

# FacLens: Transferable Probe for Foreseeing Non-Factuality in Fact-Seeking Question Answering of Large Language Models

Yanling Wang<sup>1\*</sup>, Haoyang Li<sup>2</sup>, Hao Zou<sup>1</sup>, Jing Zhang<sup>2†</sup>, Xinlei He<sup>3</sup>, Qi Li<sup>1,4†</sup>, Ke Xu<sup>1,4</sup>

<sup>1</sup>Zhongguancun Laboratory <sup>2</sup>Renmin University of China

<sup>3</sup>Hong Kong University of Science and Technology (Guangzhou) <sup>4</sup>Tsinghua University

yanlingwang777@gmail.com zhang-jing@ruc.edu.cn qli01@tsinghua.edu.cn

## Abstract

Despite advancements in large language models (LLMs), non-factual responses still persist in fact-seeking question answering. Unlike extensive studies on post-hoc detection of these responses, this work studies non-factuality prediction (NFP), predicting whether an LLM will generate a non-factual response prior to the response generation. Previous NFP methods have shown LLMs' awareness of their knowledge, but they face challenges in terms of efficiency and transferability. In this work, we propose a lightweight model named **Factuality Lens** (FacLens), which effectively probes hidden representations of fact-seeking questions for the NFP task. Moreover, we discover that hidden question representations sourced from different LLMs exhibit similar NFP patterns, enabling the transferability of FacLens across different LLMs to reduce development costs. Extensive experiments highlight FacLens's superiority in both effectiveness and efficiency.<sup>1</sup>

## 1 Introduction

Large language models (LLMs) have shown impressive abilities in understanding and generating coherent text (OpenAI, 2023; Meta, 2024; Jiang et al., 2023), yet they may provide non-factual responses in fact-seeking question answering (fact-seeking QA) (Zhang et al., 2023; Cui et al., 2024). Extensive studies have been devoted to detecting the non-factual responses, a task we name non-factuality detection (NFD) (Manakul et al., 2023; Azaria and Mitchell, 2023; Chen et al., 2024a, 2023; Min et al., 2023a). However, these post-hoc methods require response generation, which incurs significant computational overhead. Therefore, this paper studies non-factuality prediction (NFP), which predicts the likelihood of an LLM

generating a non-factual response to a fact-seeking question before the response generation. Figure 1 (a) illustrates the difference between NFD and NFP.

To solve the NFP problem, researchers have proposed making predictions by analyzing specific tokens in a question (Mallen et al., 2023; Yükselgönül et al., 2024), making these methods applicable to specific types of questions. For more general questions, approaches based on prompting or fine-tuning the LLM for NFP have been proposed (Luo et al., 2024; Kadavath et al., 2022). However, two limitations persist: (1) current NFP models can be improved in terms of effectiveness and efficiency, and (2) they are designed for an individual LLM, lacking the transferability for rapid adaptation to new LLMs.

Inspired by studies that monitor and manipulate hidden representations to improve LLMs' performance (Zou et al., 2023; Zhang et al., 2024; Chen et al., 2024b), we hypothesize that knowledge awareness has been embedded in the hidden representations of fact-seeking questions. To investigate this, we propose a lightweight model, **Factuality Lens** (FacLens), and demonstrate its ability to probe the hidden question representations for NFP. Figure 1 (b) shows the workflow of FacLens. As hidden question representations can be efficiently obtained, and the model structure is lightweight, FacLens achieves high **efficiency in both training and prediction** (see Table 2). This makes it highly suitable for real-world LLM applications, helping to reduce non-factual responses to end-users while maintaining a minimal budget and latency. To construct the training data of FacLens, we prompt the target LLM to produce responses to questions from high-quality fact-seeking datasets. We then compare the LLM-generated answers with the golden answers, assigning binary factual/non-factual labels to the responses. Nevertheless, extending FacLens to support multiple LLMs becomes resource-

\*Work done while at Zhongguancun Laboratory.

†Corresponding authors.

<sup>1</sup>Code: <https://github.com/wyl7/FacLens>.

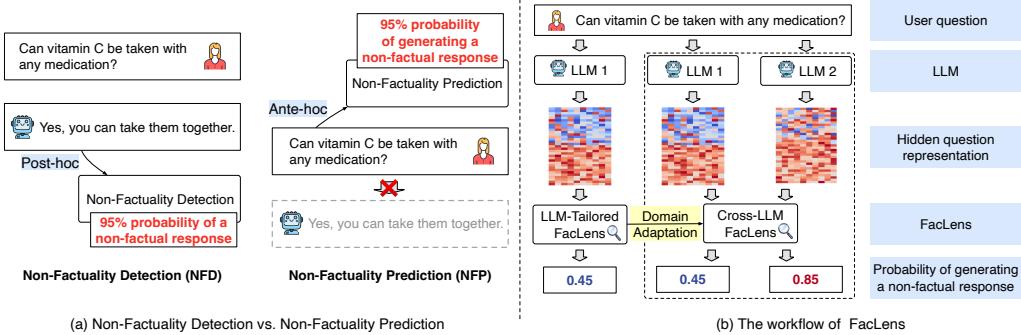


Figure 1: Illustrations of the objective and workflow of FacLens for efficient and transferable ante-hoc NFP.

intensive and time-consuming, because each LLM must conduct response generation for the training data construction. Fortunately, we discover the transferability of FacLens across different LLMs, allowing us to assign the binary labels for just one LLM and apply unsupervised domain adaptation (DA) to rapidly apply FacLens to other LLMs without collecting new labels, thereby improving the **efficiency in development**.

Investigating the transferability of FacLens is inspired by the research on human cognition (Miller et al., 2012), which shows that individuals with similar cognitive styles and encoding strategies exhibit similar brain activity when performing the same task. Given current LLMs generally follow the Transformer (Vaswani et al., 2017) architecture and share overlapping training corpora (Gao et al., 2021; com, 2024; Kocetkov et al., 2023), we hypothesize that different LLMs have similar cognitive patterns in terms of knowledge awareness (i.e., similar NFP patterns). To validate this hypothesis, we define a collection of hidden question representations sourced from a certain LLM as a data domain. Our experiments demonstrate that a FacLens trained on data from multiple domains performs similarly to the one trained on a single domain, suggesting that different individual domains do not have a significant concept shift (Moreno-Torres et al., 2012).<sup>2</sup> Building on this insight, we can quickly apply a trained FacLens to a new LLM through unsupervised domain adaptation (DA) (Smola et al., 2007; Liu et al., 2022). Additionally, we propose a question-aligned strategy to enhance the mini-batch-based DA.

This paper makes the following contributions:

- **Findings.** We verify that hidden question rep-

<sup>2</sup>Not having a significant concept shift implies highly consistent conditional distributions  $P(y|X)$  between domains.

resentations in an LLM contain valuable information for NFP in fact-seeking QA, i.e., LLMs’ activation during question understanding mostly reveals whether they know the facts. Moreover, we show that different LLMs exhibit similar NFP patterns to enable a cross-LLM NFP model.

- **Method.** We propose a lightweight and transferable NFP model named FacLens, enabling efficient development and application. To our knowledge, this is a pioneer work to train a transferable NFP model for multiple LLMs.
- **Experiments.** We conduct extensive experiments, involving four open-source LLMs and three widely used fact-seeking QA datasets. The results show that FacLens outperforms baselines in terms of AUC metric and runtime.

## 2 Related Work

**Hidden Representations in LLMs.** Hidden representations in LLMs have been shown to encode valuable information that can be leveraged for various tasks (Zou et al., 2023). In terms of LLMs’ factuality, studies like SAPLMA (Azaria and Mitchell, 2023) and MIND (Su et al., 2024) leverage hidden representations of LLM-generated responses for post-hoc NFD. TruthX (Zhang et al., 2024) edits hidden representations of LLM-generated responses via an edit vector to enhance the LLM’s truthfulness. Activation Decoding (Chen et al., 2024b) reveals that an LLM’s responses are closely tied to the representations of input entities but do not validate their effectiveness in NFP.

**Non-Factuality Prediction in LLMs.** We categorize the studies into token-based and non-token-based methods. The entity popularity-based method (Mallen et al., 2023) focuses on input entity

tokens, assuming that LLMs are more familiar with questions about popular entities and estimating entity popularity based on Wikipedia page views. However, not every question contains entities that exactly match a Wikipedia entry. SAT Probe (Yükselgönül et al., 2024) predicts based on the LLM’s attention to specific constraint tokens. The authors restrict the types and formats of questions to facilitate the identification of the constraint tokens. However, extracting constraint tokens from free-form questions is non-trivial. Without focusing on specific tokens, Self-Familiarity (Luo et al., 2024) estimates an LLM’s familiarity with the requested facts through multi-round conversations with the LLM, requiring carefully crafted prompts to engage the LLM multiple times, resulting in low prediction efficiency. Alternatively, researchers fully fine-tune the LLM for NFP (Kadavath et al., 2022) (termed Self-Evaluation). However, this approach incurs significant computational costs and may hinder the LLM’s generalization ability (Yang et al., 2024b). In contrast, FacLens exhibits good applicability and high efficiency. LLM Factoscope (He et al., 2024) estimates whether the first generated word matches a known fact using the representation of the prompt’s last token. By comparison, FacLens does not restrict the LLM’s output format and demonstrates that, beyond the representation of the prompt’s last token, question-level representations captured through alternative forms are also effective. Most importantly, unlike prior work, our method is distinguished by its ability to support cross-LLM domain adaptation.<sup>3</sup>

### 3 Preliminary

#### 3.1 Problem Definition

**Definition 1 Non-Factual Response.** *Given an LLM  $m \in \mathcal{M}$  and a fact-seeking question  $q \in \mathcal{Q}$ ,  $m$  generates an answer  $s$ . If the answer  $s$  fails to convey the queried fact, it is a non-factual response.*

**Problem 1 Non-Factuality Prediction in an LLM (NFP).** *Given an LLM  $m \in \mathcal{M}$  and a fact-seeking question  $q \in \mathcal{Q}$ , the objective is to learn a function  $f(m, q) \rightarrow y$ , where  $y = 1$  if  $m$  will generate a non-factual response to  $q$  and  $y = 0$  otherwise.*

**Problem 2 Transferable Cross-LLM NFP.** *Given*

<sup>3</sup>Hidden question representation has been utilized for predicting an LLM’s self-consistency (Liang et al., 2024). Notably, self-consistency does not equate to factuality; for instance, an LLM can consistently produce incorrect answers.

*LLMs  $m_1, m_2 \in \mathcal{M}$  and a fact-seeking question set  $\mathcal{Q}$ , NFP labels have been constructed based on  $\mathcal{Q}_{train} \subset \mathcal{Q}$  for  $m_1$ , deriving a training set  $\{((m_1, q_i), y_{1,i})\}_{q_i \in \mathcal{Q}_{train}}$ . The goal is to utilize the training set and  $m_2$  to learn a function  $f(m, q) \rightarrow y$ , where  $m \in \{m_1, m_2\}$  and  $q \in \mathcal{Q}$ .*

#### 3.2 NFP Datasets

**Dataset Construction.** Given an LLM  $m$  and a fact-seeking QA dataset, for each question  $q \in \mathcal{Q}$ , we assign a binary label  $y$  to the  $(m, q)$  pair, where  $y = 1$  if  $m$  fails to generate the golden answer for  $q$ , and  $y = 0$  otherwise. Our goal is to predict the labels prior to answer generation. Notably, a fact-seeking question asks for objective and verifiable information, such as dates, locations, and entities. Examples include “In which year was the Eiffel Tower built?” and “Which city is the capital of France?”. Their answers are naturally short and precise. Therefore, we follow previous work (Mallen et al., 2023) to mark an LLM’s response as non-factual (i.e.,  $y = 1$ ) if no sub-string of the response matches any of the gold answers.<sup>4</sup> We consider four LLMs and three QA datasets in the main body of the paper, deriving  $4 \times 3 = 12$  NFP datasets. In each NFP dataset, consisting of samples in the form of  $((m, q), y)$ , we randomly sample 20% data for training, 10% data for validation, and use the remaining data for testing.

**LLMs & QA Datasets.** LLaMA2-7B-Chat (Touvron et al., 2023), LLaMA3-8B-Instruct (Meta, 2024), Mistral-7B-Instruct-v0.2 (Jiang et al., 2023), and Qwen2-1.5B-Instruct (Yang et al., 2024a) are used for experiments. These LLMs have been instruction-tuned for conversational engagement. We pose questions from three widely-used QA datasets: PopQA (PQ) (Mallen et al., 2023), Entity Questions (EQ) (Sciavolino et al., 2021), and Natural Questions (NQ) (Kwiatkowski et al., 2019). Detailed statistics of these datasets are provided in Appendix A. To ensure reproducibility, we set each LLM’s decoding strategy to greedy search rather than top- $p$  or top- $k$  sampling. We have also run the sampling-based decoding, and find that the paper’s experimental conclusions still hold true.

<sup>4</sup>The labeling method ensures accurate labels of all positive samples. We randomly sample 20 negative samples from each NFP dataset, deriving  $20 \times 12 = 240$  negative samples, and manually checked their labels’ quality. Given that all positive samples constitute 72.2% of the dataset, the ratio of correct labels is 97.0%.

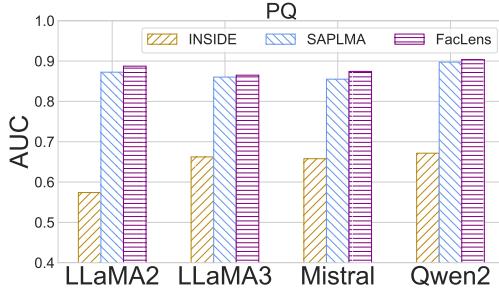


Figure 2: Performance comparison of FacLens (ante-hoc NFP) with INSIDE and SAPLMA (post-hoc NFD). Trends are consistent on EQ and NQ (see Appendix D.1).

## 4 Methodology

### 4.1 FacLens

Given an LLM  $m$  and a fact-seeking question  $q$ , we can quickly acquire the hidden states corresponding to input tokens. In a certain layer, we use the hidden states corresponding to the last input token as the question’s hidden representation  $\mathbf{x}$ . Then we use an encoder  $g_{enc}$  to transform the question’s hidden representation into a latent feature space, where we presume that the NFP patterns are represented. Afterwards, a linear classifier  $g_{clf}$  is set upon  $g_{enc}$  for classification. Formally, based on the  $\ell$ -th hidden layer of  $m$ , FacLens predicts by,

$$\mathbf{p} = g_{clf}(g_{enc}(m_{\leq \ell}(q))) = g_{clf}(g_{enc}(\mathbf{x})) \quad (1)$$

where  $m_{\leq \ell}(\cdot)$  denotes the function composed of the  $\ell$ -th transformer layer and its preceding layers,  $g_{enc}$  is implemented by a lightweight multi-layer perceptron (MLP)<sup>5</sup>,  $g_{clf}$  is implemented by a linear layer with the Softmax function, and  $\mathbf{p}$  is a two-dimensional vector revealing the probability of (not) producing non-factual responses. Based on a set of labeled NFP instances  $\{(m_{\leq \ell}(q_i), y_i)\}_{q_i \in Q_{train} \cup Q_{val}}$ , where  $Q_{train}$  and  $Q_{val}$  denote question sets used for training and validation, respectively, we can train a FacLens for  $m$  with the classic cross-entropy (CE) loss.

**Observation: While post-hoc NFD methods leverage more information, the ante-hoc FacLens has the potential to perform better.** In Figure 2, we compare FacLens with SAPLMA (Azaria and Mitchell, 2023) and INSIDE (Chen et al., 2024a),

<sup>5</sup>Our goal is to verify that hidden question representations contain useful patterns for the NFP task. Exploring other model architectures for the NFP pattern extraction is beyond the scope of this paper.

two representative post-hoc methods that leverage hidden representations of LLM-generated responses to identify non-factual responses. While post-hoc methods leverage more information (i.e., the LLM-generated responses), FacLens shows comparable and even better performance.

### 4.2 Transferability of FacLens

When it comes to multiple LLMs, the construction of NFP training data becomes resource-intensive and time-consuming, because each LLM needs to conduct costly response generation (see Section 3.2). Fortunately, we discover the transferability of FacLens, which allows us to label training data for just one LLM and adapt the FacLens to support other LLMs. In Appendix B, we illustrate the more efficient process of training data construction enabled by the transferability of FacLens.

**Why Domain Adaptation is Effective for Transferring FacLens Across LLMs.** Domain adaptation (DA) is an approach in transfer learning that transfers information from a source domain to improve performance in a target domain (Ben-David et al., 2006; Liu et al., 2022; Moreno-Torres et al., 2012). The premise of DA is that the source and target domains have distinct marginal probability distributions  $P(\mathbf{X})$ , but share similar conditional probability distributions  $P(\mathbf{y}|\mathbf{X})$  (i.e., no significant concept shift) (Liu et al., 2022; Moreno-Torres et al., 2012). Here we refer to the domain as,

**Remark 1** Let the variable  $\mathbf{X}$  represent the hidden question representation in an LLM. A data domain  $D$  refers to a collection of hidden question representations sourced from a certain LLM.

Different domains naturally have different  $P(\mathbf{X})$ . If  $P(\mathbf{y}|\mathbf{X})$  of different domains exhibit similar forms, we can perform DA to apply FacLens to other LLMs without new labels for training.

Now we verify that different data domains indeed have similar conditional distributions  $P(\mathbf{y}|\mathbf{X})$  by introducing a mixture domain  $D_{mix}$ , whose joint probability distribution is,

$$P_{mix}(\mathbf{X}, \mathbf{y}) = \sum_{i=1}^M \alpha_i \cdot P_{m_i}(\mathbf{X}, \mathbf{y}) \quad (2)$$

$$\text{s.t. } \sum_{i=1}^M \alpha_i = 1$$

where  $M$  is the number of individual data domains (i.e. the number of different LLMs),  $m_i$  denotes the  $i$ -th LLM, and  $0 < \alpha_i < 1$  represents the

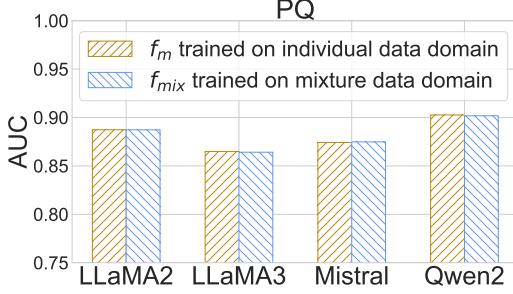


Figure 3: Performance comparison between  $f_m$  and  $f_{mix}$ . Similar performance suggests no significant concept shift across different domains. Trends are consistent on EQ and NQ (See Appendix D.2)

proportion of  $D_i$  in the mixture domain. Here we set  $\alpha_i = \frac{1}{M}$ .

Therefore, the FacLens trained on the mixture domain follows the conditional distribution,

$$P_{mix}(\mathbf{y}|\mathbf{X}) = \sum_{i=1}^M \beta_i(\mathbf{X}) \cdot P_{m_i}(\mathbf{y}|\mathbf{X}),$$

$$\beta_i(\mathbf{X}) = \frac{\alpha_i \cdot P_{m_i}(\mathbf{X})}{\sum_{j=1}^M \alpha_j \cdot P_{m_j}(\mathbf{X})} \quad (3)$$

It is readily derived that  $\sum_{i=1}^M \beta_i(\mathbf{X}) = 1$ , and  $0 < \beta_i(\mathbf{X}) < 1$  if  $P_{m_1}(\mathbf{X}), P_{m_2}(\mathbf{X}), \dots, P_{m_M}(\mathbf{X})$  are not disjoint. If there are no concept shifts between individual data domains, we have,

$$P_{mix}(\mathbf{y}|\mathbf{X}) = P_{m_1}(\mathbf{y}|\mathbf{X}) = P_{m_2}(\mathbf{y}|\mathbf{X}) \quad (4)$$

$$= \dots = P_{m_M}(\mathbf{y}|\mathbf{X})$$

Conversely, if significant concept shifts exist between individual domains, Eq. 4 is not valid, as there must exist at least a domain  $D_i$  where  $P_{mix}(\mathbf{y}|\mathbf{X})$  is very different from  $P_{m_i}(\mathbf{y}|\mathbf{X})$ . Consequently, on the test set of domain  $D_i$ ,  $f_{mix}$  will noticeably underperform  $f_{m_i}$ , where  $f_{mix}$  is trained in  $D_{mix}$ , and  $f_{m_i}$  is trained in  $D_i$ . For simplicity, we use  $f_m$  to denote a FacLens trained on an individual domain.

Given a specific fact-seeking QA dataset, we have four individual domains as our experiments consider four LLMs. Each individual domain has its training, validation, and test sets. The training sets of all the individual domains form the training set of the mixture domain. Notably, the hidden dimension of Qwen2-1.5B-Instruct is different from that of the other three LLMs, so we introduce an additional linear layer to reshape Qwen2's hidden question representations to match the dimension of the other three LLMs.

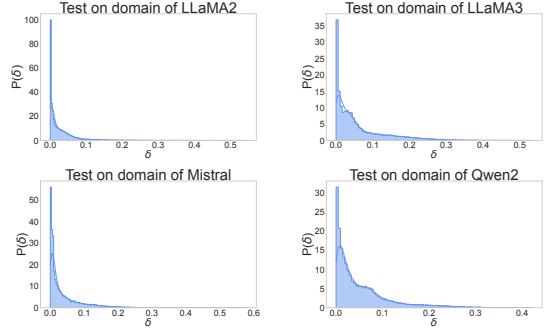


Figure 4: Distribution of  $\delta$  between an individual-domain FacLens and the mixture-domain FacLens overall questions (questions from PQ, EQ, and NQ).

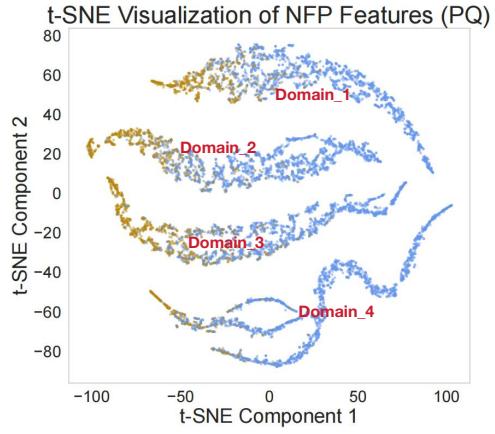


Figure 5: Visualization of NFP features extracted from different LLMs' hidden question representations, where each domain corresponds to an LLM, and the blue and dark yellow points denote the positive and negative samples, respectively. Trends are consistent on EQ and NQ (see Appendix D.2).

**Observation 1.** Figure 3 shows the results on the PQ dataset, and we can observe that  $f_{mix}$  exhibits comparable performance to  $f_m$  on the test set of the corresponding individual domain, indicating similar  $P(\mathbf{y}|\mathbf{X})$  across different domains. Hence, **we can conduct DA to derive a FacLens used by different LLMs**.

**Observation 2.** We measure concept shift between an individual domain and the mixture domain by  $\delta = \|\mathbf{p}^m(y=1|\mathbf{x}) - \mathbf{p}^{mix}(y=1|\mathbf{x})\|$ , where  $\mathbf{p}^m$  is computed by  $f_m$  and  $\mathbf{p}^{mix}$  by  $f_{mix}$ . The values of  $\delta$  are mostly near zero (see Figure 4), indicating that  $P_{m_1}(\mathbf{y}|\mathbf{X}), P_{m_2}(\mathbf{y}|\mathbf{X}), \dots, P_{m_M}(\mathbf{y}|\mathbf{X})$  and  $P_{mix}(\mathbf{y}|\mathbf{X})$  have similar distributions.

**Observation 3.** Given hidden question representations from different LLMs, we use the encoder of  $f_{mix}$  to extract the NFP features and visualize them with t-SNE. The results on PQ are shown in Fig-

ure 5, where the positive and negative samples are represented by points of different colors. Although these points are sourced from different LLMs, we can see that a unified classification boundary can be applied to them, further demonstrating the similar  $P(\mathbf{y}|\mathbf{X})$  across different domains.

### 4.3 Cross-LLM FacLens

We have verified that we can use DA to train a cross-LLM FacLens for LLM  $m_j$  leveraging label information from LLM  $m_i$ . Here the distribution shift between a source domain  $D_S$  and a target domain  $D_T$ , is due to the difference of LLMs.  $D_S$  has labeled data, yet  $D_T$  has no label information.

**Unsupervised Domain Adaptation.** The classic Maximum Mean Discrepancy (MMD) loss (Gretton et al., 2012) is used to find a domain-invariant NFP feature space, based on which FacLens predicts the labels. The MMD loss calculates the distance between two distributions in the reproducing kernel Hilbert space (RKHS) (Smola et al., 2007). We denote the NFP features in the source and target domains as  $Z_S = \{\mathbf{z}_{S,i}\}_{i=1}^{N_S}$  and  $Z_T = \{\mathbf{z}_{T,j}\}_{j=1}^{N_T}$ , respectively, where  $\mathbf{z}_{S,i} = g_{enc}(\mathbf{x}_{S,i})$ , and  $\mathbf{z}_{T,j} = g_{enc}(\mathbf{x}_{T,j})$ . The encoder  $g_{enc}$  in FacLens is optimized by minimizing the MMD loss,

$$\begin{aligned} \mathcal{L}_{\text{MMD}}(Z_S, Z_T) &= \frac{1}{N_S^2} \sum_{i,j=1}^{N_S} k(\mathbf{z}_{S,i}, \mathbf{z}_{S,j}) \\ &\quad + \frac{1}{N_T^2} \sum_{i,j=1}^{N_T} k(\mathbf{z}_{T,i}, \mathbf{z}_{T,j}) \\ &\quad - \frac{2}{N_S N_T} \sum_{i=1}^{N_S} \sum_{j=1}^{N_T} k(\mathbf{z}_{S,i}, \mathbf{z}_{T,j}) \end{aligned} \quad (5)$$

where  $N_S = N_T = |\mathcal{Q}_{train}|$  is the number of questions for training, and  $k(\cdot)$  denotes a kernel function. We extract hidden question representations from the LLM’s middle layer.

Importantly, we also use the CE loss to optimize  $g_{enc}$  and  $g_{clf}$  with the labeled data in  $D_S$ , which collaborates the MMD loss to find the latent feature space for NFP. Finally, the loss function is,

$$\mathcal{L}_{\text{DA}} = \mathcal{L}_{\text{MMD}} + \frac{1}{N_S} \sum_{i=1}^{N_S} \mathcal{L}_{\text{CE}}(g_{clf}(\mathbf{z}_{S,i}), y_{S,i}) \quad (6)$$

Notably, if LLMs have distinct hidden dimensions, we introduce an additional linear layer to reshape

the target domain’s hidden question representations to match the dimension of the source domain’s hidden question representations. We demonstrate that FacLens can transfer across LLMs of distinct hidden dimensions in Figure 6 and Appendix D.5. Besides, we discuss the choice of kernel function for MMD loss in Appendix D.6.

**Question-Aligned Mini-Batch Training.** In order to address GPU out-of-memory issues, FacLens employs mini-batch training for DA. In each mini-batch, we sample two question sets,  $\bar{\mathcal{Q}}_S$  and  $\bar{\mathcal{Q}}_T$ , from  $\mathcal{Q}_{train}$ , for two domains. This raises a question: are  $\bar{\mathcal{Q}}_S$  and  $\bar{\mathcal{Q}}_T$  identical? Given a range of questions, the distribution  $P(\mathbf{Z})$  should be determined by the LLM. In a mini-batch, the number of sampled questions is limited, so the estimation of  $P_S(\mathbf{Z})$  and  $P_T(\mathbf{Z})$  within the mini-batch is likely to be affected by the sampling process. Hence, we propose to use the same question set for two domains in each mini-batch, i.e.,  $\bar{\mathcal{Q}}_S = \bar{\mathcal{Q}}_T$ , to alleviate the influence of sampling process in estimating the true distance between  $P_S(\mathbf{Z})$  and  $P_T(\mathbf{Z})$ .

## 5 Experiments

### 5.1 Experimental Setup

**Baselines.** We compare FacLens with existing NFP methods, which have been introduced in Section 2, including **Entity-Popularity** (Mallen et al., 2023), **SAT Probe** (Yüksekgönül et al., 2024), **Self-Familiarity** (Luo et al., 2024), and **Self-Evaluation** (Kadavath et al., 2022). As Self-Evaluation fully fine-tunes the LLM for NFP, we adopt **LoRA** (Hu et al., 2022) as an additional baseline to conduct parameter-efficient fine-tuning. We also consider a **Prompting**-based method, which directly asks the LLM whether the LLM knows the factual answer to the given question. Moreover, inspired by using the perplexity to evaluate the factual precision of responses (Min et al., 2023b), we consider **perplexity (PPL)** on the input question as a baseline (see Appendix C.1).

**Hyper-Parameter Settings.** Our experiments are conducted based on 4 \* 80G NVIDIA Tesla A800 GPUs. We implement the encoder  $g_{enc}$  of FacLens by a 3-layer MLP, setting the dimension of each MLP layer to 256. We use the Adam optimizer with weight decay 1e-4. The hyper-parameters determined on the validation set include: the training epochs (set the maximum epochs to 100), and the

	LLaMA2			LLaMA3			Mistral			Qwen2		
	PQ	EQ	NQ									
PPL	72.5	67.1	56.4	69.8	65.5	53.9	69.1	67.2	57.7	74.1	64.6	57.4
Prompting	72.7	67.8	58.1	70.6	64.9	57.2	72.2	66.0	65.5	73.0	74.7	57.1
Entity-Popularity	79.0	—	—	75.9	—	—	77.6	—	—	67.9	—	—
SAT Probe	85.1	79.3	—	83.4	81.5	—	84.4	81.9	—	88.5	81.9	—
Self-Familiarity	59.1	64.9	55.8	61.8	68.4	52.0	57.1	64.9	54.2	54.1	61.8	57.6
LoRA (Parameter-Efficient FT)	88.2	84.8	67.0	86.1	83.8	63.2	84.1	81.8	65.7	90.0	85.1	73.5
Self-Evaluation (Fully FT)	88.5	85.2	68.8	85.7	<b>85.8</b>	63.9	83.5	80.9	61.9	89.7	<b>86.6</b>	71.3
FacLens-ent (avg, last layer)	76.0	79.6	60.4	75.8	77.7	57.4	76.8	77.8	59.2	84.6	77.7	65.2
FacLens-ent (avg, 2 <sup>nd</sup> to last layer)	77.9	80.5	60.4	76.2	79.0	58.0	77.1	78.3	60.5	84.5	78.6	65.1
FacLens-ent (avg, middle layer)	81.7	81.2	60.6	79.2	81.0	58.6	81.4	82.4	61.5	87.0	82.2	65.4
FacLens-ent (last token, last layer)	81.4	81.7	60.6	78.9	79.6	55.3	80.9	80.9	59.3	87.4	81.7	64.4
FacLens-ent (last token, 2 <sup>nd</sup> to last layer)	82.3	82.1	60.1	78.1	79.7	57.8	81.6	81.9	59.7	87.6	81.7	63.9
FacLens-ent (last token, middle layer)	83.5	81.4	61.2	79.9	81.0	60.0	82.9	82.8	60.5	88.0	81.5	63.5
FacLens (last token, last layer)	88.7	84.9	69.1	86.1	84.1	64.7	86.1	84.4	71.7	90.0	85.9	<b>74.0</b>
FacLens (last token, 2 <sup>nd</sup> to last layer)	<b>88.8</b>	85.0	67.7	86.1	84.1	65.6	87.0	<b>85.7</b>	<b>72.1</b>	<b>90.7</b>	85.6	72.4
FacLens (last token, middle layer)	88.7	<b>85.6</b>	<b>69.5</b>	<b>86.5</b>	85.0	<b>68.9</b>	<b>87.4</b>	85.4	71.4	90.3	86.4	71.6

“—” means the method is not suitable for the QA dataset. We give the detailed explanation in the appendix C.3. “avg” refers to the averaged hidden representation of the input entities’ tokens or a question’s tokens. “last token” refers to the hidden representation of the last token in the input entities or the question. The question consists of a chat template and the original question, where the chat template can prompt the LLM to better respond. Due to space limitation, we show the performance of FacLens (avg) in Appendix D.3.

Table 1: Prediction performance of different NFP methods (AUC %).

learning rate  $\in \{1e-3, 1e-4\}$  for single-LLM FacLens. Considering that the number of training questions from NQ are relatively small, we set the learning rate of FacLens to  $1e-4$  on NFP datasets derived from NQ. The default learning rate of cross-LLM FacLens is set to  $1e-5$ . Due to the memory limitation, we minimize the MMD loss via mini-batch training with a batch size of 64. We provide the hyper-parameter settings of baselines in Appendix C.2.

**Evaluation Metric.** As the number of positive samples is larger than that of negative samples (see Table 3), we adopt AUC, a common metric for imbalanced binary classification, as the evaluation metric.

## 5.2 Experimental Results

**Probing hidden question representations for NFP in fact-seeking QA yields promising results.** As reported in Table 1, FacLens exhibits promising performance compared to the baselines, and it favors hidden representations from LLMs’ middle layers. Specifically, Self-Evaluation and LoRA can be regarded as two special representation-based NFP methods. While both perform well, FacLens, with much fewer trainable parameters, offers higher efficiency and yields results comparable to, or even surpassing, those of Self-Evaluation and LoRA. Compared to SAT Probe, FacLens demonstrates that hidden representation could be more useful than attention weights for NFP. Furthermore, we

conduct main experiments on HotpotQA (Yang et al., 2018), a more complex multi-hop QA dataset, in Appendix D.4. The results show that FacLens continues to perform well. We also developed a demo to collect more questions and provide the case study in Appendix E.

**Question-level modeling is more effective than focusing on specific tokens.** Inspired by entity-centric studies, we propose FacLens-ent, which feeds the representations of input entity tokens into FacLens.<sup>6</sup> As shown in Table 1, FacLens consistently surpasses FacLens-ent, Entity-Popularity, Self-Familiarity, and SAT Probe, suggesting that overemphasizing specific tokens of the input question may mislead the predictions.

**FacLens stands out for its efficiency.** Taking the LLaMA2-PQ NFP dataset as an example, which includes 2,272 questions for training, 1,136 questions for validation, and 7,952 questions for testing, Table 2 reports the average training time per epoch and the average prediction time per question of each method. We can see that fine-tuning an LLM (i.e., Self-Evaluation and LoRA) incurs significant computational costs. Importantly, if the LLM has been fine-tuned for a specific task, its ability on other tasks can be compromised (Yang et al., 2024b). As for prediction, FacLens runs much faster than Self-Familiarity because Self-

<sup>6</sup>Entities, e.g., persons, locations, organizations, are identified using the Stanza NLP Package (Qi et al., 2020).

	Training-Free	Transferable	Training Time Per Epoch (avg)	Prediction Time Per Question (avg)
Self-Familiarity	Yes	–	–	5.838s
Prompting	Yes	–	–	0.115s
PPL	Yes	–	–	0.044s
LoRA (Parameter-Efficient FT)	No	No	116.500s (1 * 80G A800)	0.038s
Self-Evaluation (Fully FT)	No	No	184.778s (4 * 80G A800)	0.028s
SAT Probe	No	No	0.010s (1 * 80G A800)	0.037s
FacLens	No	Yes	0.012s (1 * 80G A800)	0.016s

Note: Here FacLens denotes FacLens (last token, middle layer). Before training FacLens, we extract hidden question representations of training questions, a process that takes 71.856s. SAT Probe involves a feature extraction step, where attention weights are extracted for probing, taking 132.860s.

Table 2: Efficiency evaluation of FacLens (seconds).

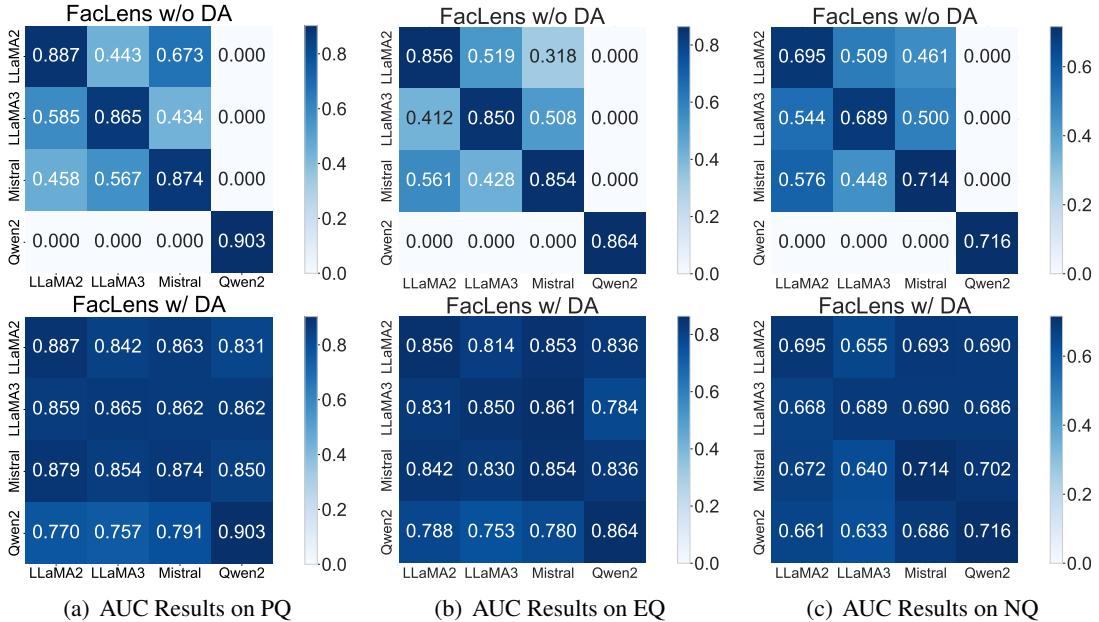


Figure 6: Performance of cross-LLM FacLens w/o and w/ DA. In each heatmap, the element in the  $i$ -th row and  $j$ -th column represents the performance on the  $j$ -th target domain, with label information transferred from the  $i$ -th source domain.

Familiarity involves multiple conversations with the LLM. As FacLens uses the middle-layer hidden question representations, it runs faster than Prompting, PPL, LoRA, and Self-Evaluation that involve more layers in the LLM. SAT Probe extracts attention weights across all layers and attention heads, so FacLens runs faster during prediction.

**Unsupervised domain adaptation performs well for cross-LLM FacLens.** Given an LLM, we train FacLens with the training data of the corresponding domain and directly test it on the test data of another domain. The results in the upper part of Figure 6 are unsatisfactory. After unsupervised DA (MMD loss with linear kernel), the cross-LLM FacLens can work much better in the target domain, as depicted in the the lower part of Figure 6. Furthermore, we observe that FacLens shows better

transferability between LLMs of similar scales. In future work, we will explore more effective methods to enhance FacLens’s transferability between LLMs of very different scales.

**Question-aligned strategy is necessary to mini-batch training of cross-LLM FacLens.** Figure 7 shows that our question-aligned strategy for mini-batch training significantly enhances the performance of cross-LLM FacLens. Particularly on the NQ dataset released by Google, which consists of questions from real users and covers more diverse questions, the estimation of  $P_S(\mathbf{Z})$  and  $P_T(\mathbf{Z})$  is more likely to be influenced by the sampling process in a mini-batch. Hence, integrating the question-aligned strategy fosters the training process more on NQ.

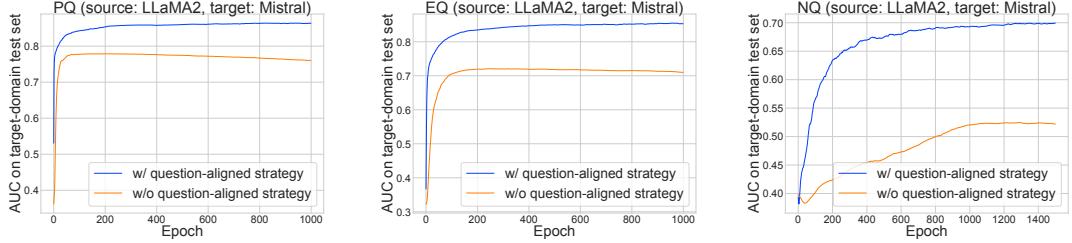


Figure 7: Evaluation of the question-aligned mini-batch training strategy. Similar trends appear on other pairs of source and target domains.

## 6 Conclusion

In this paper, we find that the hidden representation of a fact-seeking question contains valuable information for identifying potential non-factual responses (i.e., NFP). We also discover that similar NFP patterns emerge in hidden question representations sourced from different LLMs. These findings support our lightweight and transferable NFP model, FacLens, which enables more efficient development and application. We hope this work can inspire future research on LLMs' factuality.

## Limitations

This paper assumes access to the parameters of LLMs, which limits the application of FacLens to API-based black-box LLMs. With the growing availability of advanced open-source LLMs, research on white-box methods is becoming increasingly important, making our work meaningful. However, we also recognize the necessity of designing effective and efficient NFP models for black-box LLMs in our future work.

## Acknowledgments

This work was supported in part by the National Science Foundation for Distinguished Young Scholars of China under No. 62425201, the Key Program of the National Natural Science Foundation of China under No. 62132011, and the Science Fund for Creative Research Groups of the National Natural Science Foundation of China under No. 62221003. We would like to thank the reviewers for their valuable comments.

## References

2024. Common crawl. <https://commoncrawl.org/>.

Amos Azaria and Tom M. Mitchell. 2023. The internal state of an LLM knows when it’s lying. In *Findings of EMNLP*, pages 967–976.

Jeanine Banks and Tris Warkentin. 2024. [Gemma: Introducing new state-of-the-art open models](#).

Shai Ben-David, John Blitzer, Koby Crammer, and Fernando Pereira. 2006. Analysis of representations for domain adaptation. In *NeurIPS*, pages 137–144.

Chao Chen, Kai Liu, Ze Chen, Yi Gu, Yue Wu, Mingyuan Tao, Zhihang Fu, and Jieping Ye. 2024a. INSIDE: LLMs’ internal states retain the power of hallucination detection. In *ICLR*.

Jifan Chen, Grace Kim, Aniruddh Sriram, Greg Durrett, and Eunsol Choi. 2023. [Complex claim verification with evidence retrieved in the wild](#). *CoRR*, abs/2305.11859.

Shiqi Chen, Miao Xiong, Junting Liu, Zhengxuan Wu, Teng Xiao, Siyang Gao, and Junxian He. 2024b. In-context sharpness as alerts: An inner representation perspective for hallucination mitigation. In *ICML*.

Tianyu Cui, Yanling Wang, Chuanpu Fu, Yong Xiao, Sijia Li, Xinhao Deng, Yunpeng Liu, Qinglin Zhang, Ziyi Qiu, Peiyang Li, Zhixing Tan, Junwu Xiong, Xinyu Kong, Zujie Wen, Ke Xu, and Qi Li. 2024. [Risk taxonomy, mitigation, and assessment benchmarks of large language model systems](#). *CoRR*, abs/2401.05778.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. 2021. [The pile: An 800gb dataset of diverse text for language modeling](#). *CoRR*, abs/2101.00027.

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander J. Smola. 2012. A kernel two-sample test. *J. Mach. Learn. Res.*, 13:723–773.

Jinwen He, Yujia Gong, Zijin Lin, Cheng’ an Wei, Yue Zhao, and Kai Chen. 2024. LLM factoscope: Uncovering llms’ factual discernment through measuring inner states. In *Findings of ACL*, pages 10218–10230.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. Lora: Low-rank adaptation of large language models. In *ICLR*.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. [Mistral 7b](#). *CoRR*, abs/2310.06825.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, Deep Ganguli, Danny Hernandez, Josh Jacobson, Jackson Kernion, Shauna Kravec, Liane Lovitt, Kamal Ndousse, Catherine Olson, Sam Ringer, Dario Amodei, Tom Brown, Jack Clark, Nicholas Joseph, Ben Mann, Sam McCandlish, Chris Olah, and Jared Kaplan. 2022. [Language models \(mostly\) know what they know](#). *CoRR*, abs/2207.05221.

Denis Kocetkov, Raymond Li, Loubna Ben Allal, Jia Li, Chenghao Mou, Yacine Jernite, Margaret Mitchell, Carlos Muñoz Ferrandis, Sean Hughes, Thomas Wolf, Dzmitry Bahdanau, Leandro von Werra, and Harm de Vries. 2023. The stack: 3 TB of permissively licensed source code. *Trans. Mach. Learn. Res.*, 2023.

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

Yuxin Liang, Zhuoyang Song, Hao Wang, and Jiaxing Zhang. 2024. [Learning to trust your feelings: Leveraging self-awareness in llms for hallucination mitigation](#). *CoRR*, abs/2401.15449.

Xiaofeng Liu, Chae Hwa Yoo, Fangxu Xing, Hyejin Oh, Georges El Fakhri, Je-Won Kang, and Jonghye Woo. 2022. [Deep unsupervised domain adaptation: A review of recent advances and perspectives](#). *CoRR*, abs/2208.07422.

Junyu Luo, Cao Xiao, and Fenglong Ma. 2024. Zero-resource hallucination prevention for large language models. In *Findings of EMNLP*, pages 3586–3602.

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 *ACL*, pages 9802–9822.

Potsawee Manakul, Adian Liusie, and Mark J. F. Gales. 2023. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. In *EMNLP*, pages 9004–9017.

Meta. 2024. Introducing meta llama 3: The most capable openly available llm to date. <https://ai.meta.com/blog/meta-llama-3/>.

Michael B Miller, Christa-Lynn Donovan, Craig M Bennett, Elissa M Aminoff, and Richard E Mayer. 2012. Individual differences in cognitive style and strategy predict similarities in the patterns of brain activity between individuals. *Neuroimage*, 59(1):83–93.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettle-

moyer, and Hannaneh Hajishirzi. 2023a. [Factscore: Fine-grained atomic evaluation of factual precision in long form text generation](#). *CoRR*, abs/2305.14251.

Sewon Min, Kalpesh Krishna, Xinxin Lyu, Mike Lewis, Wen-tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023b. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. In *EMNLP*, pages 12076–12100.

Jose G. Moreno-Torres, Troy Raeder, Rocío Alaíz-Rodríguez, Nitesh V. Chawla, and Francisco Herrera. 2012. A unifying view on dataset shift in classification. *Pattern Recognit.*, 45(1):521–530.

OpenAI. 2023. [GPT-4 technical report](#). *CoRR*, abs/2303.08774.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A python natural language processing toolkit for many human languages. In *ACL (demo)*, pages 101–108.

Christopher Sciavolino, Zexuan Zhong, Jinhyuk Lee, and Danqi Chen. 2021. Simple entity-centric questions challenge dense retrievers. In *EMNLP*, pages 6138–6148.

Alexander J. Smola, Arthur Gretton, Le Song, and Bernhard Schölkopf. 2007. A hilbert space embedding for distributions. In *ALT*, volume 4754 of *Lecture Notes in Computer Science*, pages 13–31.

Weihang Su, Changyue Wang, Qingyao Ai, Yiran HU, Zhijing Wu, Yujia Zhou, and Yiqun Liu. 2024. [Unsupervised real-time hallucination detection based on the internal states of large language models](#). *Preprint*, arXiv:2403.06448.

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

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NeurIPS*, pages 5998–6008.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jian-hong Tu, Jianwei Zhang, Jianxin Ma, Jianxin Yang, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Xuejing Liu, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, Zhifang Guo, and Zhihao Fan. 2024a. [Qwen2 technical report](#). *CoRR*, abs/2407.10671.

Haoran Yang, Yumeng Zhang, Jiaqi Xu, Hongyuan Lu, Pheng-Ann Heng, and Wai Lam. 2024b. [Unveiling the generalization power of fine-tuned large language models](#). *CoRR*, abs/2403.09162.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *EMNLP*, pages 2369–2380.

Mert Yüksekgönül, Varun Chandrasekaran, Erik Jones, Suriya Gunasekar, Ranjita Naik, Hamid Palangi, Ece Kamar, and Besmira Nushi. 2024. Attention satisfies: A constraint-satisfaction lens on factual errors of language models. In *ICLR*.

Shaolei Zhang, Tian Yu, and Yang Feng. 2024. Truthx: Alleviating hallucinations by editing large language models in truthful space. In *ACL*, pages 8908–8949.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, et al. 2023. [Siren’s song in the ai ocean: A survey on hallucination in large language models](#). *CoRR*, abs/2309.01219.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, Shashwat Goel, Nathaniel Li, Michael J. Byun, Zifan Wang, Alex Mallen, Steven Basart, Sanmi Koyejo, Dawn Song, Matt Fredrikson, J. Zico Kolter, and Dan Hendrycks. 2023. [Representation engineering: A top-down approach to AI transparency](#). *CoRR*, abs/2310.01405.

## A Statistics of QA Datasets

During the NFP dataset construction, to reduce the false negative samples, we exclude multiple-choice questions because their LLM-generated responses are likely to mention both correct and incorrect answers. Moreover, we exclude questions where the golden answers are three characters or fewer,

	PQ		EQ		NQ	
	Pos	Neg (factual)	Pos	Neg (factual)	Pos	Neg (factual)
LLaMA2-7B-Chat	74.9	25.1	70.3	29.7	57.2	42.8
LLaMA3-8B-Instruct	65.5	34.5	61.6	38.4	48.2	51.8
Mistral-7B-Instruct-v0.2	73.0	27.0	68.2	31.8	55.5	44.5
Qwen2-1.5B-Instruct	86.2	13.8	80.1	19.9	75.9	24.1

Table 3: Positive and negative sample ratios in different NFP datasets (%). A NFP dataset is built based on an LLM and a QA dataset. A positive (non-factual) sample indicates the LLM  $m$  cannot provide the queried facts in response to the fact-seeking question  $q$ , whereas a negative (factual) sample indicates the LLM  $m$  can provide the queried facts in response to the fact-seeking question  $q$ .

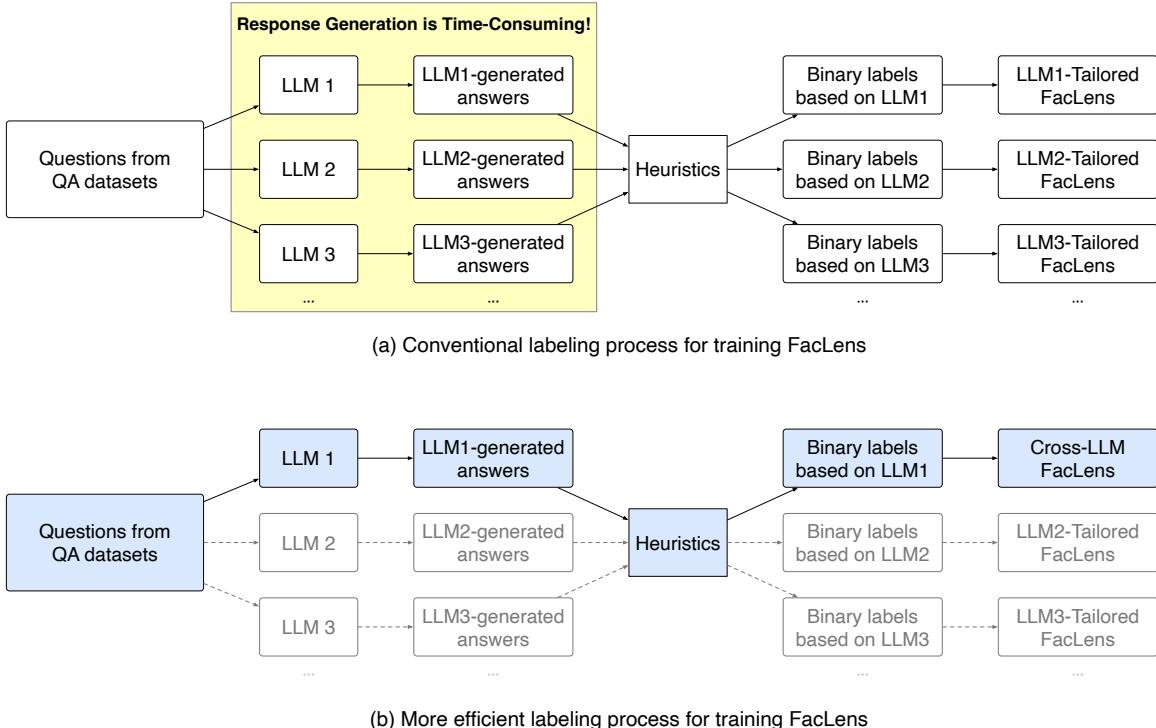


Figure 8: Illustration of different labeling processes for training FacLens.

as such short strings are likely to appear as substrings within unrelated words. PQ initially contains 14,267 questions. After eliminating duplicates and removing the above special questions, 11,360 unique questions remained. EQ contains 100K questions. We randomly sample 7,200 questions from EQ, ensuring uniform coverage across all question topics. After eliminating duplicates and removing special questions, we retain 7,159 questions from EQ. The full NQ dataset is huge (about 42GB), so we download a simplified development set of NQ. In this paper, we focus on the case of short answers, so we select questions whose answers are comprised of 30 characters or less. Hence, we include 1,244 questions sourced from NQ. Table 3 shows the ratios of positive and negative samples in each NFP dataset, where a pair

of QA dataset and LLM corresponds to an NFP dataset.

## B More Efficient Process of Training Data Construction

In Section 4.2, we explained that the transferability of FacLens can reduce overall development costs of FacLens for multiple LLMs by lowering the costs of obtaining labels for training. Figure 8 provides the illustration, where the gray dashed lines indicate that the corresponding steps are omitted.

## C Experimental Settings

### C.1 Perplexity (PPL) on a Question

We regard the PPL of a fact-seeking question as a baseline. In specific, we predict  $y = 1$  if the PPL value exceeds a certain threshold. We extend the

calculation of PPL to be conducted in each layer to obtain multiple PPL values for a text and determine the layer based on the NFP performance on labeled data. Formally, PPL on a question calculated in the  $\ell$ -th layer is formulated as,

$$PPL = \exp \left( \frac{1}{|q|} \sum_{v_k \in q} -\log(p_\ell(v_k | v_{<k})) \right) \quad (7)$$

$$p_\ell(v_k | v_{<k}) = \text{Softmax}(m_{\leq \ell}(v_{<k}) W_U)_{v_k} \quad (8)$$

where  $q$  is a fact-seeking question,  $v_k$  is the  $k$ -th token in  $q$ ,  $v_{<k}$  represents the set of tokens preceding the  $k$ -th token, and  $W_U$  is the pre-trained unembedding matrix of the LLM  $m$  that converts the hidden token representations into distributions over the vocabulary.

## C.2 Hyper-Parameter Settings

In terms of baselines, we adopt hyper-parameter settings recommended by their authors. Since we extend PPL to be calculated in each hidden layer, we determine the specific layer according to PPL’s performance on the labeled data. We introduce the Prompting-based method, which encourages an LLM to answer whether it knows the factual responses via prompt “Question: {question}\nCan you provide a factual response to the above question? If you can, please reply yes or Yes. If you can not, please reply no or No.\nAnswer: {label}\n”. The probabilities of predicting tokens “yes”, “Yes”, “no” and “No” are normalized for prediction.

For the Self-Evaluation (Fully FT), we train the model on 4\*80G A800 GPUs, with a learning rate of 1e-6, batch size of 32, and epochs of 12, and we also determine the training epochs based on the performance on the validation set. Self-Evaluation (Fully FT) needs to fully fine-tune an LLM. Therefore, to mitigate overfitting, the learning rate scheduler employs a cosine decay strategy with 5% of the training steps dedicated to linear warm-up. Additionally, the final learning rate is set to one-tenth of its initial value. For LoRA, we integrate adapters on all “q\_proj”, “k\_proj”, “v\_proj”, and “o\_proj” layers, while maintaining the original weights of the language model unchanged. The configuration is as follows: we specify a rank of 128 and an alpha of 256, with a learning rate of 1e-4, a batch size of 32, and the training is conducted over 32 epochs. We employ the same learning

rate scheduler as used in Fully SFT. Because LoRA is a parameter-efficient fine-tuning technique, the training process requires only a single 80G A800 GPU. Note that we chose hyperparameters ( $r=128$  and  $\alpha=256$ ), which are larger than those used in the original LoRA paper, to introduce more trainable parameters, thereby enhancing the modeling capacity of the LoRA adapters. Although these values are larger than those used in the original LoRA paper, the additional trainable parameters remain significantly smaller than those of the original LLM. For instance, with LLaMA2, LoRA adapters with  $r=128$  and  $\alpha=256$  introduce only 1.95% trainable parameters.

## C.3 Why Certain Baselines Are Unsuitable for the QA Datasets

In Table 1, Entity-Popularity and SAT Probe are not suitable for certain datasets. Here, we explain the reasons. Entity-Popularity uses Wikipedia page views to approximate the entity’s popularity. However, EQ and NQ datasets do not provide relevant Wikipedia page views, and not every subject entity in the two datasets can be uniquely matched to a Wikidata entity. As a result, Entity-Popularity is unsuitable for EQ and NQ. For the baseline SAT Probe, each question is assumed to contain constraint tokens, and the model extracts LLMs’ attention to the constraint tokens to probe factuality. The authors of SAT Probe have restricted the formats of questions to directly identify the constraint tokens. However, extracting constraint tokens from free-form questions can be challenging. For PQ and EQ, which are template-based, extracting constraint tokens is relatively straightforward. However, SAT Probe is not suitable for NQ, as questions in NQ come from real users and exhibit diverse structures.

The core of the SAT Probe is using an LLM’s attention weights to constraint tokens within a question to reflect the LLM’s factual accuracy. In the original paper, the SAT probe is implemented by a linear layer, optimized by logistic regression. To compare the effectiveness of hidden representations and attention weights, we employ the same MLP structure and CE loss for both SAT Probe and FacLens.

## D Supplementary Experiments

### D.1 Ante-Hoc NFP vs. Post-Hoc NFD

In Figure 9, we supplement the performance comparison between hidden representation-based ante-hoc and post-hoc methods on EQ and NQ. The

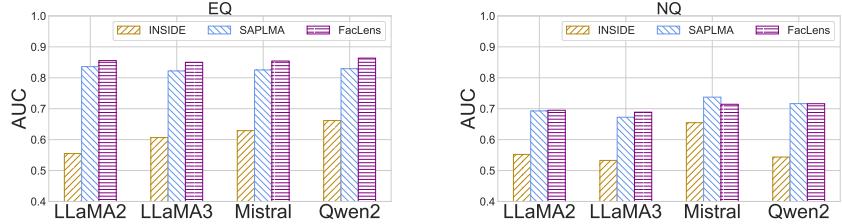


Figure 9: Performance comparison of FacLens (ante-hoc NFP) with INSIDE and SAPLMA (post-hoc NFD) on EQ and NQ.

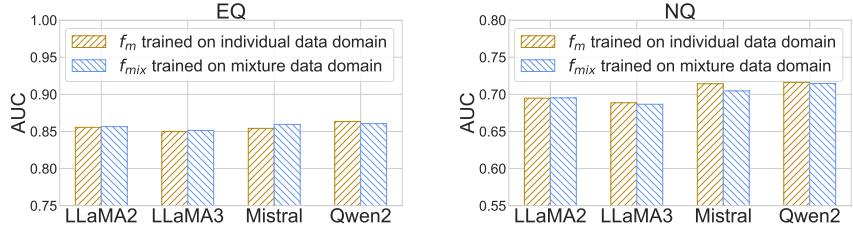


Figure 10: Performance comparison between  $f_m$  and  $f_{mix}$  on EQ and NQ. Similar performance suggests no significant concept shift across different domains.

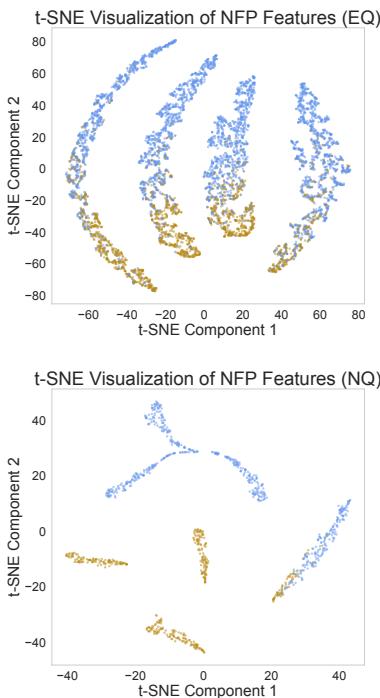


Figure 11: Visualizations of NFP features.

results further highlight that FacLens (our ante-hoc NFP method) has the potential to outperform the post-hoc NFD methods.

## D.2 Demonstrations of FacLens’s Transferability

In Section 4.2, we compare FacLens  $f_m$ , trained on an individual domain, with FacLens  $f_{mix}$ , trained

on the mixture domain. The results show that  $f_{mix}$  exhibits comparable performance to  $f_m$  on the test set of the corresponding individual domain. We supplement the results based on questions from EQ and NQ in Figure 10, further demonstrating no significant concept shifts between domains.

Besides, we supplement more visualizations of the NFP features in Figure 11, further demonstrating that a unified classification boundary can be applied to the NFP task for different LLMs.

## D.3 Performance of FacLens (avg)

We use the averaged hidden representation of all tokens in a question as input to FacLens, denoted as FacLens (avg). Comparing the results in Table 1 and Table 4, we observe that FacLens (last token) performs more stably. Therefore, we recommend using the hidden representation of the last token in a question as the hidden question representation.

## D.4 Evaluation on HotpotQA

Table 5 shows the prediction performance of different NFP methods on HotpotQA. The observations are consistent with that in Table 1, indicating FacLens can also handle the more complex fact-seeking questions.

## D.5 Cross-LLM FacLens for LLMs of Distinct Hidden Dimensions

Both Qwen2-1.5B-Instruct (Yang et al., 2024a) and Gemma-7B-it (Banks and Warkentin, 2024) have

	LLaMA2			LLaMA3			Mistral			Qwen2		
	PQ	EQ	NQ	PQ	EQ	NQ	PQ	EQ	NQ	PQ	EQ	NQ
FacLens (avg, last layer)	87.9	84.8	63.8	84.2	82.4	60.6	86.9	85.3	63.3	90.1	84.8	70.6
FacLens (avg, 2 <sup>nd</sup> to last layer)	87.5	85.1	59.9	84.5	83.0	54.9	87.4	85.8	64.9	89.6	84.4	70.7
FacLens (avg, middle layer)	88.5	85.9	66.0	85.5	84.8	62.8	87.5	84.7	67.6	89.0	86.2	70.8

Table 4: Prediction performance of FacLens (avg) (AUC %).

	LLaMA2	LLaMA3	Mistral	Qwen2
PPL	55.2	55.2	54.4	53.5
Prompting	62.5	61.0	63.0	62.1
Entity-Popularity	—	—	—	—
SAT Probe	—	—	—	—
Self-Familiarity	55.3	56.8	54.5	53.7
LoRA (Parameter-Efficient FT)	72.9	68.0	66.9	70.8
Self-Evaluation (Fully FT)	75.0	69.3	71.1	72.1
FacLens (last token, last layer)	74.3	<b>68.6</b>	74.1	72.1
FacLens (last token, 2 <sup>nd</sup> to last layer)	74.7	68.3	74.7	<b>72.7</b>
FacLens (last token, middle layer)	<b>75.5</b>	66.9	<b>74.9</b>	71.4

Table 5: Prediction performance of different NFP methods on HotpotQA (AUC %).

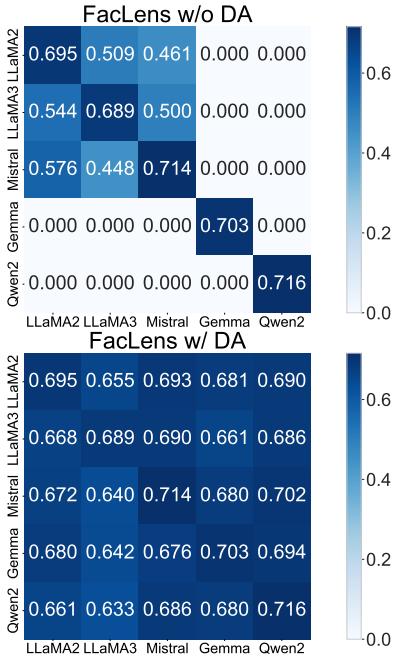


Figure 12: Evaluation of cross-LLM FacLens on LLMs of different hidden dimensions. Questions are from NQ.

different hidden dimensions compared to LLaMA2-7B-Chat, LLaMA3-8B-Instruct, and Mistral-7B-Instruct-v0.2. The hidden dimension of Qwen2-1.5B-Instruct is 1536, and the hidden dimension of Gemma-7B-it is 3072, while the hidden dimension of LLaMA2-7B-Chat, LLaMA3-8B-Instruct, and Mistral-7B-Instruct-v0.2 is 4096. A FacLens specially trained for the source-domain LLM cannot be directly used for a target-domain LLM whose

hidden dimension is distinct from that of the source-domain LLM. Hence we introduce a linear layer to reshape the target-domain hidden question representations to match the dimension of the source domain’s hidden question representations, and still adopt Eq. 6 to conduct domain adaptation. In Figure 12, we can see that although two LLMs have different hidden dimensions, the cross-LLM FacLens can work well.

## D.6 Kernel Selection for MMD Loss

In the MMD loss, the data features are mapped into a reproducing kernel Hilbert space (RKHS) determined by a kernel function. Then distribution distance between different data domains is measured within the RKHS. We minimize the MMD loss to find a domain-invariant NFP feature space. Here, we evaluate two commonly employed kernel functions: the linear kernel and the Gaussian kernel (Gretton et al., 2012). As depicted in Figure 13, the linear kernel tends to perform better. This suggests that the features extracted by  $g_{enc}$  for NFP tasks are already inherently discriminative.

## E Demo & Case Study

**Human Evaluation via Demo.** We have implemented a demo of FacLens. In the demo, a user can choose a specific LLM, and then enter a fact-seeking question in the text box. After submitting the question, FacLens will return whether the LLM knows the factual answer. Then the user can decide

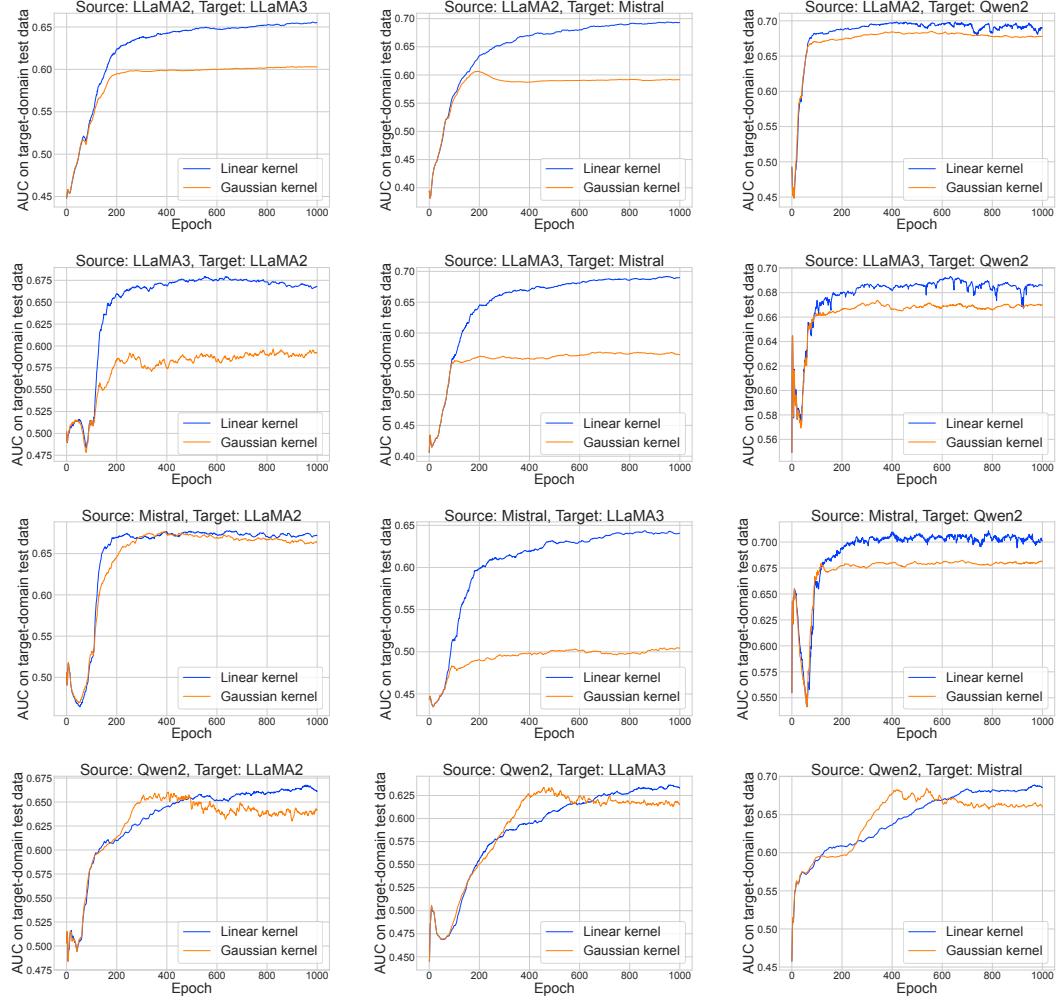


Figure 13: Evaluation of kernels used by MMD loss for training cross-LLM FacLens. The results are derived from the NQ dataset. Similar trends are observed on other QA datasets.

whether to call the LLM to generate the response. If the user decides to query the LLM, the demo will provide the response generated by the LLM. According to the prediction of FacLens and the LLM’s response, the user can score the performance of FacLens. We recruited 22 volunteers, consisting of 11 females and 11 males with bachelor’s degrees or higher, to use our demo and rate its performance. 3 points indicate that the prediction of FacLens is correct, 2 points indicate that FacLens acknowledges its lack of confidence in the prediction result, and 1 point indicates that the prediction of FacLens is incorrect. We received 680 de-duplicated user queries, with 127 (18.7%) receiving 1 point, 70 (10.3%) receiving 2 points, and 483 (71.0%) receiving 3 points.

**FacLens in the Demo.** Taking LLaMA2-7B-Chat (abbreviated as LLaMA2) as the example, we integrate its NFP datasets, i.e., LLaMA2-

PQ, LLaMA2-EQ, and LLaMA2-NQ, to train a FacLens. Specifically, we use instances from LLaMA2-PQ, and LLaMA2-EQ for training, and instances from LLaMA2-NQ for validation. That is because NQ, released by Google, consists of questions posed by real users. We set the learning rate to 1e-3, and determine the training epochs based on the performance on the validation set. On the validation set, we use FacLens to predict the probability  $p(y=1|x)$  for each instance. Then we calculate the averaged probability  $\bar{p}(y=1|x)_{pos}$  based on positive instances in the validation set, as well as the averaged probability  $\bar{p}(y=1|x)_{neg}$  based on negative instances in the validation set.  $\bar{p}(y=1|x)_{pos}$  and  $\bar{p}(y=1|x)_{neg}$  are used as the thresholds for predicting whether the LLM knows the factual answers. Clearly, given a question, FacLens predicts the probability of LLaMA2 generating a non-factual response. If the probabil-

ity is larger than  $\bar{\mathbf{p}}(y = 1|\mathbf{x})_{pos}$ , the demo outputs “The LLM does not know the factual answer”; If the probability is smaller than  $\bar{\mathbf{p}}(y = 1|\mathbf{x})_{neg}$ , the demo outputs “The LLM knows the factual answer”; otherwise, the demo outputs “I am not sure if the LLM knows the factual answer”. Similarly, we train FacLens for LLaMA3-8B-Instruct, Mistral-7B-Instruct-v0.2, and Qwen2-1.5B-Instruct, respectively.

**Case Study and Future Work.** Taking LLaMA2-7B-Chat as an example, we show some good cases at the end of this paper, based on questions from our human evaluation. These questions cover a range of topics, such as movies, TV series, music, brands, games, and science. We observe that FacLens works well on most questions, but it is more likely to mistakenly predict on questions related to periodic events. The recurrence of similar events could impact FacLens’s judgments. In other words, LLMs could have limited self-awareness in recognizing the nuances of periodic events. We plan to enhance FacLens’s ability to better handle such questions in future work.

## Good Case

### User Question:

Who was the screenwriter for The Toy?

### Prediction from FacLens:

The LLM does not know the factual answer.

### Main Response from LLM:

The screenwriter for the 1982 film “The Toy” is:

Tom Mankiewicz

### Note:

The LLM’s response is incorrect. The screenwriter is not Tom Mankiewicz.

## Good Case

### User Question:

Who is the director of TV series The Happy Family?

### Prediction from FacLens:

The LLM does not know the factual answer.

### Main Response from LLM:

I apologize, but I don’t have access to information on the director of a specific TV series called “The Happy Family.” Could you please provide more context or details about the series, such as its release date, network, or any other relevant information? This will help me provide a more accurate answer.

### Note:

The LLM acknowledges that it does not know the factual answer.

## Good Case

### User Question:

Who wrote the lyrics for the song I Can Do It With A Broken Heart?

### Prediction from FacLens:

The LLM does not know the factual answer.

### Main Response from LLM:

I apologize, but I don’t have access to the lyrics of the song “I Can Do It With A Broken Heart” as it may be a private or unreleased song. Additionally, I cannot provide lyrics for songs that are not publicly available or have not been released.

### Note:

The LLM acknowledges that it does not know the factual answer.

## Good Case

### User Question:

Which TV series does the character Sheldon Lee Cooper come from?

### Prediction from FacLens:

The LLM knows the factual answer.

### Main Response from LLM:

The character Sheldon Lee Cooper is from the popular TV series “The Big Bang Theory.”

### Note:

The LLM’s response is correct.

### Good Case

#### User Question:

Who founded KFC?

#### Prediction from Faclens:

The LLM knows the factual answer.

#### Main Response from LLM:

KFC, or Kentucky Fried Chicken, was founded by Colonel Harland Sanders.

#### Note:

The LLM's response is correct.

### Good Case

#### User Question:

Which is heavier, 1kg iron or 1kg cotton?

#### Prediction from Faclens:

The LLM knows the factual answer.

#### Main Response from LLM:

Both 1 kg of iron and 1 kg of cotton weigh the same amount, 1 kilogram.

#### Note:

The LLM's response is correct.

### Bad Case

#### User Question:

Which team won the women's volleyball championship at the 2004 Athens Olympics?

#### Prediction from Faclens:

The LLM knows the factual answer.

#### Main Response from LLM:

The United States women's volleyball team won the gold medal at the 2004 Athens Olympics.

#### Note:

The LLM's response is incorrect. The correct answer is the Chinese women's volleyball team.

### Bad Case

#### User Question:

Who won the Turing Award in 2016?

#### Prediction from Faclens:

The LLM knows the factual answer.

#### Main Response from LLM:

The Turing Award for 2016 was awarded to Yoshua Bengio, Geoffrey Hinton, and Yann LeCun for their contributions to the development of deep learning.

#### Note:

The LLM's response is incorrect. The correct answer is Tim Berners-Lee.

### Bad Case

#### User Question:

Where was the 2006 World Expo held?

#### Prediction from FacLens:

The LLM knows the factual answer.

#### Main Response from LLM:

The 2006 World Expo was held in Aichi, Japan, from March 25 to September 25, 2005.

#### Note:

The LLM's response is incorrect. The correct answer is Chiang Mai, Thailand.

### Bad Case

#### User Question:

Who won the Nobel Prize in Physics in 1903?

#### Prediction from FacLens:

The LLM knows the factual answer.

#### Main Response from LLM:

The Nobel Prize in Physics was not awarded in 1903. The Nobel Prize in Physics was first awarded in 1901, and the first Nobel Prize ceremony was held in 1904. Therefore, there was no Nobel Prize in Physics awarded in 1903.

#### Note:

The LLM's response is incorrect. The correct answer is Henri Becquerel, Pierre Curie, and Marie Curie.