NOT *Whether*, BUT *Which*: CONCEPT-LEVEL DEFICIENCIES BEHIND LLM INCONSISTENCY

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models (LLMs) exhibit inconsistency: semantically equivalent inputs elicit contradictory factual outputs, eroding their reliability in downstream applications. Prior work has documented this phenomenon but leaves a more fundamental question unanswered: which knowledge deficiencies produce inconsistency? Knowing that a model is inconsistent is of limited use unless we know which knowledge it lacks. In this paper, we posit that inconsistency stems from identifiable, localizable gaps in concept-level knowledge. To test this hypothesis, we introduce `KnowledgeMap`, a pipeline that pinpoints such gaps by automatically parsing inputs into lexical concepts, applying calibrated perturbations, and then dynamically profiling concept-level proficiency via consistency measurements. Evaluated across large-scale datasets and diverse LLMs, `KnowledgeMap` reliably isolates specific knowledge deficiencies and yields actionable guidance for targeted model training. Codes are available at `https://anonymous.4open.science/r/Memocon-F4D6/`.

1 INTRODUCTION

Consistency has become a fundamental requirement for the practical adoption of LLM-powered applications Kim et al. (2024); Stureborg et al. (2024). It refers to the models' ability when processing inputs that are semantically identical or share equivalent intent, the outputs should maintain logical coherence and factual alignment.

Current research attributes inconsistency to several causes, such as the knowledge conflicts between internal and contextual knowledge Xu et al. (2024); Su et al. (2024), misinformation in training data Pan et al. (2023); Liu et al. (2024), or memorization-generalization tradeoffs Bansal & Pruthi (2022); Feldman & Zhang (2020). While offering in-depth insights, we observed that inconsistency in these studies is operated at the parametric level, or remains at a general theoretical level Yang et al. (2024a); Wang et al. (2024a). Crucially, these works establish that LLMs are inconsistent and *why* in the aggregate, but not *which* knowledge is responsible, providing neither intuitive explanations nor actionable insights for downstream improvement. Such abstraction obscures the root causes and impedes targeted mitigations.

Given the inherent dependence of LLM outputs on knowledge integrity, we posit that inconsistency fundamentally signals a knowledge gap, and more importantly, a gap that can be explicitly located. This perspective remains unexplored despite extensive knowledge evaluation studies in factual domains Yu et al. (2024), ethics Jiang et al. (2025), and technical fields Wang et al. (2025). Given the critical importance of knowledge, prior studies typically categorize knowledge as either present or absent based solely on answer correctness and subsequently reduce knowledge levels to several basic metrics, lacking the fine granularity needed to precisely identify knowledge gaps and a continuous measure of proficiency to quantify their severity.

Motivated by this research gap, *this paper seeks to provide an intuitive and explicit interpretation of the occurrence of inconsistency resulting from the perspective of knowledge deficiency in LLMs.* We propose `KnowledgeMap`, a framework for interpreting inconsistency by detecting LLM's knowledge gaps at a fine-grained concept level. Knowledge within the input sample is parsed into lexical concepts for knowledge systematization Medelyan et al. (2013), while multi-intensity perturbations, used to measure consistency, dynamically update the model's proficiency for each knowledge concept. `KnowledgeMap` fully considers the correlation between knowledge and the diverse manifesta-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

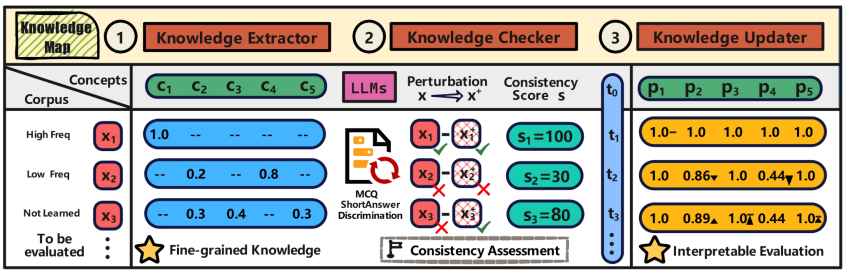


Figure 1: Overview of KnowledgeMap, a concept-level framework for interpreting LLM inconsistency. **(1) Knowledge Extractor** parses each input into lexical concepts $c = (c_1, \ldots, c_n)$ with normalized importance weights w, $\sum_{i=1}^{n} w_i = 1$. **(2) Knowledge Checker** applies m knowledge-invariant perturbations $x^+ = f(x)$ and assigns a consistency score $s \in [0, 1]$ from the correctness outcomes before and after perturbation. **(3) Knowledge Updater** uses w and s to update the per-concept proficiency $p_t \rightarrow p_{t+1}$.

tions of consistency. Through testing on two structurally distinct datasets totaling over 200k samples and three widely used LLMs, KnowledgeMap's capability to discriminate knowledge gaps has been rigorously validated, while also generating a substantial amount of informative data for in-depth inconsistency attribution analysis. It benefits various subsequent tasks such as the knowledge evolution mechanism, targeted fine-tuning, task-agnostic performance prediction, etc. In summary, our contributions are as follows:

- We propose KnowledgeMap, an interpretable framework that identifies LLMs' fine-grained knowledge gaps behind inconsistency. It decomposes input knowledge into lexical concepts and dynamically evaluates concept-level proficiency through multi-intensity perturbation tests, while accounting for knowledge correlations and diverse consistency manifestations.
- Leveraging KnowledgeMap, we precisely pinpoint the specific deficient knowledge components responsible for overall inconsistency, characterizing LLMs' inconsistency status quo. These localized deficiencies constitute concrete, model-specific targets for subsequent data augmentation or fine-tuning.
- KnowledgeMap reveals, for the first time, how varying degrees of knowledge deficits produce distinct inconsistency patterns. Our framework also enables precise attribution of inconsistencies to knowledge confusion and demonstrates how localized deficiencies undermine the observed consistency.

2 Related Work

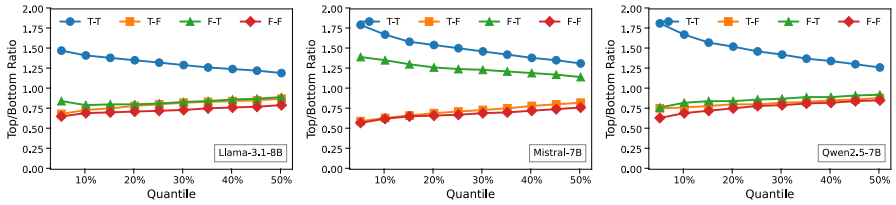
An extended related work is provided in Appendix A.

How knowledgeable are LLMs? Previous research explores LLM knowledge through multiple perspectives, including whether models have learned Yu et al. (2024) or are familiar Pletenev et al. (2025) with knowledge, and the conditions for the effective use of knowledge Yin et al. (2024). An essential focus is the model's awareness of unknown knowledge to mitigate hallucinations Cheng et al. (2024). Some studies treat knowledge as concrete text Carlini et al. (2023), images Wang et al. (2024a), or facts Petroni et al. (2019). Others might abstract it into structural or parametric network representations Bansal & Pruthi (2022); Wang et al. (2024a); Dankers & Titov (2024), thus classifying knowledge by its parametric traits.

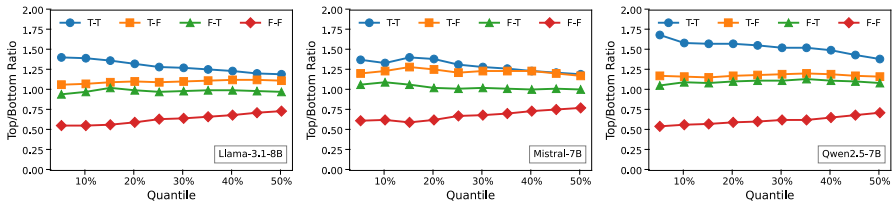
Understanding a model's internalized knowledge is crucial for improving its performance. Research shows that larger LLMs possess greater knowledge capacity Lu et al. (2024). They acquire knowledge rapidly in early pre-training stage, while the learning difficulty increases exponentially in the later stage Li et al. (2020). To compensate for limited capacity, models enhance reasoning by blending learned knowledge with generalization Dankers & Titov (2024); Wang et al. (2024b). However, introducing new knowledge during fine-tuning can trigger hallucinations Gekhman et al. (2024). Notably, precise knowledge memorization can harm generalization Bansal & Pruthi (2022), while

retaining even irrelevant data paradoxically improves it Feldman & Zhang (2020); Brown et al. (2021).

Why does inconsistency occur? Research reveals that LLMs exhibit non-trivial inconsistency Elazar et al. (2021); Dong et al. (2023) and undermines reliance on LLMs' downstream tasks. Intrinsically, inconsistency is not simply related to accuracy Raj et al. (2023) or model size Jang et al. (2022). Two primary explanations emerge: the first is the knowledge conflict between parametric knowledge and context (e.g., time/place/relationships) Xu et al. (2024); Xie et al. (2024). Models are proven to be highly context-sensitive Su et al. (2024), especially with conflicting pretrained data Longpre et al. (2021). The second is the reasoning deficits when semantic misunderstanding is replaced by pattern imitation Bender et al. (2021); Yang et al. (2024b). Mitigation strategies include self-feedback Liang et al. (2024), improved decoding Tomani et al. (2024), and model editing Yang et al. (2024a).



(a) CONFLICTBANK



(b) TRIVIAQA (Sum Titles' Popularities)

Figure 2: Sample size comparison under different consistency groups by knowledge popularity in *top-x%* vs. *bottom-x%* quantiles. The x-axis represents quantile thresholds, and the y-axis shows the number of samples per group within a quantile. A line positioned higher indicates that more samples within the corresponding group are concentrated toward the top of the popularity ranking. This suggests that *samples with higher popularity tend to exhibit relatively better consistency performance*, offering strong evidence supporting the assumption that knowledge deficiency may relate to inconsistency.

3 METHODOLOGY

An overview of the `KnowledgeMap` is presented in Figure 1. Assume there is a finite set of knowledge concepts $\mathbb{C}$ of size N. At a time t with a given input x and its associated output y. The workflow is formalized as follows, with detailed explanations of each step presented in the rest of this section:

- **Step 1.** A sample x is first parsed through the *Knowledge Extractor* $\mathcal{E} : x \mapsto (\boldsymbol{c}, \boldsymbol{w})$ to retrieve the input's associated *concepts* $\boldsymbol{c} = (c_1, c_2, \ldots, c_n), n \leq N, c_i \in \mathbb{C}$ and the *importance* of each *concept* $\boldsymbol{w} = (w_1, w_2, ..., w_n)$, where $w_i \in [0, 1], \forall i \in \{1, \ldots, n\}$ and $\sum_{i=1}^{n} w_i = 1$.
- **Step 2.** Then, x is sent to the *Knowledge Checker* $C : (x, \{x^+{}_i\}_{i=1}^m) \mapsto s, s \in [0, 1]$ by first generating m perturbed samples $x^+ = f(x)$ using the perturbation function $f(\cdot)$, followed by generating the consistency score s according to the consistency outcome resulting from the testing model under the raw input and its perturbed variants $(y, \{y^+{}_i\}_{i=1}^m)$, where $y, y^+{}_i \in \{\text{True}, \text{False}\}$.
- **Step 3.** With concepts' relevance $\boldsymbol{w}$ and the resulting consistency score s performed by the model, the *Knowledge Updater* $\mathcal{U} : (\boldsymbol{w}, s, \boldsymbol{p}^t) \mapsto \boldsymbol{p}^{t+1}$ is able to update the model's *proficiency* $\boldsymbol{p}^{t+1} = (p_1{}^{t+1}, p_2{}^{t+1}, ..., p_n{}^{t+1})$ in the related concepts.

3.1 THE *Knowledge Extractor*

The *Knowledge Extractor* is responsible for deriving knowledge concepts from the dataset and constructing a concrete structure of the knowledge Fidel & Efthimiadis (1995). In our framework, a lexical knowledge *concept* is a fundamental, atomic unit of information (e.g., a named entity like "London", a technical term like "Transformer") that constitutes a building block of a larger knowledge system Medelyan et al. (2013); Yi & Choi (2012). This module is designed to parse an input into its constituent concepts (c_i) and assign a relevance importance weight (w_i) to each, thereby systematizing the knowledge required to accurately answer the query. The concepts for constructing the dataset knowledge are first extracted by an extractor model, prior to any consistency evaluation of the tested model. The resulting concept list is subsequently used as a static external framework to evaluate the tested model.

We utilize LLMs for performing concept extraction for each test input. This is inspired by the superior ability of LLMs in semantic understanding and paragraph summarization, which has been well demonstrated in previous works Dubey et al. (2024); van Schaik & Pugh (2024); Ramprasad et al. (2024), as well as the versatility of LLMs to construct domain knowledge, which has also been verified Han et al. (2024); Chen et al. (2024); Mihindukulasooriya et al. (2023); Wang et al. (2021).

We use a structured prompt to extract concepts and their importance weights. The output is in the form of *concept (importance%)*. To control the quality of LLM output and ensure the usability of the framework, the model's valid output is filtered according to predefined conditions (details in Appendix B.2/C.4).

3.2 THE *Knowledge Checker*

The *Knowledge Checker* aims to assess the consistency *score* with the specific question-related *concepts* through perturbations at various intensities. The intensity for the i-th perturbation $v_i \in [0, 1]$ is computed by the cosine similarity between inputs through semantic embedding using the texts' embeddings of `Sentence-BERT`, i.e., $v_i = 1 - d_{\cos}(x, x^+{}_i)$. Higher similarity corresponds to lower perturbation intensity. Therefore, a significant performance drop under high similarity (low-intensity perturbation) strongly indicates fragile knowledge mastery (Appendix B.4/B.5).

There are four types of consistency outcomes for each perturbation based on the correct (**T** as True) and incorrect (**F** as False) results before (x) and after perturbation (x^+). They accordingly constitute four consistency outcomes $\mathcal{H} = \{\text{T-T}, \text{T-F}, \text{F-T}, \text{F-F}\}$. Each perturbation will result in one of these outcome types and is assigned a predefined case score $\{s_h\}_{h\in\mathcal{H}}, s_h \in [0, 1]$. The intuition is that if certain knowledge is sufficiently mastered by the LLM, there is a higher possibility that the LLM's performance on questions pertaining to this knowledge should remain consistent between the original and the perturbed version. We made the following hypotheses and inferences:

- **T-T** If the model can answer correctly on both x and x^+, it is insensitive to perturbations and exhibits strong consistency. It implies that the model has a small knowledge gap in the corresponding knowledge domain.
- **T-F** If the model answers correctly on x but wrong on x^+, it indicates that the model possesses certain knowledge related to the task but has not fully mastered it. The frequency of inconsistencies reflects the extent of the knowledge gap.
- **F-T** If the model answers x incorrectly but succeeds on x^+, we interpret this case as reflecting a fragile or uncertain knowledge state rather than stable knowledge possession. Since the model can recover the correct answer under perturbation, the relevant knowledge may exist but fail to be reliably activated for the original query. Conversely, the correct response to x^+ may arise from a lucky guess or context-dependent knowledge activation.
- **F-F** If the model cannot make a correct answer on both x and x^+, it lacks any relevant knowledge to such an extent that not even reasoning or guessing can enable it to provide a correct answer. The knowledge gap under this condition is assumed to be the largest.

We refer to the method of knowledge-invariant perturbation methods in PERTEVAL Li et al. (2024). With this, we additionally introduced a new options method and an irrelevant context method, thereby increasing the perturbation intensity. A showcase of perturbations is presented in Table 1.

Table 1: The applied perturbations and a perturbed example of **MCQ** and **SAQ**.

Data	Perturbation	Showcase (x / x^+)
MCQ	Add Option + Index Caesar + Option Permutation + Option Format	**Q:** *Where does Lady Gaga reside?* A. `Vienna` B. `Portuguese Ceuta` C. `Uncertain` D. `Los Angeles`✓ --- {I} `Dresden` {J} `Uncertain` {K} `Vienna` {L} `Los Angeles`✓ {M} `Humppila` {N} `Portuguese Ceuta`
SAQ	Rewrite + Irrelevant Context	**Q:** *In 1999 Anna Kournikova signed a lucrative contract to model what?* --- **Q:** *(In 1994 and 1995, gangs of armed youth destroyed the homes of foreign nationals living in Johannesburg, demanding that the police work to repatriate them to their home countries.) What did Anna Kournikova sign a lucrative contract to model in 1999?*

We design five levels of perturbation, with details presented in Table 11. For multiple-choice questions (**MCQ**), we add more distractor options matching the *relation* field in WIKIDATA (e.g., institution/date/region) and change the options structure (e.g., IDs/format/order) across five intensity levels. For the short-answer question (**SAQ**), we first rewrite the question while keeping the knowledge invariance by leveraging an LLM prompt (Figure 10). Then, we prepend unrelated context which is randomly drawn from the other Wiki page context. We consider five levels of perturbation intensity, determined by the added context length, which is allowed to vary by ±10 words.

Ultimately, the consistency score is measured based on these pairs of perturbation intensity and consistency outcome. The consistency score s is computed as either the average (s_{avg}) or the maximum ($s_{\max}$) intensity that the model could resist. Given a set of perturbations applied to a sample x, the model outputs are $\{y, y_1^+, \ldots, y_m^+\}$. Let $\{h_i = (y, y_i^+) | h_i \in \mathcal{H}\}_{i=1}^m$ be the set of consistency outcomes according to the correctness before and after perturbations.

$$s_{\text{avg}} = \frac{1}{m} \sum_{i=1}^{m} \left(v_i \cdot s_{h_i} \right), \quad s_{\max} = \max_{1 \le i \le m} \left(v_i \cdot s_{h_i} \right). \tag{1}$$

3.3 THE *Knowledge Updater*

The *Knowledge Updater* dynamically modifies the concept's *proficiency* $\boldsymbol{p}$ based on the consistency score. Underperforming concepts trigger score reductions, scaled by concept *importance*, while strong performance leads to proportional increases. The update process is formalized in Algorithm 1.

The algorithm takes a sample's associated *concepts* $\boldsymbol{c}$ with *importance* $\boldsymbol{w}$, as well as the inconsistency *score* s as input. The proficiency is increased if the consistency score exceeds a threshold θ, and decreased otherwise. The adjustment of *proficiency* Δp_i is computed via

$$\Delta p_i = \begin{cases} \eta_i \cdot (s - \theta) \cdot w_i \cdot (1 - p_i), & s \ge \theta \\ \eta_i \cdot (s - \theta) \cdot w_i \cdot p_i, & s < \theta. \end{cases} \tag{2}$$

The learning rate η_i is set per *concept* i and jointly influenced by v_i and r_i. Let t be the number of times the concept is evaluated. Here $v_i = \sigma({\Delta p_i}^{t-k:t})$ is the standard deviation of the k historical adjustments of p_i, reflecting volatility, and $r_i = \mu({s_i}^{t-k:t})$ the mean of historical scores s, reflecting accuracy. Then, a *sigmoid* operator is applied to smooth the influence of δ_i. Consider the testing process as a static inspection of the model's knowledge level, the proficiency vector is assumed to eventually converge at a stable point. Therefore, when mastery of a concept exhibits high volatility (larger v_i) and low accuracy (smaller r_i), η_i is expected to accelerate adjustment. Conversely, when volatility is low (smaller v_i) and accuracy is high (larger r_i), η_i is expected to decrease to stabilize results. To achieve that, v_i and r_i are joined together in a multiplicative way to synergistically form the learning rate: $\eta_i = \eta_{\min} + (\eta_{\max} - \eta_{\min}) \cdot v_i' \cdot r_i'$.

Algorithm 1 Knowledge Updating Algorithm

Input: Concept index set $\mathcal{I}_{\mathbb{C}}$, concept importance $\boldsymbol{w}^t = \{w_i^t\}_{i\in\mathcal{I}_{\mathbb{C}}}$, consistency score s^t, current proficiency $\boldsymbol{p}^t = \{p_i^t\}_{i\in\mathcal{I}_{\mathbb{C}}}$, parameters θ, p_{init}, $\eta_{\min}$, $\eta_{\max}$, k, a, b, v_0, r_0.
Output: Updated proficiency $\boldsymbol{p}^{t+1} = \{p_i^{t+1}\}_{i\in\mathcal{I}_{\mathbb{C}}}$.

For new concept c_i, initialize $p_i = p_{\text{init}}$;
for each concept c_i **do**
  if $w_i^t > 0$ **then**
    Compute learning rate:
    $v_i \leftarrow \sigma(\Delta {p_i}^{t-k:t})$, $r_i \leftarrow \mu({\boldsymbol{s}_i}^{t-k:t})$;
    $v_i' \leftarrow \frac{1}{2}(1 + \text{Sigmoid}(a(v_i - v_0)))$;
    $r_i' \leftarrow 1 - \frac{1}{2}(1 + \text{Sigmoid}(b(r_i - r_0)))$;
    $\eta_i \leftarrow \eta_{\min} + (\eta_{\max} - \eta_{\min}) \cdot v_i' \cdot r_i'$;
    Proficiency update:
    if $s^t \geq \theta$ **then**
      $\Delta p_i^{t+1} \leftarrow \eta_i (s^t - \theta)\, w_i^t (1 - p_i^t)$;
    else
      $\Delta p_i^{t+1} \leftarrow \eta_i (s^t - \theta)\, w_i^t\, p_i^t$;
    end if
    $p_i^{t+1} \leftarrow p_i^t + \Delta p_i^{t+1}$;
  end if
end for
return $\boldsymbol{p}^{t+1}$

4 Experiment Settings

4.1 Ground Truth Dataset

We leverage the *pageviews* of Wikipedia entities as a plausible proxy of the ground truth of model knowledge. *Pageviews* is an important indicator in the field of Wikipedia and its derived knowledge graphs (such as DBpedia and Wikidata) have also been widely used in past research as a key metric for model knowledge Sun et al., entity occurrence frequency Sciavolino et al. (2021), popularity Mallen et al. (2023), etc. In our setting, the model knowledge should be inherently related to what the model has learned, but the training corpus of LLMs is huge and unknowable. Hence, we use *pageviews* to best approximate but not as a perfect measure of the *concept*'s *popularity*. The higher the *popularity*, the greater the probability that the *concept* appears in the training data. Consequently, with sufficient learning iterations, the model is expected to have a better mastery of that *concept*.

The *pageviews* dataset is the number of all Wiki article visits dumped each day, which is in the form {*title*: *pageviews*}. We select dumps from the first day of each month, spanning from March 2024 to March 2025, to enhance coverage and accuracy. Then the *pageviews* from 12 different dumps are aggregated via summation.

4.2 Evaluation Dataset

We employ two benchmark datasets linked to Wikipedia *pageviews* via the *title*. We retain only samples with complete *pageview* data, see Appendix B.1.1 for the matching and filtering procedure.

- ConflictBank, proposed by Su et al. (2024), is a multiple-choice question dataset (**MCQ**) made from Wiki facts with a total number of 81,445 samples. We use this dataset to evaluate model consistency under format-level perturbations. Each question has a unique Wiki *title*, and thus is linked with the *pageviews*.
- TriviaQA, proposed by Joshi et al. (2017), which is a short-answer question dataset (**SAQ**) of 138,383 samples ideal for testing content-level perturbations through its word/phrase-oriented QA format. The difference is that each question here is associated with multiple linked Wiki titles. We first assign pageviews to each individual title, then compute an overall value through averaging, summation, or taking the maximum. Appendix B.3 shows that different aggregation methods exhibit similar patterns.

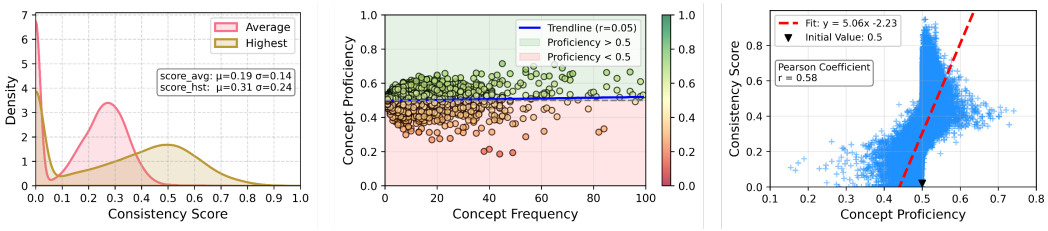


Figure 3: (**Left**) Consistency score distribution of TRIVIAQA under `LLaMA`. (**Middle**) Concept proficiency-frequency. (**Right**) The relationship between concept mastery and average consistency.

4.3 MODELS

We utilize three state-of-the-art open-source language models: `LLaMA-3.1-8B` Touvron et al. (2023), `Qwen2.5-7B-Instruct` Bai et al. (2023), and `Mistral-7B` Jiang et al. (2023), allowing for controlled, computationally feasible large-scale experiments. As these models were released in 2024, their knowledge is expected to be broadly compatible with the WIKIPEDIA *pageview* collection window. Moreover, since both datasets consist predominantly of static and enduring facts whose relative popularity remains temporally stable, WIKIPEDIA *pageviews* provide a reasonable proxy for the distribution of model knowledge exposure. A detailed validation is provided in Appendix B.1.2.

4.4 PARAMETER SETTINGS

We define case scores as $s_{\mathcal{H}} = \{(\text{T-T}, 1.0), (\text{T-F}, 0.5), (\text{F-T}, 0.0), (\text{F-F}, 0.0)\}$, reflecting decreasing knowledge levels: full knowledge (T-T), partial knowledge (T-F), and no knowledge (F-T and F-F).

The complete idea for selecting parameters of Algorithm 1 is detailed in Appendix B.6. We adopt $p_0 = 0.5$ as the initial proficiency for each concept, representing a neutral prior that assumes no prior knowledge advantage or disadvantage. Following standard practice in iterative learning algorithms, we set the learning rate bounds to $\eta_{\min} = 0.1$ and $\eta_{\max} = 1.0$. The choice of a relatively large $\eta_{\max}$ is motivated by the observed experimental fact that most concepts receive fewer than 10 updates; a smaller value would hinder timely differentiation of proficiency levels. The update threshold is set to $\theta = 0.31$, which equals the empirical mean of $s_{\max}$. This value reflects the average knowledge level of the model and provides an optimal decision boundary for distinguishing fine-grained variations in concept mastery. We use a history window of size $k = 20$ to compute volatility and correctness. Finally, the smoothing parameters are configured as $a = b = 5.0$, $v_0 = 0.3$, and $r_0 = 0.5$. These parameters are designed to produce a symmetric learning rate response surface in terms of volatility (v) and correctness (r), ensuring balanced adaptation in cases of both high correctness with high volatility and low correctness with low volatility.

4.5 BASELINE INCONSISTENCY PERFORMANCE

We first emphasize that each component of `KnowledgeMap` has been independently validated for stability and usability in Appendix C, where human evaluation further shows strong alignment with GPT-4 automatic assessment. In this section, we therefore present only the overall consistency performance of the selected models.

Table 2: Proportion of consistent group samples across different models.

	MCQ				**SAQ**			
	T-T	**T-F**	**F-T**	**F-F**	**T-T**	**T-F**	**F-T**	**F-F**
`LLaMA`	47.45%	17.74%	11.35%	23.45%	48.14%	12.70%	5.44%	33.71%
`Mistral`	39.74%	15.64%	9.13%	35.49%	40.01%	15.02%	4.15%	40.82%
`Qwen`	35.15%	15.14%	9.47%	40.24%	38.46%	11.17%	5.12%	45.24%

The baseline consistency situation is given in Table 2. These results support the core research hypothesis that knowledge deficiency is strongly associated with model inconsistency and that knowledge gaps vary significantly and show different patterns across different models. Except for `Qwen`,

Table 3: Significance of *popularity* mean differences under *t*-test across consistency groups.

	LLaMA				Mistral				Qwen			
	mean	significance			mean	significance			mean	significance		
MCQ												
T-T	3,280	**T-T**	**T-F**	**F-T**	3,647	**T-T**	**T-F**	**F-T**	3,891	**T-T**	**T-F**	**F-T**
T-F	2,302	***			2,136	***			2,563	***		
F-T	2,655	***	*		3,376	NS	***		2,380	***	NS	
F-F	2,246	***	NS	**	1,947	***	NS	***	2,018	***	***	**
SAQ												
T-T	15,895	**T-T**	**T-F**	**F-T**	16,425	**T-T**	**T-F**	**F-T**	17,811	**T-T**	**T-F**	**F-T**
T-F	13,875	***			14,819	NS			14,395	***		
F-T	13,441	**	NS		14,079	NS	NS		14,096	***	NS	
F-F	9,481	***	***	***	9,825	***	***	***	9,196	***	***	***

✶ Significance levels are denoted as: *** ($p<0.001$), ** ($p<0.01$), * ($p<0.05$), and NS (not significant, $p\geq0.05$).

the T-T group proportions of the remaining models in both tasks are higher than those of the other groups, indicating that the models can resist perturbations to a certain extent. LLaMA demonstrates the best overall consistency performance, with the highest T-T group proportions in both tasks. The T-T group proportions of Mistral and Qwen decrease successively, while their F-F group proportions increase, reflecting that they face larger knowledge gaps to varying degrees. The T-F group proportions remain relatively stable across models, indicating that all models exhibit a certain degree of partial knowledge mastery, where perturbations can trigger inconsistency.

5 Interpretation of Inconsistency

5.1 Inconsistency does stem from knowledge deficiency.

We analyze whether inconsistency arises from knowledge gaps. High-popularity samples reflect common knowledge that is likely well-learned by LLMs, while low-popularity data indicate niche concepts with weaker knowledge internalization. We analyze *popularity* differences between samples where LLMs show varying consistency patterns. Table 3 presents average popularity per group with pairwise *t*-test results. The divergence in popularity across different consistency groups is visualized in Figure 2. The x-axis shows popularity quantiles of all samples ranging from 5% to 50% in 5% intervals. The y-axis plots the ratio of the number of samples from different consistency groups within the top-x% and the bottom-x% quantile. The higher this ratio is, the more samples are concentrated in the high *popularity* quantile, indicating better consistency.

The results reflect that, across all datasets and models, the T-T group has significantly higher *popularity* compared to other groups ($p<0.001$), confirming that high-popularity knowledge yields stronger consistency. In ConflictBank, the T-F, F-T, and F-F groups show overlapping popularity ratios, revealing the model's confusion and uncertainty under a lack of knowledge perturbations. An exception arises in Mistral, where the F-T group exhibits higher *popularity* than some non-T-T groups, though still below T-T; this anomaly is absent in TriviaQA. One possible explanation is that the model possesses the relevant knowledge (as evidenced by correct responses under perturbation), but its retrieval is context-dependent in Mistral. Regarding the TriviaQA dataset, the distinction among the four groups is more pronounced, following a clear hierarchy: T-T >T-F >F-T >F-F. Therefore, we can confirm that T-T indicates the presence of knowledge, T-F and F-T suggest ambiguity or partial knowledge, and F-F implies minimal or no knowledge. Given that the open-ended short-answer format of TriviaQA has a lower likelihood of correct random guessing compared to the multiple-choice format of ConflictBank, we posit that TriviaQA provides a more accurate reflection of the underlying knowledge state.

5.2 Which knowledge? Compound and confused concepts.

Based on the evaluation in Table 2 and Figure 2 from the previous sections, we observe that LLaMA serves as the best-performing model and the open-ended answers in TriviaQA demonstrate better discriminative ability for assessing model knowledge. The analysis in the following sections will therefore focus on LLaMA and TriviaQA.

To further interpret the characteristics of inconsistency, we investigate the influence of compound knowledge on the model's consistency performance. Co-occurrence concepts are defined as the number of concepts simultaneously extracted by the *Knowledge Extractor* within a single sample. Figure 4 depicts the average consistency *score* of the samples under different numbers of co-occurring concepts. It demonstrates a clear downward trend, declining from 0.3342 of 1,454 single-concept samples to 0.2082 of 308 10-concept samples. Moreover, when the co-occurrence concepts exceed 15, the trend of decreased consistency becomes even more pronounced. Therefore, it indicates that the more co-occurrence concepts there are, the more knowledge concepts need to be integrated to answer the question correctly, and the more severe the inconsistency becomes.

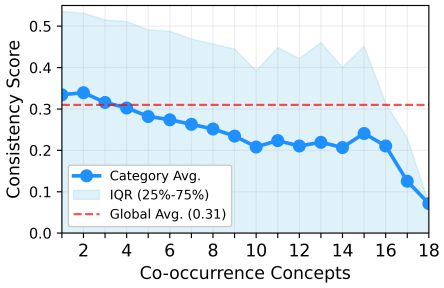


Figure 4: Consistency decreases with concept complexity increases.

5.3 Locating the culprits: local deficiencies contaminate overall consistency.

Previous results have demonstrated that LLMs exhibit greater inconsistency when processing unfamiliar and multi-concept knowledge.

We then examine how low-proficiency concepts influence consistency when co-occurring with other concepts. The core hypothesis states that samples containing deficient knowledge should exhibit significantly lower consistency than those without such gaps, directly linking concept-level deficiencies to the overall inconsistency patterns. According to the middle of Figure 3, we select 218 concepts with $p < 0.4$ as low-proficiency concepts and analyze their impact on overall consistency. For each concept with low *proficiency* ($\mathbf{P}_1$), we find all related samples that contain that concept to obtain the average consistency ($\mathbf{S}_1$) with average *popularity* ($\mathbf{O}_1$) as the ground truth proxy. For comparison, for every other concept that used to co-occur with low-proficiency concepts, we take its proficiency ($\mathbf{P}_2$), and find all its related samples that exclude any low-proficiency concepts, followed by computing its consistency ($\mathbf{S}_2$) and *popularity* ($\mathbf{O}_2$). Here, the *popularity* of TriviaQA sample is taken as the largest one among the *popularity* of multiple titles.

Table 4: Consistency drop caused by low-proficiency concepts. Low: samples containing low-proficiency concepts ($p < 0.4$); Other: co-occurring concepts' samples excluding them.

Metric	Low_1	Other_2	Avg. | (Δabs/Δrel)
Proficiency (**P**)	0.3174	0.4664	−0.1490 / −46.93%
Consistency (**S**)	0.2264	0.2780	−0.0516 / **−22.77%**
Popularity (**O**)	11,518	12,686	−1,168 / **−10.14%**

✶ Δabs = absolute difference, Δrel = relative change.

From Table 4, the selected low-proficiency concepts exhibit an average *proficiency* 46.93% lower than other concepts. The analysis reveals that when these low-proficiency concepts co-occur with normal concepts, they reduce sample consistency by approximately 22.77%. By referencing the *popularity* as the proxy of ground truth, we further find that samples containing low-proficiency concepts show 10.14% lower *popularity* on average compared to samples with only normal concepts. These findings demonstrate that the model's overall inconsistency is fundamentally correlated to insufficient learning of specific knowledge concepts within the entire knowledge system.

6 Conclusion

In this work, we have proposed `KnowledgeMap` to study LLM inconsistency from the perspective of knowledge deficiency, a challenge that constantly undermines the trustworthiness and reliability of LLM-powered applications. By leveraging `KnowledgeMap` at scale on diverse datasets and models, we have provided empirical evidence for a robust association between knowledge gaps and inconsistencies. We have identified the specific knowledge deficiencies at the concept level and quantified their non-negligible impact on overall consistency performance. Our work extends the question from *whether* LLMs are inconsistent to *which* knowledge makes them so, yielding concrete and actionable targets for practitioners to intervene.

AI USE STATEMENT

In this work, we used generative AI tools to improve grammar and check for typos. All AI-suggested edits were manually checked by the authors to preserve the original intent. The final wording and framing of the paper were decided by the authors. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work analyzes the knowledge gaps underlying LLM inconsistency and does not involve collecting new data. All datasets used are publicly available and widely adopted benchmarks in the community: CONFLICTBANK, constructed from Wikidata facts, and TRIVIAQA, whose questions are linked to Wikipedia entities. Both were used in accordance with their respective licenses and intended research purposes. The popularity signal is also derived from the public, aggregated Wikimedia pageview dumps, which contain only article-level view counts and no user-level or personally identifiable information. The irrelevant contexts used for perturbation are likewise drawn from public Wikipedia pages.

REPRODUCIBILITY STATEMENT

Several measures are in place to ensure the reproducibility of our work. The complete source code is available via the anonymous link provided at the end of the abstract. The method as well as all hyperparameters, including case scores, update threshold, learning-rate bounds, history window, and smoothing parameters, are clearly listed in Section 4.4. All prompts used for concept extraction, answer generation, and question rewriting are provided verbatim in Figures 8–11, together with the output validation and retry rules in Appendix B.1. Further, Appendix C reports the robustness of the concept extractor across 10 repeated runs, which characterizes the variance introduced by LLM-based components.

LIMITATIONS

Our study focuses on evaluating inconsistency at the level of manifested conceptual knowledge. However, inconsistency in LLMs is multifactorial and may arise either from alignment-induced behaviors introduced through fine-tuning or from reasoning over their knowledge. To isolate this phenomenon from knowledge effects as cleanly as possible, we restrict our analysis to neutral factual QA tasks, where alignment-related behaviors such as safety refusal or stylistic overfitting are substantially less prevalent. In addition, responses violating the predefined output format are excluded from evaluation, ensuring that inconsistencies are measured only among valid, task-relevant generations. Under this controlled setting, the strong association between inconsistency and concept popularity (Figure 2) further suggests that disparities in knowledge exposure during pre-training remain a substantial contributor to observed failures. Consequently, our framework should be interpreted as a diagnostic lens for identifying fragile regions of model knowledge, independent of whether such fragility originates from pre-training deficiencies or downstream alignment effects. A structured investigation into the multiple causes of inconsistency is a worthwhile direction for future work.

Another difficulty is the lack of a ground-truth dataset. As explained in Section 4, we take WIKIPEDIA entity *pageviews* as an approximation of LLMs' inherent knowledge. This might introduce bias to our measurement result, as web popularity does not perfectly align with the training distribution, and that training distribution is not essentially equivalent to the LLMs' truly internalized knowledge. While acknowledging the existence of such a bias, *pageviews* remain a feasible approximation of model knowledge that is currently available, and we provide a detailed discussion and validation in Appendix B.1.2. Furthermore, our critical finding suggests that models are, in fact, more consistent on concepts that are more common in the world, which is an intuitive and meaningful result. There is a statistically significant difference in popularity between the T-T group and all other groups (Table 3). Also, the strong correlation (r=0.58, Figure 3-Right) between our

model-derived proficiency scores and this external popularity metric provides convergent validity for our method.

REFERENCES

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

Rachit Bansal and Danish Pruthi. Measures of Information Reflect Memorization Patterns. In *NeurIPS*, 2022.

Anna Bavaresco, Raffaella Bernardi, Leonardo Bertolazzi, Desmond Elliott, Raquel Fernández, Albert Gatt, Esam Ghaleb, Mario Giulianelli, Michael Hanna, Alexander Koller, et al. Llms instead of human judges? a large scale empirical study across 20 nlp evaluation tasks. In *ACL*, 2025.

Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. On the dangers of stochastic parrots: Can language models be too big? In *ACM FAccT*, 2021.

Gavin Brown, Mark Bun, Vitaly Feldman, Adam Smith, and Kunal Talwar. When Is Memorization of Irrelevant Training Data Necessary for High-Accuracy Learning? In *ACM STOC*, 2021.

Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. Quantifying Memorization Across Neural Language Models. In *PMLR ICLR*, 2023.

Hanzhu Chen, Xu Shen, Qitan Lv, Jie Wang, Xiaoqi Ni, and Jieping Ye. Sac-kg: Exploiting large language models as skilled automatic constructors for domain knowledge graph. In *ACL*, 2024.

Qinyuan Cheng, Tianxiang Sun, Xiangyang Liu, Wenwei Zhang, Zhangyue Yin, Shimin Li, Linyang Li, Zhengfu He, Kai Chen, and Xipeng Qiu. Can AI Assistants Know What They Don't Know? In *ICML*, 2024.

Verna Dankers and Ivan Titov. Generalisation First, Memorisation Second? Memorisation Localisation for Natural Language Classification Tasks. In *ACL*, 2024.

Qingxiu Dong, Jingjing Xu, Lingpeng Kong, Zhifang Sui, and Lei Li. Statistical knowledge assessment for large language models. In *NeurIPS*, 2023.

Avinava Dubey, Zhe Feng, Rahul Kidambi, Aranyak Mehta, and Di Wang. Auctions with llm summaries. In *ACM SIGKDD*, 2024.

Yanai Elazar, Nora Kassner, Shauli Ravfogel, Abhilasha Ravichander, Eduard Hovy, Hinrich Schütze, and Yoav Goldberg. Measuring and improving consistency in pretrained language models. *Transactions of the Association for Computational Linguistics*, 9:1012–1031, 2021.

Vitaly Feldman and Chiyuan Zhang. What Neural Networks Memorize and Why: Discovering the Long Tail via Influence Estimation. In *NeurIPS* , volume 33, pp. 2881–2891. Curran Associates, Inc., 2020.

Raya Fidel and Efthimis N Efthimiadis. Terminological knowledge structure for intermediary expert systems. *Information Processing & Management*, 31(1):15–27, 1995.

Zorik Gekhman, Gal Yona, Roee Aharoni, Matan Eyal, Amir Feder, Roi Reichart, and Jonathan Herzig. Does Fine-Tuning LLMs on New Knowledge Encourage Hallucinations? In *ACL EMNLP*, 2024.

Jiuzhou Han, Nigel Collier, Wray Buntine, and Ehsan Shareghi. Pive: Prompting with iterative verification improving graph-based generative capability of llms. In *ACL*, 2024.

Myeongjun Jang, Deuk Sin Kwon, and Thomas Lukasiewicz. BECEL: Benchmark for Consistency Evaluation of Language Models. In *ICCL COLING*, 2022.

Albert Qiaochu Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, L'elio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.

Han Jiang, Xiaoyuan Yi, Zhihua Wei, Ziang Xiao, Shu Wang, and Xing Xie. Raising the Bar: Investigating the Values of Large Language Models via Generative Evolving Testing, 2025.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. TriviaQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension. In *ACL*, 2017.

Sunnie S. Y. Kim, Q. Vera Liao, Mihaela Vorvoreanu, Stephanie Ballard, and Jennifer Wortman Vaughan. "I'm Not Sure, But...": Examining the Impact of Large Language Models' Uncertainty Expression on User Reliance and Trust. In *ACM FAccT*, 2024.

Jiatong Li, Renjun Hu, Kunzhe Huang, Yan Zhuang, Qi Liu, Mengxiao Zhu, Xing Shi, and Wei Lin. Perteval: Unveiling real knowledge capacity of llms with knowledge-invariant perturbations. In *NeurIPS*, 2024.

Zhong Li, Jiequn Han, Qianxiao Li, et al. On the curse of memory in recurrent neural networks: Approximation and optimization analysis. *arXiv preprint arXiv:2009.07799*, 2020.

Xun Liang, Shichao Song, Zifan Zheng, Hanyu Wang, Qingchen Yu, Xunkai Li, Rong-Hua Li, Yi Wang, Zhonghao Wang, Feiyu Xiong, and Zhiyu Li. Internal Consistency and Self-Feedback in Large Language Models: A Survey, 2024.

Aiwei Liu, Qiang Sheng, and Xuming Hu. Preventing and detecting misinformation generated by large language models. In *ACM SIGIR*, 2024.

Shayne Longpre, Kartik Perisetla, Anthony Chen, Nikhil Ramesh, Chris DuBois, and Sameer Singh. Entity-Based Knowledge Conflicts in Question Answering. In *ACL EMNLP*, 2021.

Xingyu Lu, Xiaonan Li, Qinyuan Cheng, Kai Ding, Xuanjing Huang, and Xipeng Qiu. Scaling Laws for Fact Memorization of Large Language Models. In *ACL EMNLP*, 2024.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. When Not to Trust Language Models: Investigating Effectiveness of Parametric and Non-Parametric Memories. In *ACL*, 2023.

Olena Medelyan, Ian H Witten, Anna Divoli, and Jeen Broekstra. Automatic construction of lexicons, taxonomies, ontologies, and other knowledge structures. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 3(4):257–279, 2013.

Nandana Mihindukulasooriya, Sanju Tiwari, Carlos F Enguix, and Kusum Lata. Text2kgbench: A benchmark for ontology-driven knowledge graph generation from text. In *Springer ISWC*, 2023.

Yikang Pan, Liangming Pan, Wenhu Chen, Preslav Nakov, Min-Yen Kan, and William Yang Wang. On the risk of misinformation pollution with large language models. *arXiv preprint arXiv:2305.13661*, 2023.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. Language Models as Knowledge Bases? In *ACL EMNLP-IJCNLP*, 2019.

Sergey Pletenev, Maria Marina, Daniil Moskovskiy, Vasily Konovalov, Pavel Braslavski, Alexander Panchenko, and Mikhail Salnikov. How Much Knowledge Can You Pack into a LoRA Adapter without Harming LLM? In *ACL NAACL*, 2025.

Harsh Raj, Vipul Gupta, Domenic Rosati, and Subhabrata Majumdar. Semantic consistency for assuring reliability of large language models. *arXiv preprint arXiv:2308.09138*, 2023.

Sanjana Ramprasad, Elisa Ferracane, and Zachary C Lipton. Analyzing llm behavior in dialogue summarization: Unveiling circumstantial hallucination trends. In *ACL*, 2024.

Christopher Sciavolino, Zexuan Zhong, Jinhyuk Lee, and Danqi Chen. Simple entity-centric questions challenge dense retrievers. In *ACL EMNLP*, 2021.

Rickard Stureborg, Dimitris Alikaniotis, and Yoshi Suhara. Large Language Models are Inconsistent and Biased Evaluators. In *ACL*, 2024.

Zhaochen Su, Jun Zhang, Xiaoye Qu, Tong Zhu, Yanshu Li, Jiashuo Sun, Juntao Li, Min Zhang, and Yu Cheng. ConflictBank: A Benchmark for Evaluating the Influence of Knowledge Conflicts in LLM. In *NeurIPS*. arXiv, 2024. doi: 10.48550/arXiv.2408.12076.

Kai Sun, Yifan Xu, Hanwen Zha, Yue Liu, and Xin Luna Dong. Head-to-Tail: How Knowledgeable are Large Language Models (LLMs)? A.K.A. Will LLMs Replace Knowledge Graphs? In *ACL NAACL*.

Christian Tomani, Kamalika Chaudhuri, Ivan Evtimov, Daniel Cremers, and Mark Ibrahim. Uncertainty-Based Abstention in LLMs Improves Safety and Reduces Hallucinations, 2024.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Tempest A van Schaik and Brittany Pugh. A field guide to automatic evaluation of llm-generated summaries. In *ACM SIGIR*, 2024.

Chenguang Wang, Xiao Liu, Zui Chen, Haoyun Hong, Jie Tang, and Dawn Song. Zero-shot information extraction as a unified text-to-triple translation. In *EMNLP*, 2021.

Dawei Wang, Geng Zhou, Xianglong Li, Yu Bai, Li Chen, Ting Qin, Jian Sun, and Dan Li. The Digital Cybersecurity Expert: How Far Have We Come? In *IEEE S&P*, 2025.

Wenhao Wang, Adam Dziedzic, Michael Backes, and Franziska Boenisch. Localizing Memorization in SSL Vision Encoders. In *NeurIPS*, 2024a.

Xinyi Wang, Antonis Antoniades, Yanai Elazar, Alfonso Amayuelas, Alon Albalak, Kexun Zhang, and William Yang Wang. Generalization v.s. Memorization: Tracing Language Models' Capabilities Back to Pretraining Data. In *ICML FM-Wild Workshop*, 2024b.

Jian Xie, Kai Zhang, Jiangjie Chen, Renze Lou, and Yu Su. Adaptive Chameleon or Stubborn Sloth: Revealing the Behavior of Large Language Models in Knowledge Conflicts, 2024.

Rongwu Xu, Zehan Qi, Zhijiang Guo, Cunxiang Wang, Hongru Wang, Yue Zhang, and Wei Xu. Knowledge Conflicts for LLMs: A Survey. In *ACL EMNLP*, 2024.

Jingyuan Yang, Dapeng Chen, Yajing Sun, Rongjun Li, Zhiyong Feng, and Wei Peng. Enhancing Semantic Consistency of Large Language Models through Model Editing: An Interpretability-Oriented Approach. In *Findings of the Association for Computational Linguistics ACL 2024*, 2024a.

Sohee Yang, Elena Gribovskaya, Nora Kassner, Mor Geva, and Sebastian Riedel. Do large language models latently perform multi-hop reasoning? In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2024b.

Sangyoon Yi and Jinho Choi. The organization of scientific knowledge: the structural characteristics of keyword networks. *Scientometrics*, 90(3):1015–1026, 2012.

Xunjian Yin, Xu Zhang, Jie Ruan, and Xiaojun Wan. Benchmarking Knowledge Boundary for Large Language Models: A Different Perspective on Model Evaluation. In *ACL*. arXiv, 2024. doi: 10.48550/arXiv.2402.11493.

Jifan Yu, Xiaozhi Wang, Shangqing Tu, Shulin Cao, Daniel Zhang-Li, Xin Lv, Hao Peng, Zijun Yao, Xiaohan Zhang, Hanming Li, Chunyang Li, Zheyuan Zhang, Yushi Bai, Yantao Liu, Amy Xin, Nianyi Lin, Kaifeng Yun, Linlu Gong, Jianhui Chen, Zhili Wu, Yunjia Qi, Weikai Li, Yong Guan, Kaisheng Zeng, Ji Qi, Hailong Jin, Jinxin Liu, Yu Gu, Yuan Yao, Ning Ding, Lei Hou, Zhiyuan Liu, Bin Xu, Jie Tang, and Juanzi Li. KoLA: Carefully Benchmarking World Knowledge of Large Language Models. In *ICLR*, 2024.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. In *NeurIPS*, 2023.

A Extended Related Work

A.1 Knowledge representation and utilization

To explore the model's ability to handle diverse knowledge, knowledge is further refined into whether the model has learned Yu et al. (2024) or is familiar Pletenev et al. (2025) with it, and under what conditions can knowledge be effectively utilized Yin et al. (2024). In addition, it is also essential to study whether the model itself is aware of the knowledge it does not know Cheng et al. (2024) for mitigating hallucination. In most studies, knowledge is conceptualized as concrete textCarlini et al. (2023), imageWang et al. (2024a), or factual informationPetroni et al. (2019), while another branch of research abstracts knowledge into structural and parametric representations within networks Bansal & Pruthi (2022); Wang et al. (2024a); Dankers & Titov (2024), characterizing different types of knowledge by their parametric features.

A.2 Knowledge capacity and internalization

Research has been conducted to investigate how knowledgeable LLMs are in areas of factual knowledgeYu et al. (2024), ethical issuesJiang et al. (2025), specific technical fieldsWang et al. (2025), etc. Current research has confirmed that larger models usually have a greater capacity for knowledge Lu et al. (2024). Moreover, during the early stage of pre-training, the model learns knowledge at a faster rate, and as the training progresses, new knowledge requires more neurons for approximation, and the difficulty increases exponentially Li et al. (2020). Simultaneously, the model's reasoning ability is enhanced to compensate for its limited knowledge capacity, and the output is essentially a mixture of the learned knowledge with generalization Dankers & Titov (2024); Wang et al. (2024b). Under such observation, new knowledge without pre-learned knowledge at a later fine-tuning stage becomes a reason for hallucination Gekhman et al. (2024). It has also been observed that precise learning of training samples could impede generalization Bansal & Pruthi (2022), but when even irrelevant data are retained, the model generalizes better Feldman & Zhang (2020); Brown et al. (2021). Therefore, understanding the extent of a model's internalized knowledge is crucial for improving performance.

A.3 Causes and mitigation of inconsistency

Substantial research has demonstrated that LLMs could exhibit a non-trivial degree of inconsistency Elazar et al. (2021); Dong et al. (2023). Yet posing a trust crisis for various applications reliant on LLMs, this issue is generally not directly related to accuracy Raj et al. (2023), nor can it be mitigated by simply enlarging the model size Jang et al. (2022). Regarding how inconsistencies occur, a branch of research attributes such to knowledge conflict, including the conflict between or within parametric knowledge and context information Xu et al. (2024); Xie et al. (2024). This line of research creates conflicts in different aspects of the sample, such as time, place, characters, and relationships, and then evaluates the characteristics exhibited by the model under different conflicts. They demonstrate that LLMs are quite sensitive to context information Su et al. (2024), and popular but conflicting context data is more likely to affect the model when the model has already learned the sample during training Longpre et al. (2021). Another group of researchers holds that inconsistency is related to the model's ability of reasoning. They believe that inconsistencies arise as the semantics of the samples have not been truly understood, but rather mechanically learned rules or patterns from the training data Bender et al. (2021); Yang et al. (2024b). To overcome inconsistency, efforts have been made by adopting advanced methods such as self-feedback Liang et al. (2024), decoding strategy Tomani et al. (2024), model editing Yang et al. (2024a), etc.

B EXPERIMENT DETAILS

B.1 PROCESSING AND VALIDITY OF *pageviews* AS STABLE KNOWLEDGE PROXY

In this section, we first describe the procedure for matching *pageviews* to dataset samples, and then assess the robustness of *pageviews* as a stable proxy for the models' knowledge.

B.1.1 MATCHING AND FILTERING

Regarding the mapping from samples to WIKIPEDIA entities, both datasets are derived from WIKIPEDIA. The *title* field in TRIVIAQA and the *subject* field in CONFLICTBANK correspond directly to standard WIKIPEDIA entity titles, which match the page titles in the *pageview* dumps directly.

However, some samples may not have a pageview values in dumps. For these samples, we discard any sample for which a valid *pageview* cannot be retrieved. For TRIVIAQA samples associated with multiple WIKIPEDIA titles, we retain a sample only when *pageview* data are available for all of its titles, excluding any sample with incomplete data. Consequently, all results reported in this paper are based exclusively on the subset of samples with complete *pageview* information.

B.1.2 *Pageviews* AS STABLE PROXY

As the training data of these models is inherently unknowable, we do not aim to reconstruct this opaque corpus directly. Instead, we use the *pageviews* of WIKIPEDIA entities as a proxy for their popularity across the internet, which in turn approximates the distribution of the large-scale web corpora from which LLMs acquire knowledge.

Following the practice from previous work Sun et al.; Sciavolino et al. (2021); Mallen et al. (2023), we treat WIKIPEDIA *pageviews* as a ground-truth proxy for concept popularity rather than as a perfect mirror of an LLM's true training distribution. Several factors nonetheless support the validity of this proxy. The evaluated models are relatively recent. `LLaMA-3.1-8B` was released in July 2024, `Qwen2.5-7B` in September 2024, and `Mistral-7B` in December 2023, therefore their training corpora are temporally close to our *pageview* collection window (March 2024 to March 2025). In addition, both evaluation datasets, TRIVIAQA (released in 2017) and CONFLICTBANK (sourced from an April 2024 WIKIDATA snapshot), consist predominantly of static, enduring factual knowledge. In the following, we evaluate that for the vast majority of such concepts, both their relative popularity and their distribution in training corpora remain stable over time.

To examine potential post-treatment bias and the temporal stability of entity *pageviews*, we further conduct a robustness check comparing *pageview* data from 2021 to 2025. For each year, we use the first daily *pageviews* file (`pageviews-202{i}0101-000000.gz`, $i \in \{1, \ldots, 5\}$) and the first test split of TRIVIAQA (`test-00000-of-00001.parquet`), which contains 7,993 samples with 14,229 extracted entities. Across the five years, we successfully match 8,528, 8,311, 8,240, 8,290, and 8,243 TRIVIAQA entities, respectively. As a showcase, Table 5 illustrates the *pageview* values of the first 10 entities as a showcase. *High-popularity entities remain dominant throughout the observation windows, indicating a high degree of temporal stability.*

Considering that the overall scale of pageviews may exhibit interannual fluctuations due to expanding internet growth, this study converts raw pageviews into rankings to eliminate bias caused by aggregate drift. By calculating the Spearman's rank correlation coefficients across the five data groups, we evaluated the relative consistency of pageviews; the detailed analytical results are presented in Table 6. We found a high Spearman correlation ($\rho > 0.8$) across comparisons among five groups, *suggesting that the relative pageviews of concepts is a stable structural feature rather than a transient artifact*.

At last, we showcase the top-10 concepts from CONFLICTBANK with the highest popularity.

B.2 PROMPT TEMPLATES AND EXTRACTION FILTERING CONDITIONS.

This section details the prompts used in `KnowledgeMap`. Four prompts are used (Figure 8-11), corresponding to the following four stages: **P1**-knowledge extraction, (**P2**) **MCQ** answer generation,

Table 5: A showcase of *pageviews* value of the first 10 entities. The *pageviews* of the same entity on the same date across different years exhibit only minor fluctuations.

Entity	Pageviews				
	2021	2022	2023	2024	2025
Andrew Lloyd Webber	36	36	33	36	32
Prime Minister of the UK	40	54	41	41	45
Arthur Balfour	5	6	9	6	1
Kiss You All Over	2	–	1	–	1
Kathleen Ferrier	3	2	3	1	1
Lauren Bacall	57	49	54	39	36
Michael Jackson	226	173	234	158	157
Tanzania	63	31	35	44	44
Mount Kilimanjaro	23	15	21	20	38
Flag of Libya	5	4	3	5	5

Table 6: Spearman correlation matrix of *pageviews* across years. This indicates that, once the overall year-to-year drift in internet traffic is removed, the rank ordering of entity *pageviews* remains highly consistent across years.

	2021	2022	2023	2024	2025
2021	1.0000	0.8263	0.8105	0.8051	0.8017
2022	0.8263	1.0000	0.8102	0.8021	0.8008
2023	0.8105	0.8102	1.0000	0.8158	0.8059
2024	0.8051	0.8021	0.8158	1.0000	0.8219
2025	0.8017	0.8008	0.8059	0.8219	1.0000

(**P3**) **SAQ** question rephrasing, and (**P4**) **SAQ** answer generation. All prompts are provided in a one-shot example format to ensure the model generates structured responses, facilitating subsequent pattern-based data extraction.

To ensure the quality of the generated content, we validate the format of each LLM response when extracting structured information. If the response does not adhere to the instructed output format, we regenerate the response with a maximum of 5 retries. The validation process differs for each stage and is as follows. The result is only considered valid if all the above conditions are satisfied.

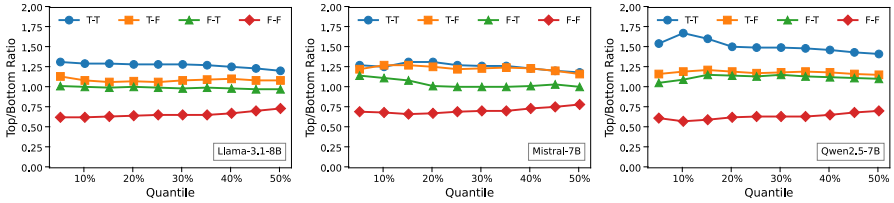
- *Knowledge Extraction* (**P1**): First, verify that the response format matches the specified structured template. Then, validate the data content against the following three constraints:
  - Each "key" must not exceed `WORD_LIMIT` words in length, which is set to 2;
  - The sum of "values" must fall within the range of 100 ± 10;
  - The total number of key-value pairs must not exceed `CONCEPT_LIMIT`, which is set to 10.
- *Knowledge Checker* (**P2-4**): We check the response format against the specified structural requirements. If it matches, the result is deemed valid.

B.3 Effects of titles' *popularity* aggregation methods.

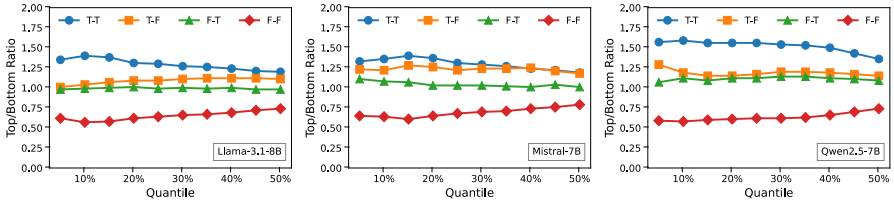
This is a supplement about the method of aggregating multiple titles' *popularity* corresponding to TriviaQA samples when they contain multiple titles. We test three aggregation methods of *popularity*: the average, maximum, and sum value, which respectively correspond to Figure 5a, Figure 5b, and Figure 2b. It can be found that no matter which aggregation method is adopted, the results clearly show an obvious T-T ¿ T-F ¿ F-T ¿ F-F pattern. This further indicates the robustness and rationality of using *popularity* as the ground truth. Ultimately, we presented the result of the summation method in the main body, as this method takes into account the contributions of the associated titles equally.

Table 7: Examples of *popularity* that conform to common sense.

Rank	***Concept***	***Popularity***
1	Kamala Harris	1,030,211
2	YouTube	888,739
3	Elon Musk	680,561
4	Katie Ledecky	633,645
5	Cristiano Ronaldo	595,247
6	Cynthia Erivo	492,521
7	Bob Dylan	473,261
8	Vontae Davis	472,216
9	Kris Kristofferson	457,217
10	Macaulay Culkin	390,194



(a) TRIVIAQA (Average Titles' Popularities)



(b) TRIVIAQA (Maximum Titles' Popularities)

Figure 5: Continuation of Figure 2: A comparison of knowledge mastery under different popularity aggregation methods for multiple titles in the TRIVIAQA sample.

B.4 FIVE LEVELS OF INTENSITY.

This section shows the rule for setting different levels of perturbation for MCQ and SAQ. Table 11 demonstrates examples of perturbing original questions through progressively intensified perturbation operations (**L1**-**L5**).

For MCQ, we perform format-level perturbation to progressively increase the complexity of strategies:

- **L1**. Change option indices to numerical labels and randomly change option format.
- **L2**. Add one extra option, randomly change option format, and randomly permute the order and format of options.
- **L3**. Add two extra options, randomly change option format, use numerical indices and random permutation.
- **L4**. Add three extra options, randomly change option format, apply random Caesar-shift indices with random formatting.
- **L5**. Add three extra options, randomly change option format, combine random Caesar-shift indices with numbered labels, and perform random permutation.

For SAQ, contexts of various lengths ([30, 50, 70, 90, 120] from **L1** to **L5**) are appended ahead of the original question. These distracting contexts are selected randomly from WIKIPEDIA pages. The selection of these lengths is based on our initial tests on the perturbation-resistant ability of large models. We found that the model is almost never disturbed by content with a length of less

than 30 words. To maintain the integrity of the sentence, the length of the context does not strictly equal these lengths but has a margin of ±1 words.

B.5 Measures of perturbation intensity.

As introduced in Section 3.2, the perturbation intensity is derived from the text similarity between samples before and after perturbation. To determine the most appropriate calculation method, we considered two aspects. First, the scope of similarity calculation, which compares either the question alone (**QSim**) or the question with the prompt (**PSim**). Second, the metric calculation approach, for which we experimented with the following four measures:

- **TFIDF**. This method first uses the TF-IDF vectorizer to convert the texts into numerical vectors, and then computes their cosine similarity. This is a surface-level matching approach based solely on vocabulary.
- **BERT**. This method utilizes a pre-trained Sentence-BERT model (`all-MiniLM-L6-v2`) to encode texts to their embeddings and then computes their cosine similarity, which can achieve in-depth semantic understanding.
- **BLEU**. This method operates by comparing the overlapping n-grams (contiguous word sequences) between texts, which stresses words' order most.
- **W2V**. This method uses the Word2Vec model pre-trained on Google News, and then takes the average of the vectors of each word in the sentence to obtain the sentence vector. Similarity is obtained by computing their cosine similarity. It has better semantic awareness compared to TF-IDF and is more lightweight compared to BERT, but it is not as good as BERT in terms of deep semantic understanding.

The results are presented in Table 12 and Table 13 for ConflictBank and TriviaQA, respectively. Results show that regardless of the dataset, the BERT method has the best discrimination and the most uniform value dispersion within the interval of [0,1]. Then, comparing QSim and PSim using BERT, PSim of ConflictBank is slightly higher than that of QSim because the perturbation of ConflictBank is at the option format level. However, the PSim of TriviaQA is significantly higher than that of QSim, and as the interference intensity increases, the difference between PSim and QSim gradually decreases. Considering that the model's understanding of the input also includes the understanding of prompts, we thus chose PSim+BERT as the calculation method for subsequent analysis.

B.6 Algorithm design and parameter selection.

This section provides additional details on the design of Algorithm 1 and the rationale behind its parameter selection. The overall objective is to assess the knowledge level of the LLMs while maintaining the assumption that the model's knowledge remains static during evaluation. Thus requiring the algorithm to converge. Since the adjustment rule is described in Section 3.3, here we focus primarily on the adjustment of the learning rate.

In Algorithm 1, v_i reflects the volatility of *proficiency* and r_i reflects the models' response correctness for that concept. When a concept shows high volatility in proficiency and low correctness, the learning rate should increase to accelerate adjustment towards the LLMs' true proficiency. Conversely, when volatility is small and correctness is high, the learning rate decreases to stabilize the result. The two factors form a synergistic adjustment through multiplication.

In the face of the collaborative design of v_i and r_i, and considering stable convergence, we require learning rate to have a reasonable response in different situations:

- *Large volatility + low correctness → maximum learning rate*;
- *Small volatility + high correctness → minimum learning rate*;
- *Large volatility + high correctness → mild learning rate*;
- *Small volatility + low correctness → mild learning rate*.

The learning rate is relatively large when there is large volatility and high correctness, and relatively small when there is small volatility and low correctness. This means that when both v_i r_i are high,

the learning rate should be relatively high. When both are small at the same time, the learning rate should remain mild. High values in both dimensions work together to raise the learning rate, while low values jointly suppress it.

To achieve that, we adopt a sigmoid-based coupling function as the algorithmic foundation. The adjustment rate explicitly follows the above principles, and the sigmoid function provides smooth transitions, preventing sharp changes in the learning rate. Finally, the parameter set that yields the most symmetric learning-rate response surface is selected via grid search. The learning rate response according to volatility and correctness is depicted in Figure 6.

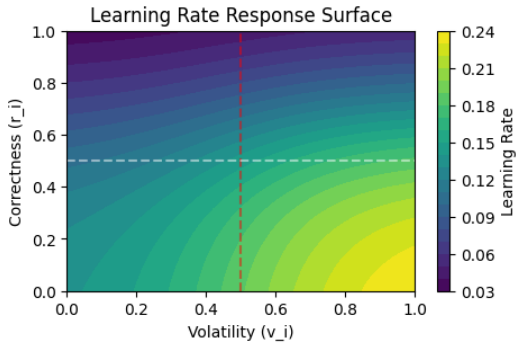


Figure 6: The response plane of the learning rate in Algorithm 1.

B.7 Popularity distribution of consistency outcome.

Table 7 is a supplementary detail for Figure 2 and Table 3. It shows the results of the `Llama3-8B` model on the ConflictBank dataset. The x-axis represents the *popularity* value in logarithmic form, while the y-axis represents the sample size corresponding to each popularity level. The red line indicates the mean popularity within the group.

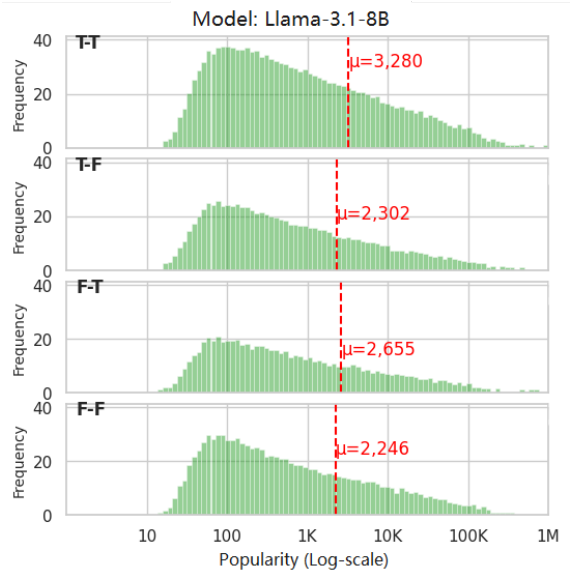


Figure 7: The distribution of *popularity* among different consistency group.

C KnowledgeMap EVALUATION

In this section, we evaluate the validity of KnowledgeMap step by step to show that KnowledgeMap could effectively discriminate knowledge gaps in LLMs. We conduct an independent assessment of the three components of KnowledgeMap to validate its robustness, as well as the evaluation of knowledge extraction from humans and GPT-4.

C.1 ROBUSTNESS OF *Knowledge Extractor*.

We demonstrate the robustness of the prompt-based method by quantifying the performance and the characteristics of *Knowledge Extractor*. We repeatedly extract the {*concept*: *importance*} dictionary 10 times with *word_limit* set to 2 and *concept_limit* set to 10 for each sample question in CONFLICTBANK and TRIVIAQA. To maintain fairness, the results are generated using the models' default generation configurations, which are shown in Table 8. In general, LLaMA and Qwen share similar features, whereas Mistral's outputs are completely identical. And the robustness is determined by both the characteristics of the tested data and the preference of the underlying model.

Table 8: Robustness of *Knowledge Extractor* tested on different LLMs.

Model	% of Valid Extraction[(a)]	# of Different Concepts [(b)]	% of Title Include[(c)]	*Concept* Freqency[(d)]	*Importance* Std.[(e)]	Jaccard Similarity[(f)]
MCQ						
LLaMA	85.48%	2.79	86.98%	67.53%	7.64	0.40
Mistral	100.00%	1.09	87.04%	100.00%	0.00	1.00
Qwen	89.86%	2.17	87.49%	84.38%	2.17	0.78
SAQ						
LLaMA	86.95%	3.69	62.32%	69.15%	7.86	0.32
Mistral	100.00%	1.90	43.16%	100.00%	0.00	1.00
Qwen	85.68%	3.94	55.80%	76.45%	2.77	0.57

In average percentage of valid concepts[(a)] among samples in 10 extractions, all models achieve ≥85% successful extraction. The average number of different concepts[(b)] extracted from each sample ranges from 1.09 to 2.79 for the **MCQ** average of 7.67 question words, and 1.90 to 3.94 for the **SAQ** sample average of 13.08 question words.

Whether the samples' *titles* are included in the extracted *concepts*[(c)] implies the accuracy of the method to some extent. The inclusion rates of the three models are relatively consistent on the CONFLICTBANK, averaging approximately 87%. However, when evaluated on TRIVIAQA, the inclusion rate for LLaMA reaches 62.32%, whereas it is only 43.16% for Mistral and 55.8% for Qwen, which is much lower than CONFLICTBANK. This is because the questions from TRIVIAQA are more indirect and nuanced (e.g., "What rock band was fronted by the late great Ian Curtis?" vs. a direct fact), requiring the extractor to identify underlying concepts ("Ian Curtis," "rock band," "Joy Division") rather than just the title. While the questions in CONFLICTBANK are directly constructed based on the title (entity), thus having a higher inclusion rate compared to TRIVIAQA.

For similarity comparisons, the *concept* frequency[(d)] is computed as the average frequency of occurrence of each concept. From the results, different datasets have no apparent effect on the *concepts*' robustness, but Qwen is more uniform than LLaMA. The standard deviation of importance[(e)] is the average standard deviation of the importance values (on a percentage scale) for each concept. In this regard, Qwen exhibits lower volatility than LLaMA, although both remain minimal. Then, we evaluate the dictionaries as a whole and use the Jaccard similarity[(f)] to quantify their degree of similarity. The Jaccard index ranges between [0, 1], with a higher value indicating a larger similarity. Each concept-importance pair is treated as an element in the set, and the global Jaccard similarity is derived by averaging the Jaccard similarity of all dictionary pairs. According to this metric, knowledge extraction on CONFLICTBANK is more robust compared to TRIVIAQA, and the similarity under Qwen is significantly higher than that of LLaMA.

C.2 PERTURBATION EFFECT BY *Knowledge Checker*.

The *Knowledge Checker* calculates the consistency score of the model on certain knowledge by measuring the maximum or average intensity of interference that the model can withstand under multiple perturbations. We show the effect of the applied perturbation of `LLaMa` on the TRIVIAQA dataset. The similarity is computed between the pair of prompted inputs. The accuracy under different perturbation levels is presented in Table 9. It can be observed that as the length of the interfering context increases, the growth of interference intensity gradually slows down under similarity-based metrics. Starting from the L5 level of interference, increments become minor and declines in accuracy become less pronounced.

The resulting distribution of consistency *scores* is illustrated on the left of Figure 3. The mean and standard deviation of the consistency scores calculated using the average intensity and highest intensity were 0.19 ± 0.14 and 0.31 ± 0.24, respectively. It can be observed that these scores exhibit a slightly right-skewed normal distribution in their non-zero portions.

Table 9: The overall perturbation intensity and the associated accuracy of `LLaMa` in TRIVIAQA respecting various prepending context lengths.

	–	L1	L2	L3	L4	L5
Intensity	0.00%	13.28%	23.00%	33.28%	39.40%	43.27%
Accuracy	62.27%	55.49%	55.02%	54.29%	53.67%	53.37%

C.3 VALIDITY OF *Knowledge Updater*.

To validate whether the *Knowledge Checker* could effectively discriminate concepts' proficiency according to the degree of inconsistency, we test `KnowledgeMap` on TRIVIAQA and obtain the knowledge proficiency at the concept level. The consistency score takes $s_{\max}$ and the case score takes $s_{\mathcal{H}}$.

The TRIVIAQA dataset contains 74,508 concepts across 95,704 valid samples, with an average occurrence of 3.99. The most frequently extracted concepts are geographical, such as *United States*(834), *England*(755), *London*(636), *UK*(621), *France*(458). Regarding concept coverage of knowledge, the top 20 concepts are associated with 8,421 sample titles, while the top 100 concepts cover as many as 14,548 sample titles. The middle of Figure 3 demonstrates the distribution of *proficiency*. It can be observed that as the samples undergo increasing update iterations, their values disperse from the initial $y = 0.5$ toward both extremes. The right of Figure 3 shows the relationship between concept *proficiency* and the related samples' average consistency *score*. The results present a funnel-shaped pattern centered along the fit line with correlation $r = 0.58$. Therefore, the proficiency level of a concept is positively associated with its response consistency.

C.4 EVALUATION FROM HUMANS AND GPT-4.

To further verify the reliability of LLMs in the knowledge extraction task, we conduct additional evaluations with human annotators and GPT-4. While direct alignment via explicit human concept extraction represents a plausible alternative, we employ an acceptability-based evaluation protocol due to several practical challenges.

First, the factual queries in our benchmark span a broad spectrum of popularity and domain specificity, making direct concept extraction particularly challenging in long-tail or niche domains where annotators may lack sufficient expertise, potentially resulting in incomplete or inconsistent annotations. Second, LLMs are capable of identifying implicit yet foundational concepts that are not explicitly stated in the query. For example, given "*What rock band was fronted by the late great Ian Curtis?*", the extractor identifies "*Joy Division*" in addition to "*Ian Curtis*" and "*rock band*", where "*Joy Division*" serves as the critical conceptual bridge despite not appearing in the input. Such implicit concepts are often overlooked in human extraction, especially outside annotators' areas of expertise.

To this end, we adopt an acceptability-based validation Bavaresco et al. (2025) to evaluate the soundness of the extracted concepts, and report the agreement between human and GPT-4 judgments following Zheng et al. (2023). We randomly select 120 samples (60 in each) from the CONFLICTBANK

and TRIVIAQA datasets. These samples are stratified across three popularity levels: high, medium, and low, with 20 per level. Evaluation is conducted from the following dimensions:

- **Concept Clarity**. Are the concept names semantically clear and unambiguous for representation?
- **Concept Relevance**. Are the extracted concepts genuinely relevant to the input question?
- **Weight Rationality**. Do the assigned importance weights reasonably reflect the relative significance of each concept within the context of the question?

Each dimension is scored from one to three points, corresponding to the grades of unsatisfactory, satisfactory, and excellent. We calculated inter-human agreement using *Fleiss' Kappa* κ and measured the consistency between human and GPT-4 automated evaluations via Pearson correlation coefficients r.

For human review, we invite five graduate students with computer science backgrounds as evaluators, each independently assessing all 120 samples. Prior to the evaluation, we provide examples of both valid and invalid outputs to ensure that the evaluators can accurately apply the assessment criteria (Figure 12). For automatic evaluation, we use the leading LLM GPT-4 for assessment. We designed an evaluation prompt with assessment criteria provided in advance and required GPT-4 to evaluate the extraction results from the same dimensions as humans (Figure 13).

The results of the collaborative evaluation between human and GPT-4 are presented in Table 10. It shows that the extraction quality has an average score of 2.6, and the consistency among human evaluators was high (κ=0.73). The GPT-4 assessment was highly correlated with manual scoring (r=0.81), confirming the reliability of LLM-based knowledge extraction.

Table 10: Humans and GPT-4 evaluation results on knowledge extraction quality.

Criteria	Human	GPT-4	Agreement (κ)	Correlation (r)
Concept Clarity	2.8 ± 0.3	2.7 ± 0.4	0.78	0.82
Concept Relevance	2.6 ± 0.4	2.5 ± 0.5	0.72	0.79
Weight Rationality	2.4 ± 0.5	2.3 ± 0.6	0.68	0.76
Overall	2.6 ± 0.3	2.5 ± 0.4	0.73	0.81

D APPLICABILITY

We demonstrate the practicality of KnowledgeMap in terms of scalability and generalizability.

D.1 SCALABILITY

KnowledgeMap can handle large concept sets efficiently (e.g., 74k in TRIVIAQA) and has the potential to expand to assess a broader scope of knowledge. Regarding computation, because the proficiency update given in Algorithm 1 is performed per sample, it only involves the n concepts present in that sample. Regarding memory usage, only the correctness and volatility of each concept within k time windows are recorded, resulting in a linear increase in memory usage with the number of concepts. Therefore, the entire system is designed for scalable evaluation, which does not impose a significant computational burden on modern computing devices.

D.2 GENERALIZABILITY

KnowledgeMap is a flexible pipeline that can be extended to various tasks, not just multiple-choice questions and short-answer questions. Let us illustrate how KnowledgeMap achieves concept-level inconsistency attribution for generative tasks.

For example, Raj et al. (2023) (Raj et al., 2023) extended the study of lexical consistency in responses to long-sequence consistency. They proposed two consistency metrics for a set of long responses based on token quality and semantic clustering entropy to measure semantic consistency across multiple outputs. In this case, the *Knowledge Extractor* simultaneously extracts concepts associated with the samples and their importance from a set of prompts and integrates the results.

Then, the consistency score in the *Knowledge Checker* is directly replaced with the long-response consistency metric proposed by Raj et al. Finally, the *Knowledge Updater* updates the proficiency of each concept based on both concept importance and consistency scores. The idea of KnowledgeMap can be extended to more types of tasks.

KnowledgeMap can also be extended to ethical, commonsense, or technical domains. The primary adaptation required is in the *Knowledge Extractor* module to ensure the extracted concepts are relevant and meaningful within the target domain. We list several viable pathways:

- Refine the prompt used by the LLM in the *Knowledge Extractor*. By explicitly specifying the domain, we can guide the model to focus on concepts at an appropriate level of granularity and relevance. This method leverages the LLM's broad knowledge while constraining it to a specific context, building directly upon the automated methodology presented in our paper.
- Integrate KnowledgeMap with existing, expert-built knowledge structures, such as the Gene Ontology for biology or the ACM Computing Classification System for computer science.
- Anchor the knowledge concepts in large-scale, domain-specific knowledge graphs such as Wikidata, DBpedia, or domain-specific equivalents. However, this approach could still be limited by the inherent structure of the underlying knowledge base.

You are a professional text analysis tool specialized in extracting Wikipedia-style concepts from text. Your task is to identify the most specific, notable entities (people, events, works, etc.) that could have their own Wikipedia pages, and evaluate their importance percentages according to these rules:
1. Extract only specific, notable concepts that would qualify as Wikipedia article titles (e.g., "Bill Clinton" not just "President").
2. Prioritize proper nouns and specific named entities over general terms.
3. Each concept should be exactly as it would appear as a Wikipedia title (proper capitalization, full names when available).
4. Assign an importance percentage to each concept (sum must be 100%).
5. Concepts should typically be 1-{WORD_LIMIT} words unless the full proper name requires more.
6. Maximum {CONCEPT_LIMIT} concepts.

Response format MUST be:
1. ¡*Concept*$_1$¿ (¡*Importance*$_1$¿%)
2. ¡*Concept*$_2$¿ (¡*Importance*$_2$¿%)
...
NO additional text or explanations.

Example:
Input: The Mona Lisa, painted by Leonardo da Vinci during the Italian Renaissance, is displayed at the Louvre Museum in Paris.
Output:
1. Mona Lisa (40%)
2. Leonardo da Vinci (30%)
3. Italian Renaissance (15%)
4. Louvre (15%)

Extract Wikipedia-style concepts from:
{QUESTION}

Figure 8: **(P1)** Knowledge extraction prompt used for *Knowledge Extractor*.

Table 11: Perturbation intensity setting and examples.

Level	Perturbation	MCQ	Perturbation	SAQ
-	Original	**Q:** *Where does Lady Gaga reside?* A `Vienna` B `Portuguese Ceuta` C `Uncertain` D `Los Angeles`	Original	**Q:** *In 1999 Anna Kournikova signed a lucrative contract to model what?*
L1	Option Indice + Option Format (rnd.)	**Q:** *Where does Lady Gaga reside?* 1. `Vienna` 2. `Portuguese Ceuta` 3. `Uncertain` 4. `Los Angeles`	Rewrite + Irrelevant Context (30 words)	**Q:** (*In 1994 and 1995, gangs of armed youth destroyed the homes of foreign nationals living in Johannesburg, demanding that the police work to repatriate them to their home countries.) In 1999 Anna Kournikova signed a lucrative contract to model what?*
L2	Add Option (1) + Option Permutation (rnd.) + Option Format (rnd.)	**Q:** *Where does Lady Gaga reside?* {E} `Dresden` {D} `Uncertain` {A} `Vienna` {C} `Los Angeles` {B} `Portuguese Ceuta`	Rewrite + Irrelevant Context (50 words)	**Q:** *(The Amazon basin is home to the largest tropical rainforest in the world, covering over 7 million square kilometers. It hosts unparalleled biodiversity, including thousands of species of fish, birds, mammals, and plants. The river's freshwater output significantly influences ocean salinity and climate patterns in the region.) What did Anna Kournikova sign a lucrative contract to model in 1999?*
L3	Add Option (2) + Index Indice + Option Permutation (rnd.) + Option Format (rnd.)	**Q:** *Where does Lady Gaga reside?* (2) `Los Angeles` (3) `Humppila` (5) `Portuguese Ceuta` (1) `Dresden` (4) `Uncertain` (6) `Vienna`	Rewrite + Irrelevant Context (70 words)	**Q:** *(Leonardo's artistic approach was characterized by sfumato—a technique of blending colors and tones subtly. He often left works unfinished, pursuing perfection across multiple disciplines. His designs for flying machines, armored vehicles, and urban planning were centuries ahead of their time, though few were built during his lifetime.) What was the lucrative modeling contract that Anna Kournikova signed in 1999?*
L4	Add Option (3) + Index Caesar (rnd.) + Option Permutation (rnd.) + Option Format (rnd.)	**Q:** *Where does Lady Gaga reside?* ~17~ `Dresden` ~14~ `Uncertain` ~16~ `Vienna` ~13~ `Los Angeles` ~15~ `Humppila` ~19~ `Portuguese Ceuta` ~18~ `Buenos Aires`	Rewrite + Irrelevant Context (90 words)	**Q:** *(The sound's tides and currents create dynamic mixing zones that nurture plankton blooms and sustain food webs. It is a major hub for ferry transportation, connecting islands and peninsulas to the mainland. Indigenous Coast Salish peoples have lived along its shores for millennia, relying on its resources for sustenance and cultural practices. Today, the region balances economic development with conservation, aiming to preserve the sound's ecological integrity for future generations.) In 1999, Anna Kournikova signed a lucrative contract to model a particular product—what was it?*
L5	Add Option (3) + Index Caesar (rnd.) + Option Indice + Option Permutation (rnd.) + Option Format (rnd.)	**Q:** *Where does Lady Gaga reside?* 10) `Humppila` 8) `Uncertain` 6) `Philadelphia` 7) `Vienna` 11) `Dresden` 9) `Portuguese Ceuta` 5) `Los Angeles`	Rewrite + Irrelevant Context (120 words)	**Q:** *(The first successful Mars flyby was by Mariner 4 in 1965, which returned images of a cratered surface. Later missions like Viking, Curiosity, and Perseverance have searched for signs of ancient microbial life and collected soil samples for eventual return to Earth. Seasonal changes include dust storms that can engulf the entire planet. Terraforming proposals imagine modifying Mars's environment to support human life, though such projects remain speculative. Mars continues to captivate public imagination as humanity's next potential frontier beyond Earth, symbolizing both scientific curiosity and the enduring dream of interplanetary exploration.) Which product did Anna Kournikova sign a lucrative contract to model in 1999?*

Please select the correct option from the following options given the question:
Question: {QUESTION}
{OPTION_ID$_1$} {OPTION$_1$}
{OPTION_ID$_2$} {OPTION$_2$}
{OPTION_ID$_3$} {OPTION$_3$}
...
Your output must strictly follow this format and SHOULD NOT include any other text in the response:
{"answer": ¡*Selected Option*¿}

Figure 9: **(P2) MCQ** answer generation prompt used for *Knowledge Checker*.

Please rewrite the question WITHOUT explanations, punctuation, or extra text.
Your output must strictly follow this format:"{"rewrites": ¡*Rewritted Setence*¿}"
Example: At which university did Joseph Goebbels become a doctor of philosophy?
Output: {"answer": "Which university awarded Joseph Goebbels a PhD in philosophy?"}.
Now rewrite this sentence: {QUESTION}

Figure 10: **(P3) SAQ** question rewrite prompt used for *Knowledge Checker*.

Please answer the question WITHOUT explanations, punctuation, or extra text.
Your output must strictly follow this format:"{"answer": ¡*Your Answer*¿}"
Example: Which port lies between Puget Sound and Lake Washington? Output: {"answer": "Seattle"}.
Now answer this question: {QUESTION}

Figure 11: **(P4) SAQ** answer generation prompt used for *Knowledge Checker*.

You'll be presented with a series of questions on LLM knowledge extraction. For each question, please assess the extraction quality:

Extracted Results:
Question: [Question Text]
Extracted Results: [Concept1] ([Weight]%), [Concept2] ([Weight]%), ...
Please evaluate:

1. Concept Clarity (1–3 points):
❏ 1 - Unsatisfactory (naming ambiguous or unclear)
❏ 2 - Satisfactory (naming generally clear but could be more precise)
❏ 3 - Excellent (naming semantically clear and unambiguous)

2. Concept Relevance (1–3 points):
❏ 1 - Unsatisfactory (concept irrelevant or counter-intuitive)
❏ 2 - Satisfactory (concept relevant but possibly incomplete)
❏ 3 - Excellent (concept fully relevant and intuitive)

3. Weight Rationality (1–3 points):
❏ 1 - Unsatisfactory (weight clearly unreasonable)
❏ 2 - Satisfactory (weights roughly reasonable)
❏ 3 - Excellent (weights highly reasonable)

Please provide a score of 1–3 for each dimension (1 = unsatisfactory, 2 = satisfactory, 3 = excellent), along with a brief justification.

Figure 12: Instructions for human evaluation.

You are a rigorous academic evaluation expert. Please assess the quality of the knowledge extraction results based on the following three dimensions:

1. Concept Clarity: Whether the generated concept names are semantically clear and unambiguous, conveying precise meaning without vagueness or confusion.
2. Concept Relevance: Whether the concepts are directly relevant to the question and align with human cognitive expectations.
3. Weight Rationality: Whether the assigned weight percentages reasonably reflect the relative importance of the concepts in the context of the question.

Please provide a score of 1–3 for each dimension (1 = unsatisfactory, 2 = satisfactory, 3 = excellent).

You should generate the scores in order and in the following format:
{"Format Standardization": ¡*Score*¿,
"Concept Relevance": ¡*Score*¿,
"Weight Rationality": ¡*Score*¿}

Now score the results:
Input: {`INPUT`}
Extracted Results: {`Concept1` (`Importance%`), `Concept2` (`Importance%`), ...}

Figure 13: Prompt for GPT-4 automatic evaluation.

Table 12: Perturbation intensity under different measures (CONFLICTBANK).

Perturbation		QSim+TFIDF	QSim+BERT	QSim+BLEU	QSim+W2V	PSim+TFIDF	PSim+BERT	PSim+BLEU	PSim+W2V
	mean	1.0000	0.9486	0.5363	0.9153	1.0000	0.9800	0.7767	0.9827
	std	0.0000	0.0235	0.1021	0.0375	0.0000	0.0063	0.0144	0.0033
	min	1.0000	0.7962	0.0000	0.6311	1.0000	0.9228	0.7366	0.9681
L1	25%	1.0000	0.9361	0.4763	0.8941	1.0000	0.9769	0.7668	0.9804
	50%	1.0000	0.9526	0.5441	0.9226	1.0000	0.9815	0.7753	0.9829
	75%	1.0000	0.9657	0.6091	0.9408	1.0000	0.9845	0.7859	0.9847
	max	1.0000	0.9947	0.7946	0.9889	1.0000	0.9918	0.8376	0.9939
	mean	0.8951	0.9300	0.4476	0.8876	0.9735	0.9723	0.7042	0.9743
	std	0.0561	0.0318	0.0946	0.0473	0.0140	0.0112	0.0236	0.0081
	min	0.5061	0.6762	0.0000	0.4641	0.8964	0.8661	0.6068	0.8945
L2	25%	0.8674	0.9140	0.3899	0.8612	0.9669	0.9668	0.6898	0.9713
	50%	0.9056	0.9356	0.4517	0.8941	0.9773	0.9743	0.7056	0.9760
	75%	0.9356	0.9524	0.5113	0.9203	0.9827	0.9801	0.7198	0.9794
	max	1.0000	0.9926	0.7455	0.9889	1.0000	0.9949	0.7974	0.9939
	mean	0.8229	0.8783	0.3848	0.8659	0.9490	0.9567	0.6550	0.9654
	std	0.0727	0.0452	0.0870	0.0535	0.0207	0.0133	0.0287	0.0110
	min	0.3764	0.5523	0.0000	0.4123	0.7765	0.8254	0.5278	0.8335
L3	25%	0.7817	0.8521	0.3300	0.8363	0.9375	0.9499	0.6374	0.9601
	50%	0.8336	0.8842	0.3870	0.8724	0.9517	0.9588	0.6572	0.9673
	75%	0.8757	0.9110	0.4453	0.9026	0.9635	0.9661	0.6751	0.9729
	max	1.0000	0.9876	0.6773	0.9797	1.0000	0.9875	0.7721	0.9907
	mean	0.7678	0.8738	0.3373	0.8483	0.9257	0.9386	0.6122	0.9541
	std	0.0843	0.0466	0.0802	0.0574	0.0264	0.0169	0.0324	0.0155
	min	0.2944	0.5794	0.0000	0.3523	0.7597	0.7422	0.4680	0.8303
L4	25%	0.7188	0.8472	0.2885	0.8157	0.9105	0.9301	0.5924	0.9460
	50%	0.7765	0.8799	0.3382	0.8544	0.9282	0.9411	0.6142	0.9567
	75%	0.8277	0.9072	0.3919	0.8872	0.9440	0.9502	0.6344	0.9651
	max	1.0000	0.9775	0.6376	0.9744	1.0000	0.9809	0.7721	0.9895
	mean	0.6663	0.8570	0.3372	0.8502	0.7723	0.9425	0.6121	0.9558
	std	0.0904	0.0508	0.0802	0.0566	0.0310	0.0150	0.0325	0.0142
	min	0.2881	0.5234	0.0000	0.4479	0.6440	0.7990	0.4722	0.8339
L5	25%	0.6032	0.8273	0.2885	0.8182	0.7510	0.9343	0.5923	0.9482
	50%	0.6570	0.8636	0.3379	0.8564	0.7671	0.9444	0.6141	0.9580
	75%	0.7259	0.8934	0.3919	0.8886	0.7882	0.9530	0.6344	0.9657
	max	0.9423	0.9801	0.6240	0.9754	0.9201	0.9809	0.7721	0.9908

Table 13: Perturbation intensity under different measures (TRIVIAQA).

Perturbation		QSim+TFIDF	QSim+BERT	QSim+BLEU	QSim+W2V	PSim+TFIDF	PSim+BERT	PSim+BLEU	PSim+W2V
	mean	0.2122	0.4691	0.0000	0.4720	0.7436	0.8075	0.5369	0.9108
	std	0.1112	0.1497	0.0607	0.1554	0.0109	0.1014	0.0395	0.0164
	min	0.0854	0.3187	0.0000	0.3049	0.7332	0.6938	0.4914	0.8948
L1	25%	0.1717	0.3945	0.0000	0.4019	0.7380	0.7670	0.5243	0.9024
	50%	0.2579	0.4704	0.0000	0.4990	0.7428	0.8402	0.5571	0.9101
	75%	0.2755	0.5443	0.0526	0.5556	0.7489	0.8643	0.5596	0.9189
	max	0.2931	0.6181	0.1051	0.6121	0.7549	0.8885	0.5621	0.9276
	mean	0.1892	0.5564	0.0000	0.5383	0.6448	0.7675	0.4464	0.8912
	std	0.1023	0.0866	0.0000	0.1430	0.0571	0.1017	0.0592	0.0199
	min	0.0828	0.4887	0.0000	0.4149	0.5887	0.6502	0.3923	0.8698
L2	25%	0.1404	0.5076	0.0000	0.4600	0.6158	0.7353	0.4147	0.8823
	50%	0.1979	0.5266	0.0000	0.5051	0.6428	0.8204	0.4371	0.8947
	75%	0.2424	0.5903	0.0000	0.6000	0.6728	0.8261	0.4734	0.9019
	max	0.2868	0.6539	0.0000	0.6950	0.7028	0.8318	0.5097	0.9090
	mean	0.1018	0.5951	0.0000	0.4126	0.5618	0.5684	0.3148	0.8228
	std	0.0509	0.1215	0.0000	0.1067	0.0093	0.1458	0.0047	0.0482
	min	0.0440	0.4609	0.0000	0.2948	0.5532	0.4051	0.3107	0.7754
L3	25%	0.0828	0.5439	0.0000	0.3676	0.5569	0.5097	0.3123	0.7984
	50%	0.1215	0.6270	0.0000	0.4405	0.5606	0.6142	0.3138	0.8213
	75%	0.1307	0.6622	0.0000	0.4716	0.5661	0.6500	0.3169	0.8465
	max	0.1399	0.6974	0.0000	0.5027	0.5717	0.6857	0.3200	0.8718
	mean	0.1753	0.6169	0.0142	0.5000	0.5203	0.4636	0.2840	0.8203
	std	0.1135	0.1108	0.0247	0.1265	0.0761	0.0914	0.0443	0.0807
	min	0.0608	0.5167	0.0000	0.3691	0.4612	0.3702	0.2502	0.7334
L4	25%	0.1191	0.5574	0.0000	0.4392	0.4773	0.4189	0.2589	0.7838
	50%	0.1774	0.5980	0.0000	0.5093	0.4933	0.4677	0.2677	0.8343
	75%	0.2326	0.6669	0.0214	0.5654	0.5498	0.5103	0.3010	0.8637
	max	0.2878	0.7359	0.0427	0.6216	0.6062	0.5528	0.3342	0.8931
	mean	0.1379	0.5465	0.0135	0.4511	0.5533	0.5744	0.3481	0.8565
	std	0.1024	0.0906	0.0235	0.1288	0.0778	0.1216	0.0539	0.0276
	min	0.0402	0.4734	0.0000	0.3077	0.4770	0.4342	0.3039	0.8264
L5	25%	0.0846	0.4959	0.0000	0.3982	0.5137	0.5365	0.3181	0.8442
	50%	0.1290	0.5183	0.0000	0.4888	0.5504	0.6388	0.3323	0.8620
	75%	0.1867	0.5831	0.0203	0.5229	0.5915	0.6445	0.3702	0.8715
	max	0.2444	0.6479	0.0406	0.5570	0.6325	0.6501	0.4081	0.8809