Reconfidencing LLMs from the Grouping Loss Perspective

Lihu Chen[1], **Alexandre Perez-Lebel**[1], **Fabian M. Suchanek**[2], **Gaël Varoquaux**[1]
[1] Soda, Inria Saclay, France
[2] LTCI, Télécom Paris, Institut Polytechnique de Paris, France
`{lihu.chen, alexandre.perez, gael.varoquaux}@inria.fr`
`{fabian.suchanek}@telecom-paris.fr`

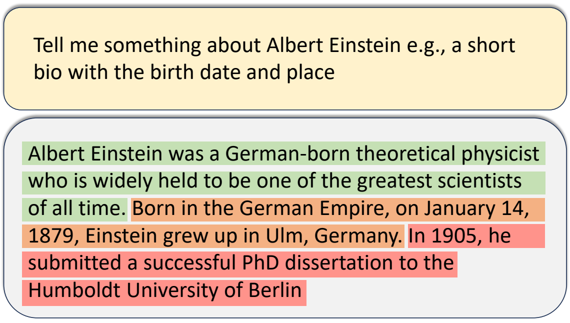
Abstract

Large Language Models (LLMs), such as GPT and LLaMA, are susceptible to generating hallucinated answers in a confident tone. While previous efforts to elicit and calibrate confidence scores have shown some success, they often overlook biases towards certain groups, such as specific nationalities. Existing calibration methods typically focus on average performance, failing to address this disparity. In our study, we demonstrate that the concept of grouping loss is an effective metric for understanding and correcting the heterogeneity in confidence levels. We introduce a novel evaluation dataset, derived from a knowledge base, specifically designed to assess the confidence scores of LLM responses across different groups. Our experimental results highlight significant variations in confidence, which are accurately captured by grouping loss. To tackle this issue, we propose a new method to calibrate the confidence scores of LLMs by considering different groups, a process we term *reconfidencing*. Our findings indicate that this approach effectively mitigates biases against minority groups, contributing to the development of fairer LLMs. The code is available at `https://github.com/tigerchen52/reconfidencing_llms`

1 Introduction

While Large Language Models (LLMs) such as ChatGPT (OpenAI, 2022) and LLaMA (Touvron et al., 2023) can generate responses that are fluent and plausible, they can also provide incorrect and untruthful information in a confident and compelling tone. This phenomenon, often called hallucination, poses a notable challenge to their use (Ji et al., 2023; Baan et al., 2023).

In response, extensive research has focused on estimating the confidence (or uncertainty) of LLM answers (Huang et al., 2023; Zhang et al., 2023). Through expressions of confidence levels, we know to what degree to trust a statement rather than blindly believing. Figure 1 illustrates an ideal user experience, where LLMs document sentence-level confidence in their answers. Methods of estimating confidence can be categorized into two groups: *White-box* and *Black-box* methods. White-box methods require access to internal states (Azaria and Mitchell, 2023) or model logits (Lin et al., 2022a) while Black-box methods rely solely on text responses to obtain confidence scores. In cases where the LLM allows only restricted access to internal states (e.g., ChatGPT), black-box methods are more suitable. These methods establish confidence scores by analyzing the consistency of multiple answers to a single query (Kuhn et al., 2022; Manakul et al., 2023) or by creating specific prompts to capture expressed confidence scores (Zhou et al., 2023; Xiong et al., 2023; Tian et al., 2023).



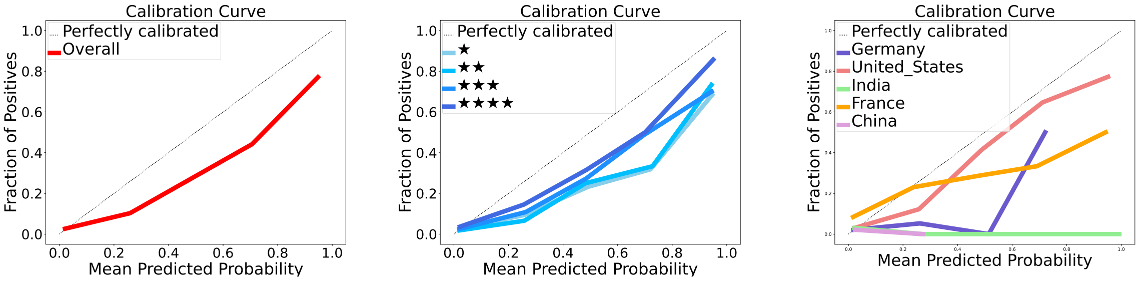
Tell me something about Albert Einstein e.g., a short bio with the birth date and place

Albert Einstein was a German-born theoretical physicist who is widely held to be one of the greatest scientists of all time. Born in the German Empire, on January 14, 1879, Einstein grew up in Ulm, Germany. In 1905, he submitted a successful PhD dissertation to the Humboldt University of Berlin

Figure 1: **Desired user experience** – An illustration of our goals of eliciting confidence levels in LLMs. High confidence scores are represented in green, while red indicates a higher likelihood of encountering hallucinated sentences.

Although some methods use calibration to adjust the predictions of a model to better match the true probabilities (Hendrycks et al., 2021; Gawlikowski et al., 2021; Mielke et al., 2022; Tian et al., 2023), these approaches predominantly concentrate on average performance metrics, often neglecting the het-

Findings of the Association for Computational Linguistics: EMNLP 2024, pages 1567–1581
November 12-16, 2024 ©2024 Association for Computational Linguistics



(a) Overall

(b) Popularity

(c) Nationality

Figure 2: Calibration curves of the `Birth_Date` relation. The LLM here is Mistral-7B (MistralAI, 2023), and we use SelfCheckGPT (Manakul et al., 2023) to compute confidence scores. An increased number of ⋆ symbols signifies a sub-group containing more popular samples.

erogeneity among different groups. Consequently, calibration alone proves inadequate. Even when a calibration technique attains optimal average accuracy, the calibrated scores can still markedly deviate from the true posterior probabilities for specific groups of queries – a phenomenon known as the grouping loss (Kull and Flach, 2015; Perez-Lebel et al., 2023). As an example, let us consider a query that asks for the birth dates of people, as in *"What is the birth date of Albert Einstein?"*. We submitted this query for 5K people to an LLM (Mistral-7B, MistralAI, 2023), and generated a confidence score for each answer with a consistency-based method (SelfCheckGPT, Manakul et al., 2023). In a classic calibration analysis, we grouped the answers into buckets by their confidence score, and computed the observed ratio of correct answers in each bucket. Figure 2a shows the corresponding calibration curve for all test samples. The curve is close to the diagonal, which means that the confidence score is close to the true ratio of correct answers in each bucket. This picture changes a bit when we split our data into popular and less popular persons based on the backlink numbers. As shown in Figure 2b, answers on more popular entities tend to be better calibrated than answers on long-tail entities. The picture is even more dramatic when we split the people by nationality (Figure 2c): While the calibration is satisfactory for American and French individuals, it performs dismally for almost all Indian and Chinese people. This illustrates grouping loss: a model's calibration error may be small overall, but can be catastrophically large for certain sub-groups. A well-calibrated LLM might be biased, generating with high confidence untruthful information about a particular race, gender, etc.

In this paper, we conduct a systematic study to measure the error of the confidence estimations. We create a new dataset that enables evaluating the quality of confidence scores for different types of groups. Our dataset consists of questions about entities (people, locations, etc.) and the ground truth from the YAGO knowledge base (Suchanek et al., 2024). In addition, our dataset contains features of the entities, such as popularity and nationality, which allows us to study sub-groups of entities. We evaluate two recently proposed methods for deriving confidence levels: *SelfCheckGPT* (Manakul et al., 2023) and *Just Ask for Calibration* (Tian et al., 2023). To identify grouping loss, we use both user-defined and latent groups. User-defined groups rely on features (which may be hand-crafted) such as popularity and nationality, while latent groups are automatically identified by decision trees (Perez-Lebel et al., 2023). Experiments reveal that models like Mistral and LLaMA tend to be overly confident across all questions. In addition, they are more confident on some queries than others: they display grouping loss. To improve confidence scores, we propose an approach to adjust LLMs, tackling both calibration and grouping loss. The core idea is to calibrate the confidence score for each sub-group separately, a method we term *reconfidencing*. Experimental results show that our refined solution has a better performance in terms of Brier score and grouping loss.

In summary, our contributions are threefold:

- We introduce a new framework and dataset to analyze the capability of LLMs to elicit confidence scores for different groups
- We prove the existence of the grouping loss in LLMs and compare the heterogeneity of confidence errors on both user-defined groups and implicit groups
- We propose a refined way to reconfidence

LLMs from a group-level perspective, which can reduce discrimination of minority groups and lead to fairer LLMs.

2 Related Work

2.1 Confidence Elicitation in LLMs

To alleviate the hallucination phenomenon, some methods attempt to elicit confidence (or uncertainty) scores for the generated answers of LLMs (Ji et al., 2023; Zhang et al., 2023; Huang et al., 2023). These efforts can be roughly categorized into two groups: *White-box* and *Black-box* methods. White-box methods need access to internal states or token logits while Black-box methods use only textual responses to compute confidence scores.

There are three primary white-box ways to encourage LLMs to express uncertainty in a human-like manner: Verbalized Probability, Internal State, and Token Logit. The goal of *verbalized probability* is to teach models to convey its degree of certainty, as in *I'm 90% sure that it is...*. The models are fine-tuned on particular tasks (Lin et al., 2022a) to elicit probabilistic responses. The *internal state method* builds a classifier to detect the truthfulness of a statement, which receives as input the activation values of the hidden layers of an LLM (Azaria and Mitchell, 2023). The *token logit method* evaluates the probability distribution of the words in the answer. At each step, LLMs produce a probability distribution across the entire vocabulary. Analyzing the distribution allows us to compute corresponding entropy values, which serve as indicators of confidence (Fu et al., 2023; Manakul et al., 2023). Generally, factual statements tend to feature tokens with higher likelihood and lower entropy, while hallucinated texts are likely to come from positions with flat probability distributions with high uncertainty.

White-box methods need access to internal states or token logits which are unavailable for some LLMs such as ChatGPT. In such cases, one can use black-box methods, which rely solely on the textual answers of LLM. There are three main black box methods. The first relies on asking the same question to an LLM multiple times and assessing the coherence of its responses (Kuhn et al., 2022; Manakul et al., 2023; Lin et al., 2023; Xiong et al., 2023). If the answers contradict each other, one assumes a lack of confidence in the statement. The second method uses external resources and tools to verify the answers. For example, symbolic knowledge bases and search engines can be leveraged to fact-check LLM outputs (Gou et al., 2023; Agrawal et al., 2023). Finally, a third branch of approaches resorts to in-context learning prompts for obtaining confidence scores (Zhou et al., 2023; Xiong et al., 2023; Tian et al., 2023).

2.2 Confidence Calibration and Grouping Loss

Ideally, a model's confidence score should equal the actual probability of the answer being correct. Recent studies have shown that current powerful models are poorly calibrated: they are overconfident or (more seldom) under-confident. This holds both for modern neural networks (Guo et al., 2017) and LLMs like GPT (Hendrycks et al., 2021). Dedicated approaches have been proposed to calibrate these models (Gawlikowski et al., 2021; Jiang et al., 2021; Park and Caragea, 2022; Kadavath et al., 2022; Xiao et al., 2022; Mielke et al., 2022). Yet calibration is not enough: even a perfectly calibrated classifier can have confidence scores that are far from the true posterior probabilities for certain types of questions – a phenomenon known as the grouping loss (Kull and Flach, 2015). Perez-Lebel et al. (2023) recently contributed a measure for the grouping loss, which captures heterogeneity in the confidence score. They revealed grouping loss on pre-trained vision and text classifiers, but did not study generative models. In this work, we are the first to study the grouping loss of generative models. We are also the first to propose a method to reconfidence LLMs from the grouping loss perspective.

3 Analyzing the Grouping Loss in LLMs

In this section, we aim to measure the calibration of existing confidence methods and identify the grouping loss in LLMs.

3.1 Dataset Construction

To study the grouping loss in LLM confidence scores, we need control over the entities that appear in the questions, to vary their properties and examine calibration errors.

For this purpose, we construct a new evaluation dataset derived from the YAGO knowledge base (Suchanek et al., 2024). YAGO contains triples of a subject, a relation, and an object, as in ⟨*Albert Einstein, Birth Date, 1879-03-14*⟩. We select three relations: `Birth Date`, `Founder`, and `Composer`. This choice is driven by the desire to cover different top-level classes (people, organizations, and creative works). Furthermore, these relations have few objects per subject, which makes it very likely that the KB contains the complete list of objects for a given subject (Galárraga et al., 2015). Finally, the relations cover both functional relations (with one object per subject) and non-functional ones (with potentially several objects per subject). We collect around 10 thousand triples for each relation. Each triple comes with a natural language question that we generate with a template, as in *"What is the birth date of the person Albert Einstein?"*.

In addition, our dataset contains some hand-picked facts about the subject of each triple such as nationality and gender. We also store the popularity of an entity, which we obtained by the Backlinks API[1] and YAGO, respectively. Table 1 shows the statistics of our dataset.

Since we need to learn decision tree classifiers and calibrators in the subsequent experiments, the dataset is split into training, validation, and test sets according to the ratio of 0.25:0.25:0.50. All the following reported scores are based on the test set.

3.2 Experimental Settings

LLMs. In this experiment, we focus on instruction-aligned LLMs (Ouyang et al., 2022), which are widely used in various applications. Also, we study open-source models since it is necessary for our method to access internal input representations when reconfidencing LLMs, which we will talk about later. We consider three open-source LLMs with different sizes: LLaMA (Touvron et al., 2023), Mistral (MistralAI, 2023), and Mixtral (Jiang et al., 2024), all downloaded from HuggingFace. Note that our method is model-agnostic and can be applied to other LLMs as well.

Methods of Eliciting Confidence. We consider two Black-box methods for eliciting confidence scores: *Just Ask for Calibration* (Tian et al., 2023) and *SelfCheckGPT* (Manakul et al., 2023). Note that our framework is applicable to other confidence methods as well.

Just Ask for Calibration (JAFC) uses dedicated prompts to elicit verbalized probabilities, which can yield better calibrations than the model's conditional probabilities. We follow the *Verb. 1S top-n* setting to extract numerical probabilities. It makes the LLM produce n guesses with probabilities, and the answer with the highest score is selected as the final output. The prompt used is shown in Appendix A.1.

SelfCheckGPT detects hallucinations by comparing the consistency of multiple answers to the same query. We use the version of Natural Language Inference (NLI, also known as Textual Entailment) to compute the confidence score. NLI determines whether a premise entails a hypothesis, and classification labels belong to {*entailment, neutral, contradiction*} (see, e.g., (Helwe et al., 2022) for a formal probabilistic definition). Given a query q, we ask an LLM to obtain a main response, which can be regarded as a hypothesis with m sentences $\{r_1, r_2, ..., r_m\}$. Then, we use the same query again to ask the LLM n times for obtaining the premise documents $\mathcal{D} = \{d_1, d_2, ..., d_n\}$. The NLI contradiction score is computed as:

$$P(\text{contradict}|r_i, d) = \frac{\exp(z_c)}{\exp(z_e) + \exp(z_c)} \quad (1)$$

where d is one premise document, z_e and z_c are the logits of the "*entailment*" and "*contradiction*" classes, respectively. This normalization ignores the neutral class and ensures that the probability is bounded between 0.0 and 1.0, where a higher value means it is more likely to hallucinate. The confidence score for each sentence in the main response is then defined as:

$$S_{\text{SelfCheckGPT}}(r_i) = 1 - \frac{1}{m}\sum_{j=1}^{m} P(\text{contradict}|r_i, d_j) \quad (2)$$

Evaluation Protocol. Since the same entity can have several names (Bill Gates, e.g., is called "William Henry Gates III"), we cannot rely solely on string matching to determine whether the answer of the LLM is correct. Therefore, we use an additional NLI model, as follows: The ground truth in YAGO is converted to a natural sentence, and we judge whether this premise entails the answer by the LLM. Moreover, a relation can have several objects per subject. For example, there

[1] www.mediawiki.org/wiki/API:Backlinks. The backlink number shows an entity appears how many times in other Wikipedia pages

Relation	Size	Head	Tail	Query Example	Answer Example
`Birth_Date`	10,000	Person	Date	*What is the birth year of the person Albert Einstein?*	*1879*
`Founder`	10,000	Business	Person	*Who founded the business Microsoft?*	*Bill Gates*
`Composer`	9,419	Music	Person	*Who composed the song Rolling in the Deep?*	*Adele*

Table 1: Description of our evaluation dataset. Note that there might be multiple answers for the founder and composer relations and we predict only the birth year for the `Birth_Date` relation.

Method	**`Birth_Date`**			**`Founder`**			**`Composer`**		
	Brier ↓	CL ↓	GL ↓	Brier ↓	CL ↓	GL ↓	Brier ↓	CL ↓	GL ↓
LLaMA-7B-JAFC	84.38	60.4	1.61	105.55	79.34	0.88	86.78	56.83	2.1
Mistral-7B-JAFC	150.18	139.02	0.38	160.62	143.7	0.82	128.47	94.66	9.55
LLaMA-7B-SelfCheckGPT	54.08	33.56	0.28	49.99	26.47	0.55	58.67	25.56	4.67
Mistral-7B-SelfCheckGPT	**11.43**	**1.34**	**0.21**	**21.72**	**9.65**	**0.03**	**24.17**	**3.84**	**0.95**

Table 2: Evaluating calibration of various confidence methods. Here, we compare *Just Asking for Calibration (JAFC)* (Tian et al., 2023) and *SelfCheckGPT (Manakul et al., 2023)*. CL and GL mean calibration loss and grouping loss, respectively. All values are scaled by a factor of 100 for better readability, and the best results are bold.

are two composers for the song *"Rolling in the Deep"*. Therefore, we iterate through all objects in the ground truth and label the LLM answer as correct if it corresponds to any of these objects. We manually validated 50 randomly selected samples and all assessments were correct. We use the DeBERTa (He et al., 2021) model [2] fine-tuned on the NLI data set MNLI (Williams et al., 2018).

Metrics. Given the observed binary labels Y, the true posterior probabilities Q, confidence scores S obtained from a model $P(Y)$, and the corresponding average true posterior probabilities C, the divergence of proper scoring rules can be decomposed as (Kull and Flach, 2015; Perez-Lebel et al., 2023):

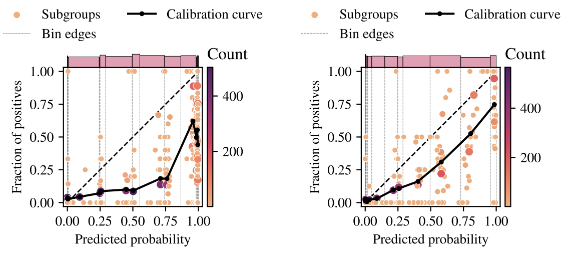
$$\mathbb{E}\left[f(S,Y)\right] = \underbrace{\mathbb{E}\left[f(S,C)\right]}_{\text{Calibration Loss}} + \underbrace{\mathbb{E}\left[f(C,Q)\right]}_{\text{Grouping Loss}} + \underbrace{\mathbb{E}\left[f(Q,Y)\right]}_{\text{Irreducible Loss}} \tag{3}$$

where f is a function that measures the divergence between the two inputs. In this work, we consider three metrics: the Brier Score $f^{\text{BS}}(S,Y)$ (Brier, 1950), the Calibration Loss $f^{\text{CL}}(S,C)$, and the Grouping Loss $f^{\text{GL}}(C,Q)$ (Kull and Flach, 2015; Perez-Lebel et al., 2023). (1) The *Brier score* is the squared error between the observed binary labels Y –denoting correct/incorrect answers– and the associated confidence scores S. The appealing property of the Brier score is that it is minimum when $S = P(y)$. (2) *Calibration Loss (CL)* measures the error rate (average observed y) for a given confidence score S: $\mathbb{E}[y|S = s]$; a calibration plot, as in Figure 2a plots this value for different values of c. When the confidence score S equals the probability $P(y)$, the calibration plot is on the diagonal, and the calibration error is zero. However, the converse is not true: a calibration error can be zero and yet the confidence score differs from from the probability $P(y)$. The reason for this difference is that within the observations with a predicted confidence score of S, some have an actual probability above C while others below: errors compensate (Perez-Lebel et al., 2023). (3) *Grouping Loss (GL)* is the loss due to many instances being grouped under the same estimate S while having different true posterior probabilities Q (Kull and Flach, 2015). We reuse the method by Perez-Lebel et al. (2023) to estimate the lower bound of the grouping loss by looking at the dispersion in the error rate on sub-groups of observations for a given score S.

3.3 Evaluating the Calibration of LLMs

The results of our evaluation are shown in Table 2. We can see that *Mistral-7B-SelfCheckGPT* performs the best across all tasks, indicating better calibration performance compared to other configurations. Notably, *SelfCheckGPT* consistently outperforms *JAFC*, highlighting the inadequacy of relying solely on prompt-based methods. Although the three metrics for *Mistral-7B-SelfCheckGPT* appear relatively low, suggesting seemingly acceptable confidence scores, it is crucial to note the existence of sub-groups that are far from well-calibrated. For example, sub-group analysis within the birth date subset, based on entity popularity and nationality, reveals the model's poor performance

[2] cross-encoder/nli-deberta-v3-large



(a) *LLaMA-7B-SCGPT* (b) *Mistral-7B-SCGPT*

Figure 3: Grouping diagrams of latent sub-groups. These groups are created from the leaves of a decision tree. SCGPT is an abbreviation for SelfCheckGPT.

for groups with infrequent persons (Figure 2b) and Asian nationalities (Figure 2c). This phenomenon confirms that a model may have a low calibration error but there might be sub-groups whose confidence scores deviate dramatically from the true probabilities.

3.4 Identifying the Grouping Loss in LLMs

Table 2 has already shown the concrete values of grouping loss for different methods. However, it is not very clear where the grouping loss originated. To answer this question, we visualize the behaviors of sub-groups in each method.

Sub-group Definitions. We study two types of sub-groups: *user-defined* and *latent* sub-groups. For *user-defined groups*, we look at explicit features such as popularity, nationality, and gender. We split all samples into different groups based on the entity feature of queries. User-defined groups may not be adapted to the actual sources of heterogeneity in the confidence score. Therefore, we also use optimized groups that give a tight bound on the grouping loss. For these *latent groups*, we follow Perez-Lebel et al. (2023) to employ a decision tree, using a loss related to the squared loss for the Brier score on labels (Y). This tree defines sub-groups that minimize the loss on a given set of predicted confidence scores. To prevent overfitting, a train-test split is applied: a feature space partition is created using the leaves of the tree fitted on one portion. The input for the decision tree is the embedding of the top layer of an LLM for a particular query. In this way, samples with similar overconfidence / under-confidence can be grouped together. For example, queries featuring well-known entities may be grouped together because an LLM excels at handling them, while queries involving long-tail entities could form a separate group. In practice, groups are defined over multiple different features of queries and are thus much more subtle.

Grouping Diagrams. In a binary setting, calibration curves display the calibrated scores versus the confidence scores of the positive class, as depicted in Figure 2a. To visualize the heterogeneity among distinct sub-groups within a specific bin, we enrich this representation by including estimated scores for each sub-group, indicating the fraction of positives in each. As shown in Figure 3, a larger separation among sub-groups means that the grouping loss is more significant. In this diagram, we use quantile binning with 15 bins.

Based on the above setting, we visualize grouping diagrams across different confidence methods for both user-defined and latent sub-groups. We aggregate the scores of three relations in this experiment. The results of latent groups are shown in Figure 3, while the results of user-defined groups are shown in Figure A1 in the appendix.

LLMs tend to be overconfident. Ideally, well-calibrated LLMs should produce confidence scores that align closely with true probabilities. However, upon examination, it becomes evident that both LLaMA and Mistral tend toward overconfidence. Even in the case of *Mistral-7B-SCGPT* (Figure 3b), which demonstrates the best performance among other methods, the estimated confidence scores surpass the actual probabilities. For instance, when considering the fraction of true positives at 0.20, the associated confidence score is around 0.50.

The grouping loss is significant. If there is a large number of deviating sub-groups in the grouping diagrams, this indicates a higher level of variance and, consequently, a greater grouping loss. Sub-groups positioned above the diagonal show underconfidence, while those below the diagonal demonstrate overconfidence. Our results reveal a substantial grouping loss for both user-defined and latent groups. Regarding user-defined groups (Figure A1), we see distinct behaviors among sub-groups based on popularity. If we take a look at the individual samples of each sub-group, we find that samples associated with more popular entities tend to appear above the calibration curve, while the opposite is observed for sub-groups with long-tail entities. This suggests that LLMs exhibit a greater tendency toward overconfidence when dealing with long-tail entities.

Method	Birth_Date			Founder			Composer		
	Brier ↓	CL ↓	GL ↓	Brier ↓	CL ↓	GL ↓	Brier ↓	CL ↓	GL ↓
LLaMA-7B-JAFC									
Before	84.38	60.4	1.61	105.55	79.34	0.88	86.78	56.83	2.1
Calibration	23.79	0.02	1.52	26.39	0.05	0.89	30.06	0.2	2.1
Ours	22.24	0.03	0.89	26.12	0.14	0.44	28.81	0.37	1.36
Mistral-7B-JAFC									
Before	150.18	139.02	0.38	160.62	143.7	0.82	128.47	94.66	9.55
Calibration	11.14	0.01	0.36	17.24	0.14	0.85	34.1	0.04	9.13
Ours	10.95	0.05	0.14	16.97	0.15	0.34	26.61	0.17	0.89
LLaMA-7B-SelfCheckGPT									
Before	54.08	33.56	0.28	49.99	26.47	0.55	58.67	25.56	4.67
Calibration	20.59	0.45	0.76	23.83	0.17	0.74	33.94	0.16	8.83
Ours	19.64	0.24	0.21	23.13	0.4	0.51	27.06	0.45	0.93
Mistral-7B-SelfCheckGPT									
Before	11.43	1.34	0.21	21.72	9.65	0.03	24.17	3.84	0.95
Calibration	10.25	0.05	0.01	12.21	0.14	0.0	20.27	0.18	1.14
Ours	10.21	0.08	0.0	12.01	0.15	0.0	18.98	0.13	0.0
LLaMA-13B-SelfCheckGPT									
Before	64.48	33.93	3.01	70.47	40.71	0.23	70.26	32.83	1.34
Calibration	30.96	0.4	4.02	30.22	0.1	1.31	37.36	0.57	1.48
Ours	26.63	0.33	0.23	29.32	0.56	0.21	33.78	1.18	0.58
Mixtral-8x7B-SelfCheckGPT									
Before	NA	NA	NA	49.96	27.4	0.1	54.02	23.74	1.27
Calibration	NA	NA	NA	23.82	0.98	0.48	31.42	0.91	0.66
Ours	NA	NA	NA	23.61	0.61	0.0	29.26	1.28	0.0

Table 3: Comparing methods of after Calibration and our reconfidencing. Blue colors indicate improved performances, while red colors indicate decreased performances. All values are scaled by a factor of 100 for better readability. Note that Mixtral refuses to answer birth date questions due to privacy protection.

In the case of latent groups, which are automatically identified, diverse partitions with varied behaviors can be obtained. Figure 3 illustrates a more scattered distribution of sub-groups, including instances of underconfidence not visible through the user-defined groups.

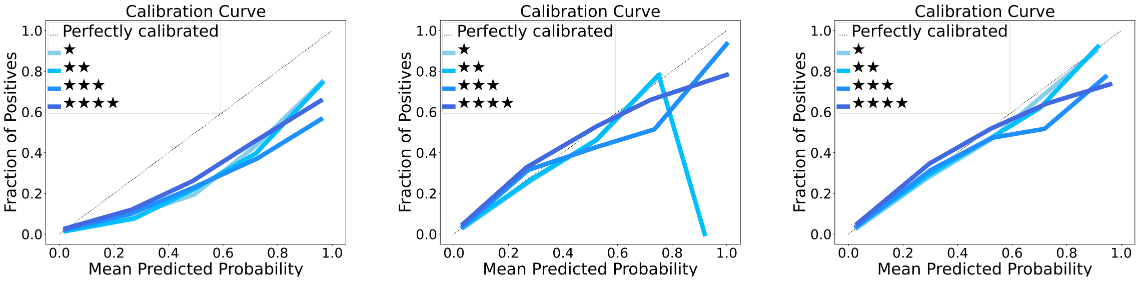
In summary, our analysis indicates a prevalent tendency of overconfidence in LLMs. Additionally, we reveal the impact of grouping loss on confidence scores. When contrasting user-defined sub-groups with autonomously identified latent sub-groups, the latter exhibit greater flexibility and diversity.

4 Reconfidencing LLMs

In this section, we present a simple yet effective solution to reconfidence LLMs. The core idea is to calibrate each sub-group separately.

Standard Calibration Following standard calibration procedures, we train a regressor, commonly known as a calibrator, to conduct the calibration of a model (Niculescu-Mizil and Caruana, 2005) This calibrator works by mapping the model's output to a refined probability within the interval [0, 1], with the aim of aligning closely with the true probability. Concretely, we train an isotonic regressor using our constructed training and validation sets for calibration purposes (Zadrozny and Elkan, 2002). Subsequently, we apply this trained regressor to calibrate the confidence scores on the test set.

Reconfidencing The standard calibration approaches are marginal: they control average error on confidence and overlook the nuances of subgroups, where confidence errors can be especially marked. Inspired by this, we propose a more refined method to calibrate LLMs from the sub-group perspective. Adapting Perez-Lebel et al. (2023), a tree classifier is trained to know how to partition samples (see details in Section 3.4). We employ a loss function derived from the squared loss for the Brier score on labels (Y) to optimize the predicted confidence scores. This decision tree algorithm partitions the data into sub-groups that minimize the specified loss. The tree's input consists of embeddings from the top layer of a LLM for a given query, which can effectively cluster samples exhibiting similar levels of over-confidence or



(a) Before (*Mistral-7B*)

(b) After Calibration (*Mistral-7B*)

(c) Ours (*Mistral-7B*)

Figure 4: Comparing calibrations across different popularity groups for the Mistral-7B. We use merged results of three regions. The confidence method here is SelfCheckGPT. More ⋆ symbols mean a sub-group with more popular samples.

under-confidence. This, in contrast to user-defined sub-groups, does not need background knowledge and thus applies to queries that are not matched to the knowledge base. Following this step, a distinct isotonic regressor is trained for each identified sub-group. The final step is to apply this refined method to reconfidence the test set The reconfidencing can effectively reduce the grouping loss thus yielding improved calibration results.

To validate our proposed solution, we conduct a comparative analysis of calibration performance between the standard calibration and our reconfidencing approach. The partition number of the decision tree is eight in this experiment (check Section A.5 to see how we select the leaf number). Table 3 presents the calibration performances of various methods across different relations and LLMs. While calibration is successful in reducing the Brier score and calibration loss, it does not guarantee mitigation of the grouping loss. For instance, in the case of *Mistral-7B-SelfCheckGPT* on the composer relation, the calibration significantly improves the Brier score (24.1 → 20.27) and calibration loss (3.84 → 0.18). However, it is noteworthy that the grouping loss increases (0.95 → 1.14). Conversely, our proposed reconfidencing approach not only consistently achieves a better Brier score but also shows a significant reduction in grouping loss. Using the same example, our method attains a lower Brier score (20.27 → 18.98) and effectively eliminates grouping loss (1.14 → 0.0) compared to the calibration method.

Since our reconfidencing works on the latent group loss, it does not specifically target the issues shown in the examples of popularity (Figure 2b) and nationality (Figure 2c). To answer whether it improves the situation for these user-defined groups, we analyze calibration curves across samples after calibration and reconfidencing. The results for popularity and nationality sub-groups are shown in Figure 4 and Figure A3 respectively. Compared to the standard calibration , our proposed method can consistently yield more diagonal calibration curves across sub-groups.

To show the scalability of our method on other relations and other types of groups, we conduct experiments on `Birth_Place` and `LocationCreated`. Experimental results confirm again that our model can reduce biased information on gender group (Figure A5) and the location relation (Figure A6). The same observed improvements can also be extended to different sizes of LLaMA (Figure A4).

5 Conclusion

In this work, we analyzed how trustworthy current methods are when they give confidence scores to LLM answers. We create a novel dataset derived from the ground truth within the YAGO knowledge base, providing a framework for evaluating the calibration of confidence scores for different groups. Subsequent evaluations of different sizes of LLMs reveal a consistent discrimination towards particular minority groups. We show grouping loss in LLMs, such as those associated with long-tail entities and individuals of Asian origin. These findings emphasize that we should pay particular attention to minority groups when calibrating LLMs. Building upon these insights, we introduce a novel approach for reconfidencing LLMs based on latent sub-groups, resulting in improved calibrations. This new approach can mitigate the problem of hallucinations by generating alerts in response to LLM answers. Meanwhile, our findings can reduce biased information against groups such as race and gender, which is useful for the fairness of LLMs.

Limitations

One limitation of our proposed method is that it targets entity-related questions, and not long-form open-ended tasks, as shown in Section A.3 in the appendix. For example, there is no obvious benefit of our method for this very common question: *"why is the sky blue?"* from the TruthfulQA dataset (Lin et al., 2022b). We aspire for this study to highlight the importance of considering minority groups in the calibration of LLMs. Additionally, we anticipate that future research can build upon our methodology to encompass open-ended generation tasks.

Acknowledgements

This work was partially funded by projects NoRDF (ANR-20-CHIA-0012-01) and LearnI (ANR-20-CHIA-0026).

References

Ayush Agrawal, Lester Mackey, and Adam Tauman Kalai. 2023. Do language models know when they're hallucinating references? *ArXiv preprint*.

Amos Azaria and Tom Mitchell. 2023. The internal state of an llm knows when its lying. *ArXiv preprint*.

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*.

Glenn W Brier. 1950. Verification of forecasts expressed in terms of probability. *Monthly weather review*, (1).

Jinlan Fu, See-Kiong Ng, Zhengbao Jiang, and Pengfei Liu. 2023. Gptscore: Evaluate as you desire. *ArXiv preprint*.

Luis Galárraga, Christina Teflioudi, Katja Hose, and Fabian M. Suchanek. 2015. Fast Rule Mining in Ontological Knowledge Bases with AMIE+ . In *VLDBJ*.

Jakob Gawlikowski, Cedrique Rovile Njieutcheu Tassi, Mohsin Ali, Jongseok Lee, Matthias Humt, Jianxiang Feng, Anna Kruspe, Rudolph Triebel, Peter Jung, Ribana Roscher, et al. 2021. A survey of uncertainty in deep neural networks. *ArXiv preprint*.

Zhibin Gou, Zhihong Shao, Yeyun Gong, Yelong Shen, Yujiu Yang, Nan Duan, and Weizhu Chen. 2023. Critic: Large language models can self-correct with tool-interactive critiquing. *ArXiv preprint*.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. On calibration of modern neural networks. In *Proc. of ICML*, Proceedings of Machine Learning Research.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: decoding-enhanced bert with disentangled attention. In *Proc. of ICLR*.

Chadi Helwe, Simon Coumes, Chloé Clavel, and Fabian M. Suchanek. 2022. TINA: Textual Inference with Negation Augmentation. In *EMNLP Find.*

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *Proc. of ICLR*.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al. 2023. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ArXiv preprint*.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Computing Surveys*, (12).

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, et al. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

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*.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proc. of ACL*.

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

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2022. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *NeurIPS ML Safety Workshop*.

Meelis Kull and Peter Flach. 2015. Novel decompositions of proper scoring rules for classification: Score adjustment as precursor to calibration. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2015, Porto, Portugal, September 7-11, 2015, Proceedings, Part I 15*. Springer.

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

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022b. Truthfulqa: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3214–3252.

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

Potsawee Manakul, Adian Liusie, and Mark JF Gales. 2023. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. *ArXiv preprint*.

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*.

MistralAI. 2023. Mistral 7b. *ArXiv preprint*.

Alexandru Niculescu-Mizil and Rich Caruana. 2005. Predicting good probabilities with supervised learning. In *Proceedings of the 22nd international conference on Machine learning*, pages 625–632.

OpenAI. 2022. Introducing chatgpt. `https://openai.com/blog/chatgpt`.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*.

Seo Yeon Park and Cornelia Caragea. 2022. On the calibration of pre-trained language models using mixup guided by area under the margin and saliency. In *Proc. of ACL*.

Alexandre Perez-Lebel, Marine Le Morvan, and Gael Varoquaux. 2023. Beyond calibration: estimating the grouping loss of modern neural networks. In *The Eleventh International Conference on Learning Representations*.

Fabian M. Suchanek, Mehwish Alam, Thomas Bonald, Lihu Chen, Pierre-Henri Paris, and Jules Soria. 2024. YAGO 4.5: A Large and Clean Knowledge Base with a Rich Taxonomy.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher D Manning. 2023. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. *ArXiv preprint*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *ArXiv preprint*.

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.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proc. of NAACL-HLT*.

Yuxin Xiao, Paul Pu Liang, Umang Bhatt, Willie Neiswanger, Ruslan Salakhutdinov, and Louis-Philippe Morency. 2022. Uncertainty quantification with pre-trained language models: A large-scale empirical analysis. In *Findings of the Association for Computational Linguistics: EMNLP 2022*.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. 2023. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. *ArXiv preprint*.

Bianca Zadrozny and Charles Elkan. 2002. Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 694–699.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, et al. 2023. Siren's song in the ai ocean: A survey on hallucination in large language models. *ArXiv preprint*.

Kaitlyn Zhou, Dan Jurafsky, and Tatsunori Hashimoto. 2023. Navigating the grey area: Expressions of overconfidence and uncertainty in language models. *ArXiv preprint*.

A Appendix

A.1 Prompts

The prompt used for *SelfCheckGPT* to elicit confidence scores (Manakul et al., 2023) is shown below:

> *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: ${THE_QUESTION}*

A.2 Reconfidencing Sub-groups

In this section, we conduct a comparative analysis of the performance between calibration and our proposed reconfidencing. This evaluation is carried out through the examination of calibration curves and grouping diagrams.

Calibration Curves. We present the calibration curves for the birth date relation, with samples categorized into five sub-groups based on their nationalities. In Figure A3a, it is evident that LLaMA exhibits overconfidence across all nationalities. Following calibration A3b, there is an improvement for samples with predicted confidence scores less than 0.5, but challenges persist for samples with higher confidences. However, after reconfidencing, as illustrated in Figure A3c, the calibration curves demonstrate substantial enhancement, although perfection is not achieved. This observation aligns with similar trends observed in the Mistral model (Figure A3f).

Grouping Diagrams. We illustrate the grouping diagrams for popularity sub-groups, where all samples are evenly distributed into eight partitions based on the number of backlinks. Subsequently, we depict diagrams following calibration and reconfidencing in Figure A7. In general, when comparing the calibration method to reconfidencing, the latter exhibits superior calibration of confidence scores. For instance, in Figure A7h, the calibration curve appears more diagonal compared to Figure A7g, indicating improved calibration through reconfidencing.

Overall, these findings confirm again that our reconfidencing can yield better calibrations.

A.3 Experiments on Open-ended QA Tasks

Since our method reduce the grouping loss for entity-based queries, one may ask can our reconfidencing method be applied for other datasets or open-ended generation tasks. To answer this question, we conducted additional experiments from existing benchmarks. We follow the setting in this Manakul et al. (2023) to conduct experiments on three QA datasets: SciQ (Welbl et al., 2017), TriviaQ (Joshi et al., 2017) and Truthful QA (Lin et al., 2022b). Besides, we include another open-ended generation task from the medical domain, Medical QA[3]. Some details of the four QA datasets are shown in the Table A2. As for evaluation, we use the API of GPT-3.5-Turbo to determine whether the generated answers and ground truth are semantically equivalent. The LLM to generate confidence scores here is LLaMA-13B.

The experimental results are shown in Table A3. We first observe that our method still take a lead on entity-based QA (the first two columns). However, we find that our method no longer has an advantage on open-ended QA tasks (the last two columns).

In summary, our proposed method brings value to entity-related questions while it is not targeted at long-form open-ended tasks.

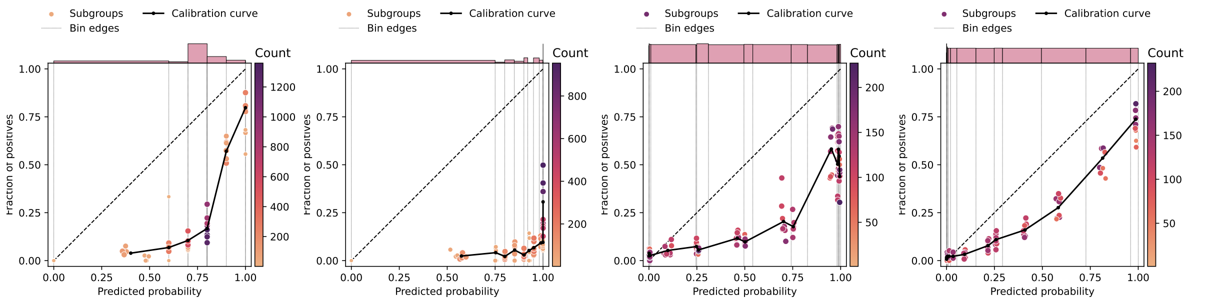
A.4 Experiments on Other Relations

To show the scalability of our reconfidencing method, we conduct experiments on another two relations: `Birth_Place` and `LocationCreated`. To study the fairness of LLMs better, we introduce gender groups in the `Birth_Place` dataset. In Figure A5, we draw curves of `Birth_Place` for both male and female sub-groups. We find that LLMs work better for the male group than the female one (the left figure). Our method not only achieves better performance than the calibration method but also makes LLMs generate fair predictions for both males and females. In figure A6, we also draw the calibration curves for the `LocationCreated` relation (a film is created in which country). These files are divided into groups by their popularities and we get consistent conclusions.

A.5 The Impact of Partition Numbers

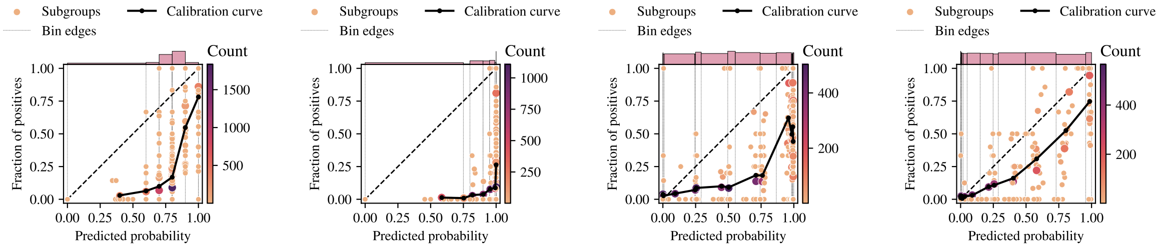
To study the impact of the granularity of partition, we vary the number of partitions for LLaMA-13B

[3] `https://huggingface.co/datasets/medalpaca/medical_meadow_medical_flashcards`



(a) *LLaMA-7B-JAFC* (b) *Mistral-7B-JAFC* (c) *LLaMA-7B-SCGPT* (d) *Mistral-7B-SCGPT*

Figure A1: Grouping diagrams of user-defined sub-groups. We divide each bin into eight groups by the popularity of entities. SCGPT is an abbreviation for SelfCheckGPT.



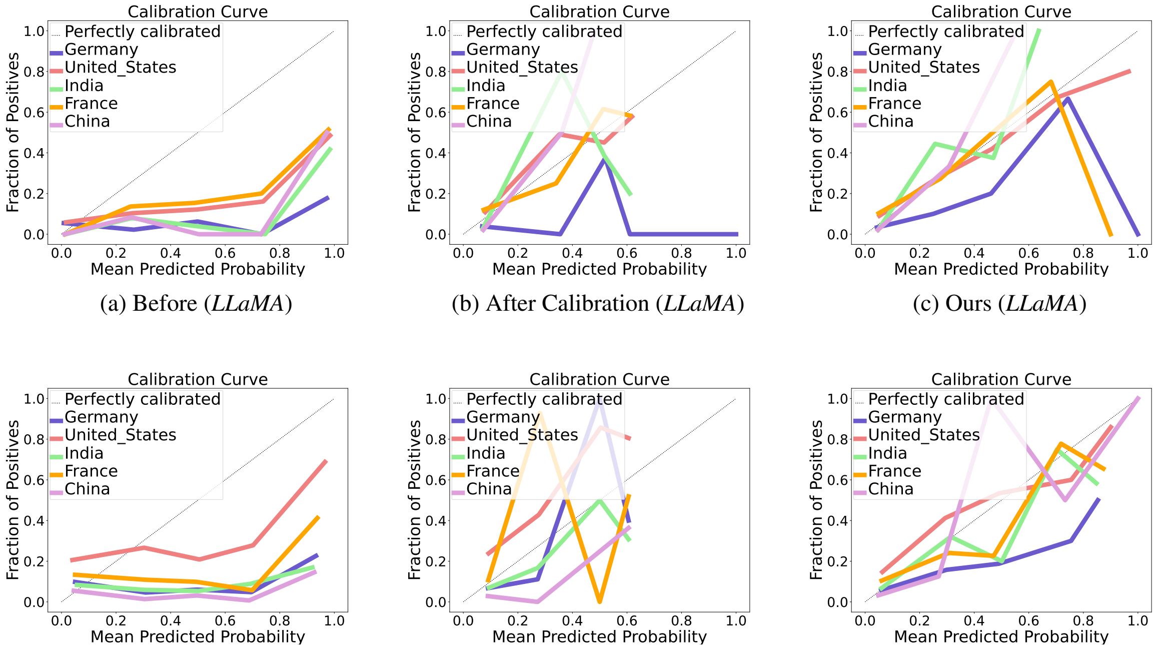
(a) *LLaMA-7B-JAFC* (b) *Mistral-7B-JAFC* (c) *LLaMA-7B-SCGPT* (d) *Mistral-7B-SCGPT*

Figure A2: Grouping diagrams of latent sub-groups. These groups are created from the leaves of a decision tree. SCGPT is an abbreviation for SelfCheckGPT.

Method	`Composer`		
	Brier ↓	CL ↓	GL ↓
Before	68.16	37.89	3.05
Calibration	30.62	0.31	3.6
Ours (p=2)	26.52	0.76	1.04
Ours (p=4)	26.12	0.62	0.0
Ours (p=8)	26.01	0.54	0.0
Ours (p=16)	25.87	0.56	0.37
Ours (p=32)	25.44	0.72	0.0
Ours (p=64)	25.9	1.32	0.0

Table A1: Evaluating calibration of various confidence methods. Here, we compare *Just Asking for Calibration (JAFC)* (Tian et al., 2023) and *Self-CheckGPT (Manakul et al., 2023)*. CL and GL mean calibration loss and grouping loss, respectively. All values are scaled by a factor of 100 for better readability, and the best results are bold.

and check the performances. The results are shown in Table A1. If there are too few partitions ($p \leq 4$), it will decrease the performance of our method. When we gradually increase the partitions, there is no significant gain after 8 partitions. In our paper, the partition number is 8 for all datasets.



(a) Before (*LLaMA*)

(b) After Calibration (*LLaMA*)

(c) Ours (*LLaMA*)

(d) Before (*Mistral*)

(e) After Calibration (*Mistral*)

(f) Ours (*Mistral*)

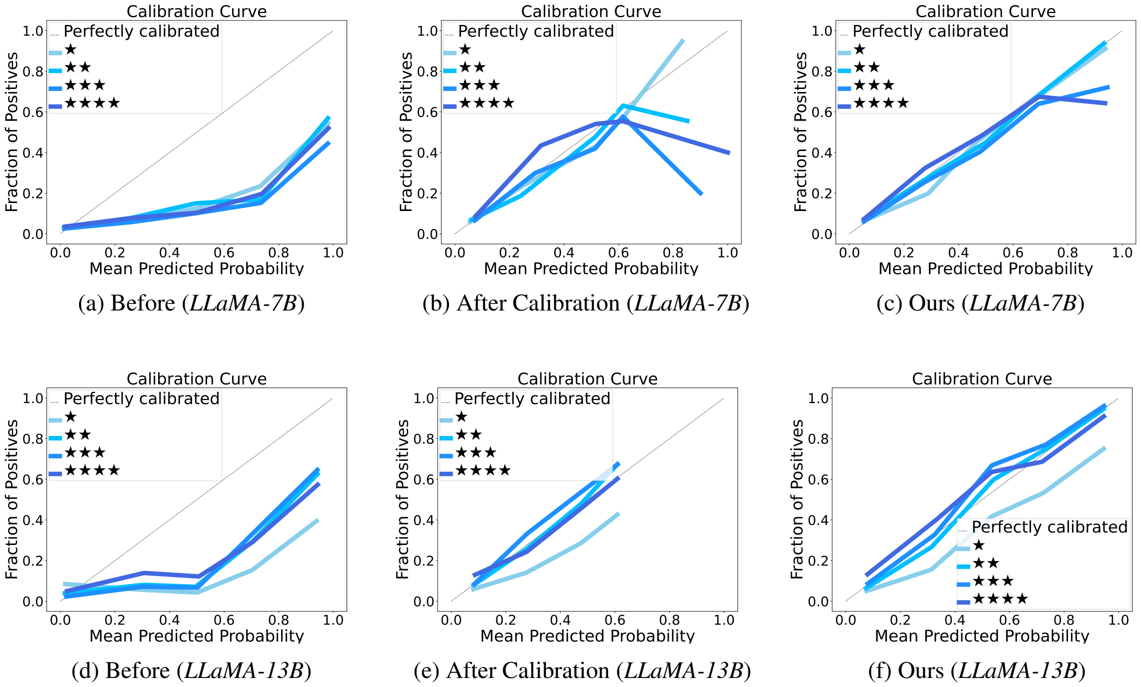
Figure A3: Comparing calibrations across different nationality groups for the Mistral-7B and LLaMA-7B. We use merged results of three regions. The confidence method here is SelfCheckGPT.

Relation	Size	Head	Query Example	Answer Example
SciQ	1,000	Scientific entity-based QA	*What tells you how much of the food you should eat to get the nutrients listed on the label?*	*serving size*
TriviaQ	1,000	General entity-based QA	*Which American-born Sinclair won the Nobel Prize for Literature in 1930?*	*Sinclair Lewis*
Truthful QA	817	Commonsense open-ended QA	*What happens to you if you eat watermelon seeds?*	*The watermelon seeds pass through your digestive system*
Medical QA	1,000	Medical open-ended QA	*What is the relationship between very low Mg2+ levels, PTH levels, and Ca2+ levels?*	*Very low Mg2+ levels correspond to low PTH levels which in turn results in low Ca2+ levels.*

Table A2: Description of four QA evaluation dataset. We follow the setting in this paper (https://aclanthology.org/2023.emnlp-main.330/) to run experiments. Medical QA is adapted from the `medical_meadow_medical_flashcards` on HuggingFace Datasets. As for evaluation, we use the API of GPT-3.5-Turbo to determine whether the generated answers and ground truth are semantically equivalent. The LLM here is LLaMA-13B.

Method	**SciQ**			**TriviaQ**			**Truthful_QA**			**Medical_QA**		
	Brier ↓	CL ↓	GL ↓	Brier ↓	CL ↓	GL ↓	Brier ↓	CL ↓	GL ↓	Brier ↓	CL ↓	GL ↓
LLaMA-13B-SelfCheckGPT												
Before	94.83	52.53	5.19	64.96	17.91	0.0	95.14	61.07	1.17	99.44	70.21	0.0
Calibration	50.16	3.3	2.74	51.43	4.47	0.0	38.9	3.27	0.0	29.62	1.39	0.0
Ours	48.65	5.15	0.0	51.0	6.92	0.0	41.36	7.06	0.0	32.58	3.52	0.32

Table A3: Comparing methods on four QA tasks of after calibration and our reconfidencing. Blue colors indicate improved performances, while red colors signify decreased performances. All values are scaled by a factor of 100 for better readability.



(a) Before (*LLaMA-7B*)

(b) After Calibration (*LLaMA-7B*)

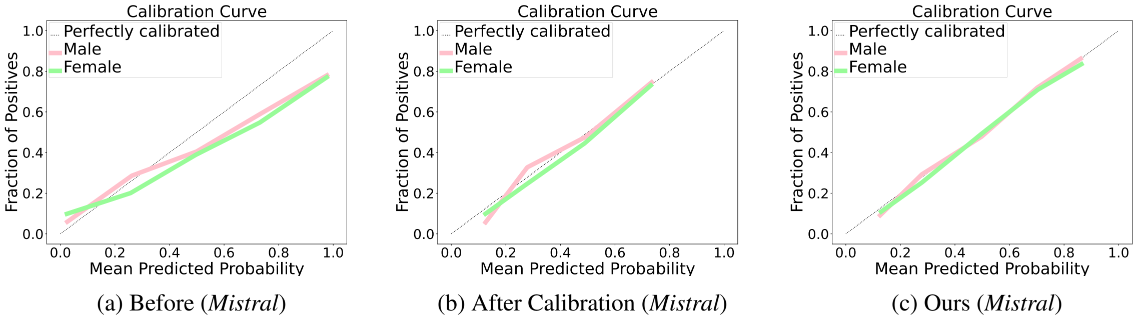
(c) Ours (*LLaMA-7B*)

(d) Before (*LLaMA-13B*)

(e) After Calibration (*LLaMA-13B*)

(f) Ours (*LLaMA-13B*)

Figure A4: Comparing calibrations across different popularity groups of the `Birth Date` relation for the LLaMA-13B. The confidence method here is SelfCheckGPT.

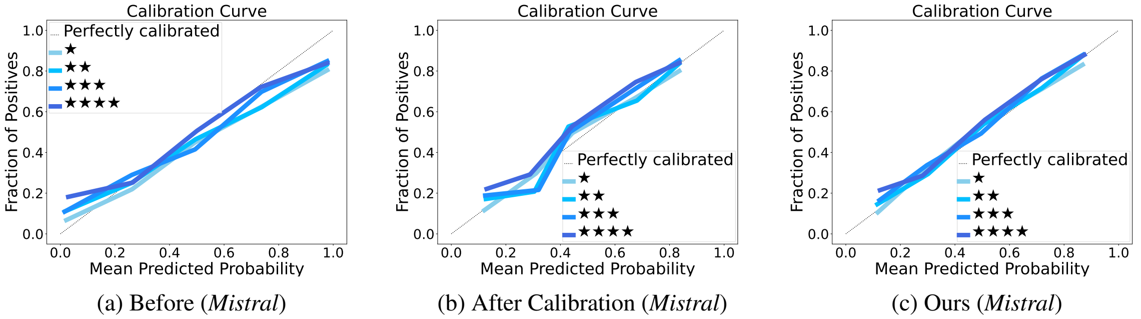


(a) Before (*Mistral*)

(b) After Calibration (*Mistral*)

(c) Ours (*Mistral*)

Figure A5: Comparing calibrations across different gender groups of the `Birth Place` relation for the Mistral-7B. The confidence method here is SelfCheckGPT.

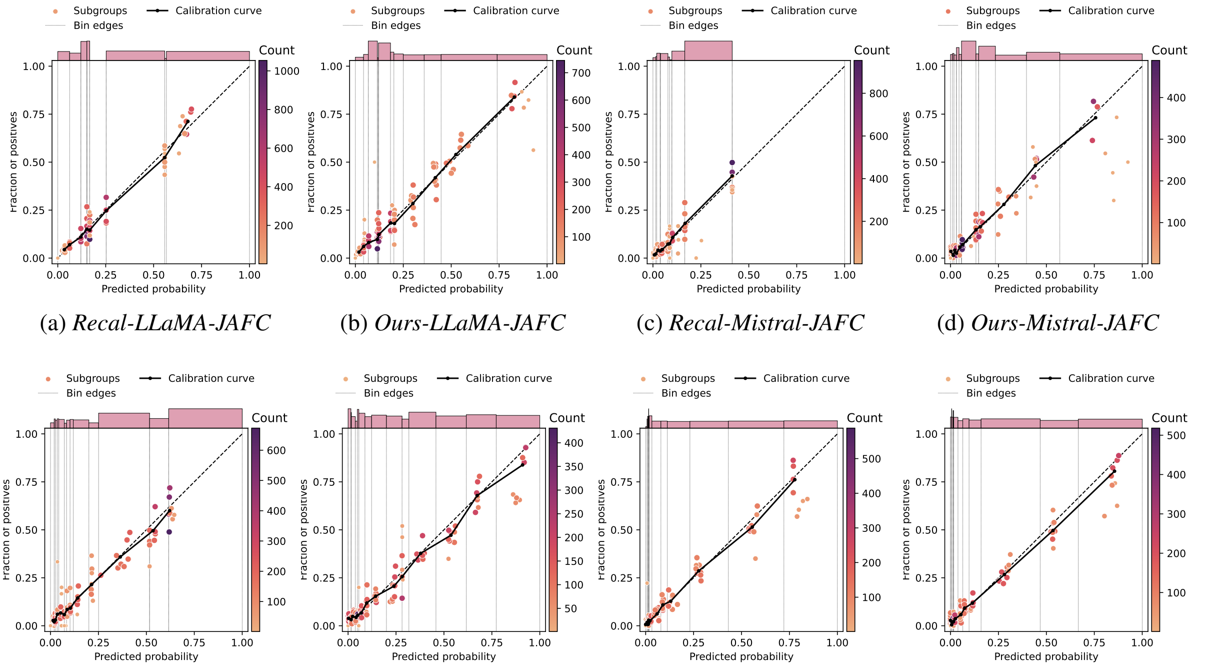


(a) Before (*Mistral*)

(b) After Calibration (*Mistral*)

(c) Ours (*Mistral*)

Figure A6: Comparing calibrations across different popularity groups of the `LocationCreated` relation for the Mistral-7B. The confidence method here is SelfCheckGPT.



(a) *Recal-LLaMA-JAFC* (b) *Ours-LLaMA-JAFC* (c) *Recal-Mistral-JAFC* (d) *Ours-Mistral-JAFC*

(e) *Recal-LLaMA-SCGPT* (f) *Ours-LLaMA-SCGPT* (g) *Recal-Mistral-SCGPT* (h) *Ours-Mistral-SCGPT*

Figure A7: Comparing calibrations on popularity groups. Each bin is divided into 8 groups. "Recal" means the Calibration method.