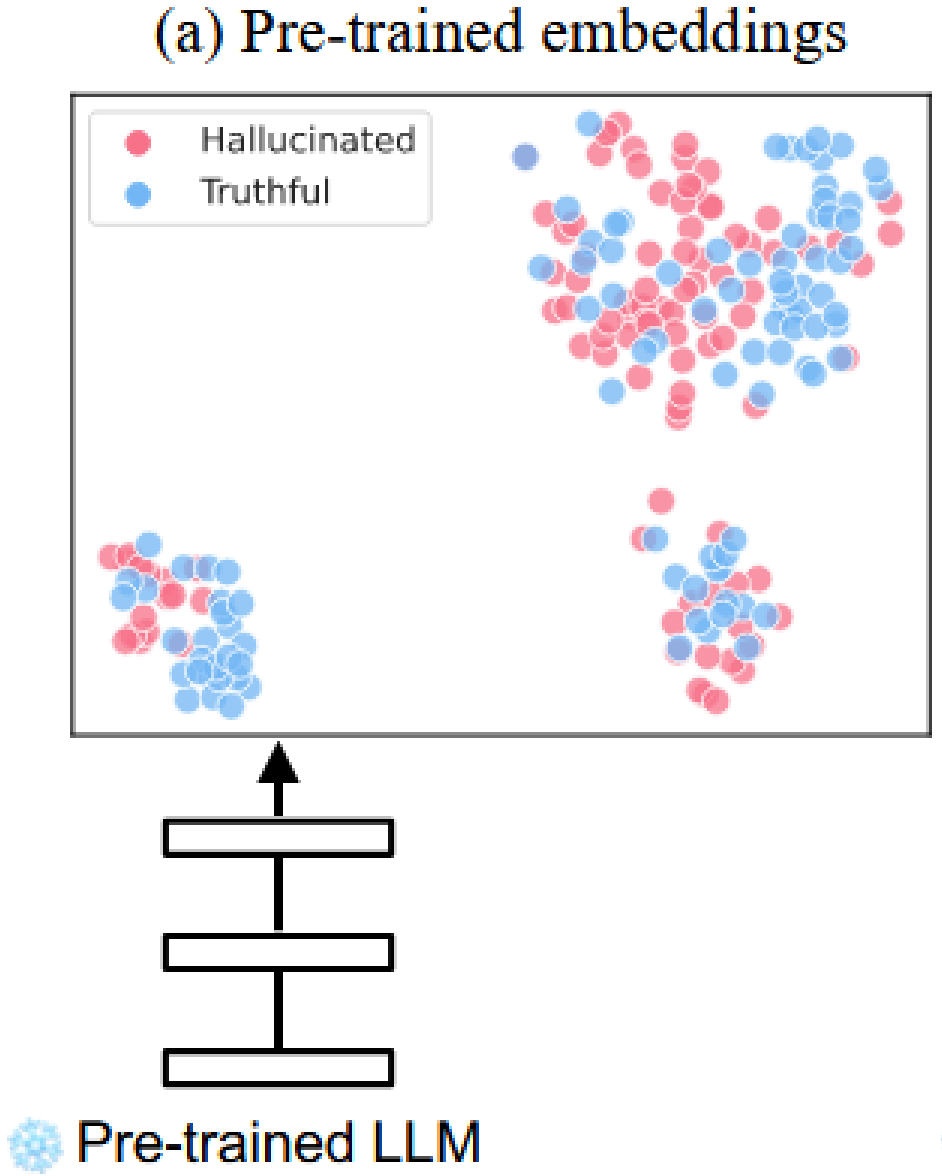
Steer LLM Latents for Hallucination Detection

Seongheon Park [1] **Xuefeng Du** [1] **Min-Hsuan Yeh** [1] **Haobo Wang** [2] **Yixuan Li** [1]

[1]Department of Computer Sciences, University of Wisconsin-Madison [2]School of Software Technology, Zhejiang University. Contact: Seongheon Park <seongheon_park@cs.wisc.edu>. Correspondence to: Yixuan Li <sharonli@cs.wisc.edu>.

引言和动机

- 大型语言模型 (LLM) 虽然能力强大，然而，“幻觉”问题是其安全部署和应用的主要障碍。
- “幻觉”检测的传统方法：依赖现有的大模型的嵌入层表示，对事实和幻觉进行分类。
  - 局限性：
    - 当前预训练大模型是自回归的，更关注输出的连贯性和语法的正确性，而不是事实的准确性。
    - 因此，可能大模型的内在表征不能提供一个关于事实和幻觉内容的清晰分界。



引言和动机

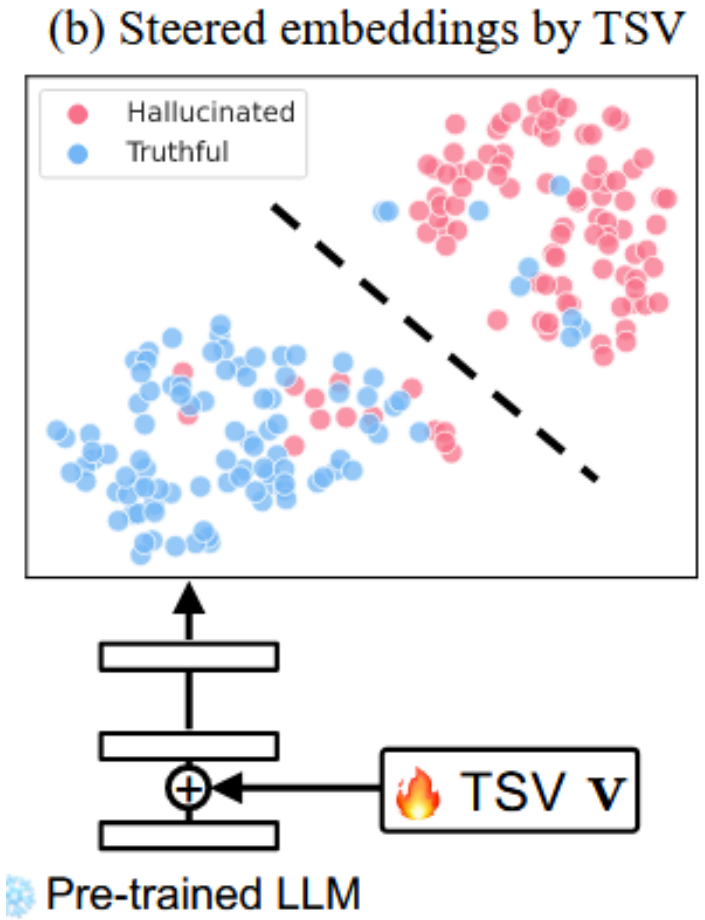
How can we shape the latent space of an LLM for hallucination detection?

微调（Fine-tuning）的缺陷：

- 计算成本高昂
- 会改变模型原有参数，可能影响其通用能力

本文提出一种高效、轻量级且不改变模型参数的幻觉检测方法，Truthfulness Separator Vector（TSV）。

- TSV是一个**可学习**、**轻量级**的引导向量
- 核心机制：“引导”潜空间。
  - 在推理时将TSV注入LLM的中间层激活。
  - 重塑LLM的表示空间，以增强真实输出和幻觉输出之间的可分离性。
  - 无需修改LLM的原始参数。



$$\mathbf{h}^{(l)} \leftarrow \mathbf{h}^{(l)} + \lambda \mathbf{v},$$

TSV如何训练？

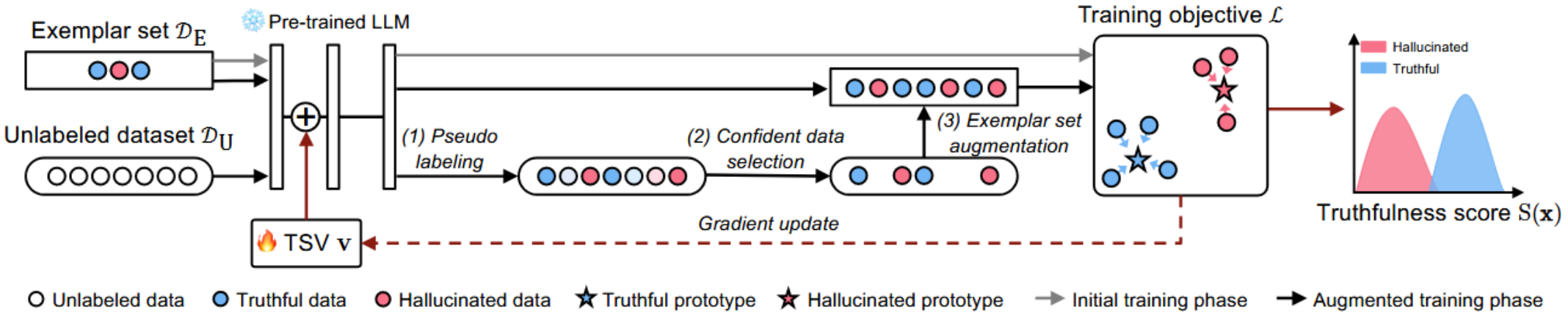


Figure 2. Overall framework. In the initial training phase, Truthfulness Separator Vector (TSV) is trained on an exemplar set. After initial training, (1) we assign soft pseudo-labels to the unlabeled data, (2) select confident pseudo-labeled samples, and (3) augment the exemplar set with selected samples. Finally, we retrain TSV with the augmented set. Best viewed in color.

面临的挑战：大规模、高质量的幻觉标注数据稀缺且昂贵

解决方案：两阶段训练框架

1. 第一阶段：基于少量“范例集”的初始训练 (有监督)
2. 第二阶段：利用无标签数据进行增强训练 (半监督思想)

第一阶段：初始训练 (利用少量有标签范例)

- 输入：一个小的“范例集”(D_E)，包含少量已标注的真实/幻觉样本(论文中**TruthfulQA**仅用N=32个样本)。
- 操作：
  - 将TSV向量v添加到LLM的中间层l的潜状态 $h^l \leftarrow h^l + \lambda v$。
  - λ是控制干预强度的超参数。

- 训练目标：
  - 通过最大似然估计 (MLE)，使得引导后的嵌入在各自类别（真实/幻觉）内形成紧凑的簇，并使不同类别间的簇尽可能分离。

第一阶段：初始训练 (利用少量有标签范例)

- 目标函数：

$$\arg\max_{\mathbf{v}} \prod_{i=1}^{|\mathcal{D}_{\mathrm{E}}|} p(c_i \mid \Phi_{\text{final}}(\mathbf{h}_i^{(l)} + \lambda \mathbf{v})),$$

- The last-token embedding at the final layer after applying TSV

$$\Phi_{\text{final}}(\mathbf{h}^{(l)} + \lambda \mathbf{v}) = \phi_L \circ \phi_{L-1} \ldots \circ \phi_{l+1}(\mathbf{h}^{(l)} + \lambda \mathbf{v}),$$

- The class conditional probability is given by,

$$p(c \mid \mathbf{r}^{\mathbf{v}}) = \frac{\exp\left(\kappa \boldsymbol{\mu}_c^\top \mathbf{r}^{\mathbf{v}}\right)}{\sum_{c'} \exp\left(\kappa \boldsymbol{\mu}_{c'}^\top \mathbf{r}^{\mathbf{v}}\right)}, \quad \text{where } \mathbf{r}^{\mathbf{v}} = \Phi_{\text{final}}(\mathbf{h}^{(l)} + \lambda \mathbf{v}) / \|\Phi_{\text{final}}(\mathbf{h}^{(l)} + \lambda \mathbf{v})\|_2$$

- 类别原型 ($\boldsymbol{\mu}_c$) 更新： 使用指数移动平均 (EMA) 更新代表真实和幻觉的类别中心。

第二阶段：增强训练 (利用无标签数据)

- **动机：** 充分利用大量易于获取的无标签LLM生成内容，以弥补标注数据的不足，提升模型泛化能力。
- **核心步骤：**

1. 伪标签分配 (Pseudo-labeling)：
   - 针对无标签数据集 (D_U)。
   - 采用基于最优传输 (Optimal Transport, OT)的算法为无标签样本分配“真实”或“幻觉”的伪标签。

2. 高置信度数据选择 (Confident Data Selection)：
   - 并非所有伪标签都是可靠的。
   - 通过计算经过第一阶段训练得到的模型的预测与所分配伪标签之间的不确定性 (如交叉熵)，筛选出置信度最高的K个伪标签样本 (D_S)。

3. 范例集增强与再训练：
   - 将筛选出的高置信度伪标签样本D_S加入到初始的范例集D_E中。
   - 使用增强后的数据集对TSV进行重新训练。

推理时幻觉检测

- **过程：**
  - 输入一个待检测的LLM生成内容。
  - 同样将训练好的TSV向量v应用于LLM的中间层。
  - 提取经过TSV引导后的最终层Token嵌入 r_{test}^{v}。
  - 计算该嵌入与已学习到的“真实”类别原型 $\mu_{truthful}$和“幻觉”类别原型 $\mu_{hallucinated}$的相似度。
- **真实性得分 $S(x')$：**
  - 基于与“真实”类别原型的相似度（归一化概率）来定义。

$$S(\mathbf{x}') = \frac{\exp\left(\kappa \boldsymbol{\mu}_{\text{truthful}}^{\top} \mathbf{r}_{\text{test}}^{\mathbf{v}}\right)}{\sum_{c'} \exp\left(\kappa \boldsymbol{\mu}_{c'}^{\top} \mathbf{r}_{\text{test}}^{\mathbf{v}}\right)}.$$

推理时幻觉检测

- **真实性得分 $S(x')$：**
  - 基于与“真实”类别原型的相似度（归一化概率）来定义。

$$S(\mathbf{x}') = \frac{\exp\left(\kappa \boldsymbol{\mu}_{\text{truthful}}^{\top} \mathbf{r}_{\text{test}}^{\mathbf{v}}\right)}{\sum_{c'} \exp\left(\kappa \boldsymbol{\mu}_{c'}^{\top} \mathbf{r}_{\text{test}}^{\mathbf{v}}\right)}.$$

- **分类：** 根据真实性得分是否高于某个阈值ζ来判断输出是真实的还是幻觉。

$$G_{\zeta}(\mathbf{x}_{\text{test}}) = \mathbb{1}\{S(\mathbf{x}_{\text{test}}) \geq \zeta\},$$

- **灵活性：** 移除TSV向量，LLM即可恢复其原始生成能力。

实验设置

- **数据集：**
  - TruthfulQA， TriviaQA， NQ Open， SciQ
- **模型：**
  - LLaMA-3.1-8B & 70B， Qwen-2.5-7B & 14B
- **基线方法：**
  - Logit-based, Consistency-based, Verbalized, Internal state-based (CCS, HaloScope, SAPLMA)
- **评估指标：** AUROC (Area Under the Receiver Operating Characteristic curve)

Evaluation

1. 将模型的生成输出视为真实的条件是：该输出与参考答案之间的相似度分数大于一个预定义的阈值
   - 本文使用BLEURT度量相似度，当其大于0.5时，认为答案为真

2. 基于大语言模型作为裁判，本文中额外用GPT-4o作为裁判来判断模型生成的答案是否真实。让其评判模型生成回答和给定的标准答案在语义上是否等价。

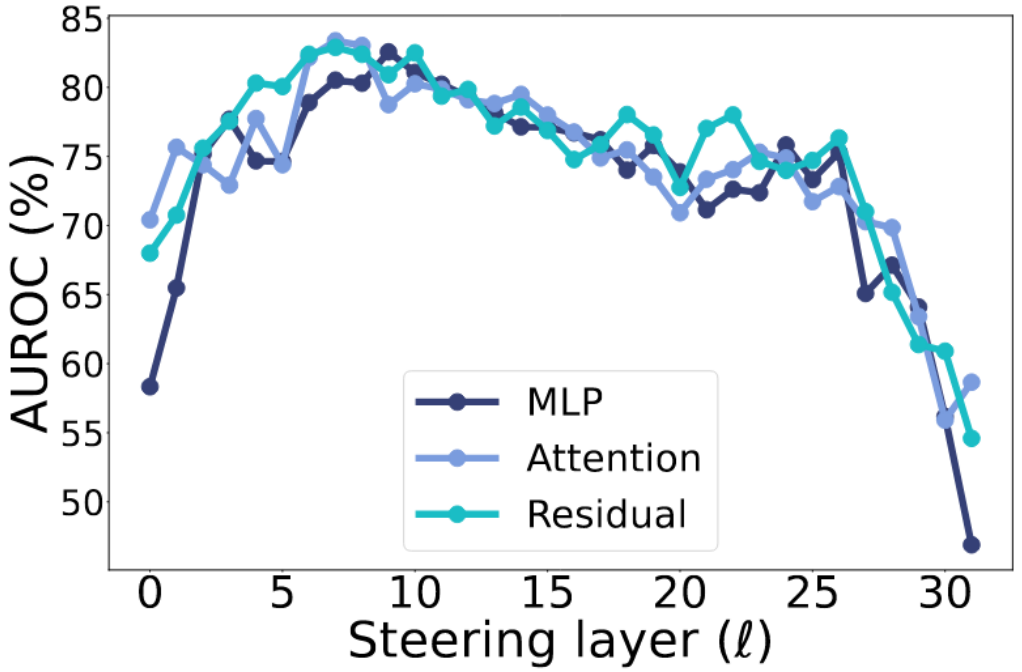
主要实验结果 (1) - 性能对比

Table 1. Main results. Comparison with competitive hallucination detection methods on different datasets. "Single sampling" indicates whether the approach requires multiple generations during inference. For our method, the mean and standard deviation are computed across three different random seeds. ♣ denotes methods trained on fully labeled datasets. All values are percentages (AUROC), and the best results are highlighted in **bold**.

Model	Method	Single Sampling	TruthfulQA	TriviaQA	SciQ	NQ Open
LLaMA-3.1-8b	Perplexity	✓	71.4	76.3	52.6	50.3
	LN-Entropy	✗	62.5	55.8	57.6	52.7
	Semantic Entropy	✗	59.4	68.7	68.2	60.7
	Lexical Similarity	✗	49.1	71.0	61.0	60.9
	EigenScore	✗	45.3	69.1	59.6	56.7
	SelfCKGPT	✗	57.0	80.2	67.9	60.0
	Verbalize	✓	50.4	51.1	53.4	50.7
	Self-evaluation	✓	67.8	50.9	54.6	52.2
	CCS	✓	66.4	60.1	77.1	62.6
	HaloScope	✓	70.6	76.2	76.1	62.7
	SAPLMA♣	✓	78.2	83.7	77.3	62.8
	TSV (Ours)	✓	**84.2**$^{\pm 0.2}$	**84.0**$^{\pm 0.5}$	**85.8**$^{\pm 0.4}$	**76.1**$^{\pm 0.7}$
	TSV♣ (Ours)	✓	**85.5**$^{\pm 0.1}$	**87.2**$^{\pm 0.2}$	**88.6**$^{\pm 0.1}$	**78.0**$^{\pm 0.2}$
Qwen-2.5-7b	Perplexity	✓	65.1	50.2	53.4	51.2
	LN-Entropy	✗	66.7	51.1	52.4	54.3
	Semantic Entropy	✗	66.1	58.7	65.9	65.3
	Lexical Similarity	✗	49.0	63.1	62.2	61.2
	EigenScore	✗	53.7	61.3	63.2	57.4
	SelfCKGPT	✗	61.7	62.3	58.6	63.4
	Verbalize	✓	60.0	54.3	51.2	51.2
	Self-evaluation	✓	73.7	50.9	53.8	52.4
	CCS	✓	67.9	53.0	51.9	51.2
	HaloScope	✓	81.3	73.4	76.6	65.7
	SAPLMA♣	✓	81.7	**82.0**	81.5	67.9
	TSV (Ours)	✓	**87.3**$^{\pm 0.4}$	79.8$^{\pm 0.9}$	**82.0**$^{\pm 0.4}$	**73.8**$^{\pm 0.7}$
	TSV♣ (Ours)	✓	**88.7**$^{\pm 0.1}$	**84.2**$^{\pm 0.5}$	**84.8**$^{\pm 0.3}$	**76.2**$^{\pm 0.3}$

主要实验结果 (2) - 消融研究与分析

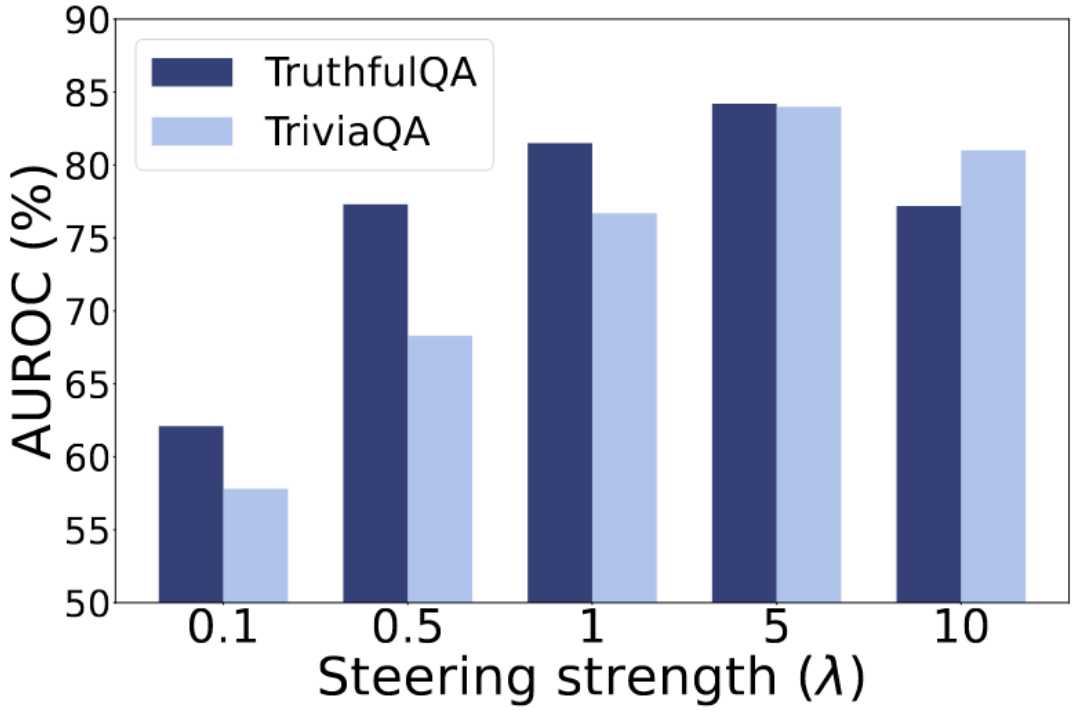
- TSV引导位置的影响:



(a) Effect of the steering location

主要实验结果 (2) - 消融研究与分析

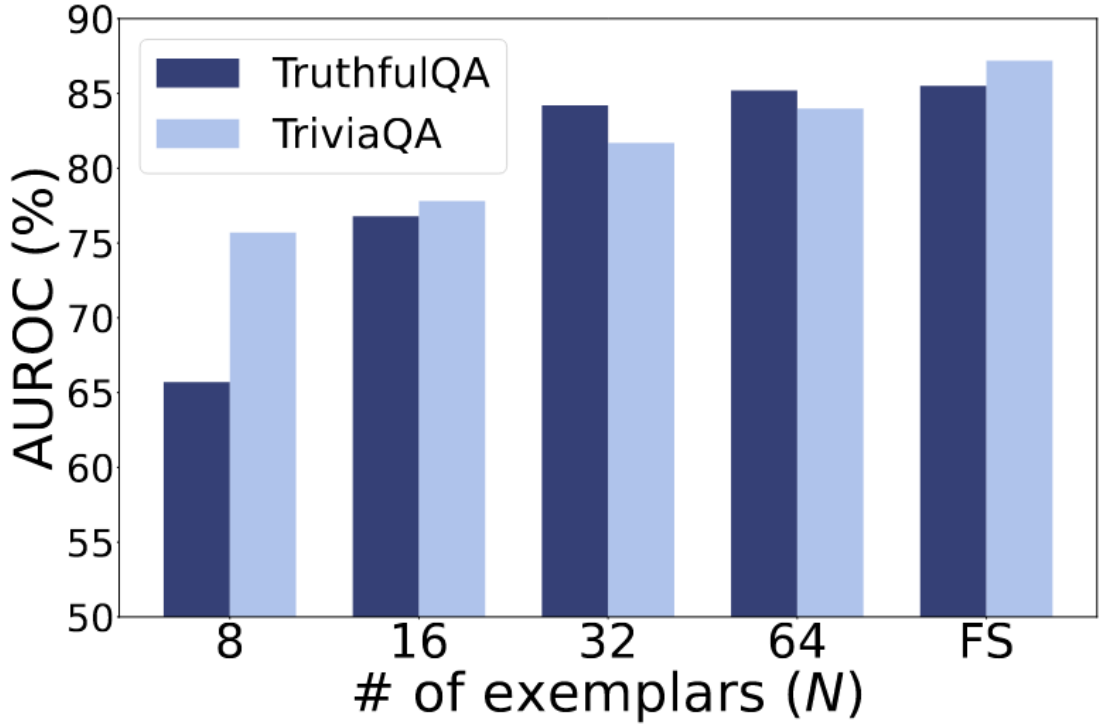
- 引导强度（Steering strength）的影响：



(b) Effect of the steering strength (λ)

主要实验结果 (2) - 消融研究与分析

- 范例集的大小（N）的影响:



(c) Effect of the number of exemplars (N)

主要实验结果 (2) - 消融研究与分析

泛化能力：

- 在一个数据集上训练的TSV，能较好地泛化到其他未见过的数据集上，表现出良好的迁移性。

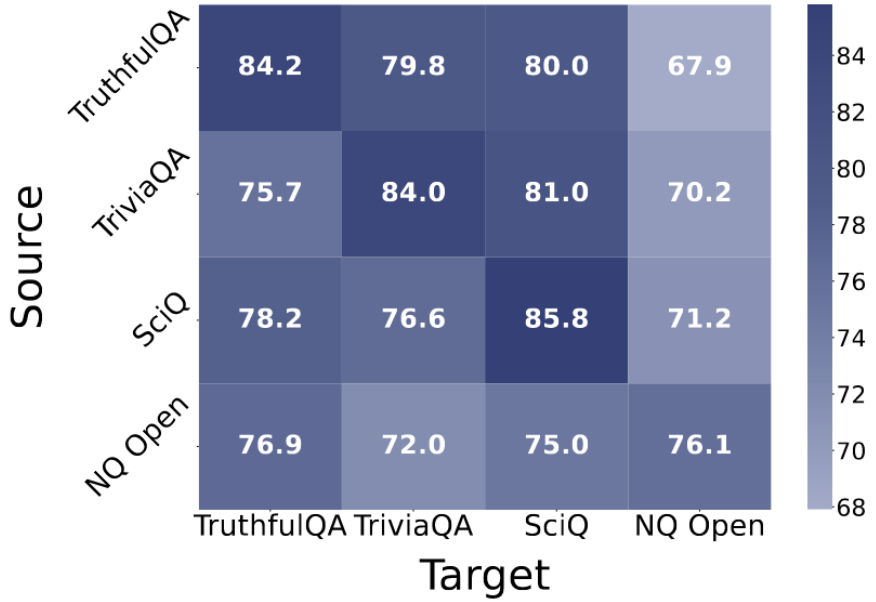


Figure 4. Generalization results on out-of-distribution datasets.

主要实验结果 (2) - 消融研究与分析

组件分析（Component analysis）：

- 每个部分（TSV、IT、AT）对于最终的效果都是有影响的，其中TSV本身的影响是最大的

Table 3. Component analysis. **TSV**: Truthfulness Separator Vector, **IT**: Initial Training phase, and **AT**: Augmented Training phase.

Index	Component			Dataset			
	TSV	**IT**	**AT**	TruthfulQA	TriviaQA	SciQ	NQ Open
(a)	✗	✓	✗	52.2	50.8	54.1	50.8
(b)	✗	✓	✓	52.0	50.2	57.1	52.1
(c)	✓	✓	✗	80.9	80.8	82.0	71.2
Ours	✓	✓	✓	**84.2**	**84.0**	**85.8**	**76.1**

Computational efficiency of TSV

Table 4. Performance comparison with PEFT methods. % Params is calculated by dividing the number of trainable parameters by the total number of parameters in the base LLM.

Model	Method	Trainable Parameters		Datasets	
		# Params	% Params	TruthfulQA	TriviaQA
Llama 3.1-8b	LoRA	3.4M	0.0424 %	83.6	82.0
	LoReFT	32K	0.0004 %	77.5	76.0
	Ours	**4K**	**0.00005%**	**84.2**	**84.0**
Qwen2.5-7b	LoRA	2.5M	0.0331 %	85.9	76.0
	LoReFT	28K	0.0004 %	81.5	79.3
	Ours	**3.6K**	**0.00005%**	**87.3**	**79.8**

总结与贡献

- 核心贡献：
  - TSV是一种轻量级、即插即用的引导向量，无需改变模型参数。
  - 结合少量有标签范例和大量无标签数据（通过最优传输进行伪标签和高置信度筛选）进行训练，就能达到全监督方法的效果。
  - 计算成本很低，泛化性良好。

Limitations and future work

- 细粒度幻觉检测：
  - 当前工作主要关注句子级别的幻觉。
  - 未来可以探索将TSV应用于更细粒度的Token或短语级别的幻觉定位。

- 长文本问答：
  - 当前实验主要基于短格式问答。
  - 可以将长文本分解为多个短问答对，再应用TSV进行检测