Beyond 1-to-1: A Metric for Probing and Editing 1-to-N Knowledge within Large Language Models

Anonymous ACL submission

Abstract

Large language models (LLMs) have demonstrated strong capabilities in encoding and applying factual knowledge, much of which follows a one-to-many (1-to-N) structure, where a single query corresponds to multiple valid answers. However, the existing metrics for evaluating 1-to-N knowledge suffer from inherent limitations, such as ignoring valid alternative answers, failing to reflect model confidence, or neglecting probability distributions. To address these limitations, we propose a new metric, named **N**-Answer **K**ullback-**L**eibler Divergence (NKL), which aligns the predicted probability distribution of an LLM with a given gold distribution (e.g. a pre-training corpus). NKL integrates both ranking and probability information, offering a more comprehensive evaluation. We also formalise 1-to-N knowledge evaluation with two criteria—*coverage* and *alignment*—under which NKL demonstrates the best overall performance. Experiments on Counterfact and SNOMED CT further validate NKL's effectiveness in knowledge probing and editing, providing new insights into LLMs' ability to represent and modify 1-to-N knowledge [1].

1 Introduction

Large language models (LLMs) have demonstrated remarkable capabilities in acquiring knowledge across diverse domains (Plaat et al., 2024; Hu et al., 2023; Wang et al., 2023). There has been growing interest in understanding and manipulating the internal knowledge of LLMs. Among these approaches, knowledge probing (Meng et al., 2022b; Sung et al., 2021) and knowledge editing (Meng et al., 2022a; Yao et al., 2023) have received considerable attention. Knowledge probing aims to evaluate an LLM's ability to recall and apply the knowledge learned during pre-training, while knowledge editing focuses on making targeted modifications to the model's internal knowledge without retraining.

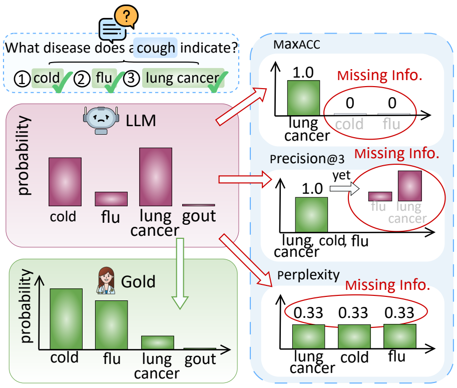

Figure 1: Limitations of evaluation metrics for 1-to-N knowledge: MaxAcc (top right) only considers the top-1 answer, missing valid alternatives like "cold" and "flu." Precision@3 (middle right) captures multiple correct answers but ignores overconfidence in lung cancer. Perplexity (bottom right) evaluates probability values without considering answer identity or ranking.

However, most existing works represent knowledge in a one-to-one format, where each query is associated with a single correct answer. This simplification often overlooks the complexity of real-world knowledge, which often exhibits a 1-to-N structure, where a query corresponds to multiple valid answers (Green, 2001; Thandi et al., 2021; Han et al., 2022). For instance, the query *"What is the treatment for hypertension?"* has several answers, including *ACE inhibitors*, *beta-blockers*, and *calcium channel blockers* (Fuchs and Whelton, 2020). This 1-to-N pattern is widespread: 40% of Counterfact (Meng et al., 2022a) and over 80% of SNOMED CT (Donnelly et al., 2006) triples follow a 1-to-N structure. While these answers are all correct, they are actually not equally common or likely. This is especially important in biomedicine, where rare but technically valid answers may be misleading. For example, given the question *"What disease does a cough indicate?"*, lung cancer is one correct answer, but far less typical than cold. This highlights the importance of evaluating not

[1] Code and datasets can be found in: `https://anonymous.4open.science/r/nkl_metrics-42E4`

only correctness of model's prediction, but also the distribution over multiple correct answers. Such considerations lead to a critical question:

How can we evaluate LLMs to effectively assess their understanding of 1-to-N knowledge?

To address this question, we first investigate existing evaluation metrics used in knowledge probing and editing that can be applied to 1-to-N knowledge, including MaxAcc (Yao et al., 2023; Li et al., 2022; Sung et al., 2021), Precision@K (Sung et al., 2021; Jiang et al., 2020), and Perplexity (Onoe et al., 2022). These metrics offer valuable insights into model performance by assessing accuracy, ranking or probability calibration. However, they still have limitations in fully capturing the structured nature of 1-to-N knowledge. As shown in Figure 1, given the predicted answer distributions and the gold distributions provided by experts, existing metrics, i.e., MaxAcc, Precision@K and Perplexity, either focus solely on the highest-ranked response or treat top answers as equally probable, and thus fail to capture the true alignment between predicted and gold distributions in knowledge representation.

To this end, we propose **N**-Answer **K**ullback-**L**eibler Divergence (NKL), a new evaluation metric designed for assessing 1-to-N knowledge probing and editing. NKL assesses the alignment between an LLM's predicted probability distribution and the gold distribution reflecting the ground-truth likelihood of each candidate answer. Compared to existing metrics, NKL provides a more comprehensive assessment by considering all valid answers, integrating both ranking and probability information, and directly aligning with the gold distribution.

To validate the effectiveness of NKL in capturing 1-to-N knowledge, we further formalise the task of 1-to-N knowledge evaluation and introduce two key evaluation criteria: *coverage*, which measures an LLM's ability to recall all valid answers, and *alignment*, which evaluates how well the predicted probability distribution matches the gold distribution of valid answers. Our empirical results indicate that NKL achieves the highest correlation scores with both criteria. We further demonstrate the utility of NKL by applying it to two key tasks: knowledge probing and knowledge editing, on two real-world datasets, namely Counterfact (Meng et al., 2022a) and SNOMED CT (Donnelly et al., 2006). These experiments allow us to gain deeper insight into how LLMs recall and update 1-to-N knowledge.

Our key contributions can be summarised as follows: (1) We propose NKL, an evaluation metric that measures the alignment between an LLM's predicted distribution and a given gold one; (2) We formally define the task of 1-to-N knowledge probing and propose its two key criteria, i.e. coverage and alignment; (3) We conduct a comprehensive evaluation of 1-to-N knowledge probing and editing, showing that existing editing methods improve the retrieval performance of edited LLMs but have limited capacity to maintain probability alignment.

2 Preliminaries

We provide a brief introduction of 1-to-N knowledge and its evaluation metrics in this section. Related work on knowledge probing and editing is discussed in Appendix A.

2.1 1-to-N Knowledge

A piece of knowledge can be considered as a query–answer pair (q, a) that captures the association between a query and a valid answer (Meng et al., 2022a; Yao et al., 2023). We define 1-to-N knowledge as a setting where a query q is associated with multiple valid answers $\mathcal{A} = \{a_1, a_2, \ldots, a_c\}$. Given a language model M, we denote its predictive probability distribution over a set of candidate answers $\mathcal{X} = \{x_1, \ldots, x_n\}$ as: $\mathbb{P}(x \mid q) = \{P_M(x_1 \mid q), \ldots, P_M(x_n \mid q)\}$, where $x_i \in \mathcal{X}$ denotes a complete candidate answer and $\mathbb{P}(x_i|q)$ is the probability that the model assigns to x_i. Similarly, we define a *gold probability distribution* over $\mathcal{X}$: $\mathbb{Q}(x|q) = \{Q(x_1|q), \ldots, Q(x_n|q)\}$, which reflects ground-truth likelihoods over candidate answers. These likelihoods can be derived from domain-specific knowledge bases or real-world prevalence statistics (see § 3.4 for details on obtaining gold distribution). While $\mathbb{P}(x \mid q)$ reflects the model's learned probabilities, $\mathbb{Q}(x \mid q)$ represents the gold distribution over candidate answers, serving as the reference for evaluation.

2.2 Existing Evaluation Metrics for 1-to-N Knowledge

To evaluate how well a language model M captures factual knowledge, there are several metrics, such as **Maximum Accuracy** (**MaxAcc**) (Yao et al., 2023; Li et al., 2022; Sung et al., 2021), **Top-K Precision** (**Precision@K**) (Sung et al., 2021; Jiang et al., 2020) and **Perplexity** (**PPL**) (Onoe et al., 2022), that are widely used for knowledge probing and knowledge editing (Yao et al., 2023; Li et al., 2022). Formally, they are defined as follows:

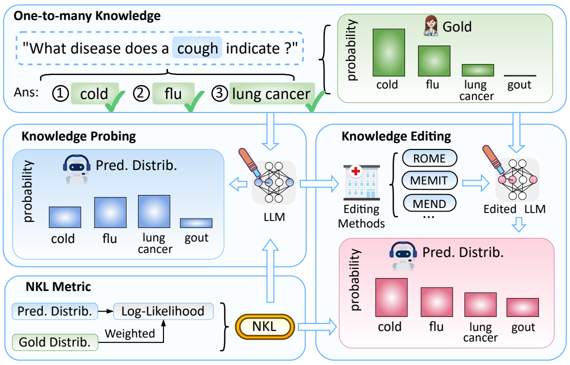

Figure 2: An overview of how NKL is used to evaluate 1-to-N knowledge. We consider two scenarios: knowledge probing, which assesses the alignment between the model's predictive distribution and the gold distribution; and knowledge editing, which evaluates how this alignment changes after the model's internal knowledge is modified. NKL quantifies distributional alignment in both settings.

(1) **MaxAcc:**

$$\mathbb{E}_{(q,\mathcal{A})\sim\mathcal{P}}\mathbb{I}\left\{\arg\max_{x\in\mathcal{X}} P_M(x \mid q) \in \mathcal{A}\right\},$$

where $\mathbb{I}(\cdot)$ is an indicator function that returns 1 if the most probable predicted answer by the model belongs to the ground-truth set $\mathcal{A}$ and 0 otherwise. The query-answer pairs $(q, \mathcal{A})$ are sampled from a data distribution $\mathcal{P}$. MaxAcc measures if the model's top-1 prediction matches any answer in $\mathcal{A}$.

(2) **Precision@K:**

$$\mathbb{E}_{(q,\mathcal{A})\sim\mathcal{P}}\ \mathbb{I}\left\{\underset{x\in\mathcal{X}}{\text{top-}K}\ P_M(x \mid q) \cap \mathcal{A} \neq \emptyset\right\},$$

where $\mathbb{I}(\cdot)$ is an indicator function that returns 1 if at least one of the top-K predicted answers belongs to the ground-truth set $\mathcal{A}$, and 0 otherwise. This metric evaluates whether the model can recall any valid answer among its top-K ranked predictions.

(3) **PPL:**

$$\mathbb{E}_{(q,\mathcal{A})\sim\mathcal{P}} \exp\left(-\frac{1}{|\mathcal{A}|}\sum_{a\in\mathcal{A}} \log P_M(a \mid q)\right),$$

where $P_M(a \mid q)$ is the probability assigned to a correct answer a. PPL assesses the average uncertainty of the model over answers in $\mathcal{A}$, with lower values indicating that the model assigns higher probabilities to correct answers.

3 The NKL Metric for Evaluating 1-to-N Knowledge

3.1 Evaluation Criteria for 1-to-N Knowledge

To effectively evaluate the ability of LLMs to represent and update 1-to-N knowledge through probing and editing, we propose two fundamental criteria that define the desired properties of an ideal evaluation metric.

Criterion 1 (Coverage)**.** *For 1-to-N knowledge, where a query q is associated with multiple valid answers $\mathcal{A}$, an LLM should maximise the size of the retrieved subset $\hat{\mathcal{A}} \subseteq \mathcal{A}$, where $\hat{\mathcal{A}}$ denotes the set of valid answers generated by the model.*

Criterion 2 (Alignment)**.** *For 1-to-N knowledge, an LLM's predicted probability distribution $\mathbb{P}(x \mid q)$ should align with the gold distribution $\mathbb{Q}(x \mid q)$.*

In knowledge probing and editing, **Criterion 1** ensures that an LLM, whether a foundational model or an edited one, retrieves a diverse set of valid answers rather than overfitting to a single dominant response, reflecting its ability to represent 1-to-N knowledge, i.e. the breadth of *coverage*. Meanwhile, **Criterion 2** ensures that a well-trained model allocates probability mass in proportion to the gold distribution of valid answers, preventing overconfidence in rare responses or underestimating frequent ones, i.e. the distribution *alignment*.

3.2 N-Answer Kullback-Leibler Divergence

Intuitively, a well-trained model should handle 1-to-N knowledge by allocating its probability mass over multiple valid answers in accordance with a specified gold distribution. This gold distribution may reflect real-world prevalence or be deliberately designed to support specific evaluation objectives. The proposed *N-Answer Kullback-Leibler Divergence (NKL)* metric quantifies the divergence between the predicted distribution of the model and the expected gold distribution, ensuring that coverage and alignment with real-world knowledge are taken into account.

Specifically, given a query q, $P_M(x \mid q)$ defines the probability that model M assigns to a candidate answer $x \in \mathcal{X}$. For an answer x_i, the predictive distribution is given by the following formula:

$$\begin{aligned} P_M(x_i|q) &= P_M(t_1, t_2, \ldots, t_k \mid q), && (1)\\ &= \prod_{j=1}^{k} P_M(t_j \mid q, t_1, t_2, \ldots, t_{j-1}), && (2)\\ &= \exp\left(\sum_{j=1}^{k} \log P_M(t_j \mid q, t_{<j})\right), && (3) \end{aligned}$$

where t_j represents the j-th token in the tokenized form of the answer x_i and k denotes the token length of x_i. Now, suppose that query q has N possible outputs and C correct answers. The probability distribution of these N possible outputs predicted by the model can be formulated as:

$$\mathbb{P}(x \mid q) = \{P_M(x_1 \mid q), \ldots, P_M(x_n \mid q)\}. \quad (4)$$

Given $\mathbb{Q}(x \mid q)$ as the gold probability distribution under the given query q. Then we have:

$$\mathbb{Q}(x \mid q) = \{Q(x_1 \mid q), \ldots, Q(x_n \mid q)\}. \quad (5)$$

We then compute the KL divergence of $\mathbb{Q}$ and $\mathbb{P}$.

$$D_{\mathrm{KL}}(\mathbb{Q}||\mathbb{P}) = \sum_{i=1}^{n} Q(x_i|q) \log \frac{Q(x_i|q)}{P_M(x_i|q)}, \quad (6)$$

$$= entropy(\mathbb{Q}) - \sum_{i=1}^{n} Q(x_i|q) \log P_M(x_i|q). \quad (7)$$

Given that $entropy(\mathbb{Q})$ is a constant. NKL can then be formulated as:

$$NKL = -\sum_{i=1}^{n} Q(x_i \mid q) \log P_M(x_i \mid q), \quad (8)$$

$$= -\sum_{x_i \in \mathcal{A}} Q(x_i \mid q) \log P_M(x_i \mid q), \quad (9)$$

where we transition from summing over all n candidate answers to summing over the set of valid answers $\mathcal{A}$, under the assumption that the probability mass of incorrect answers in the gold distribution $Q(x|q)$ is negligible. By combining Equations (3) and (9), and incorporating length normalisation for multi-token cases to ensure fair comparison across answer sequences of varying lengths [add citation], we derive the NKL formulation for the multi-token scenario:

$$NKL = -\sum_{x_i \in \mathcal{A}} \frac{Q(x_i|q)}{|x_i|} \sum_{j=1}^{|x_i|} \log P_M(t_j \mid q, t_{<j}), \quad (10)$$

where $|x_i|$ is the token length of the answer x_i.

3.3 Properties and Discussion

NKL integrates ranking and probability information. NKL simultaneously captures both ranking and probability information, making it a comprehensive measure for 1-to-N knowledge. As shown in Equation (9), by weighting the log-likelihood with $\mathbb{Q}(x)$, it ensures that predicted probabilities align with real-world relevance, enforcing a probability-sensitive ranking. Higher-probability answers in $\mathbb{Q}(x)$ contribute more to the evaluation, making NKL sensitive to both correctness and confidence. This allows NKL to provide a fine-grained assessment of the model's ability to capture the knowledge distribution in 1-to-N scenarios.

Comparison between NKL and Standard KL Divergence. NKL is derived from the standard Kullback-Leibler (KL) divergence (Kullback and Leibler, 1951), a fundamental measure of distributional difference widely used in NLP (Lang et al., 2024), but is specifically tailored for evaluating 1-to-N knowledge. The primary differences between NKL and standard KL divergence lie in two key aspects. First, as shown in Equation (9), we omit the entropy of the gold distribution, as its estimation requires computing the probability of all candidate answers for a given query, which is computationally expensive. Additionally, since this entropy is constant and irrelevant to model ranking, removing it simplifies computation without affecting the evaluation outcome. Second, compared to standard KL divergence, which considers the entire output space, NKL focuses explicitly on correct answers. As shown in Equation (9), we assume that in an ideal distribution $\mathbb{Q}(x \mid q)$, the probability of incorrect responses is negligible and can be discarded.

NKL for 1-to-1 Knowledge. While NKL is designed for evaluating 1-to-N knowledge, it remains applicable in 1-to-1 scenarios. In a 1-to-1 setting, where a query q has a single correct answer a^*, the gold distribution $\mathbb{Q}(x \mid q)$ becomes a Dirac delta distribution concentrated entirely on a^*, i.e.,

$$Q(x_i \mid q) = \begin{cases} 1, & x_i = a^* \\ 0, & \text{otherwise.} \end{cases} \quad (11)$$

Substituting this into the equation (9), we obtain:

$$NKL = -Q(a^* \mid q) \log P_M(a^* \mid q), \quad (12)$$

$$= -\log P_M(a^* \mid q), \quad (13)$$

which degenerates to the standard negative log-likelihood (NLL) commonly used for model evaluation in 1-to-1 knowledge. This confirms that NKL generalises NLL as a special case, extending its applicability from 1-to-1 to 1-to-N knowledge while maintaining consistency with traditional metrics.

3.4 Accessing the Gold Distribution

To compute NKL, we require a gold distribution $\mathbb{Q}(x|q)$ that captures the real-world prevalence of different correct answers. Inspired by previous works (Mallen et al., 2023; Kandpal et al., 2023), we construct $\mathbb{Q}(x|q)$ from corpus statistics, estimating the probability of each answer based on the co-occurrence number of subject-object pairs. Specifically, we represent knowledge as a collection of factual triples (s, r, o), where s is the subject, r is the relation, and o is the object. In our framework, the query q corresponds to the $\mathcal{T}(s, r)$, where $\mathcal{T}$ is the manual template, while the correct answers $\mathcal{A} = \{a_1, a_2, \ldots, a_{|\mathcal{A}|}\}$ correspond to the valid objects o for that relation. To estimate the gold distribution $\mathbb{Q}(x|q)$, we count how frequently each pair (s, o) appears in the corpus. These raw counts are then normalised to relative frequencies over the correct answers, so that $\mathbb{Q}(x|q)$ reflects their empirical distribution in the corpus.

4 Experiments

We conduct three main experiments to evaluate our proposed NKL. First, we perform a **simulation experiment** (§4.2) to examine whether NKL satisfies the two core criteria under controlled settings. Second, we conduct **benchmarking of probing and editing** (§4.3) on real-world datasets to evaluate how effectively LLMs capture and update 1-to-N knowledge. Third, we conduct a **probability alignment analysis** (§4.4) to examine how knowledge editing affects probability alignment in 1-to-N knowledge, as demonstrated by a case study.

Statistic	COUNTERFACT	SNOMED CT
Records	21,919	43,242
Subjects	20,391	22,753
Objects	6,482	8,232
Relations	34	24
1-to-N Instances	8,768	36,755

Table 1: Statistics of COUNTERFACT and SNOMED CT datasets. We report the number of records, subjects, objects, relations, and 1-to-N instances.

4.1 Experimental Setup

Datasets. We conduct experiments on two datasets: **COUNTERFACT** (Meng et al., 2022a) and **SNOMED CT** (Donnelly et al., 2006). To identify 1-to-N knowledge in **COUNTERFACT**, we follow Elazar et al. (2021) to extract subject-relation pairs from queries within **COUNTERFACT** and retrieve corresponding objects from Wikidata (Vrandečić and Krötzsch, 2014), revealing that approximately 40% of queries have multiple valid answers. We further estimate the real-world prevalence of these answers by following Kandpal et al. (2023), mapping query-answer pairs to pretraining documents and computing their co-occurrence number. For dataset splitting, we strictly adhere to the original splits provided by Meng et al. (2022a) to ensure consistency in evaluation. For **SNOMED CT**, we extract over 200,000 triples and identify 1-to-N knowledge by detecting subjects linked to multiple objects via the same relation. We annotate PubMed (Roberts, 2001) using PubTator (Wei et al., 2013) and perform entity linking with SapBERT (Liu et al., 2021) to compute co-occurrence number, excluding triples with zero co-occurrence due to lack of textual support. Dataset statistics are shown in Table 1, and the data splitting procedure is described in Appendix C.

Large Language Models. Following previous work (Meng et al., 2022a; Yao et al., 2023; Wang et al., 2024), we select three widely used LLMs in knowledge probing and editing: Llama3-8B (Dubey et al., 2024), Mistral-7B (Jiang et al., 2023), and GPT-J-6B (Wang and Komatsuzaki, 2021).

Knowledge Editing Methods. For knowledge editing, we evaluate the following methods using our NKL and the compared evaluation metrics. Editing details are provided in Appendix D.

- **ROME** (Meng et al., 2022a): ROME modifies an MLP layer by treating it as a key-value memory, allowing new information to be integrated. It utilises causal mediation analysis to precisely identify the optimal editing location.
- **MEMIT** (Meng et al., 2023): Building on ROME's localisation strategies, MEMIT introduces explicit parameter updates across multiple layers to embed new knowledge more effectively.
- **MEND** (Mitchell et al., 2022): MEND enables rapid, targeted updates by applying low-rank gradient transformations. It facilitates quick and localised model modifications using a single input-output example while mitigating overfitting.
- **FT** (Yao et al., 2023): FT refines model parameters via gradient descent, focusing updates on a single MLP layer identified by ROME.

Compared Evaluation Metrics. To comprehensively evaluate 1-to-N knowledge within LLMs, we compare our proposed **N-Answer Kullback-Leibler Divergence (NKL)** with several existing metrics that have been widely adopted in prior work (Yao et al., 2023; Li et al., 2022; Sung et al.,

Metric	Criterion 1	Criterion 2
Max ACC	0.4030	0.1776
Precision@K	0.4537	0.2084
Perplexity	0.3780	0.4285
NKL	**0.7352**	**0.7624**

Table 2: Pearson correlations of evaluation metrics with Criterion 1 and Criterion 2. Higher values indicate stronger alignment with coverage (Criterion 1) and probability alignment (Criterion 2) in assessing 1-to-N knowledge in LLMs.

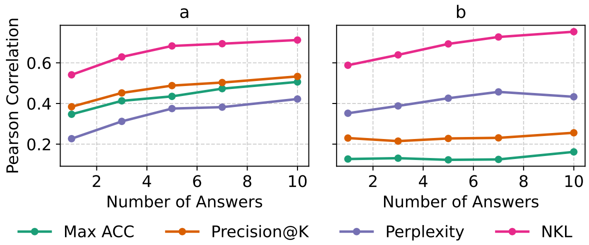

Figure 3: Pearson correlations of evaluation metrics with Criterion 1 (a) and Criterion 2 (b) across different answer quantities in the simulation experiment. Higher values indicate stronger alignment with the respective criterion.

2021; Jiang et al., 2020; Onoe et al., 2022). Specifically, we include: **Maximum Accuracy (MaxAcc)**, which captures whether the model's top-1 prediction is correct; **Top-K Precision (Precision@K)**, which considers whether any correct answer appears among the top-K predictions; and **Perplexity (PPL)**, which reflects the model's overall uncertainty over the answer space. Formal definitions and formulations of these metrics can be found in § 2.2.

4.2 Simulation Experiment

(RQ1) : Does NKL more effectively meet both evaluation criteria for measuring 1-to-N knowledge than the compared metrics?

To evaluate the effectiveness of NKL in measuring **Criterion 1** (coverage of valid answers) and **Criterion 2** (alignment with real-world probability distributions), we follow Li et al. (2020) and conduct a simulation experiment by simulating language model predictions on 1-to-N knowledge using a controlled distribution.

Simulation Setup. As illustrated in Figure 6, we design two controlled strategies to evaluate whether evaluation metrics meet two desired criteria: sensitivity to answer coverage (Criterion 1) and sensitivity to probability alignment (Criterion 2). To isolate these factors, we construct synthetic distributions sampled from a standard Gaussian. The predictive distribution simulates a language model's output by normalising scores over the full vocabulary, while the gold distribution retains the scores assigned to correct answers, normalised to reflect an ideal 1-to-N distribution. Based on these two distributions, we introduce the following strategies:

(1) Answer Masking (For Criterion 1): To simulate reduced answer coverage, we progressively remove high-probability correct answers from the predictive distribution. This yields a series of increasingly degraded distributions—from full coverage (all correct answers present) to low coverage (all correct answers removed). As shown in Figure 6, this mimics real-world cases where a model only recalls a subset of correct answers. We then compute Pearson correlation coefficients between each metric and this degradation sequence to assess how well the metric captures coverage sensitivity.

(2) Linear Interpolation (For Criterion 2): To examine whether a metric is sensitive to probability alignment, we linearly interpolate between the gold distribution $Q(x)$ and the predictive distribution $P_M(x)$, forming a continuum of intermediate distributions:

$$P_\lambda(x|q) = \lambda Q(x|q) + (1-\lambda)P_M(x|q). \quad 461$$

Varying λ from 1 to 0 transitions the distribution from perfect alignment ($\lambda = 1$) to the model's original prediction ($\lambda = 0$). We again compute Pearson correlations to evaluate how well each metric responds to increasing misalignment.

Results and Analysis. Table 2 presents the correlation of each metric with **Criterion 1** and **Criterion 2**, while Figure 3 further examines how performance varies with the number of correct answers. As shown in Table 2, NKL exhibits the highest correlation with **Criterion 1**, substantially exceeding MaxAcc and Precision@K. This suggests that NKL more effectively captures the extent to which an LLM retrieves a diverse set of valid answers, rather than overfitting to a single dominant response. In contrast, Perplexity shows a weaker correlation, indicating that it does not sufficiently emphasise coverage. Figure 3 (a) further examines the effect of varying the number of correct answers. NKL consistently demonstrates the strongest correlation with **Criterion 1** across all settings, while the correlations of MaxAcc and Precision@K remain relatively low.

Model	COUNTERFACT			
	NKL (↓)	MaxAcc (↑)	Prec@K (↑)	PPL (↓)
Llama3	**5.03**	**0.28** (–)	**0.34** (–)	**172.45** (–)
Mistral	6.38	0.24 (–)	0.27 (–)	256.32 (–)
GPT-J	7.13	0.13 (–)	0.15 (–)	325.67 (–)
Model	**SNOMED CT**			
	NKL (↓)	**MaxAcc (↑)**	**Prec@K (↑)**	**PPL (↓)**
Llama3	6.41	**0.24** (⊕1)	**0.38** (⊕1)	**145.23** (⊕1)
Mistral	**5.38**	0.23 (⊖1)	0.35 (⊖1)	238.57 (⊖1)
GPT-J	7.10	0.11 (–)	0.16 (–)	402.78 (–)

Table 3: Ranking comparison of LLM baselines on COUNTERFACT and SNOMED CT. Each column reports a metric score, with NKL as the reference. For MaxAcc, Prec@K, and PPL, we indicate whether a model's ranking on that metric differs from its NKL-based ranking: ⊕ / ⊖ indicate higher/lower rankings, with the number showing the change magnitude, and "–" denotes no change.

For **Criterion 2**, NKL again achieves the highest correlation, indicating its ability to assess whether a model appropriately distributes probability mass among correct answers in accordance with their real-world prevalence. Perplexity also exhibits a moderate correlation, as it captures probability distributions but lacks sensitivity to ranking. In contrast, MaxAcc and Precision@K show only weak correlations, reinforcing their insensitivity to probability calibration. Figure 3 (b) further supports these findings, illustrating that NKL maintains the highest correlation with **Criterion 2** as the number of valid answers increases, while the performance of other metrics remains relatively stagnant. Notably, NKL's correlation continues to rise with increasing answer count, whereas Perplexity plateaus. This indicates that NKL effectively captures additional structural information in 1-to-N knowledge, leveraging both probability alignment and ranking sensitivity. As valid answers increase, NKL more effectively captures nuanced distributions, outperforming Perplexity in modeling real-world knowledge prevalence.

4.3 Benchmarking 1-to-N Knowledge

(RQ2) : How well do existing LLMs perform in benchmarking 1-to-N knowledge probing?

To evaluate how well LLMs capture 1-to-N knowledge, we benchmark their performance on Counterfact and SNOMED CT using multiple evaluation metrics. Table 3 presents the probing performance of different LLMs. We observe that Llama3 achieves the highest MaxAcc and Precision@K on both datasets, suggesting that it retrieves more correct answers compared to Mistral and GPT-J. However, on SNOMED CT, Mistral achieves a lower NKL, indicating that while Llama3 retrieves more correct answers, Mistral's probability distribution aligns better with real-world knowledge. Meanwhile, GPT-J exhibits the highest NKL and PPL values across both datasets, suggesting that its predictive probabilities deviate the most from the expected answer distribution, making it the least aligned with real-world knowledge.

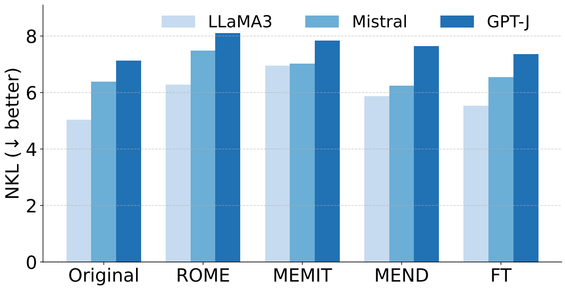

Figure 4: An evaluation of NKL on Llama3, Mistral, and GPT-J under various knowledge editing methods. NKL measures the divergence between the model's predictions and the gold distribution. The figure shows that all editing methods increase NKL relative to the original models, with ROME generally introducing larger shifts than others.

(RQ3) : How well do existing editing methods adapt to 1-to-N knowledge?

We evaluate the effectiveness of existing knowledge editing methods in handling 1-to-N knowledge. Table 4 presents the performance of various editing techniques across three base models (Llama3, Mistral, and GPT-J) on **COUNTERFACT**. We observe that all editing methods significantly improve MaxAcc and Precision@K, demonstrating their ability to make newly injected knowledge more retrievable. Among them, ROME achieves the highest improvements in both metrics, suggesting that its key-value intervention mechanism effectively strengthens the recall of edited knowledge. MEMIT also performs well, though slightly below ROME, while MEND and FT show relatively moderate improvements. Across all editing methods, Llama3 achieves the highest post-editing retrieval performance, demonstrating strong capacity for integrating new knowledge. Although GPT-J attains lower absolute scores compared to Llama3 and Mistral, it exhibits the most substantial relative improvement post-editing, indicating a high degree of responsiveness to knowledge editing despite its weaker baseline. These results highlight

Method	NKL ↓	MaxAcc ↑	Prec@K ↑	PPL ↓
Llama3	5.03	0.28	0.34	172.45
ROME	6.28 (↑1.25)	**0.68** (↑0.40)	**0.75** (↑0.41)	324.87 (↑152.42)
MEMIT	6.94 (↑1.91)	0.59 (↑0.31)	0.64 (↑0.30)	355.23 (↑182.78)
MEND	5.86 (↑0.83)	0.56 (↑0.28)	0.60 (↑0.26)	**231.14** (↑58.69)
FT	**5.52** (↑0.49)	0.38 (↑0.10)	0.47 (↑0.13)	297.52 (↑125.07)
Mistral	6.38	0.24	0.27	256.32
ROME	7.02 (↑0.64)	**0.62** (↑0.39)	**0.70** (↑0.44)	372.14 (↑115.82)
MEMIT	7.48 (↑1.10)	0.56 (↑0.33)	0.62 (↑0.35)	398.76 (↑142.44)
MEND	**6.24** (↓0.14)	0.52 (↑0.29)	0.57 (↑0.31)	**275.34** (↑19.02)
FT	6.53 (↑0.15)	0.35 (↑0.12)	0.46 (↑0.19)	312.47 (↑56.15)
GPT-J	7.13	0.13	0.15	325.67
ROME	8.10 (↑0.97)	**0.58** (↑0.46)	**0.68** (↑0.52)	448.29 (↑122.62)
MEMIT	**7.33** (↑0.20)	0.53 (↑0.40)	0.60 (↑0.45)	**384.58** (↑58.91)
MEND	7.64 (↑0.51)	0.50 (↑0.37)	0.56 (↑0.41)	391.74 (↑66.07)
FT	7.96 (↑0.83)	0.30 (↑0.17)	0.41 (↑0.26)	420.13 (↑94.46)

Table 4: Knowledge editing performance on **COUNTERFACT**. Lower NKL and PPL indicate better performance, while higher MaxAcc and Precision@K are desirable.

the effectiveness of existing editing approaches in enhancing the retrieval of injected knowledge for 1-to-N settings.

4.4 Analysing Probability Alignment in 1-to-N Knowledge

(RQ4) : Do existing editing methods maintain probability alignment in 1-to-N knowledge?

Although existing methods improve retrieval, they fail to preserve probability alignment. As shown in Table 4, NKL consistently increases post-editing, indicating that the predictive distributions diverge from the expected gold distribution. This suggests that while edited facts become more retrievable, their probability assignments no longer reflect real-world prevalence. Figure 4 further shows that ROME achieves the highest retrieval performance but also yields the largest NKL in most cases, suggesting excessive probability redistribution that overemphasises edited facts. In contrast, FT produces the lowest NKL, indicating better preservation of the probability landscape, though at the cost of weaker retrieval gains.

(RQ5) : Why does knowledge editing degrade NKL performance on 1-to-N knowledge?

As reported in Table 4 and Figure 4, NKL consistently increases, indicating greater divergence between the predicted and gold distributions after knowledge editing. To better understand this effect, we present a case study in Figure 5. The question "Which disease can cause chest pain?" has multiple correct answers with varying frequencies. After editing, the probability of the target answer (e.g., *Myocardial infarction*) increases, improving its retrievability and thereby boosting retrieval-based metrics such as MaxAcc and Precision@K. However, this improvement comes at the cost of reduced probabilities for other correct answers (e.g., *Pulmonary embolism*), leading to a less faithful alignment with the full gold distribution. This suggests that editing 1-to-N knowledge may introduce unintended interference among correct answers, which partially explains the observed degradation in NKL. These findings highlight a challenge in 1-to-N knowledge editing: existing methods prioritise recall but fail to maintain probability alignment. The NKL increase underscores the need for future approaches that not only enhance retrieval but also ensure probability distributions remain consistent with real-world knowledge.

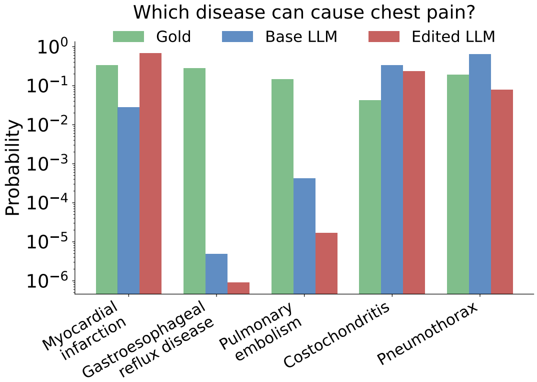

Figure 5: Case study on the question "Which disease can cause chest pain?" The gold distribution is compared with the outputs of the base LLaMA 3 and the edited Llama 3 by MEMIT. Editing improves alignment with the gold distribution, particularly for frequent answers.

5 Conclusion

We introduce N-Answer Kullback-Leibler Divergence (NKL) as a novel metric for evaluating 1-to-N knowledge in LLMs by integrating both ranking and probability alignment. Theoretical analysis confirms that NKL better satisfies key evaluation criteria for multi-answer evaluation, while knowledge probing reveals that existing LLMs tend to overfit to dominant answers. Knowledge editing experiments show that while current methods improve retrieval accuracy, they often distort probability distributions, leading to increased NKL. We also explore broader applications of NKL beyond editing, in Appendix B. We hope that NKL can serve as a faithful and comprehensive metric for evaluating LLMs' ability to represent diverse and probabilistically coherent 1-to-N knowledge.

Limitations

One limitation of our approach lies in the construction of the gold probability distribution used to compute NKL. Specifically, we approximate the gold distribution based on the co-occurrence frequencies of answers within the dataset. While this provides a practical and scalable proxy for real-world answer prevalence, it may not accurately reflect ground truth probabilities, especially in domains where frequency does not directly correspond to importance, correctness, or expert consensus. This introduces a potential source of bias in the evaluation, as models aligned with frequency-based distributions may not necessarily reflect true knowledge fidelity. Moreover, in datasets with limited coverage or long-tail distributions, co-occurrence counts may be sparse or noisy, further affecting the robustness of NKL. Future work could explore leveraging human annotations, curated ontologies, or probabilistic knowledge graphs to build more reliable gold distributions for evaluating 1-to-N knowledge representations.

References

Rie Kubota Ando and Tong Zhang. 2005. A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learning Research*, 6:1817–1853.

Galen Andrew and Jianfeng Gao. 2007. Scalable training of L_1-regularized log-linear models. In *Proceedings of the 24th International Conference on Machine Learning*, pages 33–40.

Nicola De Cao, Wilker Aziz, and Ivan Titov. 2021. Editing factual knowledge in language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6491–6506.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2022. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pages 8493–8502.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 4171–4186.

Kevin Donnelly and 1 others. 2006. Snomed-ct: The advanced terminology and coding system for ehealth. *Studies in health technology and informatics*, 121:279.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

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

Hady Elsahar, Pavlos Vougiouklis, Atanas Remaci, Christophe Gravier, Jonathon Hare, and Frédérique Laforest. 2018. T-rex: A large scale alignment of natural language with knowledge base triples. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC)*.

Flávio D Fuchs and Paul K Whelton. 2020. High blood pressure and cardiovascular disease. *Hypertension*, 75(2):285–292.

Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. 2023. Dissecting recall of factual associations in auto-regressive language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12216–12235.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2021. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5484–5495.

Gaurav Ghosal, Tatsunori Hashimoto, and Aditi Raghunathan. 2024. Understanding finetuning for factual knowledge extraction. *arXiv preprint arXiv:2406.14785*.

Rebecca Green. 2001. Relationships in the organization of knowledge: an overview. *Relationships in the organization of knowledge*, pages 3–18.

Kelvin Han, Thiago Castro Ferreira, and Claire Gardent. 2022. Generating questions from wikidata triples. In *13th Edition of its Language Resources and Evaluation Conference*.

Xuming Hu, Junzhe Chen, Xiaochuan Li, Yufei Guo, Lijie Wen, Philip S. Yu, and Zhijiang Guo. 2023. Do large language models know about facts? *CoRR*, abs/2310.05177.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, and 1 others. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Zhengbao Jiang, Antonios Anastasopoulos, Jun Araki, Haibo Ding, and Graham Neubig. 2020. X-factr: Multilingual factual knowledge retrieval from pretrained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5943–5959.

Nikhil Kandpal, Haikang Deng, Adam Roberts, Eric Wallace, and Colin Raffel. 2023. Large language models struggle to learn long-tail knowledge. In *International Conference on Machine Learning*, pages 15696–15707. PMLR.

Solomon Kullback and Richard A Leibler. 1951. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86.

Jiedong Lang, Zhehao Guo, and Shuyu Huang. 2024. A comprehensive study on quantization techniques for large language models. In *2024 4th International Conference on Artificial Intelligence, Robotics, and Communication (ICAIRC)*, pages 224–231. IEEE.

Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. 2017. Zero-shot relation extraction via reading comprehension. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL)*.

Haonan Li, Martin Tomko, Maria Vasardani, and Timothy Baldwin. 2022. Multispanqa: A dataset for multi-span question answering. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1250–1260.

Zuchao Li, Rui Wang, Kehai Chen, Masso Utiyama, Eiichiro Sumita, Zhuosheng Zhang, and Hai Zhao. 2020. Data-dependent gaussian prior objective for language generation. In *International Conference on Learning Representations*.

Fangyu Liu, Ehsan Shareghi, Zaiqiao Meng, Marco Basaldella, and Nigel Collier. 2021. Self-alignment pretraining for biomedical entity representations. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics*, pages 4228–4238.

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

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, pages 9802–9822.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022a. Locating and editing factual associations in gpt. *Advances in Neural Information Processing Systems*, 35:17359–17372.

Kevin Meng, Arnab Sen Sharma, Alex J. Andonian, Yonatan Belinkov, and David Bau. 2023. Mass-editing memory in a transformer. In *The Eleventh International Conference on Learning Representations*.

Zaiqiao Meng, Fangyu Liu, Ehsan Shareghi, Yixuan Su, Charlotte Collins, and Nigel Collier. 2022b. Rewire-then-Probe: A contrastive recipe for probing biomedical knowledge of pre-trained language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pages 4798–4810.

Eric Mitchell, Charles Lin, Antoine Bosselut, Chelsea Finn, and Christopher D. Manning. 2022. Fast model editing at scale. In *The Tenth International Conference on Learning Representations*.

Yasumasa Onoe, Michael JQ Zhang, Eunsol Choi, and Greg Durrett. 2022. Entity cloze by date: What lms know about unseen entities. *arXiv preprint arXiv:2205.02832*.

Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander Miller, and Sebastian Riedel. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Aske Plaat, Annie Wong, Suzan Verberne, Joost Broekens, Niki van Stein, and Thomas Back. 2024. Reasoning with large language models, a survey. *arXiv preprint arXiv:2407.11511*.

Mohammad Sadegh Rasooli and Joel R. Tetreault. 2015. Yara parser: A fast and accurate dependency parser. *Computing Research Repository*, arXiv:1503.06733. Version 2.

Richard J Roberts. 2001. Pubmed central: The genbank of the published literature.

Timo Schick and Hinrich Schütze. 2021. Exploiting cloze questions for few-shot text classification and natural language inference. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*.

Taylor Shin, Yasaman Razeghi, Robert Logan IV, Eric Wallace, and Sameer Singh. 2020. Autoprompt: Eliciting knowledge from language models with automatically generated prompts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Mujeen Sung, Jinhyuk Lee, Sean Yi, Minji Jeon, Sungdong Kim, and Jaewoo Kang. 2021. Can language models be biomedical knowledge bases? *arXiv preprint arXiv:2109.07154*.

M Thandi, Sharde Brown, and Sabrina T Wong. 2021. Mapping frailty concepts to snomed ct. *International Journal of Medical Informatics*, 149:104409.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85.

Ben Wang and Aran Komatsuzaki. 2021. GPT-J-6B: A 6 billion parameter autoregressive language model.

Cunxiang Wang, Xiaoze Liu, Yuanhao Yue, Xiangru Tang, Tianhang Zhang, Cheng Jiayang, Yunzhi Yao, Wenyang Gao, Xuming Hu, Zehan Qi, Yidong Wang, Linyi Yang, Jindong Wang, Xing Xie, Zheng Zhang, and Yue Zhang. 2023. Survey on factuality in large language models: Knowledge, retrieval and domain-specificity. *CoRR*, abs/2310.07521.

Song Wang, Yaochen Zhu, Haochen Liu, Zaiyi Zheng, Chen Chen, and Jundong Li. 2024. Knowledge editing for large language models: A survey. *ACM Computing Surveys*, 57(3):1–37.

Chih-Hsuan Wei, Hung-Yu Kao, and Zhiyong Lu. 2013. Pubtator: a web-based text mining tool for assisting biocuration. *Nucleic acids research*, 41(W1):W518–W522.

Yunzhi Yao, Peng Wang, Bozhong Tian, Siyuan Cheng, Zhoubo Li, Shumin Deng, Huajun Chen, and Ningyu Zhang. 2023. Editing large language models: Problems, methods, and opportunities. *arXiv preprint arXiv:2305.13172*.

Ce Zheng, Lei Li, Qingxiu Dong, Yuxuan Fan, Zhiyong Wu, Jingjing Xu, and Baobao Chang. 2023. Can we edit factual knowledge by in-context learning? In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4862–4876, Singapore. Association for Computational Linguistics.

Andy Zhou, Bo Li, and Haohan Wang. 2024. Robust prompt optimization for defending language models against jailbreaking attacks. *arXiv preprint arXiv:2401.17263*.

Wei Zhu, Aaron Xuxiang Tian, Congrui Yin, Yuan Ni, Xiaoling Wang, and Guotong Xie. 2024. Iapt: Instruction-aware prompt tuning for large language models. *arXiv preprint arXiv:2405.18203*.

Appendix

In the Appendix, we introduce more details along with related works, discussions on applications, dataset details and experimental details:

- **Appendix A**: Related Works.
- **Appendix B**: Discussion
- **Appendix C**: Dataset Details.
- **Appendix D**: Experimental Details.

A Related Work

A.1 Knowledge Probing

Knowledge probing aims to assess the factual knowledge stored within LLMs. A foundational method in this area is LAMA (Petroni et al., 2019), which uses cloze-style prompts to test whether LLMs can recover factual triples from a knowledge base. LAMA (Petroni et al., 2019) demonstrated that even without fine-tuning, models like BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019) encode substantial factual knowledge. Subsequent work expanded upon LAMA in multiple directions. For example, T-REX (Elsahar et al., 2018) and Google-RE (Levy et al., 2017) datasets introduced broader and more diverse relation types. PET (Schick and Schütze, 2021) and AutoPrompt (Shin et al., 2020) explored more flexible or learned prompt templates to improve probing accuracy. These approaches highlighted the sensitivity of probing performance to prompt phrasing.

Recent research has proposed advanced methods to enhance LLMs' factual capabilities. Instruction-Aware Prompt Tuning (IAPT) (Zhu et al., 2024) introduces a parameter-efficient mechanism using only four soft tokens per layer to generate input-specific prompts, improving performance across tasks. Ghosal et al. (2024) show that fine-tuning on obscure or less prominent facts can impair factual accuracy, even if those facts were seen during pre-training, stressing the need to consider how knowledge is encoded. To improve robustness, Zhou et al. (2024) propose Robust Prompt Optimization (RPO), which defends against jailbreaking attacks by optimizing a small, transferable prompt suffix to resist adversarial inputs. Unlike prior work, our study focuses on evaluating 1-to-N knowledge with an emphasis on probability alignment.

A.2 Knowledge Editing

Recent advancements in knowledge editing for large language models (LLMs) can be categorized into three primary strategies: memory augmentation, meta-learning, and the locate-and-modify paradigm (Yao et al., 2023).

Memory-augmented techniques integrate external memory components to enable knowledge updates without altering the core model parameters. A representative method, IKE (Zheng et al., 2023), retrieves relevant content from an attached memory bank and leverages tailored prompt demonstrations to steer the model's output accordingly. This approach emphasizes modularity and avoids direct intervention in the model's internal weights.

Meta-learning-based methods take a different route by dynamically generating weight adjustments. For instance, Knowledge Editor (KE) (Cao et al., 2021) employs a hypernetwork to synthesize updated weights in response to new knowledge. MEND (Mitchell et al., 2022) enhances this concept with low-rank gradient adaptations, offering improved efficiency. Nonetheless, these approaches often remain resource-intensive and risk modifying unrelated internal representations.

Locate-then-edit frameworks concentrate on precisely identifying and editing specific model components tied to factual knowledge. KN (Dai et al., 2022) applies attribution techniques to pinpoint influential neurons, though it struggles with fine-grained weight modification. ROME (Meng et al., 2022a) addresses this by using causal tracing to locate critical Feed Forward Network (FFN) layers—regarded as key-value storage units in transformers (Geva et al., 2021, 2023)—and directly alters their weights. MEMIT (Meng et al., 2023) scales this approach to support editing of multiple facts in parallel.

Compared to these work, we introduce a novel evaluation metric designed for the *1-to-N knowledge editing* setting, which remains underexplored in existing research.

B Discussion

This section discusses the potential applications of the proposed **N**-Answer **K**ullback-**L**eibler Divergence (NKL) metric. We highlight its utility both in knowledge editing scenarios and in domains such as clinical and biomedical reasoning, where modeling and evaluating probabilistic distributions over multiple valid answers is essential.

The proposed **N**-Answer **K**ullback-**L**eibler Divergence (NKL) metric is particularly well-suited to the evaluation of *1-to-N* knowledge representations in large language models (LLMs). A principal application of this metric lies in the domain of **knowledge editing**, wherein a model's internal distribution over semantically related factual statements must be revised in a controlled and principled manner. For instance, editing the statement "COVID-19 is caused by coronavirus" requires not only the correction of the explicit phrasing, but also a meaningful redistribution of probability across related variants, such as "COVID-19 is caused by SARS-CoV-2." The NKL metric affords a nuanced measure of the degree to which the post-edit distribution conforms to a reference distribution, thereby supporting interpretable and fine-grained evaluation. Furthermore, NKL may be employed as an optimisation objective in the formulation of editing procedures—such as constrained fine-tuning or distributionally-regularised updates—facilitating the deliberate and coherent modification of modelled knowledge.

A second critical application of NKL lies in the **clinical and biomedical domain**, where many queries naturally admit multiple valid answers, each corresponding to distinct but plausible interpretations of clinical data or medical context. In such high-stakes settings, it is insufficient for a model to merely retrieve the most likely diagnosis or treatment; rather, it must accurately represent the *full distribution* over possible alternatives. For example, when interpreting ambiguous symptoms or test results, the difference between assigning 90% versus 60% probability to a life-threatening condition can have significant consequences for downstream decision-making. NKL provides a principled framework for evaluating whether a model's probabilistic beliefs over multiple medically valid answers reflect expert-curated reference distributions, thereby enabling robust benchmarking of clinical reasoning fidelity and uncertainty calibration in LLMs.

C Dataset Details

We adopt a systematic strategy to partition both datasets—**COUNTERFACT** and **SNOMED CT**—into training, development, and test subsets. Each dataset is divided at the record level using a fixed ratio of 8:1:1. To ensure robust and fair evaluation while preventing relation-specific data leakage, we enforce the constraint that all three subsets must contain at least one instance of every relation present in the full dataset. To this end, we employ stratified random sampling based on relation type: for each relation, the corresponding triples are randomly permuted and then proportionally allocated to the three subsets in accordance with the predefined split ratio.

For **COUNTERFACT**, we identify one-to-many instances by extracting subject–relation pairs and retrieving corresponding object sets from Wikidata, revealing that roughly 40% of queries have multiple valid answers. To estimate their empirical prevalence, we follow Kandpal et al. (2023) by mapping each query–answer pair to the Wikipedia pretraining corpus and computing co-occurrence frequencies. Triples without observed co-occurrence are treated as less reliable and may be excluded from certain analyses. For dataset partitioning, we follow the original splits from Meng et al. (2022a) for

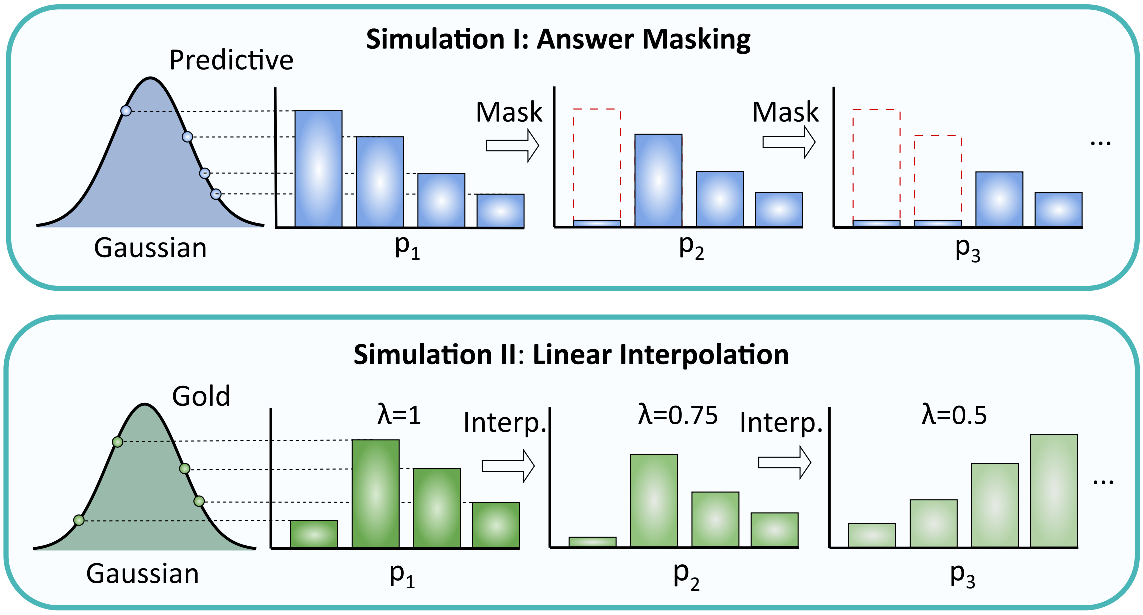

Figure 6: Illustration of two controlled strategies for simulation experiment. **Answer Masking** progressively removes probability mass from correct answers to generate increasingly degraded predictive distributions. **Linear Interpolation** constructs intermediate distributions by linearly interpolating between the gold and model-predicted distributions using decreasing λ values. Interp. is an abbreviation for Interpolation.

consistency, and apply stratified sampling where relation-level coverage is essential, ensuring each split includes all relation types.

For **SNOMED CT**, we begin by extracting over 200,000 knowledge triples from the biomedical ontology. One-to-many relations are identified by selecting subjects that are connected to multiple distinct objects through the same relation type. To assess the empirical validity of these triples, we annotate PubMed (Roberts, 2001) abstracts using PubTator (Wei et al., 2013) and apply entity linking via SapBERT (Liu et al., 2021) to compute subject–object co-occurrence frequencies. Triples lacking any observed co-occurrence in the corpus are discarded, as they are unlikely to reflect meaningful real-world associations. The filtered set is then divided into training, development, and test splits using an 8:1:1 ratio, ensuring that each split retains full coverage of the relation set.

D Experimental Details

D.1 Editing Procedures

ROME (Meng et al., 2022a): We applied ROME to Llama 3, using causal tracing to identify the optimal editing location. Layer 18 was selected as the primary target based on maximum intervention impact. We retained the default learning rate and number of editing steps from the original ROME implementation. The main tunable parameter was the scaling factor on the MLP update term within the selected layer. Edits were applied directly to test instances using these settings.

MEMIT (Meng et al., 2023): For MEMIT, we followed a similar setup to ROME, fixing the learning rate and step size as per the original paper. Based on activation analysis in Llama 3, we selected Layers 15 to 20 as editing targets. We tuned the per-layer contribution weights to balance edit success and locality, ensuring the edited facts were correctly updated without affecting unrelated outputs.

MEND (Mitchell et al., 2022): MEND was adapted to Llama 3 using default training configurations for learning rate, batch size, and epochs. We tuned the projection weights within the MEND networks, which generate low-rank updates from standard fine-tuning gradients. This enabled precise, efficient edits while minimizing interference with existing model knowledge.

Fine-Tuning (FT): For full-model fine-tuning, we used a fixed learning rate of 5e-5 across all experiments. No hyperparameter tuning was conducted beyond this, as FT was intended primarily as a performance baseline to contrast against parameter-efficient editing methods.

D.2 One-to-Many Knowledge Editing Strategy

To support one-to-many knowledge editing, we reformulate the editing objective to account for all valid answers associated with a given subject-relation pair. Specifically, we treat the full set of correct object values as editing targets, aiming to inject them into the model's internal representation. For editing methods that do not support batch updates—such as ROME (Meng et al., 2022a)—we apply edits sequentially, updating one object at a time in the order of their appearance. Each edit is performed independently, without overwriting previous modifications, allowing the model to accumulate multiple correct associations across successive interventions. In contrast, batch-editable methods like MEMIT (Meng et al., 2023) allow simultaneous updates. For these, we construct a unified batch containing all subject-relation-object triples corresponding to valid answers and perform a single joint edit. This approach ensures that all correct variants are explicitly encoded within the model's memory in a single pass, preserving interdependencies among them.

This editing protocol enables a consistent and controlled injection of 1-to-N knowledge across different methods. Moreover, it facilitates a fair comparison of post-edit generalisation, as evaluated by our NKL metric, which captures how well the edited model represents the full distribution over valid answers.