot\|\nabla\ell(w,(q_i,a_i,p,c_i))\|}. \quad (1)$$

Although TracIn uses dot product as a similarity measure, we use cosine similarity following previous work [19,20,37,47] to reduce the impact of gradient magnitude on our score, and compute the gradient with respect to the model's input embeddings w [53]. Note that the influence score captures information beyond the word embedding layer, as the gradient also chains through higher layers [53].

Influence Estimation without Verbalized Confidence. To validate our score's robustness to variations in the model's self-reported confidence, we also report results for experiments where we omit the verbalized confidence c from the completion (q, a, p). This computed influence score is then defined as:

$$\psi^{\neg c}(d|c_i) = \frac{\nabla\ell(w,d)\cdot\nabla\ell(w,(q_i,a_i,p))}{\|\nabla\ell(w,d)\|\cdot\|\nabla\ell(w,(q_i,a_i,p))\|}. \quad (2)$$

3.4 Experimental Setup

Our experiments require knowledge about the training data of LLMs. Thus, we mainly study OLMo models [15,38] that provide publicly available pre- and post-training data. Those OLMo models span different model sizes and released versions: OLMo2-13B INS (`OLMo-2-1124-13B-Instruct`), OLMo2-7B INS (`OLMo-2-1124-7B-Instruct`), OLMo-7B INS(`OLMo-7B-Instruct-hf`). Additionally, we include Llama-3.1 [14] (`Llama-3.1-Tulu-3-8B` [30]), which was post-trained with publicly available data. For each model, we also evaluate its checkpoints after post-training with supervised-fine-tuning (SFT), direct preference optimization (DPO) [40], and reinforcement learning with verified reward (RLVR) [38]. Our studied benchmark datasets are Natural Question (NQ) [28], SicQ [46], TriviaQA [22], PopQA [35] and TruthfulQA [32]. Each of the first four datasets contains 1,000 randomly selected samples, and TruthfulQA has 817. We

employ the vLLM [9] library for LLM inference and serving. All LLMs are set with a temperature of 0 to ensure consistent outputs. Experiments are conducted on NVIDIA HGX H100, requiring a total of approximately 642 GPU hours.

Fig. 2: Examples of the most influential training samples for different LLMs when generating confidence. Retrieved samples come from pre-training (Source: Pre) and post-training (Source: Post) corpora. Ground-truth answers in bold.

4 Influence Analysis of Retrieved Training Data

We analyze the influence of retrieved data examples on model confidence predictions by presenting (1) a case study of the most influential training documents retrieved from pre-training and post-training data, (2) an overview of the proportion of the source training data for the most influential retrieved data examples, and (3) an overall statistical summary of influence scores.

Case Study of the Most Influential Training Data. Fig. 2 shows the training examples with the highest estimated influence scores for two test samples. We observe that different LLMs rely on different types of examples when generating confidence estimates. For the first example (‘‘`Who was the person who escaped from Alcatraz?"`), all models are most influenced by content-related documents. However, for OLMo2-13B-INS, the top post-training example is irrelevant to the question or task, yielding a low influence score (0.029). By contrast, the most influential examples for the other two OLMo models come from pre-training data and are highly relevant, with influence scores exceeding 0.2. In contrast, for the second test example (‘‘`What genre is Foxy Brown?`’’), confidence-related examples get an influence score above 0.3 for OLMo-7B INS. This case study demonstrates that **LLMs can be affected by content-unrelated training examples that contain superficial cues, such as the keyword “confidence”.**

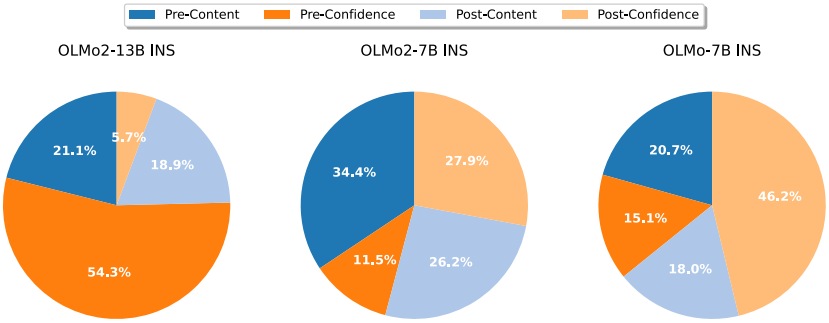
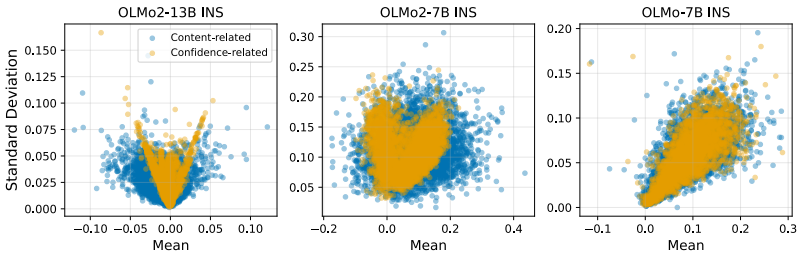
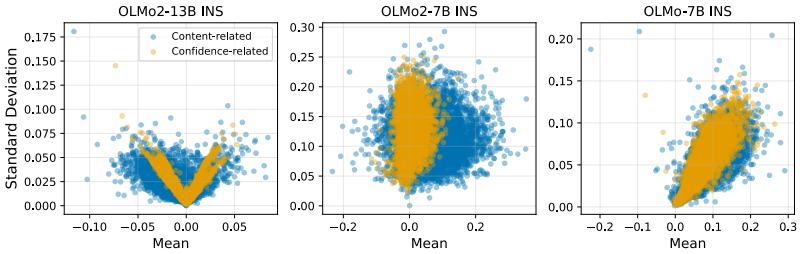


Fig. 3: Proportion of sources of the most influential training examples for each test sample. The most influential example has the highest influence score among all retrieved examples from both pre-training and post-training corpora. E.g., Pre-Content denotes a content-related example from the pre-training corpus.

Source of the Most Influential Examples. Fig. 3 shows that more than half of the most influential data for OLMo2-13B and OLMo-7B comes from the confidence-related example of pre-training data, while for OLMo2-7B, the biggest portion is from content-related data. We also observe that post-training is more influential for OLMo-7B and OLMo2-7B, but an opposite trend applies to OLMo2-13B.



(a) Pre-training data.



(b) Post-training data.

Fig. 4: Distribution of mean and standard deviation of influence scores across the 10 retrieved training examples for each test sample. Each point is a test sample.

Overall Influence Statistics. Fig. 4 shows the distribution of influence scores across models. Overall, we find similar patterns between pre-training and post-training data. For OLMo2-13B, the mean influence scores of the top 10 content-related examples are higher than those of confidence-related examples, which contradicts the earlier study based on the proportion of the most influential example. OLMo-7B shows substantial overlap between content- and confidence-related scores, while OLMo2-7B contains a larger proportion of content-related examples with higher mean scores. Notably, OLMo2-13B-INS differs from the other models by exhibiting a sizable proportion of content-related examples with relatively low mean influence scores. These results suggest that **relying solely on the most influential examples can be misleading.** Examining aggregate statistics across the top retrieved examples provides a more comprehensive view of how LLMs ground their confidence in content- versus confidence-related training data. The following sections aim to achieve this goal.

5 Explaining Verbalized Confidence of LLMs with Content Groundness

Although the previous sections provide detailed case studies and statistical analyses of influence score distributions of the retrieved data, they do not fully examine how to assess the extent to which LLMs ground their verbalized confidence in

content-related documents, nor the factors that impact such grounding. To fill this gap, the following section introduces a new metric for quantifying content groundness and analyzes its contributing factors, thereby offering deeper insights into the mechanisms underlying verbalized confidence.

5.1 Measurement of Content Groundness

We define content groundness as the property whereby content-related training data exerts a greater influence than confidence-related data on confidence generation. To quantify groundness, we introduce **ccr** (Content-over-Confidence Ratio), defined as the winning counts ratio between the con**t**ent-related set T and con**f**idence-related set F. Intuitively, a higher **ccr** indicates that the model's confidence is more strongly driven by lexically relevant content rather than by potentially misleading confidence cues in the prompt. Assume our test data contains n questions, **ccr** is computed by:

$$ccr = \frac{\sum_{i=0}^{n} \sum_{d^t \in T_i, d^f \in F_i} \mathbf{1}(\psi(d^t|c_i) > \psi(d^f|c_i))}{\sum_{i=0}^{n} \sum_{d^t \in T_i, d^f \in F_i} \mathbf{1}(\psi(d^t|c_i) < \psi(d^f|c_i))}. \tag{3}$$

We define our score as a ratio to capture relative rather than absolute importance, and use it for analysis at the aggregate level, rather than to assess individual instances. We take this approach because individual point-wise influence measurements, while inexpensive, can be noisy and have been shown to sometimes poorly predict re-training effects for deep models not trained to convergence [1,2,16]. Specifically, **ccr** aims to provide a balanced view of influence across content-related and confidence-related training data: It reduces the bias introduced by documents with extreme (the highest/lowest) influence scores while still capturing variation among the top 10 most relevant documents. For comparison, we also compute content groundness in the absence of verbalized confidence. This variant, denoted as $ccr^{\neg c}$, is defined analogously using influence estimates obtained without the confidence component (see Equation 2):

$$ccr^{\neg c} = \frac{\sum_{i=0}^{n} \sum_{d^t \in T_i, d^f \in F_i} \mathbf{1}(\psi^{\neg c}(d^t|c_i) > \psi^{\neg c}(d^f|c_i))}{\sum_{i=0}^{n} \sum_{d^t \in T_i, d^f \in F_i} \mathbf{1}(\psi^{\neg c}(d^t|c_i) < \psi^{\neg c}(d^f|c_i))}. \tag{4}$$

Interpretation of Content Groundness Score. When ccr (or $ccr^{\neg c}$) > 1, the examined LLM grounds more on content-related training data, whereas ccr (or $ccr^{\neg c}$) < 1 indicates stronger grounding in confidence-related data. We interpret a higher content groundness score as indicating that the model's confidence is more influenced by lexically relevant content. However, the threshold for determining whether the confidence of an LLM can be considered trustworthy is subjective and often depends on the specific domain or task. For instance, in high-stakes domains such as medicine, one might require a much higher level of reliability and consider a score of 1.2 insufficient.

Table 1: Result of LLMs' content groundness measured with **ccr** and **ccr**$^{\neg c}$ on different datasets. We also aggregate all the retrieved data from pre-training and post-training to compute ccr scores for Pre+Post. Values inside parentheses are insignificant (p>0.05) in their mean influence score difference between sets.

INS	**Data**	**NQ**		**SciQ**		**TriviaQA**		**TruthfulQA**		**PopQA**		**All**	
Model	**Source**	ccr	ccr$^{\neg c}$	ccr	ccr$^{\neg c}$	ccr	ccr$^{\neg c}$	ccr	ccr$^{\neg c}$	ccr	ccr$^{\neg c}$	ccr	ccr$^{\neg c}$
OLMo2-13B	Pre	0.76	0.75	0.71	0.71	0.84	0.82	0.73	0.70	0.74	0.73	0.75	0.74
	Post	0.78	0.73	0.85	0.85	0.85	0.79	0.80	0.70	0.74	0.71	0.80	0.75
	Pre+Post	0.77	0.75	0.78	0.78	0.84	0.81	0.77	0.70	0.74	0.72	0.78	0.75
OLMo2-7B	Pre	1.08	1.07	1.01	(1.02)	1.17	1.22	1.43	1.54	1.02	1.06	1.12	1.15
	Post	1.85	1.95	1.63	1.52	1.73	1.87	1.67	1.84	1.74	1.71	1.72	1.77
	Pre+Post	1.43	1.43	1.32	1.25	1.45	1.49	1.56	1.65	1.37	1.32	1.42	1.41
OLMo-7B	Pre	1.22	1.22	(0.95)	(0.96)	1.08	1.08	1.30	1.29	1.06	1.06	1.11	1.11
	Post	1.28	1.30	1.17	1.18	1.29	1.34	1.25	1.33	1.15	1.18	1.22	1.26
	Pre+Post	1.25	1.25	1.05	1.05	1.18	1.19	1.27	1.30	1.10	1.11	1.16	1.17

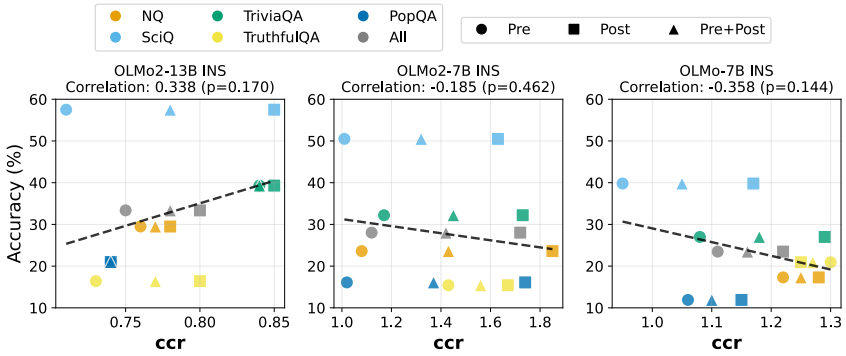


Fig. 5: Pearson correlation between content groundness and task accuracy. Each data point reflects the accuracy of the dataset and the ccr score evaluated with this dataset. Each dataset is evaluated with three settings (pre, post and pre+post), thus these three settings can have different ccr scores but the same accuracy.

5.2 What Factors Impact Content Groundness?

Table 1 and Fig. 5-7 show the results of different OLMo and Llama models evaluated on five benchmark datasets. We observe from Table 1 that the differences between **ccr** and **ccr**$^{\neg c}$ scores are small, suggesting that **our method is robust to variations in the model's self-reported confidence**.

Larger Models Are Not More Content-grounded. As shown by Table 1, confidence generation in OLMo2-13B INS is more influenced by confidence-related training data (ccr < 1), while content-related data plays a greater role in smaller-sized models OLMo-7B and OLMo2-7B INS models (ccr > 1). One possible explanation

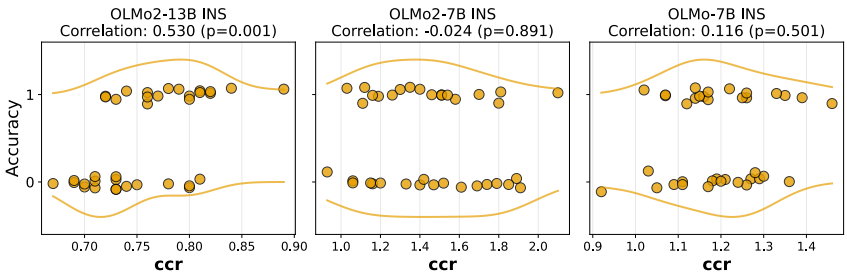


Fig. 6: Pearson correlation between content groundness and sample correctness with density estimation plot [43] (Gaussian kernel). We split the samples of each dataset into a correct set (accuracy = 1) and an incorrect set (accuracy = 0). Each point presents one set and its corresponding ccr score.

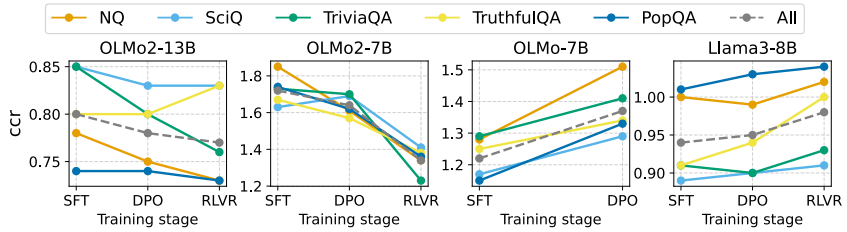


Fig. 7: Impact of different post-training schemes on the content groundness. We plot the training stages of the same model over time. The examined training samples are from the post-training data of the corresponding model. We include models of Llama3-8B as their post-training data are publicly available.

is that larger models, due to their higher capacity, are more sensitive to stylistic or superficial patterns in training data, such as linguistic markers of certainty, while smaller, less expressive models rely more directly on content-aligned signals.

Content Groundness Is Not Correlated with Task Accuracy, but can be impacted by sample correctness. [5] As shown in Fig. 5, the correlations between content groundness and task accuracy are only moderate (Pearson's $r > 0.3$) [10] and not statistically significant ($p > 0.05$). This suggests that content groundness is not correlated with task accuracy. However, we observe a significantly high correlation between content groundness and sample correctness on OLMo2-13B INS from Fig. 6. Even though this observation does not generalize to other smaller OLMo models. We hypothesize that when LLMs answer correctly, they may be drawing upon training examples that are similar to the input, leading them to ground their expressed confidence more heavily in content-relevant information.

[5] Following previous work [6,34,45,48], we use LLM-as-a-Judge, Prometheus-8x7b-v2.0 [25], to assess the semantical equivalence of the generated answers and ground truths.

Post-training Schemes Can Impact Content Groundness. Fig. 7 demonstrates that the impact of post-training schemes on content groundness can be opposite on different models. E.g., DPO or RLVR improves the content groundness for OLMo-7B and Llama3-8B but not for OLMo2-13B and OLMo2-7B models. We also observe that post-training schemes generally do not reverse the content groundness results, i.e., scores are either all under one or above one, which may indicate that pre-training schemes are more likely to determine if content groundness is greater than 1.

6 Discussions

This section offers a discussion aimed at deepening the understanding of our method and addressing the limitations.

Practical Constraints. Our findings are constrained to a limited set of LLMs, primarily the OLMo and Llama families, due to restricted access to the pre-training and post-training data of other proprietary models. Since TracVC relies on analyzing training data influence, the lack of transparency and availability of training data for widely used commercial LLMs (e.g., GPT-5) limits the generalizability of our conclusions. Note that this is not a weakness of our method but rather a practical constraint. Future work could extend this analysis as more open-source models and datasets become available.

Simplified Separation of Content- and Confidence-Related Influences. TracVC approximates content- and confidence-related influences using two disjoint sets of retrieved documents. In practice, these influences may overlap, and a single training example may encode both content- and confidence-related signals. Our decomposition should thus be viewed as a coarse-grained approximation rather than a strict separation of internal model mechanisms.

Limited Coverage of Studied Samples and Influence Estimation Methods. Our retrieval stage relies on BM25-based lexical matching, which emphasizes surface-level word overlap. As a result, semantically relevant but lexically dissimilar training examples may be omitted, while examples with shared keywords but limited relevance may be retrieved. Consequently, our influence estimates are confined to a retrieval-constrained subset of the training data rather than the full corpus. We choose to retrieve a fixed number of training examples ($k = 10$) per category. Although this choice balances computational efficiency and interpretability, it is inherently heuristic and may lead to variability in the estimates. Furthermore, we employ a gradient-based influence estimation method adapted from TracIn for scalability. While this approach is well grounded in prior work, we do not compare it against alternative influence estimation techniques, such as Hessian-based methods, which may offer greater precision but do not currently scale to LLMs of the size considered in this study. Overall, **our method should be viewed as an initial step toward understanding training-data origins of verbalized confidence, rather than a comprehensive solution.**

7 Conclusion

LLMs often exhibit overconfidence, raising the question of whether their confidence is grounded in content-relevant training data. In this paper, we introduced TracVC, a method for tracing the origins of LLM confidence back to specific types of training examples. Our analysis shows that OLMo2-13B models are more influenced by confidence-related than content-related data when estimating confidence, revealing a potential source of miscalibration and risk to trustworthiness. We also find that larger LLMs are not necessarily more content-grounded, and that content groundness can be affected by sample correctness and the extent of post-training. These findings underscore the need for future research on how pre-training and fine-tuning strategies shape confidence grounding, with the ultimate goal of developing models whose confidence reliably reflects content.

More broadly, TracVC offers a general, data-centric approach for interpreting LLM behavior through the lens of training data influence, and can be extended to study other aspects of model behavior, such as answer selection and reasoning patterns. We view TracVC as an initial step toward training-data-level explanations for more transparent and trustworthy LLMs.

Appendix

Table 2: Details of the pre- and post-training data. Given the massive scale of the pre-training corpus, we do not construct our own index but instead rely on the publicly available index provided by Elazar et al. [12] that builds on Dolma.

Model	Pre-training Data	Post-training Data
OLMo2-13B INS	dolma v1.7	tulu-3-sft-olmo-2-mixture, olmo-2-1124-13b-preference-mix, RLVR-MATH
OLMo2-13B-DPO	dolma v1.7	tulu-3-sft-olmo-2-mixture, olmo-2-1124-13b-preference-mix
OLMo2-13B-SFT	dolma v1.7	tulu-3-sft-olmo-2-mixture
Llama3-8B-INS	-	tulu-3-sft-mixture, RLVR-GSM-MATH-IF-Mixed-Constraints
Llama3-8B-DPO	-	tulu-3-sft-mix, llama-3.1-tulu-3-8b-pref-mix, llama-3.1-tulu-3-8b-pref-mix
Llama3-8B-SFT	-	tulu-3-sft-mixture
OLMo2-7B-INS	dolma v1.7	tulu-3-sft-olmo-2-mixture, olmo-2-1124-7b-preference-mix, RLVR-GSM
OLMo2-7B-DPO	dolma v1.7	tulu-3-sft-olmo-2-mixture, olmo-2-1124-7b-preference-mix
OLMo2-7B-SFT	dolma v1.7	tulu-3-sft-olmo-2-mixture
OLMo-7B-INS	dolma v1.7	tulu-v2-sft-mixture, ultrafeedback-binarized-cleaned
OLMo-7B-SFT	dolma v1.7	tulu-v2-sft-mixture

Acknowledgments. This research has been funded by the Vienna Science and Technology Fund (WWTF) [10.47379/VRG19008] "Knowledge infused Deep Learning for Natural Language Processing".

Disclosure of Interests. The authors have no relevant financial or non-financial interests to disclose.

References

1. Bae, J., Ng, N., Lo, A., Ghassemi, M., Grosse, R.B.: If Influence Functions are the Answer, Then What is the Question? Advances in Neural Information Processing Systems **35**, 17953–17967 (2022), https://proceedings.neurips.cc/paper_files/paper/2022/hash/7234e0c36fdbcb23e7bd56b68838999b-Abstract-Conference.html
2. Basu, S., Pope, P., Feizi, S.: Influence Functions in Deep Learning Are Fragile. In: International Conference on Learning Representations (2020), https://openreview.net/forum?id=xHKVVHGDOEk
3. Bejan, I., Sokolov, A., Filippova, K.: Make every example count: On the stability and utility of self-influence for learning from noisy NLP datasets. In: Bouamor, H., Pino, J., Bali, K. (eds.) Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing. pp. 10107–10121. Association for Computational Linguistics, Singapore (Dec 2023). https://doi.org/10.18653/v1/2023.emnlp-main.625
4. Chang, T.A., Rajagopal, D., Bolukbasi, T., Dixon, L., Tenney, I.: Scalable Influence and Fact Tracing for Large Language Model Pretraining (Dec 2024). https://doi.org/10.48550/arXiv.2410.17413, arXiv:2410.17413 [cs]
5. Chen, J., Mueller, J.: Quantifying uncertainty in answers from any language model and enhancing their trustworthiness. In: Ku, L.W., Martins, A., Srikumar, V. (eds.) Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 5186–5200. Association for Computational Linguistics, Bangkok, Thailand (Aug 2024). https://doi.org/10.18653/v1/2024.acl-long.283
6. Chen, T., Asai, A., Mireshghallah, N., Min, S., Grimmelmann, J., Choi, Y., Hajishirzi, H., Zettlemoyer, L., Koh, P.W.: CopyBench: Measuring literal and non-literal reproduction of copyright-protected text in language model generation. In: Al-Onaizan, Y., Bansal, M., Chen, Y.N. (eds.) Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing. pp. 15134–15158. Association for Computational Linguistics, Miami, Florida, USA (Nov 2024). https://doi.org/10.18653/v1/2024.emnlp-main.844
7. Choe, S.K., Ahn, H., Bae, J., Zhao, K., Kang, M., Chung, Y., Pratapa, A., Neiswanger, W., Strubell, E., Mitamura, T., Schneider, J., Hovy, E., Grosse, R., Xing, E.: What is your data worth to gpt? llm-scale data valuation with influence functions (2024). https://doi.org/10.48550/arXiv.2405.13954
8. Chuang, Y.S., Cohen-Wang, B., Shen, Z., Wu, Z., Xu, H., Lin, X.V., Glass, J.R., Li, S.W., Yih, W.T.: SelfCite: Self-supervised alignment for context attribution in large language models. In: Singh, A., Fazel, M., Hsu, D., Lacoste-Julien, S., Berkenkamp, F., Maharaj, T., Wagstaff, K., Zhu, J. (eds.) Proceedings of the 42nd International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 267, pp. 10839–10858. PMLR (13–19 Jul 2025), https://proceedings.mlr.press/v267/chuang25a.html
9. vLLM Contributors: vllm: A high-throughput and memory-efficient llm serving library (2023), https://github.com/vllm-project/vllm, accessed: 2025-01
10. DATAtab Team: Pearson correlation: A beginner's guide (2025), https://datatab.net/tutorial/pearson-correlation, dATAtab: Online Statistics Calculator. DATAtab e.U., Graz, Austria
11. Elastic: Elasticsearch: Distributed, restful search and analytics engine (2025), https://www.elastic.co/elasticsearch/, version 8.17.2, released February 11, 2025

12. Elazar, Y., Bhagia, A., Magnusson, I., Ravichander, A., Schwenk, D., Suhr, A., Walsh, E.P., Groeneveld, D., Soldaini, L., Singh, S., Hajishirzi, H., Smith, N.A., Dodge, J.: What's in my big data? In: ICLR (2024), https://openreview.net/forum?id=RvfPnOkPV4
13. Geng, J., Cai, F., Wang, Y., Koeppl, H., Nakov, P., Gurevych, I.: A survey of confidence estimation and calibration in large language models. In: Duh, K., Gomez, H., Bethard, S. (eds.) Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers). pp. 6577–6595. Association for Computational Linguistics, Mexico City, Mexico (Jun 2024). https://doi.org/10.18653/v1/2024.naacl-long.366
14. Grattafiori, A., Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Vaughan, A., et al.: The llama 3 herd of models (2024). https://doi.org/10.48550/arXiv.2407.21783
15. Groeneveld, D., Beltagy, I., Walsh, E., Bhagia, A., Kinney, R., Tafjord, O., et al.: OLMo: Accelerating the science of language models. In: Ku, L.W., Martins, A., Srikumar, V. (eds.) Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 15789–15809. Association for Computational Linguistics, Bangkok, Thailand (Aug 2024). https://doi.org/10.18653/v1/2024.acl-long.841
16. Grosse, R., Bae, J., Anil, C., Elhage, N., Tamkin, A., Tajdini, A., Steiner, B., Li, D., Durmus, E., Perez, E., Hubinger, E., Lukošiūtė, K., Nguyen, K., Joseph, N., McCandlish, S., Kaplan, J., Bowman, S.R.: Studying Large Language Model Generalization with Influence Functions. https://doi.org/10.48550/arXiv.2308.03296
17. Guo, C., Pleiss, G., Sun, Y., Weinberger, K.Q.: On calibration of modern neural networks. In: Precup, D., Teh, Y.W. (eds.) Proceedings of the 34th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 70, pp. 1321–1330. PMLR (06–11 Aug 2017), https://proceedings.mlr.press/v70/guo17a.html
18. Guo, H., Rajani, N., Hase, P., Bansal, M., Xiong, C.: FastIF: Scalable influence functions for efficient model interpretation and debugging. In: Moens, M.F., Huang, X., Specia, L., Yih, S.W.t. (eds.) Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. pp. 10333–10350. Association for Computational Linguistics, Online and Punta Cana, Dominican Republic (Nov 2021). https://doi.org/10.18653/v1/2021.emnlp-main.808
19. Hammoudeh, Z., Lowd, D.: Identifying a Training-Set Attack's Target Using Renormalized Influence Estimation. In: Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security. pp. 1367–1381. CCS '22, Association for Computing Machinery, New York, NY, USA (Nov 2022). https://doi.org/10.1145/3548606.3559335
20. Hammoudeh, Z., Lowd, D.: Training data influence analysis and estimation: a survey. Machine Learning **113**(5), 2351–2403 (May 2024). https://doi.org/10.1007/s10994-023-06495-7
21. Han, X., Tsvetkov, Y.: ORCA: interpreting prompted language models via locating supporting data evidence in the ocean of pretraining data. CoRR **abs/2205.12600** (2022). https://doi.org/10.48550/ARXIV.2205.12600
22. Joshi, M., Choi, E., Weld, D., Zettlemoyer, L.: TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In: Barzilay, R., Kan, M.Y. (eds.) Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 1601–1611. Association for Computational Linguistics, Vancouver, Canada (Jul 2017). https://doi.org/10.18653/v1/P17-1147

23. Kadavath, C., Askell, A., Kernion, J., Henighan, T., Mann, B., Krueger, G., Kreps, S., McKane, A., Mistry, G., Kim, J., et al.: Prompting GPT-3 to be reliable. In: Proceedings of the 11th International Conference on Learning Representations (ICLR) (2023), https://openreview.net/forum?id=K2i-xk7IUR
24. Khalifa, M., Wadden, D., Strubell, E., Lee, H., Wang, L., Beltagy, I., Peng, H.: Source-aware training enables knowledge attribution in language models. In: First Conference on Language Modeling (2024), https://openreview.net/forum?id=UPyWLwciYz
25. Kim, S., Suk, J., Longpre, S., Lin, B.Y., Shin, J., Welleck, S., Neubig, G., Lee, M., Lee, K., Seo, M.: Prometheus 2: An open source language model specialized in evaluating other language models. In: Al-Onaizan, Y., Bansal, M., Chen, Y.N. (eds.) Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing. pp. 4334–4353. Association for Computational Linguistics, Miami, Florida, USA (Nov 2024). https://doi.org/10.18653/v1/2024.emnlp-main.248
26. Koh, P.W., Liang, P.: Understanding Black-box Predictions via Influence Functions. In: Proceedings of the 34th International Conference on Machine Learning. pp. 1885–1894. PMLR (Jul 2017), https://proceedings.mlr.press/v70/koh17a.html, iSSN: 2640-3498
27. Kumar, A., Morabito, R., Umbet, S., Kabbara, J., Emami, A.: Confidence under the hood: An investigation into the confidence-probability alignment in large language models. In: Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 315–334. Association for Computational Linguistics, Bangkok, Thailand (August 2024). https://doi.org/10.18653/v1/2024.acl-long.20
28. Kwiatkowski, T., Palomaki, J., Redfield, O., Collins, M., Parikh, A., Alberti, C., Epstein, D., Polosukhin, I., Devlin, J., Lee, K., Toutanova, K., Jones, L., Kelcey, M., Chang, M.W., Dai, A.M., Uszkoreit, J., Le, Q., Petrov, S.: Natural questions: A benchmark for question answering research. Transactions of the Association for Computational Linguistics **7**, 452–466 (2019). https://doi.org/10.1162/tacl_a_00276
29. Kwon, Y., Wu, E., Wu, K., Zou, J.: Datainf: Efficiently estimating data influence in loRA-tuned LLMs and diffusion models. In: The Twelfth International Conference on Learning Representations (2024), https://openreview.net/forum?id=9m02ib92Wz
30. Lambert, N., Morrison, J., Pyatkin, V., Huang, S., Ivison, H., Brahman, F., Miranda, L.J.V., Liu, A., Dziri, N., Lyu, S., Gu, Y., Malik, S., Graf, V., Hwang, J.D., Yang, J., Bras, R.L., Tafjord, O., Wilhelm, C., Soldaini, L., Smith, N.A., Wang, Y., Dasigi, P., Hajishirzi, H.: Tulu 3: Pushing frontiers in open language model post-training (2025). https://doi.org/10.48550/arXiv.2411.15124
31. Lin, H., Long, J., Xu, Z., Zhao, W.: Token-wise influential training data retrieval for large language models. In: Ku, L.W., Martins, A., Srikumar, V. (eds.) Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 841–860. Association for Computational Linguistics, Bangkok, Thailand (Aug 2024). https://doi.org/10.18653/v1/2024.acl-long.48, https://aclanthology.org/2024.acl-long.48/
32. Lin, S., Hilton, J., Evans, O.: TruthfulQA: Measuring how models mimic human falsehoods. In: Muresan, S., Nakov, P., Villavicencio, A. (eds.) Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 3214–3252. Association for Computational Linguistics, Dublin, Ireland (May 2022). https://doi.org/10.18653/v1/2022.acl-long.229

33. Liu, J., Blanton, T., Elazar, Y., Min, S., Chen, Y.S., Chheda-Kothary, A., et al.: OLMoTrace: Tracing language model outputs back to trillions of training tokens. In: Mishra, P., Muresan, S., Yu, T. (eds.) Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations). pp. 178–188. Association for Computational Linguistics, Vienna, Austria (Jul 2025). https://doi.org/10.18653/v1/2025.acl-demo.18
34. Liu, X., Khalifa, M., Wang, L.: Litcab: Lightweight language model calibration over short- and long-form responses. In: Proceedings of the Twelfth International Conference on Learning Representations. ICLR (2024). https://doi.org/10.48550/arXiv.2310.19208
35. Mallen, A., Asai, A., Zhong, V., Das, R., Khashabi, D., Hajishirzi, H.: When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In: Rogers, A., Boyd-Graber, J., Okazaki, N. (eds.) Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 9802–9822. Association for Computational Linguistics, Toronto, Canada (Jul 2023). https://doi.org/10.18653/v1/2023.acl-long.546
36. Ni, S., Bi, K., Yu, L., Guo, J.: Are large language models more honest in their probabilistic or verbalized confidence? In: He, X., Ren, Z., Tang, R. (eds.) Information Retrieval. pp. 124–135. Springer Nature Singapore, Singapore (2025). https://doi.org/10.1007/978-981-96-1710-4_10
37. Park, S.M., Georgiev, K., Ilyas, A., Leclerc, G., Madry, A.: TRAK: Attributing Model Behavior at Scale. In: Proceedings of the 40th International Conference on Machine Learning. pp. 27074–27113. PMLR (Jul 2023), https://proceedings.mlr.press/v202/park23c.html, iSSN: 2640-3498
38. Pete, W., Luca, S., Dirk, G., Kyle, L., Shane, A., Akshita, B., et al.: 2 olmo 2 furious. arXiv preprint arXiv:2501.00656 (2024). https://doi.org/10.48550/arXiv.2501.00656
39. Pruthi, G., Liu, F., Kale, S., Sundararajan, M.: Estimating Training Data Influence by Tracing Gradient Descent. In: Advances in Neural Information Processing Systems. vol. 33, pp. 19920–19930. Curran Associates, Inc. (2020), https://proceedings.neurips.cc/paper/2020/hash/e6385d39ec9394f2f3a354d9d2b88eec-Abstract.html
40. Rafailov, R., Sharma, A., Mitchell, E., Manning, C.D., Ermon, S., Finn, C.: Direct preference optimization: Your language model is secretly a reward model. In: Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., Levine, S. (eds.) Advances in Neural Information Processing Systems. vol. 36, pp. 53728–53741. Curran Associates, Inc. (2023), https://papers.nips.cc/paper_files/paper/2023/hash/a85b405ed65c6477a4fe8302b5e06ce7-Abstract-Conference.html
41. Robertson, S.E., Walker, S., Jones, S., Hancock-Beaulieu, M., Gatford, M.: Okapi at TREC-3. In: Harman, D.K. (ed.) Proceedings of The Third Text REtrieval Conference, TREC 1994, Gaithersburg, Maryland, USA, November 2-4, 1994. NIST Special Publication, vol. 500-225, pp. 109–126. National Institute of Standards and Technology (NIST) (1994), http://trec.nist.gov/pubs/trec3/papers/city.ps.gz
42. Schioppa, A., Zablotskaia, P., Vilar, D., Sokolov, A.: Scaling up influence functions. Proceedings of the AAAI Conference on Artificial Intelligence **36**(8), 8179–8186 (Jun 2022). https://doi.org/10.1609/aaai.v36i8.20791
43. Scott, D.W.: Multivariate density estimation: theory, practice, and visualization. John Wiley & Sons (2015). https://doi.org/10.1002/9780470316849
44. Steyvers, M., Lemus, H.T., Kumar, A., Belem, C.G., Karny, S., Hu, X., Mayer, L.W., Smyth, P.: What large language models know and what people think they know. Nature Machine Intelligence **7**(2), 221–231 (2024). https://doi.org/10.1038/s42256-024-00976-7

45. Tian, K., Mitchell, E., Zhou, A., Sharma, A., Rafailov, R., Yao, H., Finn, C., Manning, C.: Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In: Bouamor, H., Pino, J., Bali, K. (eds.) Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing. pp. 5433–5442. Association for Computational Linguistics, Singapore (Dec 2023). https://doi.org/10.18653/v1/2023.emnlp-main.330
46. Welbl, J., Liu, N.F., Gardner, M.: Crowdsourcing multiple choice science questions. In: Derczynski, L., Xu, W., Ritter, A., Baldwin, T. (eds.) Proceedings of the 3rd Workshop on Noisy User-generated Text. pp. 94–106. Association for Computational Linguistics, Copenhagen, Denmark (Sep 2017). https://doi.org/10.18653/v1/W17-4413
47. Xia, M., Malladi, S., Gururangan, S., Arora, S., Chen, D.: LESS: Selecting Influential Data for Targeted Instruction Tuning (Jun 2024). https://doi.org/10.48550/arXiv.2402.04333, arXiv:2402.04333 [cs]
48. Xia, Y., Luz De Araujo, P.H., Zaporojets, K., Roth, B.: Influences on LLM calibration: A study of response agreement, loss functions, and prompt styles. In: Che, W., Nabende, J., Shutova, E., Pilehvar, M.T. (eds.) Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 3740–3761. Association for Computational Linguistics, Vienna, Austria (Jul 2025). https://doi.org/10.18653/v1/2025.acl-long.188
49. Xia, Y., Ulmer, D., Blevins, T., Liu, Y., Schütze, H., Roth, B.: Calibration is not enough: Evaluating confidence estimation under language variations. arXiv preprint arXiv:2601.08064 (2026), https://arxiv.org/abs/2601.08064
50. Xiong, M., Hu, Z., Lu, X., Li, Y., Fu, J., He, J., Hooi, B.: Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. In: Proceedings of the 12th International Conference on Learning Representations (ICLR) (2024), https://openreview.net/forum?id=gjeQKFxFpZ
51. Xu, Z., Song, T., Lee, Y.C.: Confronting verbalized uncertainty: Understanding how llm's verbalized uncertainty influences users in ai-assisted decision-making. International Journal of Human-Computer Studies **197**, 103455 (2025). https://doi.org/10.1016/j.ijhcs.2025.103455
52. Yang, D., Tsai, Y.H.H., Yamada, M.: On verbalized confidence scores for LLMs. In: ICLR Workshop: Quantify Uncertainty and Hallucination in Foundation Models: The Next Frontier in Reliable AI (2025), https://openreview.net/forum?id=CVRdNQvFPE
53. Yeh, C.K., Taly, A., Sundararajan, M., Liu, F., Ravikumar, P.: First is Better Than Last for Language Data Influence (Oct 2022). https://doi.org/10.48550/arXiv.2202.11844, arXiv:2202.11844 [cs]
54. Yin, Z., Sun, Q., Guo, Q., Wu, J., Qiu, X., Huang, X.: Do large language models know what they don't know? In: Rogers, A., Boyd-Graber, J., Okazaki, N. (eds.) Findings of the Association for Computational Linguistics: ACL 2023. pp. 8653–8665. Association for Computational Linguistics, Toronto, Canada (Jul 2023). https://doi.org/10.18653/v1/2023.findings-acl.551
55. Zhou, K., Hwang, J., Ren, X., Sap, M.: Relying on the unreliable: The impact of language models' reluctance to express uncertainty. In: Ku, L.W., Martins, A., Srikumar, V. (eds.) Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 3623–3643. Association for Computational Linguistics, Bangkok, Thailand (Aug 2024). https://doi.org/10.18653/v1/2024.acl-long.198

E. Calibration Is Not Enough: Evaluating Confidence Estimation Under Language Variations

Authors: Yuxi Xia, Dennis Ulmer, Terra Blevins, Yihong Liu, Hinrich Schütze, Benjamin Roth

Status: Under review

URL: https://arxiv.org/pdf/2601.08064

Reference: Xia et al. (2026c)

Work Division

Yuxi Xia: conceptualization, data curation, formal analysis, investigation, methodology, software, validation, visualization, original draft preparation, review and editing
Dennis Ulmer: formal analysis, methodology, original draft preparation, review and editing
Terra Blevins: conceptualization, formal analysis, methodology, review and editing
Yihong Liu: conceptualization, formal analysis, review and editing
Hinrich Schütze: review and editing
Benjamin Roth: funding acquisition, methodology, project administration, resources, supervision, review and editing.

Calibration Is Not Enough: Evaluating Confidence Estimation Under Language Variations

Yuxi Xia[1], Dennis Ulmer[3], Terra Blevins[4], Yihong Liu[5], Hinrich Schütze[5], Benjamin Roth[1,2]

[1]Faculty of Computer Science, UniVie Doctoral School Computer Science,
[2]Faculty of Philological and Cultural Studies, University of Vienna, Austria
[3]ILLC, University of Amsterdam, Netherlands
[4]Khoury College of Computer Sciences, Northeastern University, USA
[5]LMU Munich, Munich Center for Machine Learning (MCML), Germany

Correspondence: yuxi.xia@univie.ac.at

Abstract

Confidence estimation (CE) indicates how reliable the answers of large language models (LLMs) are, and can impact user trust and decision-making. Existing work evaluates CE methods almost exclusively through *calibration*, examining whether stated confidence aligns with accuracy, or *discrimination*, whether confidence is ranked higher for correct predictions than incorrect ones. However, these facets ignore pitfalls of CE in the context of LLMs and language variation: confidence estimates should remain consistent under semantically equivalent prompt or answer variations, and should change when the answer meaning differs. Therefore, we present a comprehensive evaluation framework for CE that measures their confidence quality on three new aspects: **robustness** of confidence against prompt perturbations, **stability** across semantic equivalent answers, and **sensitivity** to semantically different answers. In our work, we demonstrate that common CE methods for LLMs often fail on these metrics: methods that achieve good performance on calibration or discrimination are not robust to prompt variations or are not sensitive to answer changes. Overall, our framework reveals limitations of existing CE evaluations relevant for real-world LLM use cases and provides practical guidance for selecting and designing more reliable CE methods.

1 Introduction

Large Language Models (LLMs) are increasingly effective at answering fact-based questions across domains such as medicine, law, and education (Alqahtani et al., 2023; Alfertshofer et al., 2024; Xiao et al., 2025). In these high-stakes settings, it is paramount to improve user trust and support safer human–AI interactions (Kadavath et al., 2022; Zhang et al., 2024) in order to avoid harm and produce positive outcomes. To this end, *confidence estimation* (CE; Kadavath et al., 2023; Tian et al., 2023; Chen and Mueller, 2024) indicates how certain and therefore reliable a model's answer is.

Existing CE work (Guo et al., 2017; Osawa et al., 2019; Ovadia et al., 2019; Groot and Valdenegro Toro, 2024; Ni et al., 2025; Xiong et al., 2024) mainly focus on evaluating calibration (whether stated confidences match expected model accuracy), or discrimination (whether it ranks the right answers higher than the wrong ones). Compared to prior work in machine learning and much of the existing literature in natural language processing, this line of research overlooks a key property of language (models): Language is extremely variable, and while the surface form, i.e. a particular choice of words, might change, the meaning of an expression can stay untouched (Baan et al., 2023). However, a method may be well-calibrated yet produce confidence values that fluctuate under minor prompt variations (Xia et al., 2025a), or fail to distinguish between semantically equivalent and different answers, rendering it less useful in real-life scenarios such as user-facing QA, where the prompt variation are large across users.

This paper presents a more comprehensive evaluation paradigm for CE methods in LLMs based on language variation, including three novel aspects in addition to calibration: Intuitively, these check whether confidence estimates stay constant when only the surface form changes, and test whether estimates differ with a change in the underlying meaning. Specifically: **Robustness (P-RB)** measures whether CE methods give consistent confidence estimates for the same answer when prompted in different but semantically equivalent ways. Further, **Stability (A-STB)** requires CE methods to assign consistent confidences to semantically equivalent answers, and **sensitivity (A-SST)** measures whether confidence estimates change when the answer differs in content. In summary, our contributions are:

- We propose three complementary metrics (robustness, stability, and sensitivity) that go beyond calibration to provide a more comprehensive evaluation of CE methods for LLMs;
- Using these metrics, we conduct an extensive evaluation of five diverse CE methods across nine LLMs from five model families;
- Our results show that strong calibration or ranking performance does not guarantee robustness to prompt perturbations or effective discrimination between equivalent and different answers, highlighting important limitations of existing CE evaluations;
- We analyze how different CE methods trade off robustness, stability, and sensitivity, and discuss their implications for method selection under different evaluation priorities.

2 Related Work

Confidence Estimation in LLMs. CE has emerged as a crucial research direction for improving user trust and model interpretability (Tian et al., 2023; Chen et al., 2024; Geng et al., 2024). Currently, there are six common approaches (Geng et al., 2024; Shorinwa et al., 2025; Vashurin et al., 2025) to estimating LLM confidence: (1) *logit-based*, such as token-level probability (Zhu et al., 2023; Huang et al., 2025); (2) *linguistic confidence* (Mielke et al., 2022; Lin et al., 2022; Xiong et al., 2024; Ulmer et al., 2025), which elicits verbalized confidence directly from the model; (3) *self-evaluation* (Kadavath et al., 2022) such as P(True), where a model judges the correctness of its own answers; (4) *auxiliary-based* methods (Ulmer et al., 2024; Xia et al., 2025a), which employ an auxiliary model trained to predict the confidence of a target LLM based on answer content and predicted correctness; (5) *probes* (Liu et al., 2024; Burns et al., 2023; Kadavath et al., 2022), which comprise smaller, often linear classifiers trained on internal model representations. Lastly, (6) *consistency-based methods* (Lin et al., 2024; Manakul et al., 2023; Xiong et al., 2024) rely on multiple stochastic generations obtained by prompting the LLM with loose temperatures and diverse prompts. Confidence is then estimated based on the similarity or frequency of the generated outputs. However, these methods are expensive in practice due to the need for multiple generations. Since our proposed metrics themselves rely on comparing the confidence of multiple different responses under perturbation, consistency-based methods are not applicable in our framework, and we focus on computationally cheaper, single-pass CE methods instead.

Evaluation of Confidence Estimators. LLMs are typically evaluated using task-level metrics (Le Bronnec et al., 2024; Moe et al., 2025; Xia et al., 2025b), which are not suitable for CE methods. Instead, calibration metrics, including ECE (Guo et al., 2017) and its variants (Kumar et al., 2019; Nixon et al., 2019; Kirchenbauer et al., 2022; Błasiok and Nakkiran, 2024), and Brier score (Brier, 1950), are used to measure the alignment between predicted confidence and actual correctness (Groot and Valdenegro Toro, 2024; Zhou et al., 2024; Stengel-Eskin and Durme, 2023). Ranking-based metrics such as AUROC and AUPRC (Bradley, 1997) further evaluate whether CE methods can distinguish correct from incorrect responses (Vazhentsev et al., 2022; Tian et al., 2023; Ulmer et al., 2024), in-distribution from out-of-distribution inputs (Ovadia et al., 2019; Charpentier et al., 2020; Ulmer et al., 2022), or whether loss correlates with decreased confidence (Ulmer et al., 2022). However, existing metrics do not assess the robustness of CE methods to prompt perturbations, as studied for LLM outputs for instance by ProSA (Zhuo et al., 2024), or their ability to distinguish between similar and dissimilar responses.

3 Background

In the following, we discuss the necessary preliminaries and supply a motivating example for our new CE metrics. In the rest of the paper, we use $\mathbf{x}$ to denote an input sequence to an LLM, $\mathbf{y}$ a model generation or response and $\mathbf{y}^*$ as the intended or correct response for $\mathbf{x}$. In cases where there might exist multiple correct responses, we denote the corresponding set by $\mathcal{Y}^*$. In addition, a prompt to the language model is denoted as $\mathbf{t}$. Then, we write a confidence estimator as a function f that predicts a confidence value c conditioned on $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{t}$, written $c = f(\mathbf{y}, \mathbf{x}, \mathbf{t})$ as a shorthand. So in the case of sequence likelihood, f would involve passing $\mathbf{x}$ and $\mathbf{t}$ to the target LLM to obtain token probabilities, and e.g. in the case of an auxiliary model, f can be directly equated with the secondary model.

3.1 Preliminaries

Before demonstrating challenges with language variation, we first briefly revisit the most common

metrics for CE evaluation. In the first case, we consider the expected calibration error (Guo et al., 2017):

$$\text{ECE} = \mathbb{E}\big[|p(y = y^* \mid c) - c|\big], \qquad (1)$$

where the expectation is typically approximated by binning N test predictions into M buckets of equal size by their predicted confidence. The ECE has been widely criticized for not being a *proper scoring rule*, i.e. trivial predictors other than the true distribution can minimize it (Liu et al., 2023), as we will see in the upcoming example. An actual instance of a proper scoring rule is instead the Brier score (Brier, 1950) defined by:

$$\text{Brier} = \frac{1}{n}\sum_{i=1}^{n}\big(c_i - \mathbf{1}(y_i = y_i^*)\big)^2. \qquad (2)$$

Both ECE and Brier score evaluate whether the predicted confidence score aligns directly with (expected) correctness. However, one might also be interested in whether the confidence score helps us to distinguish between correct and incorrect predictions. For this reason, confidence scores are also commonly evaluated through the area under the receiver operating characteristic curve (AUROC; Bradley, 1997), where error detection is framed as a binary classification task, and the ratio of the true to false positive rate is measured under all possible thresholds. Alternative, one might also resort to the area under the precision-recall curve (AUPR). However, all of these example are insufficient to evaluate confidence estimation given the variability of natural language.

3.2 An Illustrative Example

Let us consider a simplified example to demonstrate the shortcomings of the current calibration metrics in the case of language generation. Let us assume that for each input $\mathbf{x}_i$ out of N total samples, we obtain M generations each, which we denote as $\mathbf{y}_i^{(1)}, \ldots, \mathbf{y}_i^{(M)}$. For our example, we sample a "difficulty" $d_i \sim \mathcal{U}[0,1]$ for every $\mathbf{x}_i$. Then, the correctness $\text{corr}_i^{(m)}$ of each $\mathbf{y}_i^{(m)}$ is simulated through sampling $\text{corr}_i^{(m)} \sim \text{Bernoulli}(1 - d_i)$, meaning that the higher the difficulty, the less likely each $\mathbf{y}_i^{(m)}$ is to be correct. Then, we define four different confidence estimators: The oracle confidence estimator f_{oracle} simply returns the actual correctness: $f_{\text{oracle}}(\mathbf{y}_i^{(m)}) = \text{corr}_i^{(m)}$. Furthermore, we define a constant estimator $f_{\text{constant}}(\mathbf{y}_i) = 1 - d_i$ that predicts the difficulty per input, ignoring the actual generation and its correctness. Similarly, let us denote a prior estimator $f_{\text{prior}}(\mathbf{y}_i) = \mathbb{E}_{d_i \sim \mathcal{U}[0,1]}[1 - d_i] = 0.5$, that stoically estimates the expected difficulty across the whole dataset. Lastly, we can define a random estimator $f_{\text{random}}(\mathbf{y}_i) \sim \text{Bernoulli}(1 - d_i)$ that samples a confidence of 1 or 0 randomly from the same distribution as $\text{corr}_i^{(m)}$, but doing so independently.

	ECE ↓	Brier ↓	AUROC ↑	STB ↑	SST ↑
f_{oracle}	0.00	0.00	1.00	1.00	1.00
f_{constant}	0.01	0.17	0.83	1.00	0.00
f_{random}	0.33	0.33	0.67	0.72	0.18
f_{prior}	0.00	0.25	0.50	1.00	0.00

Table 1: Simulation results. Our own proposed metrics are stability (STB) and sensitivity (SST).

Results. We obtain results using a simulation with $N = 100,000$ and $M = 10$ and show the results in Table 1.[1] Unsurprisingly, the oracle predictor achieves the best values across metrics, and the random predictor scores above-random AUROC, but shows the worst calibration. While f_{prior} achieves random-chance AUROC due to only predicting a single value, it achieves perfect ECE and a Brier score of 0.25, demonstrating that ECE is not a proper scoring rule. Similarly, the constant predictor achieves perfect ECE, a relatively low Brier score of 0.17 and an AUROC of 0.83. Although these predictors are deliberately simplified, the implications for confidence estimation in LLMs are staggering: A random baseline that predicts 0 and 1 according to answer difficulty alone can still achieve above-random AUROC. But more importantly, $f_{constant}$, which learns to infer question difficulty alone and ignores any information in the actual answers, can perform well in terms of both calibration and discrimination. A CE method that behaves randomly and one that predicts the same confidence, both independently of the actual answer, are both uninformative in a practical setting. They also form two extreme endpoints of a spectrum, motivating our novel metrics: Stability (STB) measures whether an estimator predicts the same confidence for semantically equivalent answers, while sensitivity (SST) captures whether an esti-

[1] To apply our metrics we introduce in the next Section 4, we make an assumption here that generations $\mathbf{y}_i^{(m)}$ with the same $\text{corr}_i^{(m)}$ have the same meaning.

mator changes its prediction upon encountering a different answer.

4 Methodology

We investigate whether estimated confidence scores are trustworthy based on three qualities: *robustness* ensures that the estimated confidence is robust towards prompt perturbation; *stability* and *sensitivity* quantify how much the CE methods respond to semantically similar and different answers. All metrics are visualized in Figure 1.

4.1 Robustness Under Prompt Perturbation

Motivation. Prompt formatting is known to influence LLM outputs (Min et al., 2022; Wang et al., 2023; Sclar et al., 2024; Feng et al., 2024), and recent work shows that it can also affect the performance of CE methods (Xia et al., 2025a). In practice, prompts are often modified across applications, users, or deployment settings. A reliable CE method should therefore produce stable confidence estimates under such prompt variations, if the resulting answer by the model otherwise stays the same. However, while existing methods using verbalized confidence (e.g. Xiong et al., 2024; Tian et al., 2023) have tested different prompt formats, the robustness of methods that rely on model outputs or internal logits under prompt changes has not been systematically examined. Measuring robustness to prompt perturbations is thus necessary to understand whether CE methods provide consistent confidence scores when prompts might be altered in practice.

Definition 4.1 (Robustness, P-RB)**.** Let $\mathcal{T} = \{\mathbf{t}_1, \ldots, \mathbf{t}_M\}$ be a set of M semantically equivalent prompts. A confidence estimator f is *robust to prompt perturbations* iff

$$\forall \mathbf{t}, \mathbf{t}' \in \mathcal{T}, \quad |f(\mathbf{y}, \mathbf{x}, \mathbf{t}) - f(\mathbf{y}, \mathbf{x}, \mathbf{t}')| \leq \varepsilon.$$

Robustness Metric. We quantify prompt robustness using the mean standard deviation ($\hat{\sigma}_r$, ↑) of estimates across different prompts. Given a prompt set $\mathcal{T} = \{\mathbf{t}_1, \ldots, \mathbf{t}_M\}$ and a dataset of N samples, $\hat{\sigma}_r$ is computed by first measuring the standard deviation $\mathrm{Std}(\cdot)$ of confidence scores across prompts before averaging over all samples:

$$\hat{\sigma}_r = 1 - \frac{1}{N} \sum_{i=1}^{N} \mathrm{Std}\Big(\{ f(\mathbf{y}_i, \mathbf{x}_i, \mathbf{t}_j) \}_{j=1}^{M} \Big). \quad (3)$$

We also prove in Section A.1 that for Definition 4.1 to hold, it must be that $\hat{\sigma}_r > (2 - \varepsilon)/2$.

4.2 Stability and Sensitivity across Answers

Motivation. Even setting aside any changes in the prompt, LLMs using stochastic decoding already naturally produce a variety of responses for a given input. A good CE method should react accordingly to changes in the answer: If the semantics of the answer remain unchanged, so should its probability of being correct, and the confidence estimate should not differ. If the answer is different in meaning, the confidence should be adjusted accordingly. As we saw in Section 3.2, a constant predictor can be well-calibrated, but becomes unreliable if it cannot adapt to semantic changes. Similarly, a random predictor would be sensitive (since the confidence estimate would be different for any new answer regardless), but lack stability when the response is still the same.

To define these notions more formally, let us define a semantic equivalence operation $E(\cdot, \cdot)$ similar to prior work on self-consistency-based methods (Kuhn et al., 2023).[2] Let us assume we have again have a set of M generated responses for a given input $\mathbf{x}$ and a (single) prompt $\mathbf{t}$, which we denote by $\mathcal{Y} = \{\mathbf{y}^{(1)}, \ldots, \mathbf{y}^{(M)}\}$. Then, given some predicted answer $\mathbf{y}_m \in \mathcal{Y}$, we can define $\hat{\mathcal{Y}} = \{\mathbf{y}' \in \mathcal{Y} \mid E(\mathbf{y}_m, \mathbf{y}')\}$ This helps us to outline two new desiderata for confidence estimation methods in the context of language generation:

Definition 4.2 (Stability, A-STB)**.** Let $\mathbf{y}^{(m)}$ be one out of M responses given a prompt $\mathbf{t}$, and $\hat{\mathcal{Y}} \subseteq \mathcal{Y}$ a set of semantically equivalent responses to $\mathbf{y}_m$. Then, a confidence estimator f is *stable* iff

$$\forall \mathbf{y}' \in \hat{\mathcal{Y}}: \quad |f(\mathbf{y}^{(m)}, \mathbf{x}, \mathbf{t}) - f(\mathbf{y}', \mathbf{x}, \mathbf{t})| \leq \varepsilon.$$

From this, we can also define the complementary notion of sensitivity, namely that semantically different answers should receive other confidence estimates:

Definition 4.3 (Sensitivity, A-SST)**.** Let $\mathcal{E}(\mathcal{Y}) = \{\hat{\mathcal{Y}}_l\}_{l=1}^{L}$ be a finite partition of $\mathcal{Y}$ s.t. $\bigcup_{l=1}^{L} \hat{\mathcal{Y}}_l = \mathcal{Y}$ and $\forall \hat{\mathcal{Y}}_l, \hat{\mathcal{Y}}_{l'} \in \mathcal{E}, l \neq l' : \hat{\mathcal{Y}}_l \cap \hat{\mathcal{Y}}_{l'} = \varnothing$. Then, given an input $\mathbf{x}$, prompt $\mathbf{t}$ and that each $\hat{\mathcal{Y}}_l \in \mathcal{E}(\mathcal{Y})$ is a set of semantically equivalent responses, a confidence estimator f is *sensitive* iff

$$\forall \hat{\mathcal{Y}}_l, \hat{\mathcal{Y}}_{l'} \in \mathcal{E}(\mathcal{Y}), l \neq l', \mathbf{y} \in \hat{\mathcal{Y}}_l, \mathbf{y}' \in \hat{\mathcal{Y}}_{l'} : \\ |f(\mathbf{y}, \mathbf{x}, \mathbf{t}) - f(\mathbf{y}', \mathbf{x}, \mathbf{t})| > \varepsilon.$$

[2] In practice, semantic equivalence can be empirically determined using bi-directional entailment classification like in Kuhn et al. (2023) or LLM-as-a-judge. In our case, we choose the latter.

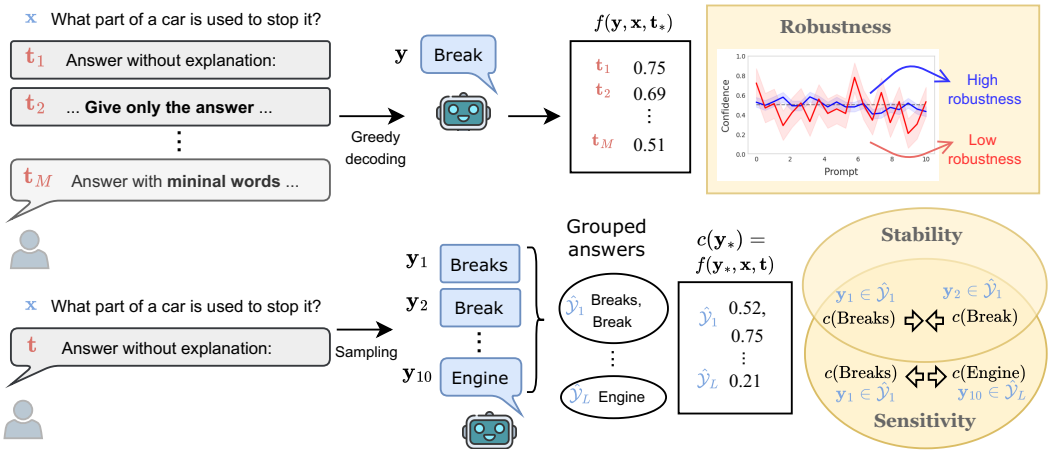


Figure 1: Design of the proposed evaluation metrics for LLM confidence. We assess *robustness* by measuring the variation in confidence scores elicited by semantically equivalent prompt perturbations. *Stability* measures confidence consistency across semantically equivalent answers, while *sensitivity* evaluates whether CE methods assign more distinct confidence scores to semantically different answers than to equivalent ones.

In practice, we do not necessarily check all the different pairs of responses from different sets of semantically equivalent answers $\hat{\mathcal{Y}}_l$, to show that this is true, but just restrict ourselves to the most salient sets.

Stability & Sensitivity Metric. For each test sample, the group with the most answers is denoted as $\hat{\mathcal{Y}}^{\max}$ and used to evaluate stability, while the group with the fewest answers is denoted as $\hat{\mathcal{Y}}^{\min}$ and used for sensitivity comparisons. *Stability* ($\hat{\sigma}_s$, $\uparrow$) measures the variability of confidence scores among semantically equivalent answers:

$$\hat{\sigma}_s = 1 - \frac{1}{N}\sum_{i=1}^{N} \mathrm{Std}\Big(\{f(\mathbf{y}_j, \mathbf{x}_i, \mathbf{t})\}_{\mathbf{y}_j \in \hat{\mathcal{Y}}_i^{\max}}\Big). \tag{4}$$

Applying the same argument as for robustness, we show in Section A.1 that $\hat{\sigma}_s > (2-\varepsilon)/2$ for Definition 4.2 to be true.

Sensitivity (λ, $\uparrow$) quantifies how effectively a CE method distinguishes semantically different answers relative to semantically equivalent ones. Formally, we define it as:

$$\lambda = \frac{1}{N}\sum_{i=1}^{N} |\Delta(\hat{\mathcal{Y}}_i^{\max}, \hat{\mathcal{Y}}_i^{\min}) - \Delta(\hat{\mathcal{Y}}_i^{\max}, \hat{\mathcal{Y}}_i^{\max})|, \tag{5}$$

where

$$c(\mathbf{y}) \equiv f(\mathbf{y}, \mathbf{x}, \mathbf{t}) \tag{6}$$

$$\Delta(\hat{\mathcal{Y}}, \hat{\mathcal{Y}}') = \frac{1}{|\hat{\mathcal{Y}}||\hat{\mathcal{Y}}'|}\sum_{\mathbf{y}\in\hat{\mathcal{Y}}}\sum_{\mathbf{y}'\in\hat{\mathcal{Y}}'} |c(\mathbf{y}) - c(\mathbf{y}')|. \tag{7}$$

When the confidence estimator is both stable and sensitive according to Definitions 4.2 and 4.3, then this metric is bounded by $\lambda > \varepsilon/N \sum_{i=1}^{N} |\hat{\mathcal{Y}}_i^{\max}|^{-1}$, as we derive in Section A.1. A high value of λ indicates that the confidence differences between semantically different answers are much larger than those among equivalent answers, and therefore the CE is sensitive to changes in meaning.

It should be noted that our metrics are in a sense unsupervised, as they do not require any information about the correctness of a response in contrast to other common ways to evaluate CE, as depicted in Figure 2. This is especially attractive in cases where automated correctness judgements are expensive, difficult or noisy, e.g. ambiguous questions (Min et al., 2020) or questions whose answers change over time (Pletenev et al., 2025).

5 Experimental Setup

Our experiments are conducted on an aggregation of four question answering datasets and nine LLMs across different model families and sizes.

Evaluated Methods. We benchmark five CE methods. (1) probability (**Prob.**): aggregates the

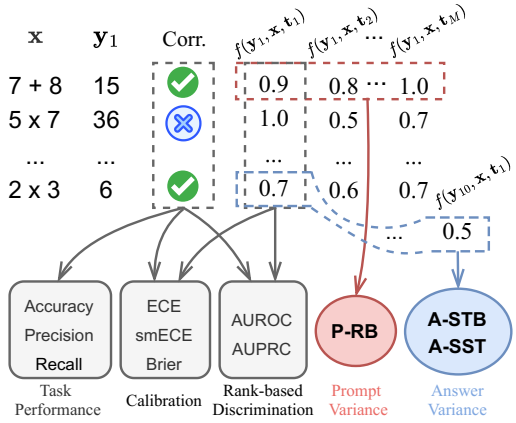


Figure 2: Comparison of different evaluation metrics for confidence estimator, based on the data they require. While other metrics use correctness labels, P-RB, A-STB and A-SST solely rely on measurements from different model answers and prompts.

token probabilities of the generated answer under greedy decoding and normalizes them by length as confidence. (2) Platt Scaling (**PS**; Platt, 1999) applies two scalars, which are obtained by minimizing the mean-squared error between model confidences and answer correctness, to scale the sequence probability. (3) Verbalized confidence (**VC**) directly elicits confidence scores from the LLM itself by prompting it to state how certain it is about its answer, following recent work on self-reported confidence (Tian et al., 2023; Xiong et al., 2024). (4) **P(True)** prompts the LLMs to judge if the response is true or false, and uses the normalized probability of the "True" token as confidence (Kadavath et al., 2022). (5) Calib-1-Focal (**Calib1**) trains an auxiliary calibration classifier (Ulmer et al., 2024; Xia et al., 2025a) with focal loss (Mukhoti et al., 2020), which uses the LLM input and responses as inputs and correctness of the responses as targets. We also employ a random baseline (**BL**) that predicts uniform random confidence.

Datasets. We perform our experiments on four open-ended question-answering benchmarks covering diverse topics. It includes 2,000 randomly sampled training examples each from Natural Questions (NQ; Kwiatkowski et al., 2019), SciQ (Welbl et al., 2017), TriviaQA (Joshi et al., 2017), and PopQA (Mallen et al., 2023). All methods are evaluated on randomly sampled 1,000 test samples from each corresponding dataset.

LLMs. We evaluate nine large language models spanning five major families. From **GPT**, we include GPT-4o (`gpt-4o-2024-11-20`; OpenAI, 2024). From **Mistral**, we use Mistral-123B (`Large-Instruct-2411`; Mistral AI, 2024). From **Llama**, we test `Llama-3.3-70B-Instruct` (Meta AI, 2024). From **Qwen**, we cover three model sizes: Qwen2.5-72B, Qwen2.5-32B, and Qwen2.5-14B (Yang et al., 2024). Finally, from **OLMo**, we assess a broad range of models (Team OLMo et al., 2024), including `OLMo-2-0325-32B-Instruct` (OLMo-32B), `OLMo-2-1124-13B` (OLMo-13B), and `OLMo-2-1124-7B` (OLMo-7B).

Experimental Design. We first use an *answer-elicit prompt* that instructs the LLM to produce an answer to the input question. At this stage, **Prob.**, **PS**, **P(True)** and ***Calib1*** can already estimate confidence scores based on the generated answer. In contrast, **VC** requires an additional *confidence-elicit prompt* to guide the LLM in verbalizing its confidence regarding the produced answer. To isolate the effect of prompt perturbations on confidence estimation, we only perturb the prompts directly involved in it, namely the *confidence-elicit prompt* for **VC** and the *answer-elicit prompt* otherwise. For measuring prompt robustness, we use greedy decoding to ensure that any variability is solely driven by prompt perturbations.

Answer-elicit Prompt Perturbation. To avoid introducing ambiguity, we keep the question fixed and perturb only the instructions that elicit the LLM's answer. Variants include forms such as "answer the question" and "provide an answer to the question." Each prompt also enforces conciseness, with phrases like "give ONLY the answer," "no other words or explanation," "without explanation," "as short as possible," and "with minimal words." In total, we designed 10 prompts incorporating these variants (see Table 4 in the appendix).

Confidence-elicitation Prompt Perturbations. To evaluate robustness, we design seven variants of the confidence-elicitation prompt (see Appendix, Table 5), grouped into three types: (i) Scale variants, which elicit confidence using numerical ranges of 0–1, 0–100%, and 0–10; (ii) Lexical variants, which vary the key term used to request a score between 0 and 1 ("probability", "certainty" and "confidence"); and (iii) Linguistic expressions, following Tian et al. (2023), which elicit confidence using predefined verbal scales from "Almost

No Chance" to "Almost certain", and a corresponding multiple-choice format (e.g., "a: Almost No Chance"). Those selected expressions are converted to numerical scores according to rules in Section A.3. All confidence outputs are normalized to a 0–1 scale for comparison.

Evaluation. We use GPT-4o as a judge model (see prompt in Section A.2) to determine the semantic equivalence between LLM responses and target answers for correctness evaluation, as well as the semantic equivalence among sampled responses when computing stability and sensitivity metrics.

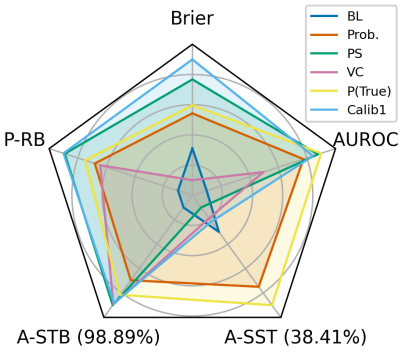


Figure 3: Averaged performance (↑) among all models. The percentage indicates the amount of data that fulfills the requirement for evaluation of the metric (more than one answer in $\hat{\mathcal{Y}}^{\max}$ for A-STB, and at least two semantically different answer groups for A-SST).

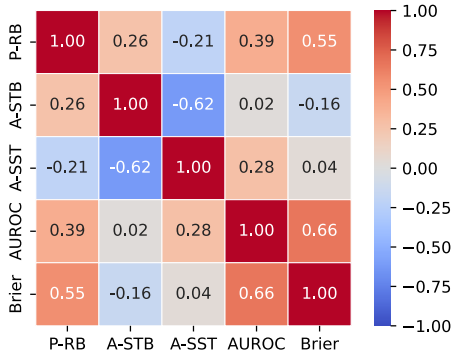


Figure 4: Pearson correlation of metrics.

6 Results and Analysis

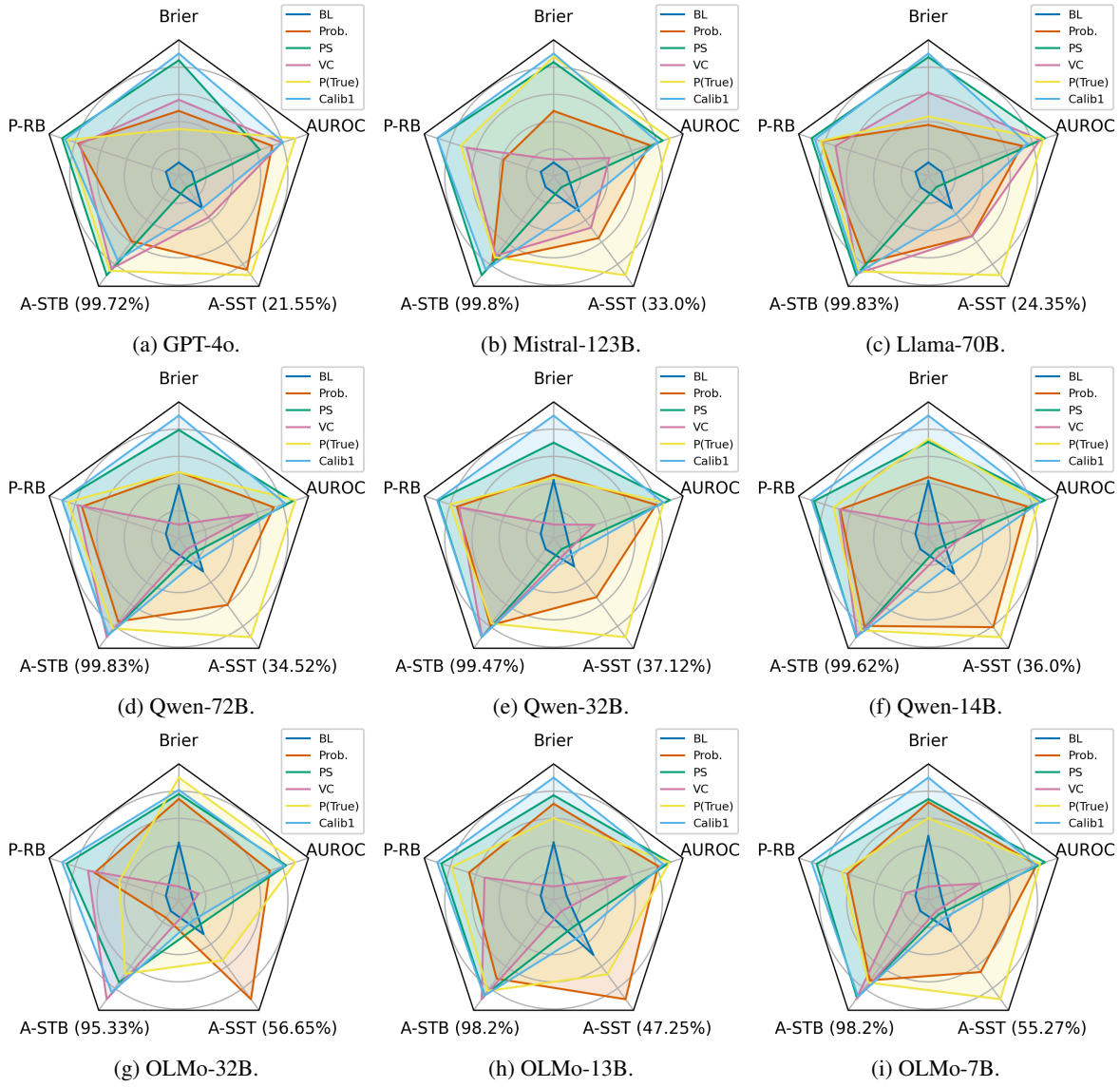
We present the average results across all evaluated models in Figure 3 and provide a detailed breakdown in Figure 5 (raw numbers are shown in Table 6). Our analysis covers five complementary perspectives: robustness (P-RB), stability (A-STB), sensitivity (A-SST), discrimination (AUROC) and calibration (Brier score). Together, these reveal nuanced trade-offs across methods and models.

Finding 1: CE methods with near-top Brier score and AUROC can still underperform on the proposed metrics. Figure 3 shows that P(True), despite achieving the highest AUROC, is sensitive to prompt perturbations and therefore performs poorly on the P-RB metric. As illustrated in Table 2, the gap between maximin and minimum confidences of P(True) is around or more than 0.5 for the test examples. PS, which ranks among the top two methods on both Brier score and AUROC, underperforms the random baseline in sensitivity. The case study in Table 3 shows that PS assigns very similar confidence scores to all answers, and is sensitive to surface-level changes such as punctuation; for example, adding a comma reduces the confidence by 0.06 in the GPT-4o case. Similarly, Calib-1, which achieves the best Brier score, underperforms P(True), Prob. and baseline in sensitivity. Overall, these results demonstrate that the proposed metrics uncover weaknesses in existing CE methods that are not revealed by standard calibration and ranking metrics.

Finding 2: VC, Prob., and P(True) are more affected by prompt perturbations. As shown in Figure 3, VC methods, which directly rely on prompting, are the most sensitive to prompt perturbations. Prob. and P(True), while indirectly influenced by prompts through their dependence on output logits, are more robust than VC but less robust than post-hoc methods (PS) and auxiliary models (Calib1).

Finding 3: Calib1, VC and PS are the most stable for equivalent answers, but only P(True) and Prob. effectively distinguish different answers. As shown in Table 3, many methods, such as VC, achieve high stability by assigning similar confidence scores to equivalent answers. But only P(True) and Prob. outperform baseline in sensitivity. High stability or sensitivity alone does not imply good performance. For example, Prob. is sensitive to surface-level changes, such as punctuation, which can lead to undesirable variability. These results suggest that an effective CE method should achieve both high stability and strong sensitivity.

Finding 4: Brier of CE methods is moderately correlated with robustness, and stability is negatively correlated with sensitivity. As shown in Figure 4, CE methods with better Brier scores tend to achieve higher P-RB scores. In contrast, methods that perform well on stability may exhibit



(a) GPT-4o. (b) Mistral-123B. (c) Llama-70B.

(d) Qwen-72B. (e) Qwen-32B. (f) Qwen-14B.

(g) OLMo-32B. (h) OLMo-13B. (i) OLMo-7B.

Figure 5: Performance of CE methods evaluated on different models.

lower sensitivity. We further observe only a weak correlation between AUROC and sensitivity. Although both metrics assess discriminative ability, they capture different aspects: AUROC measures separation between correct and incorrect responses, whereas sensitivity measures responsiveness to answer semantic changes. We take the totality of correlation coefficients in Figure 4 as evidence that our metrics outline new, meaningful axes along which we can evaluate CE methods for LLMs.

Finding 5: The robustness and sensitivity of VC vary substantially across models. Figure 5 shows that larger models, such as GPT-4o, Mistral, produce more reliable VC compared to smaller ones like OLMo models. OLMo models consistently underperform those from the Qwen family in terms of robustness and sensitivity. Other CE methods are less affected by model differences; in particular, PS and Calib-1 show relatively stable performance across models, reflecting their weaker dependence on underlying model quality.

7 Conclusion

We propose a comprehensive evaluation framework for confidence estimation in LLMs that goes beyond calibration and ranking metrics. Our three complementary metrics capture important reliability properties that are overlooked by existing CE evaluations. Experiments show that strong Brier or AUROC performance does not guarantee robustness to prompt perturbations or effective discrimination between answers. Our results suggest that method choice should depend on application priorities: post-hoc and auxiliary methods (PS, Calib1)

	What was the name of the frog in the children's TV series Hector's House? GPT-4o: Kiki. Corr.: 1				Who won season 16 on dancing with the stars? Mistral: Nicole Scherzinger. Corr.: 0			
CE	**max**	**min**	**mean**	**P-RB**	**max**	**min**	**mean**	**P-RB**
Prob.	1.00	0.80	0.96	0.94	0.13	0.01	0.07	0.94
PS	0.68	0.64	0.67	0.99	0.69	0.62	0.65	0.97
VC	0.95	0.05	0.70	0.72	1.00	0.80	0.96	0.93
P(True)	1.00	0.38	0.91	0.80	0.50	0.01	0.07	0.85
Calib1	0.32	0.28	0.30	0.99	0.44	0.43	0.44	1.00

Table 2: Case study (Left: GPT-4o, right: Mistral) of prompt robustness. We present the confidence statistics of CE methods when using perturbed prompts for the test sample.

	In humans, fertilization occurs soon after the oocyte leaves this? y^*: Ovary						What genre is Amen? y^*: Situation comedy						
	Ovary	*The ovary.*	*Ovary.*	Fallopian tube.	**A-STB**	**A-SST**	*R&B*	Gospel	Drama	Rap Rock	Reggae	**A-STB**	**A-SST**
Prob.	0.80	0.44	0.60	0.75	0.87	0.02	0.67	0.56	0.43	0.72	0.57	1.00	0.24
PS	0.70	0.62	0.66	0.69	0.97	0.01	0.63	0.61	0.57	0.64	0.61	1.00	0.06
VC	0.90	0.90	0.90	0.95	1.00	0.05	0.95	0.95	0.95	0.85	0.95	1.00	0.00
P(True)	1.00	1.00	1.00	0.00	1.00	1.00	0.03	0.10	0.75	0.11	0.05	1.00	0.72
Calib1	0.93	0.80	0.86	0.88	0.95	0.01	0.10	0.53	0.30	0.13	0.42	1.00	0.20

Table 3: Case study (Left: GPT-4o, right: Mistral) of answer stability and sensitivity. The confidence estimated for answers in $\hat{\mathcal{Y}}^{\max}$ (*italic*), $\hat{\mathcal{Y}}^{\min}$ (underlined) and more are shown. $\hat{\mathcal{Y}}^{\max}$ can contain multiple same answers.

are preferable when robustness and stability are critical, while logit-based and self-evaluation methods (Prob., P(True)) provide higher sensitivity at the cost of robustness. Overall, our framework highlights key trade-offs and offers practical guidance for designing and using CE methods in LLM deployments.

Limitations

While our study provides a comprehensive evaluation of LLM confidence estimation across robustness, stability, sensitivity, calibration, and discrimination, several limitations remain.

First, our analysis is limited to English and fact-based question answering tasks; extending to multilingual or open-ended domains (e.g., reasoning, dialogue, or creative generation) may reveal different findings. Second, although we designed a diverse set of prompting strategies, they do not exhaustively capture all linguistic and contextual variations that may influence confidence expression. In particular, for the answer-elicit prompts, it remains challenging to construct more aggressive perturbations that preserve identical semantic meaning and avoid ambiguity across all questions and datasets. Third, our evaluation relies on GPT-4o as a judge model to assess answer correctness and semantic equivalence. While this enables scalable annotation, it may introduce model-specific biases (Bavaresco et al., 2025). Fourth, we do not include consistency-based confidence estimation methods. In this work, we focus on single-pass CE methods since our metrics themselves rely on multiple samples to be computed, and is therefore incompatible with self-consistency. Finally, our evaluation is static; future work could explore dynamic or interactive settings where models can revise their confidence based on feedback or uncertainty cues.

Acknowledgments

This research has been funded by the Vienna Science and Technology Fund (WWTF)[10.47379/VRG19008] "Knowledge infused Deep Learning for Natural Language Processing", and is supported by the Dutch National Science Foundation (NWO Vici VI.C.212.053).

References

Michael Alfertshofer, Cosima C Hoch, Paul F Funk, Katharina Hollmann, Barbara Wollenberg, Samuel Knoedler, and Leonard Knoedler. 2024. Sailing the seven seas: a multinational comparison of chatgpt's performance on medical licensing examinations. *Annals of Biomedical Engineering*, 52(6):1542–1545.

Tariq Alqahtani, Hisham A Badreldin, Mohammed Alrashed, Abdulrahman I Alshaya, Sahar S Alghamdi,

Khalid Bin Saleh, Shuroug A Alowais, Omar A Alshaya, Ishrat Rahman, Majed S Al Yami, et al. 2023. The emergent role of artificial intelligence, natural learning processing, and large language models in higher education and research. *Research in social and administrative pharmacy*, 19(8):1236–1242.

Joris Baan, Nico Daheim, Evgenia Ilia, Dennis Ulmer, Haau-Sing Li, Raquel Fernández, Barbara Plank, Rico Sennrich, Chrysoula Zerva, and Wilker Aziz. 2023. Uncertainty in natural language generation: From theory to applications. *arXiv preprint arXiv:2307.15703*.

Anna Bavaresco, Raffaella Bernardi, Leonardo Bertolazzi, Desmond Elliott, Raquel Fernández, Albert Gatt, Esam Ghaleb, Mario Giulianelli, Michael Hanna, Alexander Koller, et al. 2025. Llms instead of human judges? a large scale empirical study across 20 nlp evaluation tasks. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 238–255.

Jarosław Błasiok and Preetum Nakkiran. 2024. Smooth ECE: Principled reliability diagrams via kernel smoothing. In *The Twelfth International Conference on Learning Representations*.

Andrew P. Bradley. 1997. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30(7):1145–1159.

Glenn W. Brier. 1950. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1 – 3.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. 2023. Discovering latent knowledge in language models without supervision. In *The Eleventh International Conference on Learning Representations*.

Bertrand Charpentier, Daniel Zügner, and Stephan Günnemann. 2020. Posterior network: Uncertainty estimation without ood samples via density-based pseudo-counts. *Advances in neural information processing systems*, 33:1356–1367.

Jiuhai Chen and Jonas Mueller. 2024. Quantifying uncertainty in answers from any language model and enhancing their trustworthiness. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5186–5200, Bangkok, Thailand. Association for Computational Linguistics.

Meiqi Chen, Yixin Cao, Yan Zhang, and Chaochao Lu. 2024. Quantifying and mitigating unimodal biases in multimodal large language models: A causal perspective. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 16449–16469, Miami, Florida, USA. Association for Computational Linguistics.

Wade Fagen-Ulmschneider. 2023. Perception of probability words. Manuscript, University of Illinois Urbana-Champaign.

Zijian Feng, Hanzhang Zhou, Zixiao Zhu, Junlang Qian, and Kezhi Mao. 2024. Unveiling and manipulating prompt influence in large language models. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

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. Association for Computational Linguistics.

Tobias Groot and Matias Valdenegro Toro. 2024. Overconfidence is key: Verbalized uncertainty evaluation in large language and vision-language models. In *Proceedings of the 4th Workshop on Trustworthy Natural Language Processing (TrustNLP 2024)*, pages 145–171, Mexico City, Mexico. Association for Computational Linguistics.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 1321–1330. JMLR.org.

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*, 51(2):413–429.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611.

Chethan Kadavath, Amanda Askell, Jackson Kernion, Tom Henighan, Ben Mann, Gretchen Krueger, Sarah Kreps, Aaron McKane, Gaurav Mistry, Joe Kim, et al. 2023. Prompting GPT-3 to be reliable. In *Proceedings of the 11th International Conference on Learning Representations (ICLR)*.

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, Deep Ganguli, Danny Hernandez, Josh Jacobson, Jackson Kernion, Shauna Kravec, Liane Lovitt, Kamal Ndousse, Catherine Olsson, Sam Ringer, Dario

Amodei, Tom Brown, Jack Clark, Nicholas Joseph, Ben Mann, Sam McCandlish, Chris Olah, and Jared Kaplan. 2022. Language models (mostly) know what they know. *Preprint*, arXiv:2207.05221.

John Kirchenbauer, Jacob Oaks, and Eric Heim. 2022. What Is Your Metric Telling You? Evaluating Classifier Calibration under Context-specific Definitions of Reliability. *ArXiv preprint*, abs/2205.11454.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *Proceedings of the 11th International Conference on Learning Representations (ICLR)*.

Ananya Kumar, Percy Liang, and Tengyu Ma. 2019. Verified Uncertainty Calibration. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 3787–3798.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

Florian Le Bronnec, Alexandre Verine, Benjamin Negrevergne, Yann Chevaleyre, and Alexandre Allauzen. 2024. Exploring precision and recall to assess the quality and diversity of LLMs. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11418–11441, Bangkok, Thailand. Association for Computational Linguistics.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. Teaching models to express their uncertainty in words. *Transactions on Machine Learning Research*.

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

Jeremiah Zhe Liu, Shreyas Padhy, Jie Ren, Zi Lin, Yeming Wen, Ghassen Jerfel, Zachary Nado, Jasper Snoek, Dustin Tran, and Balaji Lakshminarayanan. 2023. A simple approach to improve single-model deep uncertainty via distance-awareness. *Journal of Machine Learning Research*, 24(42):1–63.

Xin Liu, Muhammad Khalifa, and Lu Wang. 2024. Litcab: Lightweight language model calibration over short- and long-form responses. In *Proceedings of the Twelfth International Conference on Learning Representations*. ICLR.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9802–9822, Toronto, Canada. Association for Computational Linguistics.

Potsawee Manakul, Adian Liusie, and Mark J. F. Gales. 2023. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. *Preprint*, arXiv:2303.08896.

Meta AI. 2024. Introducing meta llama 3: The most capable openly available llm to date. Accessed: 2025-05-01.

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.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Rethinking the role of demonstrations: What makes in-context learning work? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11048–11064.

Sewon Min, Julian Michael, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2020. AmbigQA: Answering ambiguous open-domain questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5783–5797, Online. Association for Computational Linguistics.

Mistral AI. 2024. Mistral large instruct 2411. Accessed: 2025-05-01.

Lwin Moe, Uyen Trang Nguyen, and Boi Trinh Luu. 2025. Mitigating class imbalance in fact-checking datasets through llm-based synthetic data generation. In *Proceedings of the 4th ACM International Workshop on Multimedia AI against Disinformation*, MAD' 25, page 73–80, New York, NY, USA. Association for Computing Machinery.

Jishnu Mukhoti, Viveka Kulharia, Amartya Sanyal, Stuart Golodetz, Philip Torr, and Puneet Dokania. 2020. Calibrating deep neural networks using focal loss. *Advances in Neural Information Processing Systems*, 33:15288–15299.

Shiyu Ni, Keping Bi, Lulu Yu, and Jiafeng Guo. 2025. Are large language models more honest in their probabilistic or verbalized confidence? In *Information Retrieval*, pages 124–135, Singapore. Springer Nature Singapore.

Jeremy Nixon, Michael W. Dusenberry, Linchuan Zhang, Ghassen Jerfel, and Dustin Tran. 2019. Measuring Calibration in Deep Learning. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 38–41.

OpenAI. 2024. Hello gpt-4o. Accessed: 2025-05-01.

Kazuki Osawa, Siddharth Swaroop, Mohammad Emtiyaz Khan, Anirudh Jain, Runa Eschenhagen, Richard E Turner, and Rio Yokota. 2019. Practical deep learning with bayesian principles. *Advances in neural information processing systems*, 32.

Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, David Sculley, Sebastian Nowozin, Joshua Dillon, Balaji Lakshminarayanan, and Jasper Snoek. 2019. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. *Advances in neural information processing systems*, 32.

John Platt. 1999. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in Large Margin Classifiers*, 10(3):61–74.

Sergey Pletenev, Maria Marina, Nikolay Ivanov, Daria Galimzianova, Nikita Krayko, Mikhail Salnikov, Vasily Konovalov, Alexander Panchenko, and Viktor Moskvoretskii. 2025. Will it still be true tomorrow? multilingual evergreen question classification to improve trustworthy QA. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 8603–8620, Suzhou, China. Association for Computational Linguistics.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. 2024. Quantifying language models' sensitivity to spurious features in prompt design or: How I learned to start worrying about prompt formatting. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Ola Shorinwa, Zhiting Mei, Justin Lidard, Allen Z. Ren, and Anirudha Majumdar. 2025. A survey on uncertainty quantification of large language models: Taxonomy, open research challenges, and future directions. *ACM Comput. Surv.*, 58(3).

Elias Stengel-Eskin and Benjamin Van Durme. 2023. Calibrated interpretation: Confidence estimation in semantic parsing. *Preprint*, arXiv:2211.07443.

Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, Nathan Lambert, Dustin Schwenk, Oyvind Tafjord, Taira Anderson, David Atkinson, Faeze Brahman, Christopher Clark, Pradeep Dasigi, Nouha Dziri, Michal Guerquin, Hamish Ivison, Pang Wei Koh, Jiacheng Liu, Saumya Malik, William Merrill, Lester James V. Miranda, Jacob Morrison, Tyler Murray, Crystal Nam, Valentina Pyatkin, Aman Rangapur, Michael Schmitz, Sam Skjonsberg, David Wadden, Christopher Wilhelm, Michael Wilson, Luke Zettlemoyer, Ali Farhadi, Noah A. Smith, and Hannaneh Hajishirzi. 2024. 2 olmo 2 furious. *arXiv preprint arXiv:2501.00656*.

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. Association for Computational Linguistics.

Dennis Ulmer, Jes Frellsen, and Christian Hardmeier. 2022. Exploring predictive uncertainty and calibration in NLP: A study on the impact of method & data scarcity. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 2707–2735, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Dennis Ulmer, Martin Gubri, Hwaran Lee, Sangdoo Yun, and Seong Oh. 2024. Calibrating large language models using their generations only. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15440–15459, Bangkok, Thailand. Association for Computational Linguistics.

Dennis Ulmer, Alexandra Lorson, Ivan Titov, and Christian Hardmeier. 2025. Anthropomimetic uncertainty: What verbalized uncertainty in language models is missing. *arXiv preprint arXiv:2507.10587*.

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, Gleb Kuzmin, Artem Shelmanov, Akim Tsvigun, Evgenii Tsymbalov, Kirill Fedyanin, Maxim Panov, Alexander Panchenko, Gleb Gusev, Mikhail Burtsev, et al. 2022. Uncertainty estimation of transformer predictions for misclassification detection. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8237–8252.

vLLM Contributors. 2023. vllm: A high-throughput and memory-efficient llm serving library. `https://github.com/vllm-project/vllm`. Accessed: 2025-01.

Lean Wang, Lei Li, Damai Dai, Deli Chen, Hao Zhou, Fandong Meng, Jie Zhou, and Xu Sun. 2023. Label words are anchors: An information flow perspective for understanding in-context learning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9840–9855.

Johannes Welbl, Nelson F. Liu, and Matt Gardner. 2017. Crowdsourcing multiple choice science questions.

In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 94–106, Copenhagen, Denmark. Association for Computational Linguistics.

Yuxi Xia, Pedro Henrique Luz De Araujo, Klim Zaporojets, and Benjamin Roth. 2025a. Influences on LLM calibration: A study of response agreement, loss functions, and prompt styles. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3740–3761, Vienna, Austria. Association for Computational Linguistics.

Yuxi Xia, Klim Zaporojets, and Benjamin Roth. 2025b. Learn to pick the winner: Black-box ensembling for textual and visual question answering. In *Proceedings of the 21st Conference on Natural Language Processing (KONVENS 2025): Long and Short Papers*, pages 12–26, Hannover, Germany. HsH Applied Academics.

Dongfu Xiao, Chen Gao, Zhengquan Luo, Chi Liu, and Sheng Shen. 2025. Can llms assist computer education? an empirical case study of deepseek. *Preprint*, arXiv:2504.00421.

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 *Proceedings of the 12th International Conference on Learning Representations (ICLR)*.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

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. Association for Computational Linguistics.

Kaitlyn Zhou, Jena Hwang, Xiang Ren, and Maarten Sap. 2024. Relying on the unreliable: The impact of language models' reluctance to express uncertainty. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3623–3643, Bangkok, Thailand. Association for Computational Linguistics.

Chiwei Zhu, Benfeng Xu, Quan Wang, Yongdong Zhang, and Zhendong Mao. 2023. On the calibration of large language models and alignment. In *Conference on Empirical Methods in Natural Language Processing*.

Jingming Zhuo, Songyang Zhang, Xinyu Fang, Haodong Duan, Dahua Lin, and Kai Chen. 2024. ProSA: Assessing and understanding the prompt sensitivity of LLMs. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 1950–1976, Miami, Florida, USA. Association for Computational Linguistics.

A Appendix

A.1 Derivations

In this section, we directly relate the definitions we lay out in Section 4 to our proposed metrics. To this end, we formally prove lower bounds that our metrics have to exceed in our for the definitions to hold. We start with Definition 4.1 and the prompt robustness metric in Equation (3).

Theorem A.1. *Given a set of M semantically equivalent prompts $\mathcal{T} = \{\mathbf{t}_1, \ldots, \mathbf{t}_M\}$ and a* robust *confidence estimator f according to Definition 4.1 s.t.*

$$\forall \mathbf{t}, \mathbf{t}' \in \mathcal{T}, \quad |f(\mathbf{y}, \mathbf{x}, \mathbf{t}) - f(\mathbf{y}, \mathbf{x}, \mathbf{t}')| \leq \varepsilon, \tag{8}$$

then it must hold for the robustness metric $\hat{\sigma}_r$ defined in Equation (3) that

$$\hat{\sigma}_r > \frac{2-\varepsilon}{2}. \tag{9}$$

Proof. In the following, we will use the shorthand c for $f(\mathbf{y}, \mathbf{x}, \mathbf{t})$. Because of Equation (18) we know that any two c, c' lie more than ε apart. Since standard deviation and variance are metrics of dispersion, they are invariant to translation, meaning that they will stay constant if the data is shifted by a constant amount. Therefore, let us assume that $c \in [0, \varepsilon]$. Since we would like to find the maximum dispersion possible under Equation (18), let us further assume that k points are equal to 0, and $M - k$ equal ε. Therefore, their mean becomes $\mu = \frac{M-k}{M}\varepsilon$. Correspondingly, we can evaluate their variance as

$$\sigma^2 = \frac{1}{M}[k(0-\mu)^2 + (M-k)(\varepsilon-\mu)^2] \tag{10}$$

$$= \frac{1}{M}\left[k\frac{(M-k)^2\varepsilon^2}{M^2} + (M-k)\frac{k^2\varepsilon^2}{M^2}\right] \tag{11}$$

$$= \frac{\varepsilon^2}{M^3}[k(M-k)^2 + (M-k)k^2] \tag{12}$$

$$= \frac{\varepsilon^2}{M^3}k(M-k)M. \tag{13}$$

Next, we would like to identify the k that maximizes $k(M-k)M$, we can easily be identified as $k=\frac{M}{2}$. Reinserting it into the previous expression yields

$$= \frac{\varepsilon^2}{M^3}\frac{M}{2}\Big(M-\frac{M}{2}\Big)M \tag{14}$$

$$= \frac{\varepsilon^2}{M^3}\frac{M^3}{4} = \frac{\varepsilon^2}{4} \leftrightarrow \sigma = \frac{\varepsilon}{2}. \tag{15}$$

By plugging this result into the definition of robustness in Definition 4.1, we can show that since the biggest standard deviation for which Equation (18) still holds is $\frac{\varepsilon}{2}$ leads robustness to amount to

$$\hat{\sigma}_r > 1 - \frac{\varepsilon}{2} = \frac{2-\varepsilon}{2}. \tag{16}$$

□

Since Definition 4.2 and the stability metric in Equation (4) follow the same structure, the matching theorem follows naturally.

Theorem A.2. *Given a set of M responses given a prompt $\mathbf{t}$, $\hat{\mathcal{Y}} \subseteq \mathcal{Y}$ a set of semantically equivalent responses to $\mathbf{y}_m$ and a* stable *confidence estimator f according to Definition 4.2 s.t.*

$$\forall \mathbf{y}' \in \hat{\mathcal{Y}}: \quad |f(\mathbf{y}^{(m)}, \mathbf{x}, \mathbf{t}) - f(\mathbf{y}', \mathbf{x}, \mathbf{t})| \leq \varepsilon.$$

Then it must hold for the stability metric $\hat{\sigma}_s$ defined in Equation (3) that

$$\hat{\sigma}_s > \frac{2-\varepsilon}{2}. \tag{17}$$

Proof. The proof follows the same argument as Theorem A.1 without loss of generality. □

Lastly, we would like to prove a similar bound for the sensitivity metric in Equation (5), which we do below.

Theorem A.3. *Let $\mathcal{Y}$ denote a set of M generated responses for a given input $\mathbf{x}$ and a (single) prompt $\mathbf{t}$ s.t. $\mathcal{Y} = \{\mathbf{y}^{(1)}, \ldots, \mathbf{y}^{(M)}\}$. Also let $\mathcal{E}(\mathcal{Y}) = \{\hat{\mathcal{Y}}_l\}_{l=1}^{L}$ be a finite partition of $\mathcal{Y}$ s.t. $\bigcup_{l=1}^{L}\hat{\mathcal{Y}}_l = \mathcal{Y}$ and $\forall \hat{\mathcal{Y}}_l, \hat{\mathcal{Y}}_{l'} \in \mathcal{E}, l \neq l' : \hat{\mathcal{Y}}_l \cap \hat{\mathcal{Y}}_{l'} = \varnothing$ and let f be a* sensitive *confidence estimator according to Definition 4.3 s.t.*

$$\forall \hat{\mathcal{Y}}_l, \hat{\mathcal{Y}}_{l'} \in \mathcal{E}(\mathcal{Y}), l \neq l', \mathbf{y} \in \hat{\mathcal{Y}}_l, \mathbf{y}' \in \hat{\mathcal{Y}}_{l'}:$$
$$|f(\mathbf{y}, \mathbf{x}, \mathbf{t}) - f(\mathbf{y}', \mathbf{x}, \mathbf{t})| > \varepsilon, \tag{18}$$

and that f is stable according to Definition 4.2 s.t.

$$\forall \mathbf{y}' \in \hat{\mathcal{Y}}: \quad |f(\mathbf{y}^{(m)}, \mathbf{x}, \mathbf{t}) - f(\mathbf{y}', \mathbf{x}, \mathbf{t})| \leq \varepsilon.$$

Then it must hold for the sensitivity metric λ defined in Equation (5) that

$$\lambda > \frac{\varepsilon}{N}\sum_{i=1}^{N} |\hat{\mathcal{Y}}_i^{max}|^{-1}. \tag{19}$$

Proof. We start by restating the definition of sensitivity from Equation (5):

$$\lambda = \frac{1}{N}\sum_{i=1}^{N} |\Delta(\hat{\mathcal{Y}}_i^{\max}, \hat{\mathcal{Y}}_i^{\min}) - \Delta(\hat{\mathcal{Y}}_i^{\max}, \hat{\mathcal{Y}}_i^{\max})| \tag{20}$$

where $\hat{\mathcal{Y}}^{\max}, \hat{\mathcal{Y}}^{\min} \in \mathcal{E}(\mathcal{Y})$ are the biggest and smallest set within the partition respectively, with i corresponding to a specific input $\mathbf{x}_i$ in the dataset. We treat each of the terms in the difference separately next. Firstly, we have

$$\Delta(\hat{\mathcal{Y}}_i^{\max}, \hat{\mathcal{Y}}_i^{\min}) \tag{21}$$

$$= \frac{1}{|\hat{\mathcal{Y}}_i^{\max}||\hat{\mathcal{Y}}_i^{\min}|}\sum_{\mathbf{y}\in\hat{\mathcal{Y}}_i^{\max}}\sum_{\mathbf{y}'\in\hat{\mathcal{Y}}_i^{\min}} |c(\mathbf{y}) - c(\mathbf{y}')| \tag{22}$$

$$> \frac{1}{|\hat{\mathcal{Y}}_i^{\max}||\hat{\mathcal{Y}}_i^{\min}|}\sum_{\mathbf{y}\in\hat{\mathcal{Y}}_i^{\max}}\sum_{\mathbf{y}'\in\hat{\mathcal{Y}}_i^{\min}} \varepsilon = \varepsilon, \tag{23}$$

where the last step follows from Definition 4.3. We can also simplify $\Delta(\hat{\mathcal{Y}}_i^{\max}, \hat{\mathcal{Y}}_i^{\max})$ by noticing that from the $|\hat{\mathcal{Y}}_i^{\max}|^2$ differences considered, $|\hat{\mathcal{Y}}_i^{\max}|$ of them will amount to 0 when an element is compared to itself. Using the definition of stability in Definition 4.2, we also know that the remaining $|\hat{\mathcal{Y}}_i^{\max}|^2 - |\hat{\mathcal{Y}}_i^{\max}|$ comparisons amount to at most ε each. Therefore,

$$\Delta(\hat{\mathcal{Y}}_i^{\max}, \hat{\mathcal{Y}}_i^{\max}) \leq \frac{1}{|\hat{\mathcal{Y}}_i^{\max}|^2}\big(|\hat{\mathcal{Y}}_i^{\max}|^2 - |\hat{\mathcal{Y}}_i^{\max}|\big)\varepsilon \tag{24}$$

$$= \varepsilon - \frac{1}{|\hat{\mathcal{Y}}_i^{\max}|}\varepsilon = \frac{|\hat{\mathcal{Y}}_i^{\max}| - 1}{|\hat{\mathcal{Y}}_i^{\max}|}\varepsilon. \tag{25}$$

Putting the two arts together, we obtain

$$\lambda > \frac{1}{N}\sum_{i=1}^{N}\left|\varepsilon - \frac{|\hat{\mathcal{Y}}_i^{\max}| - 1}{|\hat{\mathcal{Y}}_i^{\max}|}\varepsilon\right| \tag{26}$$

$$= \frac{1}{N}\sum_{i=1}^{N}\left|\frac{1}{|\hat{\mathcal{Y}}_i^{\max}|}\varepsilon\right| \tag{27}$$

$$= \frac{\varepsilon}{N}\sum_{i=1}^{N} |\hat{\mathcal{Y}}_i^{\max}|^{-1}. \tag{28}$$

□

A.2 Prompts

Semantic Equivalence Judgment Prompt for Correctness

```
You are evaluating whether an answer is semantically equivalent to the reference answer. Judge meaning rather than exact wording. Two answers should be considered equivalent if they express the same meaning, even when using different phrasing, synonyms, or slightly different names referring to the same entity.

Also, treat answers as equivalent even if they include expressions of confidence or uncertainty (e.g., 'I'm pretty sure it's X' vs. 'X'). Focus on the core meaning of the answers and ignore differences in:
- wording or phrasing
- length
- fillers
- strengtheners or weakeners

Examples:
Answer: "Mount Everest" | Reference: "Everest" → YES;
Answer: "Albert Einstein" | Reference: "Einstein" → YES;
Answer: "Paris" | Reference: "The capital of France" → YES;
Answer: "Pacific Ocean" | Reference: "Atlantic Ocean" → NO;
Answer: "Jupiter" | Reference: "I think it's Jupiter" → YES;
Answer: "Mars" | Reference: "Maybe Venus" → NO.

Answer only with:
1 if semantically equivalent;
0 otherwise.

Question: <question>
Answer: <answer>
Reference answer: <target>
Equivalent:
```

Semantic Equivalence Judgment Prompt for Semantic Answer Groups

```
You are grouping 10 answers by their semantic equivalence. Your goal is to judge meaning, not exact wording. Two answers should belong to one group if they are semantically equivalent, even if they use different phrasing, synonyms, or slightly different names that refer to the same entity.

Also, treat answers as equivalent even if they include expressions of confidence or uncertainty (e.g., 'I'm pretty sure it's X' vs. 'X'). Focus on the core meaning of the answers and ignore differences in:
- wording or phrasing
- length
- fillers
- strengtheners or weakeners

Examples:
- Answer: 'Mount Everest' | Reference: 'Everest' → YES
- Answer: 'Albert Einstein' | Reference: 'Einstein' → YES
- Answer: 'Paris' | Reference: 'The capital of France' → YES
- Answer: 'Pacific Ocean' | Reference: 'Atlantic Ocean' → NO
- Answer: 'Jupiter' | Reference: 'I think it's Jupiter' → YES
- Answer: 'Mars' | Reference: 'Maybe Venus' → NO

Answer in a dict format only, answers in the same group fall into one dict, for example:
{1: answer1, 2: ...}
{4: answer4, 5: ...}

Question: <question>
Answer: <sampled answers >
```

A.3 Mapping of Linguistic Prompts

For the linguistic prompts, we followed Tian et al. (2023) to map the linguistic expressions to numerical scores. The specific mapping are ["Almost No Chance", "Highly Unlikely", "Chances are Slight", "Little Chance", "Unlikely", "Probably Not", "About Even", "Better than Even", "Likely", "Probably", "Very Good Chance", "Highly Likely", 'Almost Certain"] to [0.02, 0.05, 0.1, 0.1, 0.2, 0.25, 0.5, 0.6, 0.7, 0.7, 0.8, 0.9, 0.95] according to Fagen-Ulmschneider (2023).

A.4 Implementation Details

Robustness. To evaluate robustness, we estimate confidence scores for the same answer under different prompt formulations. For GPT-4o, full input-sequence logits are not available, which prevents computing the probability of a fixed answer across prompts. We therefore compute confidence based on the generated output sequences instead. Note that this might bring minor changes in the overall robustness results for Prob. method, it does not impact our overall findings.

Stability and Sensitivity. We set the LLM temperature to 0.7 to generate diverse candidate answers. GPT-4o is used as a judge model to assess semantic equivalence and group answers accordingly.

CE Methods. For Calib-1, we follow the setup of Xia et al. (2025a), using BERT-base-uncased as the backbone model for training.

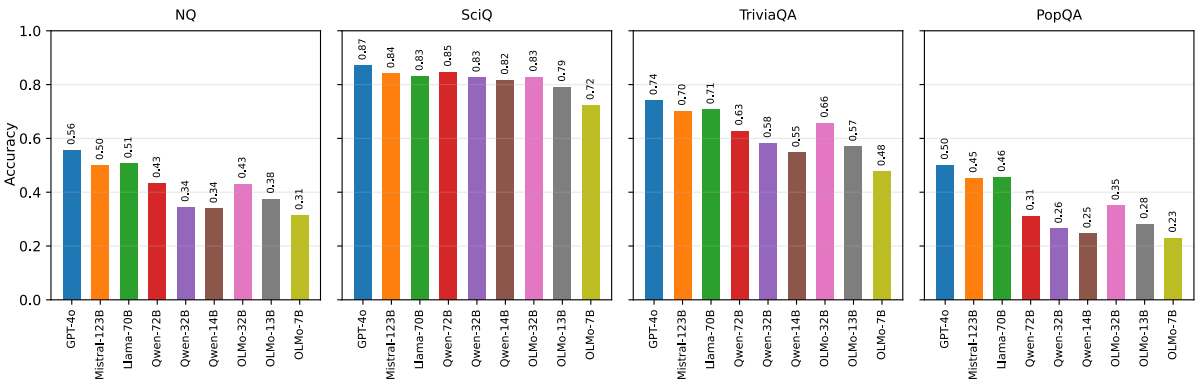


Figure 6: Accuracy of models on different datasets.

A.5 Technical Details

We use the vLLM (vLLM Contributors, 2023) library for LLM inference and serving, the temperature of all LLMs is set to 0 for greedy-encoding model outputs. All our experiments are conducted on NVIDIA HGX H100, which requires approximately 600 GPU hours to replicate.

A.6 Accuracy of Models

We present the detailed accuracy of models on each benchmark dataset in Fig. 6.

A.7 Robustness: Prompt-wise Confidence Analysis for VC

We examine the pairwise correlations of confidences between prompts for individual models (shown by Fig. 7-9). We found that larger models maintain higher stability of confidence, demonstrating their ability to respond to diverse confidence-eliciting prompts.

LLMs remain stable with the change of scales and lexical words in prompts. Fig. 7-9) shows that prompts of only scale and lexical words change tend to have more correlated verbalized confidence compared to adding linguistic expressions in the prompt. Small models (<= 32B) struggle more in maintaining stability, especially with the multi-choice linguistic expression prompts.

A.8 Robustness: Results of Answer-elicit Prompts for VC

We further conducted an experiment applying **VC** using only the answer-elicit prompts to examine how confidence values change under this setting. This design allows us to isolate the influence of answer-elicit prompts on VC and directly compare it with the other two methods under consistent perturbation conditions, even though the confidence-elicit prompt is expected to have a more direct effect on VC. As shown in Table 7, when using answer-elicit prompts (Ans-P), VC produces more stable confidence estimates (P-RB $= 0.968$) compared to when confidence-elicit prompts are used (P-RB $= 0.904$). We hypothesize that this improvement in stability arises because the generated answers remain largely consistent across perturbed answer-elicit prompts, leading to nearly identical instructions when subsequently eliciting confidence for those answers. In contrast, perturbations to the confidence-elicit prompts directly alter how the model is instructed to express its confidence, thereby introducing greater variability in the resulting confidence scores.

A.9 Sensitivity: Alternative Sets of Semantically Different Answers

Table 8 reports sensitivity results obtained by using all semantically different answers outside $\hat{\mathcal{Y}}^{\max}$ (denoted as $\hat{\mathcal{Y}}^{\notin \max}$), instead of the original setting that compares $\hat{\mathcal{Y}}^{\max}$ with $\hat{\mathcal{Y}}^{\min}$. The results remain stable under this alternative definition and do not affect the main findings of the paper.

A.10 Calibration: Results of More Evaluation Metrics

To provide a robust evaluation of calibration, we followed (Ulmer et al., 2024) and tested all the prompts and models with three more calibration metrics: ECE, smECE and AUROC. The results are shown in Table 6.

Concept	Prompt Name	Prompt
Answer	A1	Answer the question, give ONLY the answer, no other words or explanation:
	A2	Answer the question, give ONLY the answer without explanation:
	A3	Answer the question, give ONLY the answer:
	A4	Answer the question as short as possible:
	A5	Answer the question with minimal words:
Provide	P1	Provide an answer for the question, give ONLY the answer, no other words or explanation:
	P2	Provide an answer for the question without explanation:
	P3	Provide an answer for the question, give ONLY the answer:
	P4	Provide as short an answer as possible to the question:
	P5	Provide an answer with minimal words for the question:

Table 4: Detail of prompts for eliciting answers from LLMs.

Concept	Prompt Name	Prompt
Scale Variants	P(1)	Provide the probability that your answer is correct. Give ONLY the probability **between 0.0 and 1.0**, no other words or explanation.
	P(%)	Provide the probability that your answer is correct. Give ONLY the probability **between 0% and 100%**, no other words or explanation.
	P(10)	Provide the probability that your answer is correct. Give ONLY the probability **between 0 and 10**, no other words or explanation.
Lexical Variants	CF(1)	Provide the **confidence** that your answer is correct. Give ONLY the confidence between 0.0 and 1.0, no other words or explanation.
	CT(1)	Provide the **certainty** that your answer is correct. Give ONLY the confidence between 0 and 10, no other words or explanation.
Linguistic Expressions	L.	Describe how likely it is that your answer is correct as **one of the following expressions**: ['Almost No Chance', 'Highly Unlikely', 'Chances are Slight', 'Little Chance', 'Unlikely', 'Probably Not', 'About Even', 'Better than Even', 'Likely', 'Probably', 'Very Good Chance','Highly Likely', 'Almost Certain']. Give ONLY the chosen expression, no other words or explanation.
	L. MC	Describe how likely it is that your answer is correct by choosing **one of the following options**: [a: 'Almost No Chance', b: 'Highly Unlikely', c: 'Chances are Slight', d: 'Little Chance', e:'Unlikely', f: 'Probably Not', g: 'About Even', h: 'Better than Even', i: 'Likely', j: 'Probably', k; 'Very Good Chance', l: 'Highly Likely', m: 'Almost Certain']. Give ONLY the chosen option, no other words or explanation.

Table 5: Detail of prompts for eliciting confidence from LLMs.

CE	Metric	GPT-4o	Mistral-123B	Llama-70B	Qwen-72B	Qwen-32B	Qwen-14B	OLMo-32B	OLMo-13B	OLMo-7B	AVG
BL	P-RB ↑	0.728	0.728	0.728	0.728	0.728	0.728	0.728	0.728	0.728	0.728
	A-STB ↑	0.748	0.751	0.747	0.754	0.759	0.753	0.756	0.758	0.756	0.754
	A-SST ↑	0.087	0.084	0.084	0.088	0.096	0.092	0.096	0.101	0.091	0.091
	Brier ↓	0.335	0.338	0.338	0.336	0.332	0.340	0.335	0.335	0.342	0.337
	AUROC ↑	0.486	0.482	0.483	0.491	0.502	0.485	0.492	0.495	0.485	0.489
	ECE ↓	0.288	0.268	0.275	0.260	0.242	0.260	0.258	0.248	0.262	0.262
	smECE ↓	0.229	0.221	0.221	0.212	0.205	0.214	0.212	0.209	0.217	0.216
Prob.	P-RB ↑	0.949	0.821	0.967	0.937	0.940	0.918	0.908	0.909	0.899	0.916
	A-STB ↑	0.959	0.981	0.984	0.983	0.986	0.986	0.922	0.980	0.979	0.973
	A-SST ↑	0.208	0.141	0.150	0.180	0.190	0.195	0.299	0.190	0.196	0.194
	Brier ↓	0.275	0.282	0.299	0.314	0.320	0.332	0.241	0.261	0.263	0.287
	AUROC ↑	0.678	0.696	0.687	0.721	0.781	0.759	0.708	0.748	0.786	0.729
	ECE ↓	0.264	0.269	0.297	0.314	0.347	0.356	0.169	0.237	0.269	0.280
	smECE ↓	0.227	0.254	0.270	0.279	0.301	0.308	0.168	0.230	0.255	0.255
PS	P-RB ↑	0.990	0.988	0.990	0.987	0.987	0.983	0.979	0.980	0.976	0.984
	A-STB ↑	0.991	0.996	0.996	0.996	0.997	0.997	0.982	0.995	0.995	0.994
	A-SST ↑	0.046	0.032	0.034	0.041	0.044	0.044	0.071	0.045	0.047	0.045
	Brier ↓	0.216	0.228	0.230	0.245	0.251	0.254	0.230	0.245	0.255	0.239
	AUROC ↑	0.649	0.730	0.745	0.776	0.827	0.818	0.754	0.782	0.813	0.766
	ECE ↓	0.076	0.092	0.130	0.175	0.216	0.218	0.148	0.177	0.228	0.162
	smECE ↓	0.043	0.077	0.084	0.135	0.171	0.179	0.090	0.141	0.195	0.124
VC	P-RB ↑	0.945	0.915	0.930	0.949	0.934	0.920	0.925	0.870	0.751	0.904
	A-STB ↑	0.985	0.977	0.995	0.998	0.996	0.996	0.997	0.999	0.996	0.993
	A-SST ↑	0.105	0.119	0.150	0.026	0.057	0.054	0.024	0.013	0.039	0.065
	Brier ↓	0.262	0.335	0.266	0.400	0.428	0.436	0.429	0.419	0.460	0.382
	AUROC ↑	0.701	0.595	0.733	0.661	0.579	0.625	0.506	0.655	0.606	0.629
	ECE ↓	0.258	0.337	0.258	0.401	0.430	0.442	0.430	0.426	0.471	0.384
	smECE ↓	0.209	0.192	0.223	0.326	0.357	0.366	0.213	0.356	0.380	0.291
P(True)	P-RB ↑	0.976	0.929	0.967	0.972	0.957	0.935	0.845	0.953	0.912	0.939
	A-STB ↑	0.987	0.978	0.993	0.990	0.985	0.990	0.974	0.991	0.981	0.985
	A-SST ↑	0.219	0.220	0.242	0.269	0.312	0.214	0.178	0.140	0.266	0.229
	Brier ↓	0.296	0.222	0.291	0.314	0.327	0.248	0.195	0.289	0.300	0.276
	AUROC ↑	0.733	0.748	0.740	0.784	0.806	0.798	0.781	0.781	0.797	0.774
	ECE ↓	0.295	0.168	0.284	0.315	0.326	0.246	0.083	0.289	0.297	0.256
	smECE ↓	0.171	0.160	0.190	0.205	0.213	0.148	0.079	0.269	0.258	0.188
Calib1	P-RB ↑	0.982	0.988	0.982	0.987	0.990	0.990	0.990	0.990	0.989	0.987
	A-STB ↑	0.977	0.990	0.994	0.996	0.997	0.996	0.991	0.996	0.994	0.992
	A-SST ↑	0.087	0.079	0.098	0.061	0.066	0.077	0.052	0.064	0.060	0.072
	Brier ↓	0.208	0.219	0.226	0.222	0.192	0.196	0.221	0.211	0.205	0.211
	AUROC ↑	0.704	0.718	0.702	0.754	0.804	0.803	0.750	0.764	0.785	0.754
	ECE ↓	0.104	0.134	0.136	0.142	0.095	0.103	0.136	0.099	0.105	0.117
	smECE ↓	0.103	0.132	0.135	0.141	0.095	0.102	0.135	0.091	0.101	0.115

Table 6: Detailed overall results.

CE	Prompt	GPT-4o	Mistral-123B	Llama-70B	Qwen-72B	Qwen-32B	Qwen-14B	OLMo-32B	OLMo-13B	OLMo-7B	AVG
VC	Conf-P	0.945	0.915	0.930	0.949	0.934	0.920	0.925	0.870	0.751	0.904
	Ans-P	0.962	0.957	0.950	0.980	0.979	0.962	0.990	0.969	0.960	0.968

Table 7: Results of robustness of VC when using answer-elicit prompt (Ans-P) compared to the original setting in the paper that uses confidence-elicit prompt (Conf-P).

CE	$\hat{\mathcal{Y}}$	GPT-4o	Mistral-123B	Llama-70B	Qwen-72B	Qwen-32B	Qwen-14B	OLMo-32B	OLMo-13B	OLMo-7B	AVG
BL	$\hat{\mathcal{Y}}^{\text{min}}$	0.087	0.084	0.084	0.088	0.096	0.092	0.096	0.101	0.091	0.091
	$\hat{\mathcal{Y}}^{\notin\text{max}}$	0.081	0.080	0.078	0.083	0.088	0.084	0.087	0.087	0.084	0.084
Prob.	$\hat{\mathcal{Y}}^{\text{min}}$	0.208	0.141	0.150	0.180	0.190	0.195	0.299	0.190	0.196	0.194
	$\hat{\mathcal{Y}}^{\notin\text{max}}$	0.198	0.137	0.143	0.178	0.184	0.187	0.278	0.182	0.188	0.186
PS	$\hat{\mathcal{Y}}^{\text{min}}$	0.046	0.032	0.034	0.041	0.044	0.044	0.071	0.045	0.047	0.045
	$\hat{\mathcal{Y}}^{\notin\text{max}}$	0.043	0.031	0.032	0.041	0.042	0.042	0.066	0.043	0.045	0.043
VC	$\hat{\mathcal{Y}}^{\text{min}}$	0.105	0.119	0.150	0.026	0.057	0.054	0.024	0.013	0.039	0.065
	$\hat{\mathcal{Y}}^{\notin\text{max}}$	0.109	0.125	0.152	0.028	0.054	0.060	0.019	0.014	0.038	0.067
P(True)	$\hat{\mathcal{Y}}^{\text{min}}$	0.219	0.220	0.242	0.269	0.312	0.214	0.178	0.140	0.266	0.229
	$\hat{\mathcal{Y}}^{\notin\text{max}}$	0.226	0.225	0.251	0.281	0.320	0.226	0.173	0.138	0.280	0.236
Calib1	$\hat{\mathcal{Y}}^{\text{min}}$	0.087	0.079	0.098	0.061	0.066	0.077	0.052	0.064	0.060	0.072
	$\hat{\mathcal{Y}}^{\notin\text{max}}$	0.091	0.082	0.100	0.064	0.071	0.080	0.052	0.065	0.060	0.074

Table 8: Results of A-SST ↑ when using all semantically different answers that are not in $\hat{\mathcal{Y}}^{\text{max}}$ ($\hat{\mathcal{Y}}^{\notin\text{max}}$) compared to the original setting in the paper ($\hat{\mathcal{Y}}^{\text{min}}$).

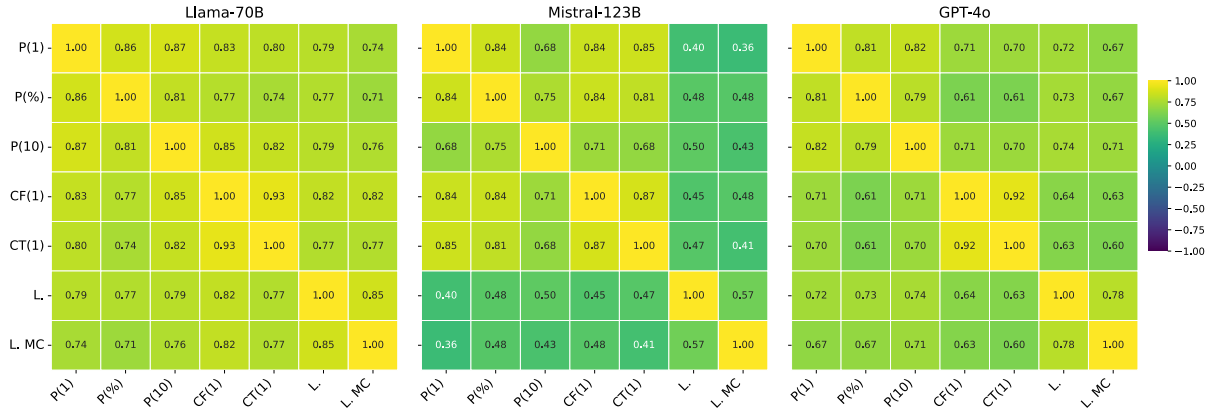


Figure 7: *Robustness*: Pearson correlations of VC when using different prompts on large models from different model families.

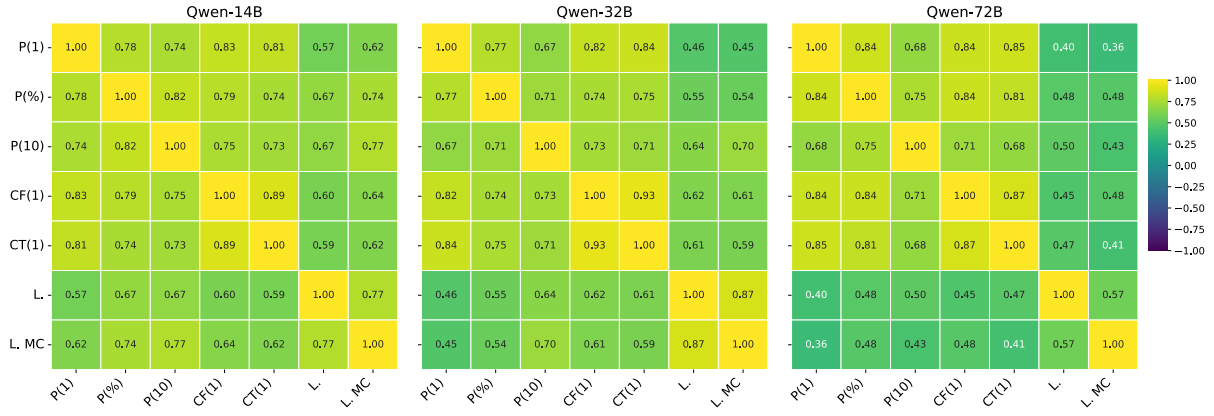


Figure 8: *Robustness*: Pearson correlations of VC when using different prompts on Qwen models with different sizes.

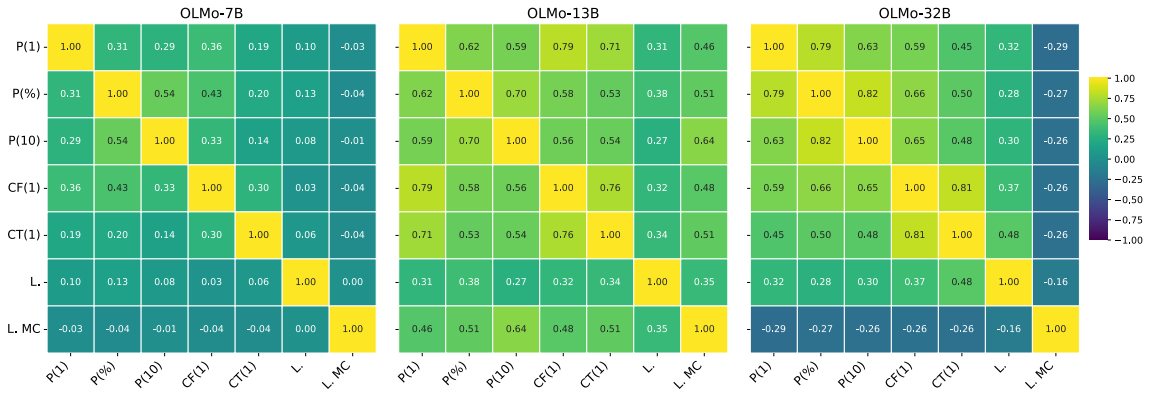


Figure 9: *Robustness*: Pearson correlations of VC when using different prompts on OLMo models with different sizes.

What was the name of the frog in the children's TV series Hector's House? y: Kiki. Corr.: 1													
		CE	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}	**P-RB**
GPT-4o	TriviaQA	Prob.	0.92	0.98	0.99	0.80	0.99	0.98	0.94	0.97	1.00	1.00	0.94
		PS	0.66	0.67	0.68	0.64	0.68	0.68	0.67	0.67	0.68	0.68	0.99
		VC	0.90	0.90	0.70	0.70	0.70	0.05	0.95	-	-	-	0.72
		P(True)	1.00	1.00	1.00	1.00	0.38	1.00	0.68	1.00	1.00	0.99	0.80
		Calib1	0.32	0.30	0.30	0.29	0.30	0.30	0.29	0.28	0.29	0.29	0.99
Who won season 16 on dancing with the stars? y: Nicole Scherzinger. Corr.: 0													
		CE	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}	**P-RB**
Mistral	NQ	Prob.	0.10	0.13	0.09	0.03	0.04	0.07	0.09	0.01	0.09	0.04	0.94
		PS	0.64	0.65	0.63	0.68	0.62	0.62	0.69	0.69	0.69	0.62	0.97
		VC	1.00	1.00	1.00	1.00	1.00	0.90	0.80	-	-	-	0.93
		P(True)	0.01	0.01	0.01	0.01	0.09	0.01	0.01	0.01	0.01	0.50	0.85
		Calib1	0.43	0.43	0.44	0.44	0.44	0.43	0.44	0.44	0.43	0.43	1.00

Table 9: Case study of prompt robustness. The confidence estimated by the difference CE methods when using perturbed 10 prompts for the same question and LLM response is shown.

In humans, fertilization occurs soon after the oocyte leaves this? y^*: Ovary													
			Ovary.	Ovary	Ovary	Ovary.	Ovary	Ovary	The ovary.	Ovary.	Fallopian tube.	Fallopian tube.	
		CE	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9	y_{10}	**A-STB**	**A-SST**
GPT-4o	SciQ	Prob.	0.58	0.85	0.79	0.61	0.78	0.78	0.44	0.61	0.75	0.75	0.87	0.02
		PS	0.65	0.71	0.70	0.66	0.70	0.70	0.62	0.66	0.69	0.69	0.97	0.01
		VC	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.9	1.00	1.00	0.05
		P(True)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.00	0.00	1.00	1.00
		Calib1	0.86	0.93	0.93	0.86	0.93	0.93	0.80	0.86	0.88	0.88	0.95	0.01
What genre is Amen? y^*: Situation comedy													
			R&B	R&B	R&B	R&B	R&B	Gospel	Gospel	Drama	Rap Rock	Reggae		
		CE	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9	y_{10}	**A-STB**	**A-SST**
Mistral	PopQA	Prob.	0.67	0.67	0.68	0.67	0.67	0.56	0.56	0.43	0.72	0.57	1.00	0.24
		PS	0.63	0.63	0.63	0.63	0.63	0.61	0.61	0.57	0.64	0.61	1.00	0.06
		VC	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.85	0.95	1.00	0.00
		P(True)	0.03	0.03	0.03	0.03	0.03	0.10	0.10	0.75	0.11	0.05	1.00	0.72
		Calib1	0.10	0.10	0.10	0.10	0.10	0.53	0.53	0.30	0.13	0.42	1.00	0.20

Table 10: Case study of answer stability and sensitivity. The confidence estimated by the difference in CE methods for 10 sampled answers to the same question, when using the same prompt, is shown. The probabilities of the same answer in GPT-4o can slightly differ even under greedy decoding (temperature = 0), likely due to nondeterminism in the model's inference pipeline.

F. Explaining Generalization of AI-Generated Text Detectors Through Linguistic Analysis

Authors: Yuxi Xia, Kinga Stańczak, Benjamin Roth

Status: Published in the Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2026) (Long Papers)

URL: https://arxiv.org/pdf/2601.07974

Reference: Xia et al. (2026b)

Work Division

Yuxi Xia: conceptualization, data curation, formal analysis, investigation, methodology, software, validation, visualization, original draft preparation, review and editing
Kinga Stańczak: conceptualization, data curation, formal analysis, software, visualization, original draft preparation
Benjamin Roth: conceptualization, funding acquisition, methodology, project administration, resources, supervision, review and editing.

Explaining Generalization of AI-Generated Text Detectors Through Linguistic Analysis

Yuxi Xia[1,2], **Kinga Stańczak**[3], **Benjamin Roth**[1,4]
[1]Faculty of Computer Science, University of Vienna, Vienna, Austria
[2]UniVie Doctoral School Computer Science, Vienna, Austria
[3]Department of Language Science and Technology, Saarland University, Germany
[4]Faculty of Philological and Cultural Studies, University of Vienna, Vienna, Austria
Correspondence: yuxi.xia@univie.ac.at

Abstract

AI-text detectors achieve high accuracy on in-domain benchmarks, but often struggle to generalize across different generation conditions such as unseen prompts, model families, or domains. While prior work has reported these generalization gaps, there are limited insights about the underlying causes. In this work, we present a systematic study aimed at explaining generalization behavior through linguistic analysis. We construct a comprehensive benchmark that spans 6 prompting strategies, 7 large language models (LLMs), and 4 domain datasets, resulting in a diverse set of human- and AI-generated texts. Using this dataset, we fine-tune classification-based detectors on various generation settings and evaluate their cross-prompt, cross-model, and cross-dataset generalization. To explain the performance variance, we compute correlations between generalization accuracies and feature shifts of 80 linguistic features between training and test conditions. Our analysis reveals that generalization performance for specific detectors and evaluation conditions is significantly associated with linguistic features such as tense usage and pronoun frequency. [1]

1 Introduction

The ability to reliably detect AI-generated text is becoming increasingly critical as large language models (LLMs) are deployed in education, media, and content moderation (Guo et al., 2024; Hu et al., 2023). While recent detectors achieve near-perfect performance on standard benchmarks (Guo et al., 2023; Wang et al., 2024a), these evaluations typically assume that training and testing data are drawn from the same distribution. In real-world scenarios, however, AI-generated text varies widely across prompts, model families, and domains, raising serious concerns about how well detectors generalize under distribution shifts.

Prior studies have begun to examine generalization in the context of unseen prompts, models, or datasets (Xu et al., 2024; Liu et al., 2024). However, these efforts largely focus on reporting performance drops without probing the underlying causes. Meanwhile, recent benchmark datasets introduce diversity in generation settings (Wang et al., 2024a; Macko et al., 2023; Li et al., 2024), but offer limited interpretability regarding the features detectors rely on. A more systematic and interpretable approach is needed to explain **why** generalization succeeds or fails.

In this paper, we propose to understand generalization through the lens of linguistic analysis. We hypothesize that changes in surface-level linguistic features, such as verb tense, syntactic complexity, or pronoun usage, can partially account for generalization behavior. To test this hypothesis, we construct a comprehensive benchmark combining 7 LLMs (e.g., Deepseek (DeepSeek-AI, 2025), Mistral (Mistral AI, 2024)), 4 domains (abstracts, news, reviews, QA), and 6 prompting strategies (e.g., few-shot, chain-of-thought (CoT)), enabling evaluation across prompt, model, and dataset generalization. We train the AI-text detectors by fine-tuning on two state-of-the-art models (XLM-RoBERTa (Conneau et al., 2020) and DeBERTa-V3 (He et al.)) for binary classification tasks. Each detector is trained with the texts generated by every possible condition (combination of prompt, model and dataset). The fine-tuned detectors are evaluated for cross-prompt, cross-model and cross-dataset generalization performance. Our results reveal substantial performance degradation under out-of-domain conditions, despite near-perfect in-domain accuracy.

To explain these generalization gaps, we perform a large-scale correlation analysis between detector generalization performance and score changes in 80 linguistic feature metrics across training and testing settings. We find that linguistic features have significant ($p<0.05$) correlations with generaliza-

[1]Code and data is available at: https://github.com/Yuuxii/Generalization-of-AI-text-Detector

tion across different test settings. While some linguistic features (e.g., passive voice, short sentence ratio) are strongly correlated (Pearson correlation > 0.7) with generalization behavior in specific configurations, there is no universal linguistic signal that explains all cases. Our findings suggest that linguistic features offer a useful, though partial, explanation of generalization, and that detectors may rely on different features depending on their training setup and the test conditions.

In summary, our main contributions are: (1) A new benchmark for evaluating AI-text detector generalization across 6 prompts, 4 domains, and 7 LLMs; (2) A comprehensive analysis linking linguistic feature shifts to detector generalization performance; (3) Insights into which linguistic features are most predictive of generalization behavior, helping to guide the development of more robust and interpretable detectors.

Ultimately, our work aims to move beyond raw performance scores toward a deeper understanding of generalization behavior in AI-text detection. While linguistic features alone do not capture the full complexity of generalization, they provide a valuable starting point for interpreting detector behavior in the wild.

2 Related Work

Datasets for AI Text Detection. Recent benchmarks have advanced AI-generated text detection by covering multiple languages (Wang et al., 2024a; Macko et al., 2025), domains (Li et al., 2024; Verma et al., 2024; Dugan et al., 2024), and generator models (Hu et al., 2023; Abassy et al., 2024; Tao et al., 2024). Several works also consider mixed-authorship settings (Yu et al., 2025; Zhang et al., 2024). However, few datasets jointly evaluate the impact of LLM type, domain, and prompt style in a systematic and controlled manner. Prompt engineering remains particularly underexplored, despite its known influence on generation behavior. To address these gaps, we introduce a new dataset that enables controlled experiments across 7 LLMs, 4 domains, and 6 prompting strategies.

AI Text Detection Models. Detection methods mainly fall into two broad categories: statistical detectors (e.g., GLTR (Gehrmann et al., 2019), DetectGPT (Mitchell et al., 2023), Binoculars (Hans et al., 2024)) and fine-tuned classifiers using pre-trained LMs (e.g., RoBERTa (Liu et al., 2019), DeBERTa (He et al., 2021), XLM-R (Conneau et al., 2020)). While statistical detectors offer interpretability, classifier-based approaches consistently achieve stronger performance on benchmark datasets such as M4GT (Wang et al., 2024a), MULTITuDE (Macko et al., 2023), and MultiSocial (Macko et al., 2025). Our study builds on this foundation by fine-tuning two top-performing detectors, XLM-RoBERTa and DeBERTa-V3, across varied training settings to assess their generalization.

Generalization in Detection. Generalization is a core challenge in AI-text detection. Prior work has shown that detectors trained on one task or domain often fail when evaluated on others (Xu et al., 2024; Li et al., 2024; Bhattacharjee et al., 2024). However, most studies focus on reporting performance gaps without offering deeper explanations. Moreover, while some papers examine prompt-based variation, they typically limit prompting strategies or focus on handcrafted or adversarial prompts (Xu et al., 2024; Zhang et al., 2024). In contrast, our work evaluates generalization comprehensively across prompt styles, model families, and content domains, includes both naturalistic generation strategies (e.g., few-shot, CoT) and handcrafted prompts (1-shot CoT, self-refine).

Explaining Generalization Behavior. Several works have proposed high-level explanations for generalization variance. For example, Xu et al. (2024) attribute success to *prompt similarity* and *human–LLM alignment*, while Li et al. (2024) explore distributional differences using linguistic metrics like POS tags and named entity counts. However, these studies stop short of identifying which specific linguistic features correlate with generalization. Our work advances this line of inquiry through a detailed correlation analysis of 80 linguistic features, covering syntactic, stylistic, and discourse-level signals. We quantify feature shifts between training and testing data and link them to generalization performance, revealing interpretable signals, e.g., shifts in pronoun frequency or passive voice usage that influence detector robustness.

3 Dataset Creation

To provide a comprehensive study of generalization of AI-text detectors against prompt, LLM and dataset changes, we create our human-written and AI-generated text dataset by incorporating 6 prompting strategies, 7 LLMs with different parameter sizes from different model families, and 4

datasets from different domains.

3.1 Human-written text

We first randomly sample the human-written text of different domains from 4 datasets: (1) Scientific paper **abstracts** from the arXiv dataset (See et al., 2017); (2) Product **reviews** from the AmazonReviews2023 dataset (Hou et al., 2024); (3) **News** articles from the CNN/Daily Mail dataset (Clement et al., 2019); (4) Question and answers (**QA**) from the ASQA dataset (Stelmakh et al., 2022).

For abstracts and news articles, we use only texts that are at least 1,000 characters long. We set the minimum text length for reviews to 350 characters because the original text is short. For the QA dataset, we sample the longest texts considering the limited size of the data. For each of the datasets, we sample 3,000 examples and split them into training, validation and testing set with a split ratio of 50:17:33.

Data cleaning for human-written text. To remove obvious features for AI-text detectors, we use the following data cleaning steps for all the human-written texts: (1) Removing duplicates; (2) Normalizing punctuation; (3) Removing duplicated whitespace; (4) Removing URLs, e-mail addresses, and emojis; (5) Artifacts such as dates of article; (6) Filtering non-English text; (7) Filter too short text, as text length can impact the difficulty of the task (Wang et al., 2024b).

3.2 AI Text Generated with LLMs

The AI-text part of the dataset consists of texts generated by 7 LLMs using 6 diverse prompting strategies for each dataset. For each human-written text, we apply every LLM and prompting strategy to generate an AI-text counterpart under the same topic. Consequently, for each source dataset, the final data include 3,000 human-written texts and, for every model–prompt combination, 3,000 corresponding AI-generated texts. In total, the dataset comprises 516,000 texts: 12,000 human-written and 504,000 AI-text. Each AI text is matched with its human-written text counterpart for performing a binary classification.

LLMs. We employ LLMs with different parameter sizes and from 5 model families: (1) **Mistral 123B** (Mistral AI, 2024): Mistral-Large-Instruct-2411; (2) **Deepseek 70B** (DeepSeek-AI, 2025): DeepSeek-R1-Distill-Llama-70B; (3) **Llama 70B** (Meta AI, 2024): Llama-3.3-70B-Instruct; (4) **Qwen 72B, Qwen 32B, Qwen 14B** (Yang et al., 2024): Qwen2.5-72B/-32B/-14B-Instruct; (5) **Solar 22B** (Upstage, 2024): solar-pro-preview-instruct.

Prompts. We use 6 different prompting strategies based on existing research on prompt engineering. The prompts include: (1) **0-shot** prompts that only provide the metadata (e.g., title, text length) of each data sample; (2) **3-shot** prompts (Brown et al., 2020) that contains 3 human-written texts from the same dataset for in-context learning; (3) **Style** prompts (Zhang et al., 2024) which require LLMs to write in a style like the given human-written text example; (4) **0-shot CoT** prompts (Kojima et al., 2022) which consist of phrase "let's think step by step."; (5) **1-shot CoT** prompts (Wei et al., 2022) that contain manually written step-by-step instructions, the instruction is based on an example of a human-written text; and (6) **Self-refine** prompts (Madaan et al., 2023) that use the LLM itself to critique and improve its own responses. Self-refine prompts are multi-stage prompts that comprise 4 stages: firstly, the LLM is prompted to generate the AI text, then it is requested to provide feedback on how to make the generated AI text more human-like. Later, the LLM needs to incorporate the feedback to improve the initially generated AI text. The improved text is final if the LLM judges it sounds more human-written than the human-written text counterpart; otherwise, it goes back to the feedback step for at most 3 iterations. For each prompting strategy, we modified the prompt template used for each dataset to suit the task of text generation. To match the text length of the AI-generated texts to their human-written counterparts, we include information about character count in the prompts. A detailed discussion of the prompts can be found in the appendix A.1, along with the prompt templates used to create our dataset (Table 6).

Data Cleaning for AI-Generated Text. To prevent detectors from exploiting superficial artifacts rather than genuine linguistic characteristics, we extensively clean the LLM-generated texts by removing elements that could trivially reveal their artificial origin. Specifically, we remove formulaic AI responses (e.g., *"Certainly!"*, *"Sure!"*), structural markers such as section titles, bullet points, and numbered lists, placeholder tokens in square brackets (e.g., *[your name]*, *[insert e-mail address]*), extraneous metadata including review

ratings, character-count information, and sentences beginning with *"Note:"* that describe the generation process, non-linguistic symbols such as asterisks (*), triple dashes (–), and hash symbols (#), as well as model-specific tags such as `\think` and any preceding text in Deepseek reasoning outputs. This cleaning step ensures that the evaluation focuses on the linguistic properties of the generated text rather than on easily detectable formatting artifacts.

4 Generalization of AI-Text Detectors

We introduce three evaluation settings to assess the generalization ability of AI-text detectors across three dimensions: prompts, LLMs, and datasets.

4.1 Generalization Testing

Assume an AI-text detector M is fine-tuned on a training set $\mathcal{D}^{\text{train}}_{i,j,k}$, consisting of AI-generated texts produced with prompt p_i by a generative model g_j for dataset d_k, along with human-written texts. We evaluate the cross-prompt, cross-model, and cross-dataset generalization of the AI-text detector under the following settings.

Cross-prompt (C-P) testing. This setting evaluates how well detectors generalize to texts generated with prompting strategies unseen during training. Each detector is evaluated on test sets produced by the same LLM and drawn from the same dataset, but generated with different prompts. This controlled setting ensures that the only changing factor is prompt strategies during evaluation. For example, a detector trained on the training split of QA texts generated by Llama 70B with 0-shot prompts is evaluated on the test split generated by Llama 70B with all 6 prompt types. Formally, the accuracy of the detector on the test data D^{test} when trained on D^{train} is denoted as $\text{Acc}(M(D^{test}|D^{train}))$. Thus, the generalization accuracies from prompt p_i to all prompts are formalized as a list:

$$\Delta^{(p_i)}_{\text{gen}} = \left\{\text{Acc}\left(M\left(\mathcal{D}^{\text{test}}_{c,j,k} \mid \mathcal{D}^{\text{train}}_{i,j,k}\right)\right)\right\}^{6}_{c=1}. \quad (1)$$

Where $\Delta^{(p_i)}_{\text{gen}}$ is a list of 6 accuracy values, with each value representing the generalization accuracy of a prompt. We carried out the test for each prompt and resulted in a 2-dimensional 6x6 vector (plot like left heatmaps in Figure 2), which presents the cross-prompt result of all prompts in one of the conditions, denoted as $\Delta^{(p)}_{\text{gen}}$.

Cross-model (C-M) testing. This setting evaluates generalization to texts generated by LLMs not seen during training. A detector fine-tuned on the training split from one LLM is evaluated on the test splits produced by other LLMs. For example, a detector trained on abstracts generated by Llama 70B is tested on abstracts generated by all 7 LLMs. We use 0-shot prompts (p_1) in this setting. Formally:

$$\Delta^{(g_j)}_{\text{gen}} = \left\{\text{Acc}\left(M\left(\mathcal{D}^{\text{test}}_{1,c,k} \mid \mathcal{D}^{\text{train}}_{1,j,k}\right)\right)\right\}^{7}_{c=1}. \quad (2)$$

Similar to cross-prompt testing, we apply the formula to all LLMs, obtaining in a 7x7 vector as cross-model results $\Delta^{(g)}_{\text{gen}}$.

Cross-dataset (C-D) testing. This setting evaluates generalization across different dataset domains. A detector fine-tuned on the training split from one dataset is evaluated on test splits from other datasets generated by the same LLM. For example, a detector trained on abstracts generated by Llama 70B is tested on news, reviews, and QA data generated by the same LLM. We use 0-shot prompts (p_1) in this setting. Formally:

$$\Delta^{(d_k)}_{\text{gen}} = \left\{\text{Acc}\left(M\left(\mathcal{D}^{\text{test}}_{1,j,c} \mid \mathcal{D}^{\text{train}}_{1,j,k}\right)\right)\right\}^{4}_{c=1}. \quad (3)$$

The corresponding cross-dataset result $\Delta^{(d)}_{\text{gen}}$ is a 4x4 vector.

4.2 Training setup

We fine-tune XLM-RoBERTa-base (Conneau et al., 2020) (referred to as RoBERTa) and DeBERTa-V3-small (He et al.) (referred to as DeBERTa) for binary classification to distinguish between human-written and AI-generated text. These architectures have achieved state-of-the-art performance in prior work (Wang et al., 2024a). We train a separate detector for each combination of prompt type, AI-text generation model, and dataset type, resulting in 168 (i.e., 7x4x6) in-domain detectors when using, for example, RoBERTa for fine-tuning.

Model fine-tuning is performed using the following hyperparameters: `learning rate` = 2e-5, `num train epochs` = 3, `weight decay` = 0.01, `train batch size` = 16. The maximum sequence length is set to 512, corresponding to the maximum input length supported by XLM-RoBERTa. All our experiments are conducted on NVIDIA HGX H100, and approximately 400 GPU hours to replicate.

5 Explaining Generalization with Linguistic Analysis

To better understand what causes the variance of generalization performance, we perform a comprehensive linguistic feature analysis by measuring the correlation of 80 different feature metrics with the generation results. We first introduce the definition and metric of each linguistic feature and present the correlation evaluation method.

5.1 Linguistic Feature Definitions and Metrics

Our studied features can be categorized into Lexical diversity, Lexical density, Sentiment, Readability, Part-of-Speech (POS), and Grammatical and Lexical analysis. We introduce the most correlated features and metrics in the main paper and the rest in Appendix A.4.

Readability. Readability refers to the ease of understanding a text. AI-generated text tends to be less readable than human-written text (Markowitz et al., 2024; Mathews et al., 2024). We use the **Gunning fog index** (Yadagiri et al., 2024) as one of the measures of readability, which is an estimated number of years needed to understand a given passage.

Part-of-Speech (POS). We use a selection of metrics from StyloMetrix (Okulska et al., 2023) to compare the frequency of **verbs, nouns, adjectives, numerals, etc**. The frequency of parts of speech is measured as the fraction of text covered by tokens representing a given part of speech. Previous research has discovered differences between human-written and AI-generated text in terms of frequency of certain POS (Georgiou, 2024). Therefore, POS analysis is relevant for gaining insights into the literary style of texts in our dataset.

Grammatical Analysis. We use StyloMetrix (Okulska et al., 2023) metrics to compare texts in terms of grammatical categories related to verbs. Human-written texts have been shown to contain more passive voice than AI-generated texts (Georgiou, 2024). We measure the **incidence of passive and active voice** as the frequency of verbs in passive or active voice. We also compare the differences between the choice of tenses. The **frequency of past, present and future tenses** is measured as the fraction of the text covered by verbs in past, present and future tenses. For example, the incidence of past tenses is the number of verbs in past simple, past continuous, past perfect or past perfect continuous divided by the total number of words in the text.

Lexical Analysis. As part of lexical analysis, we measure the **frequency of personal names**, and **adjectives in comparative and superlative degrees**. The pronoun-related metrics (Okulska et al., 2023) analyze the differences in the usage of pronouns in human-written and AI-generated text. We calculate the frequency of specific personal or reflexive pronouns and certain types of pronouns (e.g., "We", "It", "Our", "Yourself").

5.2 Correlation Between Generalization and Linguistic Features

To further investigate factors that influence generalization, we examine how changes in linguistic features correlate with generalization performance. For each linguistic feature f, we compute its shift between a given training configuration and the corresponding test configuration:

$$\Delta_f = f(\mathcal{D}^{\text{train}}) - f(\mathcal{D}^{\text{test}}) \tag{4}$$

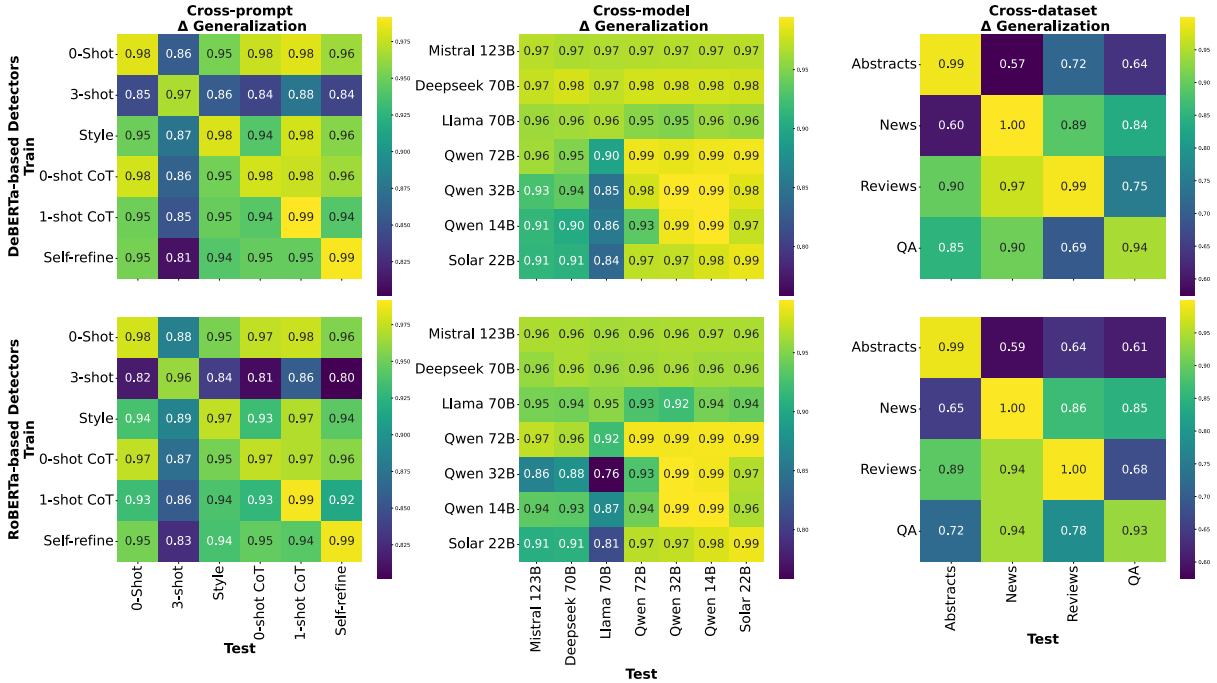
where $f(\mathcal{D}^{\text{train}})$ denotes the feature difference between AI-generated and human-written texts in the training configuration ($f(\mathcal{D}^{\text{train}}) = f(\mathcal{D}^{\text{train}}_{human}) - f(\mathcal{D}^{\text{train}}_{AI})$), and $f(\mathcal{D}^{\text{test}})$ denotes the feature difference in the corresponding test configuration. Corresponding to the generalization testing, we denote the cross-prompt, cross-model, and cross-dataset feature shift as $\Delta_f^{(p)}$, $\Delta_f^{(g)}$, and $\Delta_f^{(d)}$, respectively, which all have the same size.

We then compute the Pearson correlation between **flattened** generalization accuracy and these feature shifts under the same conditions:

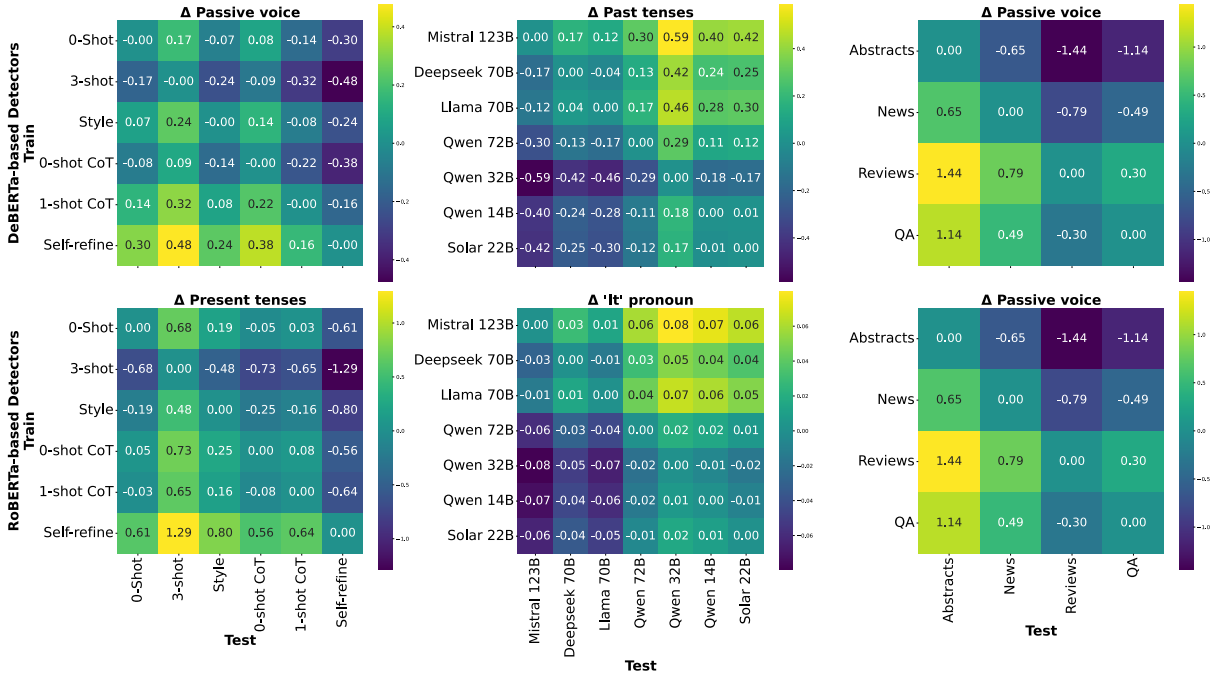
$$Corr(f) = |\text{Pearson}\big(\Delta^{(n)}_{\text{gen}}, \Delta^{(n)}_f\big)| \tag{5}$$

where n indexes each cross-prompt, cross-model, or cross-dataset comparison. The resulting value $Corr(f)$ lies in $[0, 1]$, with $0.1 \leq Corr(f) < 0.3$ indicating a low correlation, $0.3 \leq Corr(f) < 0.5$, $0.5 \leq Corr(f) < 0.7$ and $Corr(f) \geq 0.7$ as moderate, high and strong correlations (DATAtab Team, 2025). This analysis allows us to identify which linguistic features are most strongly associated with robust generalization across different test settings.

Setting-specific correlation (Table 2) is measured under a specific testing combination. For example, the setting-specific correlation of cross-prompt generalization and linguistic feature shifts



(a) The aggregated generalization performance.



(b) The aggregated feature shifts between training and test configurations.

Figure 1: Comparison of aggregated generalization performance and aggregated feature shifts across all evaluation settings. Similar patterns in the two heatmaps indicate that certain feature shifts are correlated with reduced generalization accuracy.

is only measured on texts from Llama 70B and the Abstract dataset.

Overall correlation (Table 1) is measured under all combinations. For example, the overall correlation of cross-prompt generalization and linguistic feature shifts is measured on the texts of every combination of LLMs and datasets.

Aggregation of Results. To provide a high-level summary of cross-prompt, cross-model, and cross-dataset generalization and feature shifts (as Figure 1a and 1b), we report accuracy and shift values averaged over the dimensions that are not the focus of the evaluation. For instance, when presenting overall cross-prompt results, we average accuracy

	Feature Metric	DeBERTa-based			RoBERTa-based		
		C-P	C-M	C-D	C-P	C-M	C-D
Readability	Gunning fog	0.056	**0.248**	**0.231**	0.043	**0.308**	***0.261***
Part-of-Speech	Numerals	***0.108***	***0.363***	0.086	**0.076**	**0.157**	0.031
Grammatical	Passive voice	**0.109**	**0.281**	**0.296**	0.054	**0.221**	**0.287**
	Active voice	0.010	**0.194**	0.066	0.061	***0.373***	0.014
	Present tenses	0.046	**0.147**	0.144	**0.116**	**0.328**	0.173
	Past tenses	0.018	**0.416**	0.031	0.006	**0.324**	0.049
	Future tenses	**0.062**	0.066	0.159	**0.069**	**0.172**	0.111
Lexical	Personal names	**0.076**	***0.381***	0.030	0.046	**0.182**	0.071
	Adjectives in comparative degree	0.027	0.123	***0.267***	0.023	**0.317**	***0.251***
	Adjectives in superlative degree	***0.095***	**0.239**	0.034	0.047	0.123	0.055
	"We" pronoun	0.014	0.041	0.015	***0.108***	0.030	0.037
	"It" pronoun	0.029	**0.236**	0.184	**0.077**	**0.385**	0.120
	"Our" possessive pronoun	0.004	0.007	***0.261***	***0.113***	0.063	**0.246**
	"Yourself" pronoun	0.030	**0.183**	0.157	0.061	***0.366***	0.111

Table 1: The **overall** Pearson correlation between generalization performance with different linguistic features. This table only presents the features that fall into the top 3 correlated features in one of the settings, more results are shown in Table 8 (Appendix). The **significant (p<0.05) correlation is bolded**. We underline the strongest correlation for each setting, and *italicize the other scores within the top 3 correlated features*. The results of other features that are less correlated are shown in the Appendix.

	DeBERTa-based					RoBERTa-based				
	Abstracts	News	Reviews	QA	ALL	Abstracts	News	Reviews	QA	ALL
Mistral 123B	**0.421**	**0.636**	**0.577**	**0.709**	**0.395**	0.155	**0.703**	**0.573**	**0.630**	**0.219**
Deepseek 70B	**0.342**	**0.605**	**0.415**	**0.735**	**0.276**	**0.528**	**0.506**	**0.459**	**0.728**	**0.400**
Llama 70B	**0.412**	0.209	**0.736**	**0.584**	**0.346**	0.245	0.248	**0.758**	**0.572**	**0.380**
Qwen 72B	0.072	**0.590**	0.317	0.301	0.098	0.128	**0.519**	**0.549**	**0.357**	**0.212**
Qwen 32B	0.214	**0.684**	**0.527**	**0.678**	**0.183**	**0.377**	**0.672**	**0.617**	**0.492**	**0.308**
Qwen 14B	0.275	**0.476**	**0.553**	**0.560**	**0.218**	0.264	**0.457**	**0.605**	**0.580**	**0.204**
Solar 22B	**0.563**	**0.482**	**0.517**	**0.614**	**0.303**	**0.703**	**0.688**	0.320	**0.582**	**0.426**
ALL	**0.196**	**0.231**	**0.437**	**0.255**	**0.109**	**0.246**	**0.155**	**0.486**	**0.224**	**0.116**

Table 2: The cross-prompt correlation between generation performance and the most correlated linguistic feature when evaluated on different datasets and models. The **significant correlation is bolded**. We underline the strongest correlation for each dataset. Similar cross-model and cross-dataset results are in Appendix.

scores across all 7 LLMs and 4 datasets.

6 Results and Analysis

We analyze the results in Table 1, Table 2, and Figure 1 to understand how generalization performance is shaped by shifts in linguistic features.

6.1 General Findings

Finding 1: Linguistic features have significant correlations with generalization results. Table 1 shows that several features exhibit significant correlations (bold values) with detector generalization. For example, overall *cross-model generalization* that averaged across datasets is moderately correlated (0.416) with the proportion of past-tense verbs, indicating that stylistic verb usage in training data influences transfer. In contrast, cross-prompt generalization averaged across datasets and LLMs shows only weak correlation with passive voice. These results highlight that some features play a more critical role than others in determining generalization success.

Finding 2: Certain dataset–model combinations reveal very strong feature dependencies. Although overall cross-prompt correlations appear weak in Table 1, Table 2 reveals that in specific configurations the effect is dramatic. For instance, on Llama-70B outputs for the Reviews dataset, cross-prompt generalization is strongly correlated (>0.7) with the number of short sentences. Fig 2a shows that changes in this feature align directly with sharp drops in performance when generalizing from 1-shot CoT to other prompting strategies. This demonstrates that some detectors are highly sensitive to prompt-induced shifts in linguistic structure.

Finding 3: Different detectors rely on different linguistic features. RoBERTa-based detectors and DeBERTa-based detectors do not exploit the same linguistic signals. For example, the cross-model generalization of RoBERTa models is moderately correlated (0.385) with the frequency of "It" pronouns, whereas for DeBERTa the correlation is only 0.236. This suggests that detectors may learn fundamentally different features for distinguishing human and AI text even when trained on the same data.

6.2 Linguistic Analysis of Generalization Results

Figure 4 summarizes average performance across the three generalization settings. As expected, **in-domain testing achieves near-perfect accuracy**, as shown in the diagonal cells of Figure 1a, confirming the strong baseline capabilities of our detectors. Detailed in-domain results are reported in Table 7 in the Appendix.

6.2.1 Cross-prompt Generalization

The most striking pattern is that the **3-shot prompt is consistently the hardest to generalize to and from**, with accuracy dropping to 80–89%. Other prompting strategies show relatively minor effects.

Explanation. Figure 1b shows averaged feature shifts, but strong effects can be masked by aggregation. For example, Figure 2b highlights a clear pattern: AI texts that use the "We" pronoun in similar contexts are more difficult to generalize to for the detectors. This confirms Findings 2 and 3: when we zoom in on specific dataset–model pairs, clear linguistic drivers of generalization emerge.

6.2.2 Cross-model Generalization

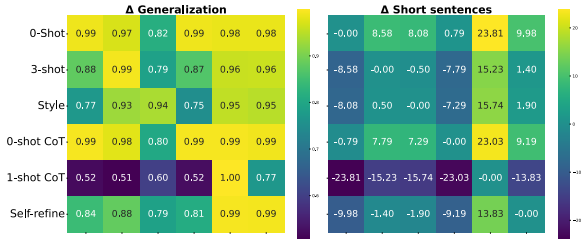
A major finding is that **detectors trained on Qwen or Solar outputs perform poorly on Llama-generated text**, whereas generalization across other LLMs is more stable.

Explanation. Figures 1a and 1b reveal that cross-model generalization is moderately influenced by shifts in past-tense usage and "It" pronoun frequency. Qwen and Solar outputs share similar linguistic profiles, which diverge from Llama's, explaining this degradation.

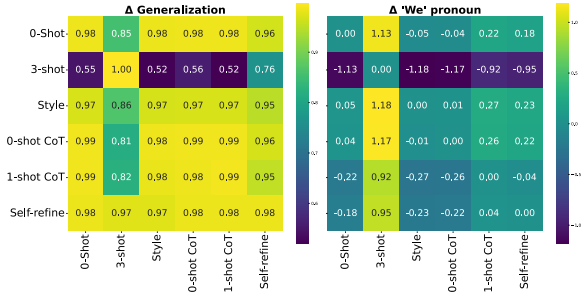
6.2.3 Cross-dataset Generalization

The most pronounced performance gap appears here: **detectors trained on abstracts generalize poorly to other datasets, achieving as low as 57% accuracy when tested on news articles**. Conversely, detectors trained on reviews or QA data transfer more successfully to news.

Explanation. Across all detectors, cross-dataset generalization shows moderate correlation with passive voice usage. Abstracts exhibit a higher rate of passive constructions, which likely makes them a poor source domain for training detectors that must generalize broadly.



(a) Llama 70B, Reviews



(b) Deepseek 70B, Abstracts

Figure 2: Cross-prompt generalization and feature shifts when evaluating on a specific model and dataset. A clearer and stronger correlation is observed than the overall cross-prompt correlation in Figure 1. We also show the specific case study of the generated texts for the above two settings in Table 9 and 10 in the Appendix.

6.3 Robustness Study

We conduct the robustness study using multiple-hypothesis corrections (Bonferroni (Goeman and Solari, 2014) and Benjamini–Hochberg FDR (Bogdan et al., 2008)), and further including Spearman (non-linear) correlations (Ali Abd Al-Hameed, 2022). The results are shown in Table 3, we find that:

(1) Key linguistic correlates (pronoun usage, verb tense, active/passive voice) remain robust for cross-prompt and cross-model generalization.

(2) A substantial number of features (>=15) remain significant in cross-model settings across various settings.

(3) Non-linear effects emerge under Spearman correlations, strengthening our interpretation of de-

		DeBERTa-based			RoBERTa-based		
		C-P	C-M	C-D	C-P	C-M	C-D
Pearson	Bonferroni	2	15	0	3	28	0
	FDR	2	37	0	6	58	0
	Top features	Numerals Passive voice	Past tense Personal names Numerals	–	"Our" pronoun Present tense "We" pronoun	"It" pronoun Active voice "Yourself" pronoun	–
Spearman	Bonferroni	3	16	0	2	25	0
	FDR	4	44	0	10	51	0
	Top features	"She" pronoun "He" pronoun short sentences	MATTR FLESCH Gunning Fog	–	"She" pronoun "Her" pronoun Present tense	Active voice "It" pronoun Function words	–

Table 3: Robustness study of applying multiple-hypothesis correction to Pearson (linear) and Spearman (non-linear) correlation. The numerical values represent the number of features that remain significant after the multiple-hypothesis correction.

tector behavior.

(4) For cross-dataset generalization, no individual features remain significant under strict multiple-hypothesis correction, suggesting that performance degradation is likely driven by broader distributional shifts rather than a single dominant linguistic cue.

Importantly, our main conclusions remain valid after these robustness checks.

6.4 Discussion

While our analysis highlights the role of linguistic features in explaining the generalization behavior of AI-text detectors, we acknowledge that these features represent only one facet of a more complex landscape. Generalization performance is likely influenced by a broader set of factors, including semantic coherence, discourse structure, and detector-specific inductive biases. Our findings should therefore be interpreted as offering a linguistic perspective rather than a comprehensive account of generalization. Nonetheless, by systematically correlating linguistic feature shifts with detection performance, our study contributes valuable insights into how stylistic and grammatical signals may impact detector robustness across prompts, models, and domains.

7 Conclusion

This work presents an interpretable investigation into the generalization behavior of AI-text detectors. While prior studies primarily report detection performance, we go further by examining why generalization succeeds or fails through a linguistic lens. Across a large-scale benchmark incorporating diverse prompts, LLMs, and domains, we show that state-of-the-art detectors, despite near-perfect in-domain accuracy, often struggle in cross-prompt, cross-model, and cross-dataset scenarios. To explain these generalization behaviors, we quantify shifts in 80 linguistic features between training and testing distributions and uncover statistically significant correlations between feature shifts and generalization performance.

Our analysis reveals that features such as pronoun usage, verb tense, and passive voice are predictive of generalization gaps, but their influence varies across detectors and settings. This suggests that detectors latch onto different linguistic signals depending on their training context, impacting their robustness in different testing scenarios.

These findings underscore two key points: (1) evaluation must extend beyond in-domain testing to realistically assess detector reliability, and (2) linguistic analysis provides a principled and interpretable path toward diagnosing and improving generalization.

Limitations

While our work offers new insights into the generalization behavior of AI-text detectors through linguistic analysis, several limitations remain.

First, our study focuses on English-language text and detectors trained on English corpora. Although our methodology can be extended to multilingual settings, the linguistic features and generalization patterns may differ significantly across languages due to variations in grammar and stylistic conventions.

Second, we rely on surface-level linguistic fea-

tures (e.g., POS tags, sentence length, voice, pronouns) that can be extracted using standard NLP tools. While these features provide interpretable signals, they may not capture deeper semantic or discourse-level properties that also influence detector decisions.

Third, the detectors we evaluate are based on fine-tuned encoder-only transformer models. Other architectures, such as generative or retrieval-augmented models, may exhibit different generalization behaviors and rely on alternative linguistic features.

Fourth, our correlation-based analysis reveals associations but does not establish causal relationships between feature shifts and performance drops. Further research using controlled interventions or counterfactual examples would be needed to verify causality.

Lastly, our dataset covers a wide but still limited set of domains, models, and prompting strategies. As the landscape of LLMs and prompting methods continues to evolve, future work should assess whether our findings hold for more recent or unseen generation techniques.

Despite these limitations, our study provides a strong foundation for more principled and interpretable evaluations of generalization in AI-text detection.

Acknowledgments

This research has been funded by the Vienna Science and Technology Fund (WWTF)[10.47379/VRG19008] "Knowledge infused Deep Learning for Natural Language Processing".

References

Mervat Abassy, Kareem Elozeiri, Alexander Aziz, Minh Ngoc Ta, Raj Vardhan Tomar, Bimarsha Adhikari, Saad El Dine Ahmed, Yuxia Wang, Osama Mohammed Afzal, Zhuohan Xie, Jonibek Mansurov, Ekaterina Artemova, Vladislav Mikhailov, Rui Xing, Jiahui Geng, Hasan Iqbal, Zain Muhammad Mujahid, Tarek Mahmoud, Akim Tsvigun, Alham Fikri Aji, Artem Shelmanov, Nizar Habash, Iryna Gurevych, and Preslav Nakov. 2024. LLM-DetectAIve: a tool for fine-grained machine-generated text detection. In Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pages 336–343, Miami, Florida, USA. Association for Computational Linguistics.

Khawla Ali Abd Al-Hameed. 2022. Spearman's correlation coefficient in statistical analysis. International Journal of Nonlinear Analysis and Applications, 13(1):3249–3255.

Amrita Bhattacharjee, Raha Moraffah, Joshua Garland, and Huan Liu. 2024. Eagle: A domain generalization framework for ai-generated text detection. ArXiv, abs/2403.15690.

Małgorzata Bogdan, Jayanta K Ghosh, Surya T Tokdar, et al. 2008. A comparison of the benjamini-hochberg procedure with some bayesian rules for multiple testing. In Beyond parametrics in interdisciplinary research: Festschrift in honor of Professor Pranab K. Sen, volume 1, pages 211–231. Institute of Mathematical Statistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In Advances in Neural Information Processing Systems, volume 33, pages 1877–1901. Curran Associates, Inc.

Colin B. Clement, Matthew Bierbaum, Kevin P. O'Keeffe, and Alexander A. Alemi. 2019. On the use of arxiv as a dataset. ArXiv, abs/1905.00075.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 8440–8451, Online. Association for Computational Linguistics.

Michael A. Covington and Joe D. McFall. 2010. Cutting the gordian knot: The moving-average type–token ratio (mattr). Journal of Quantitative Linguistics, 17(2):94–100.

DATAtab Team. 2025. Pearson correlation: A beginner's guide. https://datatab.net/tutorial/pearson-correlation. DATAtab: Online Statistics Calculator. DATAtab e.U., Graz, Austria.

DeepSeek-AI. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. Preprint, arXiv:2501.12948.

Heather Desaire, Aleesa E. Chua, Madeline Isom, Romana Jarosova, and David Hua. 2023. Distinguishing academic science writing from humans or chatgpt with over 99% accuracy using off-the-shelf machine learning tools. Cell Reports Physical Science, 4(6):101426.

Liam Dugan, Alyssa Hwang, Filip Trhlík, Andrew Zhu, Josh Magnus Ludan, Hainiu Xu, Daphne Ippolito, and Chris Callison-Burch. 2024. RAID: A shared benchmark for robust evaluation of machine-generated text detectors. In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 12463–12492, Bangkok, Thailand. Association for Computational Linguistics.

Sebastian Gehrmann, Hendrik Strobelt, and Alexander Rush. 2019. GLTR: Statistical detection and visualization of generated text. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, pages 111–116, Florence, Italy. Association for Computational Linguistics.

Georgios P. Georgiou. 2024. Differentiating between human-written and ai-generated texts using linguistic features automatically extracted from an online computational tool. Inf., 16:979.

Jelle J. Goeman and Aldo Solari. 2014. Multiple hypothesis testing in genomics. Statistics in medicine, 33(11):1946–1978.

Biyang Guo, Xin Zhang, Ziyuan Wang, Minqi Jiang, Jinran Nie, Yuxuan Ding, Jianwei Yue, and Yupeng Wu. 2023. How close is chatgpt to human experts? comparison corpus, evaluation, and detection. ArXiv, abs/2301.07597.

Xun Guo, Yongxin He, Shan Zhang, Ting Zhang, Wanquan Feng, Haibin Huang, and Chongyang Ma. 2024. Detective: Detecting ai-generated text via multi-level contrastive learning. Advances in Neural Information Processing Systems, 37:88320–88347.

Abhimanyu Hans, Avi Schwarzschild, Valeriia Cherepanova, Hamid Kazemi, Aniruddha Saha, Micah Goldblum, Jonas Geiping, and Tom Goldstein. 2024. Spotting llms with binoculars: zero-shot detection of machine-generated text. In Proceedings of the 41st International Conference on Machine Learning, ICML'24. JMLR.org.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. Debertav3: Improving deberta using electra-style pretraining with gradient disentangled embedding sharing. ICLR 2023.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention. In International Conference on Learning Representations.

Yupeng Hou, Jiacheng Li, Zhankui He, An Yan, Xiusi Chen, and Julian McAuley. 2024. Bridging language and items for retrieval and recommendation. ArXiv, abs/2403.03952.

Xiaomengc Hu, Pin-Yu Chen, and Tsung-Yi Ho. 2023. Radar: robust ai-text detection via adversarial learning. In Proceedings of the 37th International Conference on Neural Information Processing Systems, NIPS '23.

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, NIPS '22.

Yafu Li, Qintong Li, Leyang Cui, Wei Bi, Zhilin Wang, Longyue Wang, Linyi Yang, Shuming Shi, and Yue Zhang. 2024. MAGE: Machine-generated text detection in the wild. In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 36–53, Bangkok, Thailand. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. Preprint, arXiv:1907.11692.

Zeyan Liu, Zijun Yao, Fengjun Li, and Bo Luo. 2024. On the detectability of chatgpt content: Benchmarking, methodology, and evaluation through the lens of academic writing. In Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security, CCS '24, page 2236–2250, New York, NY, USA. Association for Computing Machinery.

Dominik Macko, Jakub Kopál, Robert Moro, and Ivan Srba. 2025. MultiSocial: Multilingual benchmark of machine-generated text detection of social-media texts. In Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 727–752, Vienna, Austria. Association for Computational Linguistics.

Dominik Macko, Robert Moro, Adaku Uchendu, Jason Samuel Lucas, Michiharu Yamashita, Matúš Pikuliak, Ivan Srba, Thai Le, Dongwon Lee, Jakub Simko, and Maria Bielikova. 2023. MULTITuDE: Large-Scale Multilingual Machine-Generated Text Detection Benchmark. In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pages 9960–9987.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-refine: iterative refinement with self-feedback. In Proceedings of the 37th International Conference on Neural Information Processing Systems, NIPS '23.

Anastasia Margolina and Anastasia Kolmogorova. 2023. Exploring Evaluation Techniques in Controlled Text Generation: A Comparative Study of Semantics and Sentiment in ruGPT3large-Generated and Human-Written Movie Reviews. In COMPUTATIONAL LINGUISTICS AND INTELLECTUAL TECHNOLOGIES". RSUH.

David M. Markowitz, Jeffrey T. Hancock, and Jeremy N. Bailenson. 2024. Linguistic markers of inherently false ai communication and intentionally false human communication: Evidence from hotel reviews. Journal of Language and Social Psychology, 43(1):63–82.

Daniel Mathews, Justin P Varghese, and Libin Chacko Samuel. 2024. Classifying ai-generated summaries and human summaries based on statistical features. In 2024 International Conference on Trends in Quantum Computing and Emerging Business Technologies, pages 1–5.

Akshay Mendhakar and Darshan H S. 2023. Parts-of-speech (pos) analysis and classification of various text genres. Corpus-based Studies across Humanities, 1(1):99–131.

Meta AI. 2024. Introducing meta llama 3: The most capable openly available llm to date. Accessed: 2025-05-01.

Mistral AI. 2024. Mistral large instruct 2411. Accessed: 2025-05-01.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D. Manning, and Chelsea Finn. 2023. Detectgpt: zero-shot machine-generated text detection using probability curvature. In Proceedings of the 40th International Conference on Machine Learning, ICML'23. JMLR.org.

Alberto Muñoz-Ortiz, Carlos Gómez-Rodríguez, and David Vilares. 2024. Contrasting linguistic patterns in human and llm-generated news text. Artificial Intelligence Review, 57(10).

Inez Okulska, Daria Stetsenko, Anna Kołos, Agnieszka Karlińska, Kinga Głąbińska, and Adam Nowakowski. 2023. Stylometrix: An open-source multilingual tool for representing stylometric vectors. Preprint, arXiv:2309.12810.

Chidimma Opara. 2024. Styloai: Distinguishing ai-generated content with stylometric analysis. In Artificial Intelligence in Education. Posters and Late Breaking Results, Workshops and Tutorials, Industry and Innovation Tracks, Practitioners, Doctoral Consortium and Blue Sky, pages 105–114, Cham. Springer Nature Switzerland.

Jacques Savoy. 2020. Machine Learning Methods for Stylometry: Authorship Attribution and Author Profiling. Springer International Publishing, Cham.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Ivan Stelmakh, Yi Luan, Bhuwan Dhingra, and Ming-Wei Chang. 2022. ASQA: Factoid questions meet long-form answers. In Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 8273–8288, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Zhen Tao, Zhiyu Li, Dinghao Xi, and Wei Xu. 2024. CUDRT: Benchmarking the Detection of Human vs. Large Language Models Generated Texts. arXiv preprint. ArXiv:2406.09056 [cs].

Upstage. 2024. Solar pro preview instruct. https://huggingface.co/upstage/solar-pro-preview-instruct. Instruction-tuned 22B-parameter LLM optimized for single-GPU performance. Preview released September 2024.

Vivek Verma, Eve Fleisig, Nicholas Tomlin, and Dan Klein. 2024. Ghostbuster: Detecting text ghostwritten by 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 1702–1717, Mexico City, Mexico. Association for Computational Linguistics.

Yuxia Wang, Jonibek Mansurov, Petar Ivanov, Jinyan Su, Artem Shelmanov, Akim Tsvigun, Osama Mohammed Afzal, Tarek Mahmoud, Giovanni Puccetti, Thomas Arnold, Alham Aji, Nizar Habash, Iryna Gurevych, and Preslav Nakov. 2024a. M4GT-bench: Evaluation benchmark for black-box machine-generated text detection. In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 3964–3992, Bangkok, Thailand. Association for Computational Linguistics.

Yuxia Wang, Jonibek Mansurov, Petar Ivanov, Jinyan Su, Artem Shelmanov, Akim Tsvigun, Chenxi Whitehouse, Osama Mohammed Afzal, Tarek Mahmoud, Toru Sasaki, Thomas Arnold, Alham Fikri Aji, Nizar Habash, Iryna Gurevych, and Preslav Nakov. 2024b. M4: Multi-generator, multi-domain, and multi-lingual black-box machine-generated text detection. In Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1369–1407, St. Julian's, Malta. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In Proceedings of the 36th International Conference on Neural Information Processing Systems, NIPS '22.

Han Xu, Jie Ren, Pengfei He, Shenglai Zeng, Yingqian Cui, Amy Liu, Hui Liu, and Jiliang Tang. 2024. On the generalization of training-based ChatGPT detection methods. In Findings of the Association for Computational Linguistics: EMNLP 2024, pages

7223–7243, Miami, Florida, USA. Association for Computational Linguistics.

Annepaka Yadagiri, Lavanya Shree, Suraiya Parween, Anushka Raj, Shreya Maurya, and Partha Pakray. 2024. Detecting AI-generated text with pre-trained models using linguistic features. In Proceedings of the 21st International Conference on Natural Language Processing (ICON), pages 188–196, AU-KBC Research Centre, Chennai, India. NLP Association of India (NLPAI).

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2024. Qwen2.5 technical report. arXiv preprint arXiv:2412.15115.

Hatice Yildiz Durak, Figen Eğin, and Aytuğ Onan. 2025. A Comparison of Human-Written Versus AI-Generated Text in Discussions at Educational Settings: Investigating Features for ChatGPT, Gemini and BingAI. European Journal of Education, 60(1):e70014.

Peipeng Yu, Jiahan Chen, Xuan Feng, and Zhihua Xia. 2025. Cheat: A large-scale dataset for detecting chatgpt-written abstracts. IEEE Transactions on Big Data, 11(3):898–906.

Sergio E. Zanotto and Segun Aroyehun. 2024. Human variability vs. machine consistency: A linguistic analysis of texts generated by humans and large language models. Preprint, arXiv:2412.03025.

Yuehan Zhang, Yongqiang Ma, Jiawei Liu, Xiaozhong Liu, Xiaofeng Wang, and Wei Lu. 2024. Detection vs. anti-detection: Is text generated by ai detectable? In Wisdom, Well-Being, Win-Win, pages 209–222, Cham. Springer Nature Switzerland.

A Appendix

A.1 Overview of prompting techniques

To test diverse prompting techniques, we chose six different prompt types. The prompt types are:

0-shot: Our zero-shot prompts are simple instructions for the language models that include a basic piece of information about the text and desired length of the text. For example, the template for the prompt to generate abstracts is *Write an abstract for an article titled "{title}". The abstract should be around {length} characters long.*

Chain-of-Thought (0-shot CoT): Chain-of-thought prompting is a strategy that aims to improve reasoning by dividing the task into smaller steps (Wei et al., 2022). A chain-of-thought prompt usually relies on exemplars, containing a prompt and a correct response. The example response is expressed as a series of steps that lead to a final output. This prompts the model to reason step-by-step.

For the purpose of our dataset, we adapt a **1-shot CoT** prompting strategy to the task of text generation: this prompt subtype consists of a human-written step-by-step instruction for the model. The instruction is based on an example of a human-written text from the dataset.

We also use **0-shot CoT** prompts (Kojima et al., 2022), which consist of adding *let's think step by step* to the baseline prompt to cause step-by-step reasoning.

Few-shot In-context learning (3-shot) (Brown et al., 2020): The few-shot prompt contains several examples of input-output pairs for a given task. Our few-shot prompts include three human-written examples from the dataset.

Style examples (Style): In Zhang et al. (2024), prompts with style guidelines have been effective in prompting LLMs to generate output that evades detection methods, and models have been prompted to simulate the styles of several famous writers. We adapt one of the prompts from Zhang et al. (2024) for our task. Specifically, we use the style example prompt, but with an example from the human-written part of our dataset. Even though using a text written by a famous author (e.g., Shakespeare) as a style example has been successful in preventing detection (Zhang et al., 2024), our prompt modification aims to create a more realistic setting to adapt to different datasets.

Self-refine (Madaan et al., 2023): Self-refinement prompts are prompts that use the LLM itself to critique and improve its own responses. The prompts of this kind are comprised of three stages: generating the output, generating feedback for the output and applying the feedback to the output. The second and third steps are repeated until a stopping condition is met. In this case, we base the stopping condition on the ability of the model itself to distinguish its own outputs from human-written text. To achieve this, we use evaluation prompts based on the GPT-4 evaluation prompt used in (Madaan et al., 2023). For the purpose of our task, we ask the model to decide which text sounds more human-written. The resulting set of prompts is made of the following stages:

- Initialization prompt: the same as our baseline zero-shot prompt.
- Feedback prompt: prompt used for generating feedback on how to make the text seem more human-written.
- Iterate prompt: used to get the next iteration if the model classifies its generated text as AI.
- Evaluate prompt: used to check whether the stopping condition (the model classifies the text as human-written) has been met.

A.2 Prompt templates

We demonstrate detailed prompts used in the paper in Table 6.

A.3 Detailed classification results

The results of the detailed in-domain accuracy are shown in Table 7.

A.4 Linguistic analysis

Lexical diversity. We choose Moving Average Type-Token Ratio to measure the lexical diversity of the texts, as it is independent of the text length (Covington and McFall, 2010). Additionally, we compare the texts in terms of the **number of unique words**, as this feature has been shown to be relevant in the previous research (Opara, 2024; Yildiz Durak et al., 2025).

Lexical density. Lexical density is the percentage of content words in the text. Machine-generated text is said to achieve higher lexical density (Savoy, 2020) than human-written text, which means that it contains fewer function words. The content words are adjectives, adverbs, nouns, and verbs.

Sentiment. AI-generated text has been shown to differ from human-written text in terms of sentiment: some authors have written of *positivity bias* present in AI-generated texts (Markowitz et al., 2024; Muñoz-Ortiz et al., 2024; Margolina and Kolmogorova, 2023), which means that text generated by large language models tends to contain more positive emotions compared to human-written texts. Other research suggests that human-written texts are more varied in terms of the richness of emotional content (Zanotto and Aroyehun, 2024). We use the TextBlob library to calculate the **polarity** and **subjectivity** scores for the texts in our dataset. **Polarity** is a score between -1 and 1, where -1 denotes a negative sentiment, while 1 denotes a positive sentiment. **Subjectivity** relates to the amount of personal opinion included in the text.

Readability. Readability refers to the ease of understanding a text. AI-generated text tends to be less readable than human-written text (Yadagiri et al., 2024; Markowitz et al., 2024; Mathews et al., 2024). We use the **Gunning fog index** and the **Flesch reading ease test** as measures of readability. The **Gunning fog index** is an estimated number of years needed to understand a given passage. The **Flesch reading ease test** is a metric of readability, in which texts that are easier to read receive a higher score. In the analysis of readability, we also include the **text length** in characters, as well as sentence length statistics. Machine-generated text tends to be less varied than human-written text in terms of sentence length (Desaire et al., 2023; Muñoz-Ortiz et al., 2024). Following (Desaire et al., 2023), we calculate the **average sentence length**, the **standard deviation from the average sentence length**, and the **number of very long** (35 words or more) and **very short** (10 words or less) **sentences** in each text.

Part-of-Speech (POS). We use a selection of metrics from StyloMetrix (Okulska et al., 2023) to compare the frequency of verbs, nouns, adjectives, pronouns, determiners, conjunctions and numerals across the different dataset types, models and prompts. The frequency of parts of speech is measured as the fraction of text covered by tokens representing a given part of speech. The frequency of POS has been shown to be different across different text genres, with non-fiction texts typically achieving a higher frequency of nouns than fiction texts (Mendhakar and S, 2023). A high frequency of verbs can be associated with more narrative texts, while a high frequency of adjectives is common for more descriptive texts. Additionally, previous research has discovered differences between human-written and AI-generated text in terms of frequency of certain POS (Georgiou, 2024). Therefore, POS analysis is relevant for gaining insights into the literary style of texts in our dataset.

Grammatical analysis. We use StyloMetrix (Okulska et al., 2023) metrics to compare texts in terms of grammatical categories related to verbs. In previous research, human-written texts have been shown to contain more passive voice than

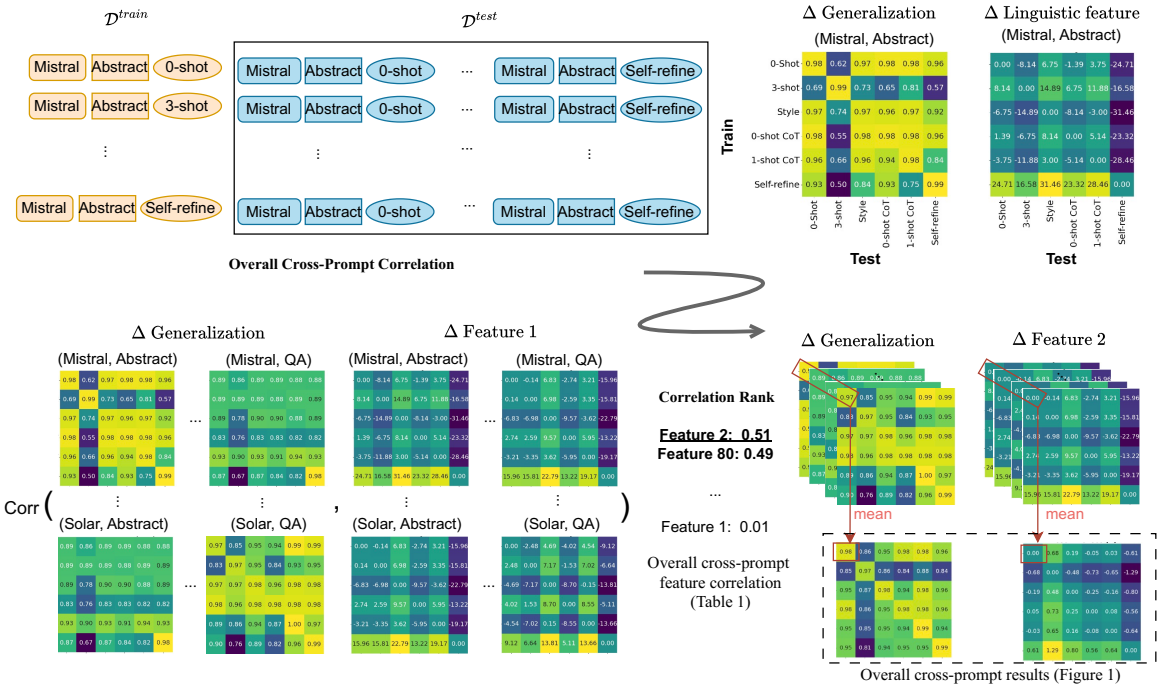


Figure 3: The workflow to get the overall cross-prompt generalization and feature shift results.

	DeBERTa-based					RoBERTa-based				
	Abstracts	News	Reviews	QA	ALL	Abstracts	News	Reviews	QA	ALL
0-shot	0.587	0.569	0.663	0.591	0.416	0.399	0.677	0.655	0.355	0.385

Table 4: The cross-model correlation between generation performance and the most correlated linguis- tic feature when evaluated on different datasets.

	DeBERTa-based 0-shot	RoBERTa-based 0-shot
Mistral 123B	0.310	0.350
Deepseek 70B	0.418	0.468
Llama 70B	0.389	0.374
Qwen 72B	0.425	0.308
Qwen 32B	0.426	0.418
Qwen 14B	0.487	0.545
Solar 22B	0.269	0.188
ALL	0.296	0.287

Table 5: The cross-dataset correlation between generation performance and the most correlated linguis- tic feature when evaluated on different models.

AI-generated texts (Georgiou, 2024). We measure the **incidence of passive and active voice** as the frequency of verbs in passive or active voice. We also compare the differences between the choice of tenses. The **frequency of past, present and future tenses** is measured as the fraction of the text covered by verbs in past, present and future tenses. For example, the incidence of past tenses is the number of verbs in past simple, past continuous, past perfect or past perfect continuous divided by the total number of words in the text.

Lexical analysis. As part of lexical analysis, we measure the **frequency of proper and personal names**, as well as the **frequency of adjectives and adverbs in positive, comparative and superlative degrees** and the **frequency of nouns in possessive case**. We use the StyloMetrix (Okulska et al., 2023) pronoun-related metrics to analyze the differences in the usage of pronouns in human-written and AI-generated text, as well as across the different prompt types, dataset types and models. We calculate not only the frequency of pronouns (POS_PRO), but also the frequency of specific personal or reflexive pronouns and the general frequency of certain types of pronouns (for example, the frequency of all first-person singular pronouns).

Prompt	Dataset	Prompt template
0-shot	Abstracts	"Write an abstract for an article in {category} with a title: \"{title}\". The abstract should be around {length} characters long."
	News	"Write a news article based on the following highlights:\n\"{highlights}\"\nYour article should be around {length} characters long."
	Reviews	"Write an Amazon review for the item \"{item_name}\" with a title \"{title}\" and a rating of {rating}. The review should be around {length} characters long."
	QA	"{question}\n Your answer should be around {length} characters long."
0-shot CoT	Abstracts	"Write an abstract for an article in {category} with a title: \"{title}\". Let's think step by step. Your answer should only include the abstract. The abstract should be around {length} characters long."
	News	"Write a news article based on the following highlights:\n\"{highlights}\"\nLet's think step by step. Your answer should only include the article. The article should be around {length} characters long."
	Reviews	"Write an Amazon review for the item \"{item_name}\" with a title \"{title}\" and a rating of {rating}. Let's think step by step. Your answer should only include the review. The review should be around {length} characters long."
	QA	"{question}\nLet's think step by step. Your answer should only include the answer to the question. The answer should be around {length} characters long."
1-shot CoT	Abstracts	"I want to write an abstract for an article in computer science. The article is titled \"ConQRet: Benchmarking Fine-Grained Evaluation of Retrieval Augmented Argumentation with LLM Judges\". \n 1. First, I introduce the context of the research and explain the motivation:\n\"Computational argumentation, which involves generating answers or summaries for controversial topics like abortion bans and vaccination, has become increasingly important in today's polarized environment. Sophisticated LLM capabilities offer the potential to provide nuanced, evidence-based answers to such questions through Retrieval-Augmented Argumentation (RAArg), leveraging real-world evidence for high-quality, grounded arguments. However, evaluating RAArg remains challenging, as human evaluation is costly and difficult for complex, lengthy answers on complicated topics. At the same time, re-using existing argumentation datasets is no longer sufficient, as they lack long, complex arguments and realistic evidence from potentially misleading sources, limiting holistic evaluation of retrieval effectiveness and argument quality.\"\nThen, I describe how I addressed the gaps in current research and give a detailed description of my methodology:\n\"To address these gaps, we investigate automated evaluation methods using multiple fine-grained LLM judges, providing better and more interpretable assessments than traditional single-score metrics and even previously reported human crowdsourcing. To validate the proposed techniques, we introduce ConQRet, a new benchmark featuring long and complex human-authored arguments on debated topics, grounded in real-world websites, allowing an exhaustive evaluation across retrieval effectiveness, argument quality, and groundedness. We validate our LLM Judges on a prior dataset and the new ConQRet benchmark.\"\nFinally, I describe the results and their implications for the research on this topic:\n\"Our proposed LLM Judges and the ConQRet benchmark can enable rapid progress in computational argumentation and can be naturally extended to other complex retrieval-augmented generation tasks.\"\nBased on the provided step-by-step instruction, write an abstract for an article in {category} titled \"{title}\"."
	News	"I want to write a news article about the following events:\nDarsh Patel, 22, was hiking with friends in the Apshawa Preserve in West Milford on Sunday when a bear started following them.\nThe group fled in different directions and when the four other hikers could not find Patel, they called police.\nPatel's body was found two hours later.\nThe 300-pound bear was circling the body and could not be scared away.\nIt was shot dead in accordance with Division of Fish and Wildlife guidelines.\nOn Saturday locals splitting wood filmed a bear rifling through their garbage.\n\n\nFirst, I introduce the event and its content to the readers:\nLocals in northern New Jersey believe they filmed a black bear hunting for food hours before a 22-year-old hiker was mauled to death in nearby woods at the weekend. Two men splitting wood on Saturday captured a video of a bear going through garbage just a few feet from where they were working, before scampering off into the woods, according to CNN. On Sunday, Darsh Patel, a senior majoring in information technology and informatics at Rutgers University, was found dead in Apshawa Preserve - about 45 miles northwest of New York City - with a 300-pound bear guarding his body. Officials say the attack was the first fatal bear-human encounter on record in New Jersey. Just a day after the footage was shot, a black bear mauled a 22-year-old student to death in the woods nearby.\nThen, I provide more details related to the event:\nPatel had been hiking with four friends in the 526-acre woods. The five friends noticed the bear beginning to follow them and ran, splitting up as they did. When they couldn't find Patel, they called police, who found his body about two hours later. The bear was about 30 yards from the body and circling, Department of Environmental Protection spokesman Larry Ragonese said, and wouldn't leave even after officers tried to scare it away by making loud noises and throwing sticks and stones. The male bear was killed with two rifle blasts and is being examined at a state lab for more clues as to why it may have pursued the group of five hikers.\nThen, I provide opinions on the event, quoting officials, experts or witnesses of the event:\nKelcey Burguess, principal biologist and leader of the state Division of Fish and Wildlife's black bear project, said the bear could have been predisposed to attack but more likely was looking for food. State and local officials stressed that bear attacks are rare even in a region of the state that may have as many as 2,400 bruins in its dense forests. \"This is a rare occurrence,\" West Milford police Chief Timothy Storbeck said, noting that his department receives six to 12 calls per week regarding bears, usually involving them breaking into trash cans. Locals: Residents in northern New Jersey often spot bears in and around their yards. There are as many 2,400 bruins in the area's dense forests, but until now had never been a fatal human-bear attack. Wildlife officials believe there is a current shortage of the acorns and berries that bears eat. The hikers had granola bars and water with them, Storbeck said. Officials don't believe the hikers provoked the bear but they may have showed their inexperience when they decided to run. The safest way to handle a bear encounter is to move slowly and not look the bear in the eye, DEP spokesman Larry Ragonese said. New Jersey Division of Fish and Wildlife guidelines direct law enforcement to euthanize \"Category I\" bears, which are deemed an \"immediate threat to human safety.\" NJ Advance Media reports that the New Jersey State Medical Examiner, the Fish and Wildlife Division of the state Department of Environmental Protection and the West Milford Police Department are looking into the circumstances of Patel's death.\nFinally, I conclude the report by highlighting the relevance of the event:\n\"Bear sightings are not unusual by any stretch in New Jersey,\" said Bob Considine, spokesperson for the Department of Environmental Protection. \"They have been seen in all 21 counties, although they're obviously most common in the northwest part of the state.\" Black bears rarely pose a threat to humans and often retreat when confronted. In 2006, a tabby cat scared a black bear up a tree in West Milford. The bear only climbed down and left after the cat's owner had called it back into the house.\n\nNow, I want to write an article on a different topic:\n{highlights}\nWrite the article, following the steps described above. Your answer should only include the article. The article should be around {length} characters long.",
	Reviews	"I want to write an Amazon review for an item called \"Haier RDG350AW 6.5 Cubic Foot Front Load Gas Dryer, White\". The rating will be 5.0.\nFirst, I choose a title for my review that describes my opinion and experience well. The title of my review will be:\n\"Very Affordable Dryer\".\nThen, I would state my initial experience with the product:\n\"I was a little worried about buying this cause it had some bad reviews, but it's a really great deal.\"\nThen I would describe the pros and cons of the product, expressing the reasons for my rating:\n\"It didn't touch my gas bill period. Yes larger loads take a while to dry, maybe up to 3 to 4 hours. It's just really energy efficent. Like I said my gas bill didn't budge with this being hooked up.\"\nNow, I want to write a review of an item called \"{item_name}\" and give it a rating of {rating}.\nFollow the directions described above to come up with the review. Your answer should only include the review and the title. The review should be one paragraph."
	QA	"If I was asked the question: \"When do the English state schools finish summer term and holiday begins?\",\nFirst, I would give a general overview of the answer:\"In the English school system, state schools run from early September to mid or late July of the following year.\"\nThen, I would go into more detail:\n\"The summer term (also known as the third term) runs from late April and finishes mid to late July with a week-long half term break in between. The summer holiday begins in late July and usually runs about six weeks long, ending in September.\"\nFinally, I would add nuance or additional context to my answer:\n\"The schools on the Trinity terms end their school year and begin summer holidays a few weeks earlier, at the end of June.\"\nFollowing the steps described above, answer the question: \"{question}\" in one paragraph."

Continued on next page

Prompt	Dataset	Prompt template
3-shot	Abstracts	"role" : "user", "content" : {title_1} "role" : "assistant", "content" : {human-written_abstract_1} "role" : "user", "content" : {title_2} "role" : "assistant", "content" : {human-written_abstract_2} "role" : "user", "content" : {title_3} "role" : "assistant", "content" : {human-written_abstract_3} "role" : "user", "content" : {title}
	News	"role" : "user", "content" : {highlights_1} "role" : "assistant", "content" : {human-written_article_1} "role" : "user", "content" : {highlights_2} "role" : "assistant", "content" : {human-written_article_2} "role" : "user", "content" : {highlights_3} "role" : "assistant", "content" : {human-written_article_3} "role" : "user", "content" : {highlights}
	Reviews	"role" : "user", "content" : {product_name_1} "role" : "assistant", "content" : {human-written_review_1} "role" : "user", "content" : {product_name_2} "role" : "assistant", "content" : {human-written_review_2} "role" : "user", "content" : {product_3} "role" : "assistant", "content" : {human-written_review_3} "role" : "user", "content" : {product}
	QA	"role" : "user", "content" : {question_1} "role" : "assistant", "content" : {human-written_answer_1} "role" : "user", "content" : {question_2} "role" : "assistant", "content" : {human-written_answer_2} "role" : "user", "content" : {question_3} "role" : "assistant", "content" : {human-written_answer_3} "role" : "user", "content" : {question}
Style	Abstracts	"As an academic paper writer, your task is to write an abstract of a research paper in a specific writing style. Write in the writing style of an example but ignore the content and topic of the example. You will be provided with the style example. You will be provided with the title for your abstract.\nStyle example: {example}\nTitle: {title}"
	News	"As a news article writer, your task is to write a news article in a specific writing style. Write in the writing style of an example but ignore the content and topic of the example. You will be provided with the style example. You will be provided with the summary of the topic for your article.\nStyle example: {example}\nSummary: {highlights}"
	Reviews	"As an Amazon review writer, your task is to write a review for an item in a specific writing style. Write in the writing style of an example but ignore the content and topic of the example. You will be provided with the style example. You will be provided with the name of the item you have to review.\nStyle example: {example}\nItem: {item_name}"
	QA	"As a highly intelligent question answering bot, your task is to answer questions in specific writing styles. Write in the writing style of an example but ignore the content and topic of the example. You will be provided with the style example. You will be provided with the question.\nStyle example: {example}\nQuestion: {question}"
Self-refine	Abstracts	1. "Write an abstract for an article in {category} with a title: \"{title}\". The abstract should be around {length} characters long." 2. "You will see an abstract for a scientific article. Your task is to provide feedback on how to make the text seem more human-like. Consider sentence length, sentence structure, vocabulary and readability.\nAbstract: {text}\nFeedback: " 3. "Based on the feedback, improve the text below:\nText: {text}\nFeedback: {feedback}." 4. "Which text sounds more human-written?\nText A: {text_a}\nText B: {text_b}\n\nPick your answer from [\"Text A\", \"Text B\", \"both\", \"neither\"]. Generate a short explanation for your choice first. Then, generate \"Text A seems more human-written\" or \"Text B seems more human-written\" or \"Both texts seem human-written\" or \"Neither of the texts sounds human-written\"
	News	1. "Write a news article based on the following highlights:\n\"{highlights}\"\nYour article should be around {length} characters long." 2. "You will see a news article. Your task is to provide feedback on how to make the text seem more human-like. Consider sentence length, sentence structure, vocabulary and readability.\nArticle: {text}\nFeedback: " 3. "Based on the feedback, improve the text below:\nText: {text}\nFeedback: {feedback}." 4. "Which text sounds more human-written?\nText A: {text_a}\nText B: {text_b}\n\nPick your answer from [\"Text A\", \"Text B\", \"both\", \"neither\"]. Generate a short explanation for your choice first. Then, generate \"Text A seems more human-written\" or \"Text B seems more human-written\" or \"Both texts seem human-written\" or \"Neither of the texts sounds human-written\"
	Reviews	1. "Write an Amazon review for the item \"{item_name}\" with a title \"{title}\" and a rating of {rating}. The review should be around {length} characters long." 2. "You will see an Amazon review. Your task is to provide feedback on how to make the text seem more human-like. Consider sentence length, sentence structure, vocabulary and readability.\nReview: {text}\nFeedback: " 3. "Based on the feedback, improve the text below:\nText: {text}\nFeedback: {feedback}." 4. "Which text sounds more human-written?\nText A: {text_a}\nText B: {text_b}\n\nPick your answer from [\"Text A\", \"Text B\", \"both\", \"neither\"]. Generate a short explanation for your choice first. Then, generate \"Text A seems more human-written\" or \"Text B seems more human-written\" or \"Both texts seem human-written\" or \"Neither of the texts sounds human-written\"

Continued on next page

Prompt	Dataset	Prompt template
	QA	1. "{question}\n Your answer should be around {length} characters long." 2. "You will see an answer to a question. Your task is to provide feedback on how to make the text seem more human-like. Consider sentence length, sentence structure, vocabulary and readability.\nAnswer: {text}\nFeedback: " 3. "Based on the feedback, improve the text below:\nText: {text}\nFeedback: {feedback}." 4. "Which text sounds more human-written?\nText A: {text_a}\nText B: {text_b}\n\nPick your answer from [\"Text A\", \"Text B\", \"both\", \"neither\"]. Generate a short explanation for your choice first. Then, generate \"Text A seems more human-written\" or \"Text B seems more human-written\" or \"Both texts seem human-written\" or \"Neither of the texts sounds human-written\"

Table 6: Prompt details.

Detector	Model	Dataset	Prompt type					
			0-Shot	3-Shot CoT	1-Shot CoT	Style	3-Shot	Self-refine
DeBERTa	Llama3.3 70b	Abstracts	0.988	0.9875	0.993	0.9795	0.9945	0.9865
		News	0.999	0.999	0.9985	0.9775	0.7935	1.0
		Reviews	0.984	0.988	0.9995	0.9615	0.9805	0.9875
		QA	0.8866	0.8966	0.9989	0.9604	0.8233	0.9509
	Qwen 14b	Abstracts	0.998	0.991	0.994	0.998	0.999	0.999
		News	1.0	0.9995	0.9995	1.0	1.0	0.999
		Reviews	0.9985	0.996	1.0	1.0	0.9965	0.9985
		QA	0.98	0.9662	0.9916	0.9926	0.9668	0.9736
	Qwen 32b	Abstracts	0.9955	0.9915	0.994	0.9965	1.0	0.999
		News	1.0	0.999	0.9995	1.0	1.0	1.0
		Reviews	0.999	0.9955	1.0	0.998	0.9985	0.9955
		QA	0.9763	0.981	0.9852	0.9942	0.9626	0.9736
	Qwen 72b	Abstracts	0.988	0.9845	0.9845	0.991	0.9955	0.991
		News	0.999	0.9995	0.9995	1.0	0.998	0.999
		Reviews	0.995	0.9975	0.9995	0.9945	0.9965	0.989
		QA	0.9699	0.9583	0.9889	0.942	0.8903	0.9662
	Solar 22b	Abstracts	0.9855	0.9915	0.992	0.989	0.9985	0.9975
		News	1.0	0.9995	0.999	1.0	0.9994	0.9995
		Reviews	0.998	0.998	0.9995	0.9975	0.9965	0.999
		QA	0.9626	0.9852	0.9947	0.9773	0.9705	0.9926
	Mistral 123b	Abstracts	0.986	0.989	0.9895	0.984	0.9895	0.9965
		News	0.9985	0.998	0.9995	0.9975	0.962	1.0
		Reviews	0.9905	0.9895	0.9945	0.993	0.9765	0.994
		QA	0.9014	0.8333	0.9299	0.8819	0.9167	0.9905
	Deepseek 70b	Abstracts	0.9885	0.989	0.992	0.9825	0.9965	0.9885
		News	0.998	0.9985	0.998	0.9985	0.999	1.0
		Reviews	0.9945	0.994	0.9945	0.99	0.9955	0.9955
		QA	0.9341	0.9019	0.9483	0.8903	0.9151	0.954
RoBERTa	Llama3.3 70b	Abstracts	0.9865	0.9765	0.9925	0.9295	0.9795	0.9835
		News	0.99	0.996	0.988	0.901	0.782	0.9965
		Reviews	0.987	0.9925	0.9995	0.938	0.99	0.9915
		QA	0.8481	0.8586	1.0	0.9578	0.8291	0.9451
	Qwen 14b	Abstracts	0.9945	0.993	0.995	0.998	0.999	0.9965
		News	0.999	1.0	0.9995	1.0	0.9985	1.0
		Reviews	0.9995	0.999	0.9995	1.0	0.998	1.0
		QA	0.9826	0.9652	0.9784	0.9852	0.9541	0.9662
	Qwen 32b	Abstracts	0.993	0.989	0.9935	0.993	0.999	0.9985
		News	1.0	1.0	1.0	0.999	0.9985	0.9975
		Reviews	0.998	0.9995	0.9965	0.9985	0.9975	0.9955
		QA	0.9789	0.9789	0.981	0.9873	0.9209	0.9789
	Qwen 72b	Abstracts	0.986	0.9825	0.9875	0.979	0.9945	0.982
		News	0.998	1.0	0.999	0.9995	0.999	0.999
		Reviews	0.996	0.9955	1.0	0.9945	0.998	0.9915
		QA	0.981	0.9399	0.9831	0.9504	0.8623	0.9699
	Solar 22b	Abstracts	0.9825	0.987	0.9795	0.9825	0.995	0.9995
		News	0.9975	0.995	0.994	0.9995	1.0	0.998
		Reviews	0.9975	0.9975	1.0	0.997	0.9935	1.0
		QA	0.9742	0.9831	0.9963	0.9821	0.9747	0.9937
	Mistral 123b	Abstracts	0.9835	0.982	0.982	0.9665	0.987	0.9925
		News	0.993	0.996	0.998	0.992	0.9825	0.999
		Reviews	0.9955	0.981	0.9915	0.991	0.9805	0.9935
		QA	0.8877	0.8318	0.9383	0.9024	0.8866	0.9826
	Deepseek 70b	Abstracts	0.9835	0.986	0.988	0.9655	0.996	0.9825
		News	0.997	0.986	0.9995	0.998	0.9985	0.998
		Reviews	0.994	0.995	0.9925	0.9925	0.9965	0.9965
		QA	0.8813	0.8734	0.9314	0.8645	0.8513	0.9185

Table 7: Accuracy of the fine-tuned detectors on in-domain data.

Linsuitic Feature	Feature Metric	DeBERTa-based C-P	C-M	C-D	RoBERTa-based C-P	C-M	C-D
Lexical diversity	MATTR	0.035	**0.312**	0.097	0.031	**0.326**	0.100
	L MATTR	0.038	**0.313**	0.087	0.021	**0.349**	0.079
	Unique words	0.004	0.118	0.116	0.044	**0.163**	0.074
Lexical density	Number of function words	0.008	**0.148**	0.104	0.042	**0.282**	0.184
Readability	FLESCH	0.006	**0.254**	**0.198**	0.006	**0.330**	**0.251**
	Sentence length	**0.069**	**0.242**	0.093	**0.073**	**0.288**	0.122
	Long sentences	0.020	**0.161**	0.099	0.061	**0.196**	0.066
	Short sentences	**0.072**	**0.223**	0.040	0.041	**0.290**	0.043
	Sentence length std	0.034	0.104	0.004	0.058	**0.210**	0.064
	Length in characters	0.002	0.117	0.083	0.045	**0.191**	0.012
Sentiment	Polarity	0.025	0.125	**0.224**	**0.106**	**0.261**	**0.187**
	Subjectivity	0.019	0.064	0.001	0.058	0.134	0.054
POS	Verbs	0.050	0.080	0.005	**0.072**	0.042	0.041
	Nouns	0.047	**0.307**	0.164	**0.066**	**0.322**	0.180
	Adjectives	0.017	**0.147**	0.038	0.037	0.014	0.092
	Adverbs	0.005	**0.201**	**0.217**	0.016	**0.213**	**0.195**
	Determiners	0.010	**0.215**	**0.218**	0.044	**0.286**	**0.241**
	Interjections	0.031	0.111	0.157	0.032	**0.169**	0.096
	Conjunctions	0.042	0.061	0.158	0.020	**0.190**	0.098
	Particles	0.027	0.012	0.026	0.006	**0.212**	0.047
	Numerals	0.008	**0.270**	0.163	0.003	**0.265**	**0.202**
	Pronouns	0.013	0.064	0.057	0.051	**0.159**	0.007
Lexical	Content words	0.023	0.128	0.064	0.039	**0.278**	0.148
	Function words	0.005	0.045	0.086	0.035	**0.251**	0.164
	Content words types	0.041	0.100	**0.195**	0.001	**0.239**	**0.238**
	Function words types	0.012	0.037	0.137	0.048	0.006	**0.194**
	Proper names	**0.070**	**0.360**	0.013	0.022	0.134	0.053
	Nouns in possessive case	0.049	0.062	0.102	0.002	0.081	0.167
	Adjectives in positive degree	0.012	**0.147**	0.039	0.043	0.018	0.097
	Adverbs in positive degree	0.003	**0.188**	**0.191**	0.025	**0.188**	0.184
	Adverbs in comparative degree	0.006	**0.191**	**0.225**	0.016	**0.206**	**0.198**
	Adverbs in superlative degree	0	**0.191**	**0.226**	0.022	**0.204**	**0.197**
	'I' pronoun	0	**0.202**	0.134	0.033	**0.327**	0.085
	'He' pronoun	0.045	**0.173**	0.045	0.031	**0.191**	0.059
	'She' pronoun	**0.086**	**0.176**	0.004	**0.080**	**0.316**	0.020
	'It' pronoun	**0.067**	0.092	0.167	**0.067**	**0.203**	**0.199**
	'You' pronoun	0.015	**0.160**	0.085	0.011	0.113	0.028
	'They' pronoun	0.015	0.072	0.101	0.018	0.128	0.128
	'Me' pronoun	0.044	0.096	0.041	0.025	0.077	0.032
	'You' object pronoun	0.016	0.077	0.153	0.050	**0.161**	0.100
	'Him' object pronoun	0.041	**0.161**	0.069	0.042	0.034	0.097
	'Her' object pronoun	0.040	0.080	0.072	**0.072**	**0.203**	0.031
	'Us' pronoun	0.041	**0.215**	0.101	0.012	0.086	0.137
	'Them' pronoun	0.052	0.030	0.073	**0.080**	0.080	0.012
	'My' pronoun	0.021	**0.214**	0.127	0.053	**0.341**	0.109
	'Your' pronoun	0.007	**0.250**	0.085	0.009	**0.254**	0.096
	'His' pronoun	0.005	0.078	0.086	0.023	0.059	0.152
	'Her' possessive pronoun	0.034	0.004	0.053	**0.063**	0.102	0.129
	'Its' possessive pronoun	**0.089**	**0.293**	**0.245**	**0.079**	**0.323**	**0.244**
	'Their' possessive pronoun	**0.065**	**0.237**	0.121	0.015	**0.228**	0.132
	'Yours' pronoun	0.026	0.001	**0.221**	0.009	0.103	0.163
	'Theirs' pronoun	0.004	0.011	**0.255**	0.008	0.084	**0.237**
	'Hers' pronoun	0.061	0.006	0.067	0.031	0.130	0.021
	'Ours' possessive pronoun	0.003	0.106	0.021	0.010	**0.186**	0.005
	'Myself' pronoun	0.023	**0.157**	0.088	0.017	**0.297**	0.073
	'Himself' pronoun	0.012	**0.332**	0.052	0.027	**0.232**	0.043
	'Herself' pronoun	0.036	**0.248**	0.025	0.032	**0.227**	0.001
	'Itself' pronoun	0.032	0.132	**0.221**	0.055	0.027	**0.204**
	'Ourselves' pronoun	0	0.103	0.073	0.018	**0.269**	0.068
	'Yourselves' pronoun	0.018	**0.177**	0.108	0.005	**0.194**	0.083
	'Themselves' pronoun	**0.064**	**0.230**	0.019	**0.100**	**0.278**	0.015
	First person singular pronouns	0	**0.202**	0.134	0.033	**0.327**	0.085
	Second person pronouns	0.014	**0.204**	0.056	0.011	**0.192**	0.005
	Third person singular pronouns	0.047	0.014	0.093	0.031	**0.198**	0.140
	Third person plural pronouns	0.050	**0.182**	0.117	0.039	0.129	0.157
General	Incidence of verbs in infinitive	0.037	0.050	0.020	**0.095**	**0.153**	0.085

Table 8: The correlation between generalization performance with different linguistic features. The **significant correlation is bolded**.

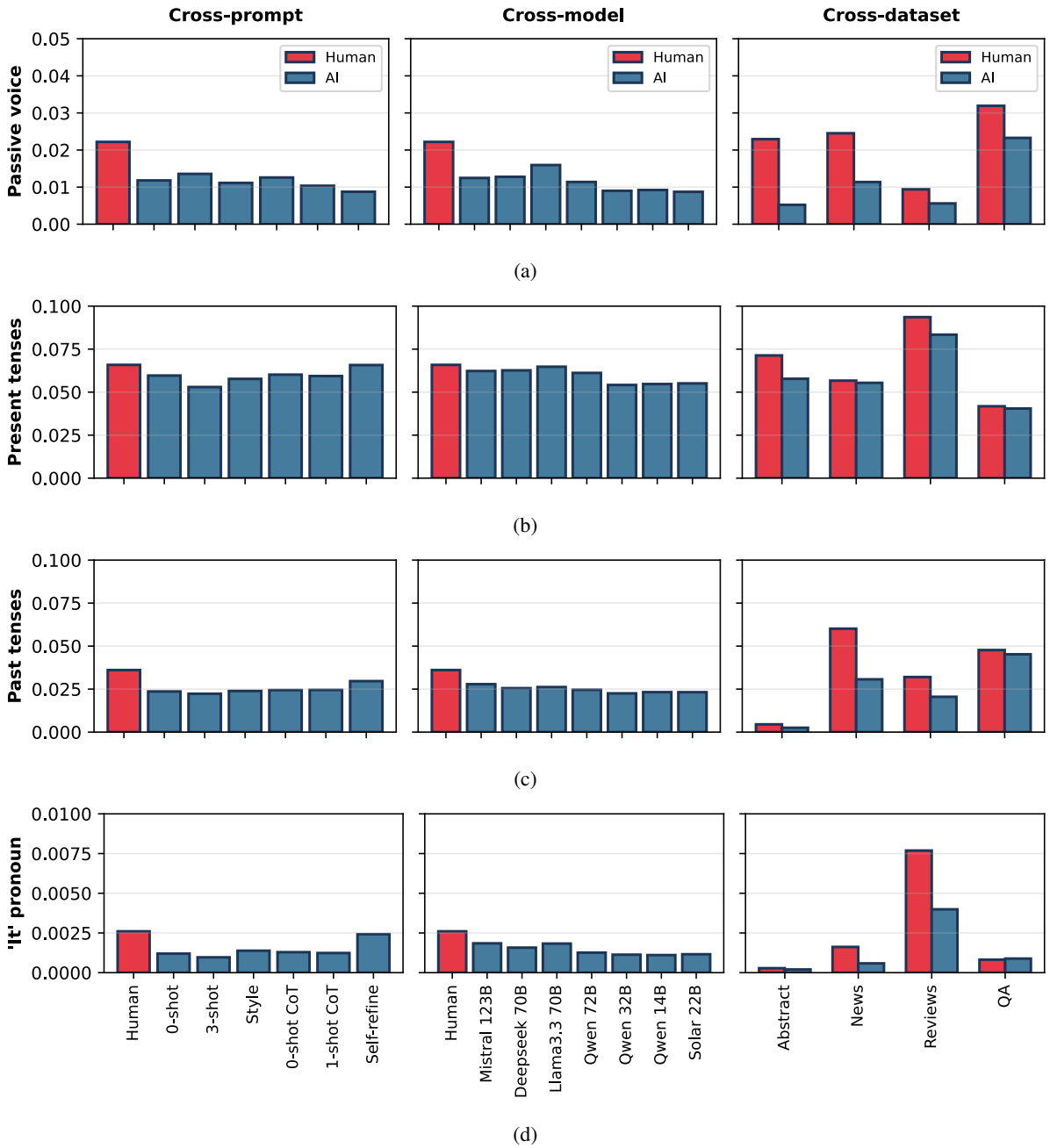


Figure 4: The more detailed comparison of different linguistic features across different configurations as well as the differences between human and AI text. We present the features that have the strongest correlations for different dimensions (underlined features in Table 4 in the main paper).

Dataset: Reviews
Item name: Homasy Amplified HDTV Antenna Indoor with Amplifier - 50 Mile Range, Built in Amplifier Signal Booster
Title: Works Well
Rating: 5
Review length in characters: 859

Human, Short sentences: 5

We got rid of satellite TV a couple of years ago. We have two TVs in the house, but only one HD over the air tuner. We're looking to get another tuner, so with a second tuner, we need a second antenna. We tested this antenna with our existing tuner. It requires a USB port to be plugged in. We didn't attach it to our wall since we plan to use it in a different room and this was just a test. My husband held it up and was able to get a clear picture with it. Our living room has a two story ceiling. Our other antenna is up pretty high on the wall. The higher it was, the better the reception. The antenna is comparable to our other one in quality and performance. Now that we have another good antenna, we will go forward with purchasing a new tuner. I received this for free in exchange for my honest & unbiased review. My thoughts & opinions are my own.

Llama70B, 0-shot, Short sentences: 2

I'm impressed with this antenna! It was easy to set up and I get great reception on all my local channels. The amplifier really makes a difference, picking up stations that my old antenna couldn't. I live about 30 miles from the broadcast towers and it works flawlessly. Highly recommend!

Llama70B, 3-shot, Short sentences: 1

I was a bit skeptical about buying an indoor antenna, but I'm glad I took the chance. The Homasy Amplified HDTV Antenna has exceeded my expectations. Setup was easy, just plug it in and scan for channels. I was able to receive over 30 channels, including major networks like ABC, CBS, NBC, FOX, and PBS, as well as some local stations. The picture quality is excellent, with no pixelation or freezing. The amplifier seems to make a big difference, as I was able to pick up channels that my old non-amplified antenna couldn't. The range is also impressive, I'm about 40 miles from the broadcast towers and I get a strong signal. The antenna is compact and lightweight, making it easy to place anywhere in the room. Overall, I'm very happy with this purchase and would recommend it to anyone looking for a reliable and affordable indoor HDTV antenna. Pros: Easy setup Excellent picture quality Good range (up to 50 miles) Compact and lightweight design Affordable price Cons: None that I can think of! Note: As with any antenna, the number of channels you'll receive will depend on your location and the strength of the signals in your area. But overall, I'm very satisfied with this product and would definitely recommend it.

Llama70B, Style, Short sentences: 2

I bought the "Ḿomasy Amplified HDTV Antenna Indoor with Amplifier"f́or my 75 year old dad because he likes to watch his local news and sports channels without relying on cable. This antenna makes it so he can. The thing I donf́ like? That you have to deal with all the extra cables the amplifier and signal booster are great, but they add clutter. So it would be nice to be able to just have a more streamlined setup. The extra cables just get tangled up behind the TV.

Llama70B, 0-shot CoT, Short sentences: 2

Works Well I'm impressed with this antenna, it works well and picks up all local channels clearly. The amplifier helps boost weak signals, giving me more channel options. Easy to set up, compact design, and great picture quality. I can receive channels within a 50-mile range without any issues. Highly recommended!

Llama70B, 1-shot CoT, Short sentences: 0

"Excellent Reception at a Great Price" I was a bit skeptical about purchasing this antenna due to mixed reviews, but it has exceeded my expectations in every way; itś a fantastic value for the money. The pros definitely outweigh the cons, as it picks up a ton of channels with crystal clear reception, and the amplifier makes a huge difference, allowing me to receive stations from over 40 miles away with perfect clarity. Yes, you may need to adjust the antennaś position for optimal signal strength, but thatś a small trade-off for the quality and number of channels you get, and the fact that itś so easy to set up and use is just icing on the cake.

Llama70B, self-refine, Short sentences: 0

I was a bit skeptical about whether this antenna would be able to pull in clear signals from 30 miles away, but I've been blown away by its performance. Setting it up was a breeze, and the built-in amplifier has made a huge difference in terms of picture quality I can now get all my local channels in crystal-clear HD, which is a big upgrade from the fuzzy reception I was getting before. What really impresses me, though, is that it's picked up a few channels I didn't even know existed, including some great local stations that I'd never been able to tune into before. Overall, I'm thoroughly impressed with this antenna and would definitely recommend it to anyone looking to boost their TV reception!

Table 9: Case study of human and AI texts for the cross-prompt generalization setting shown in Figure 2 (a).

Dataset: Abstracts
Title: Satellites and large doping- and temperature-dependence of electronic properties in hole-doped BaFe2As2
Category: Strongly Correlated Electrons
Review length in characters: 1037

Human, "We" pronoun: 0.013698

Over the last years, superconductivity has been discovered in several families of iron-based compounds. Despite intense research, even basic electronic properties of these materials, such as Fermi surfaces, effective electron masses, or orbital characters are still subject to debate. Here, we address an issue that has not been considered before, namely the consequences of dynamical screening of the Coulomb interactions among Fe-d electrons. We demonstrate its importance not only for correlation satellites seen in photoemission spectroscopy, but also for the low-energy electronic structure. From our analysis of the normal phase of BaFe2As2 emerges the picture of a strongly correlated compound with strongly doping- and temperature-dependent properties. In the hole overdoped regime, an incoherent metal is found, while Fermi-liquid behavior is recovered in the undoped compound. At optimal doping, the self-energy exhibits an unusual square-root energy dependence which leads to strong band renormalizations near the Fermi level.

Deepseek, 0-shot, "We" pronoun: 0.011628

This study investigates the electronic properties of hole-doped $BaFe_2As_2$ using angle-resolved photoemission spectroscopy (ARPES). We observe prominent satellite bands whose intensity varies significantly with doping and temperature. These satellites reveal intricate many-body interactions, suggesting coupling of electrons with lattice vibrations or spin fluctuations. Our findings indicate that doping-induced modifications to the Fermi surface and temperature-dependent renormalization play crucial roles in shaping the electronic structure. The results provide insights into the unconventional superconductivity of iron-based compounds, highlighting the importance of electron interactions in determining their unique properties.

Deepseek, 3-shot, "We" pronoun: 0.000000

The study investigates the electronic properties of hole-doped $BaFe_2As_2$, focusing on satellite features observed in spectroscopic measurements and their sensitivity to doping and temperature variations. Here's a structured summary: Material Context: Hole-Doping: Introduces excess positive charge carriers, altering the electronic structure significantly. Iron-Based Superconductor: Part of a class of materials known for high critical temperatures and complex electronic interactions. Key Observations: Satellite Features: These are secondary peaks near primary spectral lines, likely arising from many-body interactions such as electron-boson coupling. Doping Dependence: The intensity or prominence of these satellites varies with doping levels, indicating a correlation between carrier concentration and interaction strength. Temperature Dependence: Thermal effects modify the satellites, suggesting that phonon interactions or other thermally induced processes play a role. Implications for Superconductivity: Correlation with Superconducting Mechanisms: The presence and behavior of satellites may hint at the strength of pairing interactions crucial for superconductivity. Potential Optimization: Understanding how doping and temperature influence these features could guide strategies to enhance critical temperature (Tc). Methodology: Likely employs advanced spectroscopic techniques such as ARPES, supported by theoretical modeling to interpret the data accurately. Research Implications: Suggests a nuanced relationship between satellite features and superconductivity, highlighting the complexity of electronic interactions. Encourages further exploration across different superconducting materials to identify universal principles or unique mechanisms. This study contributes valuable insights into the intricate electronic dynamics of iron-based superconductors, offering potential pathways for advancing our understanding and engineering of high-Tc materials.

Deepseek, Style, "We" pronoun: 0.013605

We report a comprehensive investigation into the electronic properties of hole-doped $BaFe_2As_2$, focusing on the significant influence of doping level and temperature variations. Our studies uncover a pronounced dopingand temperature-dependent behavior in the material's electronic characteristics, with particular emphasis on the emergence of satellite features in spectroscopic measurements. By employing advanced experimental techniques combined with theoretical modeling, we demonstrate how these satellite structures correlate with changes in carrier concentration and lattice dynamics. A detailed analysis reveals that the interplay between doping-induced charge carriers and thermal fluctuations plays a critical role in modulating the material's conductivity and band structure. This work provides fundamental insights into the underlying mechanisms governing the electronic response of doped iron-based compounds, offering valuable guidance for tailoring their properties for potential technological applications. Our findings underscore the importance of considering both doping and temperature effects when designing materials for optimized performance in electronic devices.

Deepseek, 0-shot CoT, "We" pronoun: 0.006993

This study investigates the electronic properties of hole-doped $BaFe_2As_2$ using high-resolution angle-resolved photoemission spectroscopy (ARPES). We observe prominent satellite features in the spectral function, whose intensity and dispersion exhibit strong dopingand temperature-dependent behavior. These satellites, indicative of many-body interactions, become more pronounced with increased doping and lower temperatures, particularly near the Fermi level. Our analysis reveals a significant evolution of the electronic structure across the doping phase diagram, highlighting non-trivial renormalization effects and deviations from conventional quasiparticle behavior. The observed trends suggest that strong electron-electron correlations play a critical role in shaping the electronic properties of $BaFe_2As_2$. Furthermore, the dopingand temperature-dependent saturation of these satellites provides insights into the interplay between correlation effects and superconducting tendencies in iron-based superconductors. These findings challenge existing theoretical frameworks and underscore the importance of incorporating strong correlation physics in modeling the electronic states of doped iron pnictides.

Deepseek, 1-shot CoT, "We" pronoun: 0.007605

In recent years, the study of strongly correlated electron systems has gained significant attention due to their unique electronic properties, particularly in materials exhibiting unconventional superconductivity. Hole-doped $BaFe_2As_2$, a member of the iron-based superconductor family, has emerged as a model system for investigating the interplay between electronic correlations, doping, and temperature-dependent phenomena. Understanding the intricate relationships between these factors is crucial for advancing our knowledge of quantum materials and their potential applications. Despite extensive research, the dopingand temperature-dependent evolution of electronic properties in $BaFe_2As_2$ remains poorly understood, particularly in the underdoped regime where strong correlations dominate. To address these challenges, we employ advanced spectroscopic techniques to probe the electronic structure of hole-doped $BaFe_2As_2$ across a wide range of doping levels and temperatures. Our measurements reveal pronounced satellite features in the electronic spectra, which are highly sensitive to both doping and temperature. These observations suggest the presence of strong many-body interactions, whose strength varies significantly with doping and thermal fluctuations. By systematically analyzing these dependencies, we uncover universal scaling behaviors that shed light on the underlying mechanisms governing the electronic response in this class of materials. Our findings demonstrate that the dopingand temperature-dependent evolution of electronic properties in $BaFe_2As_2$ cannot be explained by conventional theories alone, pointing to the importance of incorporating strong correlation effects. Furthermore, the emergence of robust satellite structures highlights the critical role of short-range Coulomb interactions in shaping the electronic landscape. These insights not only advance our understanding of the phase diagram of iron-based superconductors but also provide a framework for exploring similar phenomena in other strongly correlated electron systems.

Deepseek, self-refine, "We" pronoun: 0.012658

Refined Hole-doped $BaFe_2As_2$ reveals fascinating insights into the interplay between strong electron correlations and unconventional superconductivity. Through advanced techniques like ARPES and transport measurements, we explored how doping and temperature affect its electronic properties. Notably, we identified significant satellite features in the valence band, which change markedly with doping and temperature, reflecting substantial particle interactions a hallmark of many-body physics. These changes align with shifts in the Fermi surface and quasiparticle behavior, underscoring how correlations redefine the material's electronic landscape. Moreover, our measurements showed variations in resistivity and Hall coefficients tied to doping and temperature, hinting at intricate interactions between charge carriers and spin fluctuations. Intriguingly, a nonlinear pattern emerged across the doping phase diagram, suggesting competing orders near quantum criticality. These findings highlight the pivotal role of strong correlations in iron pnictides and offer insights into exotic phases in similar materials. Such understanding could pave the way for innovative device technologies, bridging cutting-edge science with practical applications.

Table 10: Case study of human and AI texts for the cross-prompt generalization setting shown in Figure 2 (b).