Manifold-based Sampling for In-Context Hallucination Detection in Large Language Models

Anonymous ACL submission

Abstract

Large language models (LLMs) frequently generate factually incorrect or unsupported content, commonly referred to as hallucinations. Prior work has explored decoding strategies, retrieval augmentation, and supervised fine-tuning for hallucination detection, while recent studies show that in-context learning (ICL) can substantially influence factual reliability. However, existing ICL demonstration selection methods often rely on surface-level similarity heuristics and exhibit limited robustness across tasks and models.

We propose MB-ICL, a manifold-based demonstration sampling framework for selecting in-context demonstrations that leverages latent representations extracted from frozen LLMs. By jointly modeling local manifold structure and class-aware prototype geometry, MB-ICL selects demonstrations based on their proximity to learned prototypes rather than lexical or embedding similarity alone.

Across factual verification (FEVER) and hallucination detection (HaluEval) benchmarks, MB-ICL outperforms standard ICL selection baselines in the majority of evaluated settings, with particularly strong gains on dialogue and summarization tasks. The method remains robust under temperature perturbations and model variation, indicating improved stability compared to heuristic retrieval strategies. While lexical retrieval can remain competitive in certain question-answering regimes, our results demonstrate that manifold-based prototype selection provides a reliable and training light approach for hallucination detection without modifying LLM parameters, offering a principled direction for improved ICL demonstration selection.

1 Introduction

The rapid scaling of pre-trained language models (PLMs) has led to the emergence of new capabilities (Wei et al., 2022), particularly through in-context learning (ICL), where models perform downstream tasks by conditioning on prompts that include a small number of representative examples. Owing to its effectiveness and flexibility, ICL has become a widely adopted and efficient paradigm for utilizing PLMs. This paradigm has been successfully applied to a variety of tasks, including reasoning and code generation (Li et al., 2023a), (Bodla and Yang, 2025), mathematical reasoning (Wu et al., 2025), and question answering (Liu et al., 2023), among others, without requiring explicit supervised fine-tuning.

In this study, we investigate the use of in-context learning (ICL) as a mechanism for hallucination detection in large language models. Unlike many

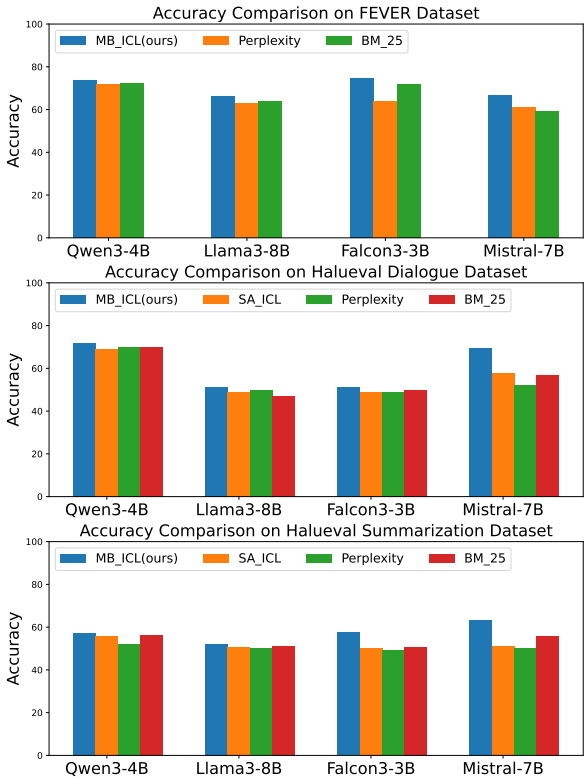


Figure 1: Accuracy comparisons of MB-ICL vs other methods

existing methods that target hallucination detection through architectural modifications or specialized training and inference (Su et al., 2024) procedures, we examine whether leveraging ICL alone can provide comparable effectiveness for this task. Our experimental findings show that Manifold-based ICL yields gains over established baselines across a wide range of benchmarks and model configurations while attaining performance levels that are comparable to those achieved by supervised fine-tuning (Appendix B). Importantly, these results are obtained with lower computational costs, highlighting the suitability of our approach for deployment in resource-constrained environments.

Existing ICL demonstration selection strategies largely rely on distance-based retrieval in a fixed embedding space, using lexical similarity, or heuristic ranking criteria. While effective in some settings, such approaches implicitly assume that the raw embedding geometry is well aligned with factual consistency, which may not hold across tasks and models. Unlike prior ICL selection methods that operate purely via distance-based retrieval in a fixed embedding space, our approach learns the geometry used for selection itself by explicitly modeling local manifolds and prototype structure. This shifts ICL selection from heuristic similarity ranking to geometry-based prototype sampling, which, to our knowledge, has not been explored for hallucination detection.

1.1 Contributions

Our contributions are summarized as follows:

- **Manifold based prototype learning.** We introduce a framework to sample In-Context-Learning (ICL) demonstrations by combining *piecewise-linear manifold learning* and *proxy-anchor based metric learning* to construct a low-dimensional sampling space. This joint formulation ensures that the sampled prototypes are semantically discriminative.

- **Manifold grounded Hallucination Detection.** We demonstrate that manifold-based sampling enhances the ability of LLMs to identify factual inconsistencies without explicit parameter updates. Our method achieves performance competitive with supervised fine-tuning (SFT) (Appendix B) while maintaining a substantially lower computational footprint. By leveraging pretrained models latent geometries, we present a scalable mechanism for hallucination detection that is robust across diverse architectures and settings.

2 Related Work

2.1 Hallucination detection

Identifying hallucinations in large language models is essential for maintaining the trustworthiness and reliability of their generated outputs, particularly in settings where factual accuracy is critical. SelfCheckGPT (Manakul et al., 2023) introduces a zero-resource, black-box framework for hallucination detection that operates without access to external knowledge bases. The method is grounded in the observation that when a model has sufficient familiarity with a topic, it tends to produce internally consistent factual statements across multiple generations, whereas responses generated for less familiar content are more likely to exhibit inconsistencies or unsupported claims. The FACTOR (Muhlgay et al., 2024) framework evaluates factuality by transforming a factual corpus into controlled multiple-choice benchmarks, generating plausible false alternatives and measuring whether a model assigns higher likelihood to true facts, which enables scalable assessment of factual accuracy without requiring external validation at generation time.

FActScore (Min et al., 2023) introduces a fine-grained evaluation metric by decomposing long-form text into atomic facts and computing the percentage of these facts supported by a trusted knowledge source, with an automated estimator that approximates human judgments efficiently. FACTOID (Rawte et al., 2024) reframes hallucination detection as a factual entailment problem, jointly identifying whether text is factually supported and localizing specific spans that contain errors, thereby improving upon traditional entailment models for fine-grained factual verification. Finally, FacTool (Chern et al., 2023) builds a task-agnostic, tool-augmented framework that extracts claims from generated outputs and verifies them using appropriate external tools such as search engines or execution environments, grounding factuality judgments in collected evidence rather than model-internal signals alone.

In (Kalai et al., 2025), the authors argue that a primary factor contributing to hallucinated yet seemingly plausible outputs in large language models is the mismatch between training and inference

objectives. Specifically, these objectives tend to incentivize confident guessing rather than explicit acknowledgment of uncertainty, which in turn complicates the identification of the underlying causes of hallucinations. To address the generation of hallucinated content, a wide range of mitigation strategies have been proposed (Sahoo et al., 2024). These approaches include, but are not limited to, data augmentation and manipulation techniques (Peng et al., 2023), (Chuang et al., 2024), alignment and fine-tuning strategies (Chen et al., 2023), (Zhang et al., 2024), as well as prompt engineering based methods (Varshney et al., 2023), (Luo et al., 2023), (Elaraby et al., 2023).

2.2 In-Context learning

In-context learning (ICL) (Brown et al., 2020) describes a paradigm in which language models solve tasks by relying on a limited set of example demonstrations provided at inference time, without modifying their internal parameters. Since these demonstrations are expressed directly in natural language, ICL enables an intuitive and interpretable mode of interaction with large language models (LLMs). This paradigm also aligns with key aspects of human cognition, particularly analogical reasoning, where prior examples inform decision-making in new situations (Winston, 1980). Unlike conventional supervised learning approaches, ICL does not require additional training or fine-tuning, allowing models to flexibly generalize to novel tasks without incurring extra computational cost.

Building on these observations, numerous unsupervised methods have been proposed to identify informative demonstrations for in-context learning, as reviewed by (Dong et al., 2024). One common line of work focuses on retrieving demonstrations that are closely related to the test input under predefined notions of similarity ((Liu et al., 2022), (Tanwar et al., 2023), Qin et al. (2024)). These approaches typically measure proximity using embedding-based distances, such as cosine similarity or Euclidean (L2) distance. Beyond similarity driven retrieval, prior studies have also explored alternative selection signals, including mutual information (Sorensen et al., 2022) and perplexity (Gonen et al., 2023), which enable effective prompt construction without requiring labeled supervision or access to a model's internal parameters.

Although widely used in practice, generic retrieval based strategies often depend on heuristic design choices and can produce suboptimal in context examples due to the absence of explicit task level supervision. To overcome these limitations, prior work has explored supervised retriever based methods for demonstration selection ((Rubin et al., 2022), (Ye et al., 2023), (Wang et al., 2024), (Zhang et al., 2022)). One representative approach is EPR proposed by (Rubin et al., 2022), which employs a two stage training procedure to learn dense retrievers tailored for demonstration retrieval. Building on this idea, (Li et al., 2023c) introduced a unified retrieval model capable of handling demonstration selection across diverse tasks. More recently, (Mavromatis et al., 2023) proposed AdaICL, a model aware framework that leverages LLM generated pseudo labels and uncertainty estimates over unlabeled data to adaptively identify informative in context demonstrations.

3 Method overview

Given a dataset D, all textual fields are concatenated with a prompt template T_p (Appendix A), resulting in a transformed dataset containing a single consolidated text field F_t paired with its corresponding label l. The consolidated text field is passed through the corresponding LLM to obtain latent representations Z. The encoded latent representations Z are obtained by taking the final transformer layer's token hidden states and computing an attention mask weighted mean across all non padding tokens to produce a single fixed dimensional vector per input text. The extracted latent vector Z is used for all the comparison baselines, including ours, to sample the ICL demonstrations with respect to each LLM.

As an initial step of our approach, we initialize a lightweight neural network h_θ (Table 2) that maps latent representations Z to transformed representations Z'. In addition, we initialize two proxy vectors θ_q, which serve as class representative prototypes within the proxy anchor loss, along with a momentum proxy vector θ_m that is updated according to the procedure outlined in Table 1. Upon completion of training, the learned momentum proxy θ_m is used to retrieve the nearest transformed representations Z', which are then selected as in context learning (ICL) demonstrations.

During training, for each mini batch $\mathcal{B} \subset D_{F_t}$, the latent representations Z are passed through h_θ to obtain transformed representations Z'. As described in Section 3.1, a manifold is then constructed for each batch based on Z'. The resulting

Algorithm: Training and Sampling Overview
Input: Dataset D_{F_T} with labels l, momentum coefficient μ
Output: Prototype representation θ_q and selected ICL demonstrations
Initialize projection network h_θ and prototype θ_q
for each batch $\mathcal{B} \subset D_{F_t}$ **do**
Obtain latent representations $Z \leftarrow \text{LLM}(T)$
$Z' \leftarrow h_\theta(Z)$, Construct manifold structure for Z' (Section 3.1)
Compute loss: $\mathcal{L} = \mathcal{L}_{\text{manifold}} + \mathcal{L}_{\text{PA}}$, eq. 2 , 1
Update parameters of h_θ and θ_q via gradient descent
Update Momentum proxy (He et al., 2020): $\theta_m \leftarrow \mu \cdot \theta_m + (1-\mu) \cdot \theta_q$
end for
Select ICL demonstrations by sampling representations Z' closest to θ_m

Table 1: Overview of the Proposed MB-ICL Algorithm.

manifold structure is used to compute the manifold point to point loss (Eq. 2), while the transformed representations Z' together with the proxy vectors θ_q are used to evaluate the proxy anchor loss (Eq. 1)

After computing both the manifold point to point loss and the proxy anchor loss, the overall training objective is formed as the sum of the two loss terms. The parameters of h_θ and the proxy vectors θ_q are then updated via gradient descent with respect to this combined loss, while the momentum proxy vector θ_m is updated following the procedure described in Table 1. Upon completion of training, the learned momentum proxy θ_m is used to retrieve the nearest transformed representations Z', which are selected as in context learning (ICL) demonstrations. The details of each parameter used and the training durations are clearly described in Appendix C.

We emphasize that our approach is not training free. Although the underlying language model remains fully frozen, our method requires training a lightweight retrieval module in the form of a projection network $h\theta$ and associated prototype representations. This training phase is performed once per dataset and model configuration and is substantially less expensive than fine tuning the language model itself. We therefore position MB-ICL as a training light rather than training free approach, operating in contrast to standard ICL methods that rely solely on heuristic or unsupervised retrieval without learned parameters.

3.1 Manifold Construction

Based on the Manifold hypothesis (Cayton, 2005), we can assume that the encoded latent representations can be locally approximated into smaller chunks of linear regions. Our approach leverages this assumption (Bhatnagar and Ahuja, 2024), (Holiday et al., 2019), (Koronaki et al., 2023), (Sonday et al., 2010) to identify representative prototypes that capture the characteristics of each action class.

Table 2: h_θ Model Architecture

Layer	**Layer Parameters**
Linear	(latent size Z, Prototype size Z')
InstanceNorm1d	Prototype size Z'
ReLU	-

Anchor Initialization and Local Clustering To approximate the piecewise linear structure of the data, we aim to construct m-dimensional linear submanifolds around selected anchor points. Given a batch B containing N data points, we randomly select n anchor points. For each anchor point $h_\theta(Z_i)$, we identify its nearest $m-1$ neighbors using Euclidean distance and form an initial neighborhood set X_i.

Manifold Expansion and Validation After initializing the local clusters, the manifold construction process proceeds iteratively by attempting to add the m-th nearest neighbor to X_i. After each addition, we compute the best fit m-dimensional submanifold using PCA to assess whether the members of X_i can be reconstructed with quality above threshold $T\%$. If so, the m-th point is retained in X_i; otherwise it is excluded. This evaluation is repeated for subsequent neighbors $N(h_\theta(x_i))_j$ for $j \in \{m+1, \ldots, k\}$, gradually constructing a local linear approximation of the manifold.

Submanifold Basis Extraction The final set X_i is comprised of all points in the anchor point's neighborhood that lie well within an m-dimensional linear submanifold. A basis for this submanifold is computed by applying PCA on X_i and extracting the top m eigenvectors. PCA is well suited for this task because it can effectively construct the lower-dimensional manifolds for locally linear regions.

3.2 Training Objectives

Proxy Anchor Loss: We use a modified version of proxy anchor loss (Kim et al., 2020) with Euclidean distance instead of cosine similarity:

$$\mathcal{L}_{\text{PA}} = \frac{1}{|\Theta_+|} \sum_{\theta_q \in \Theta_+} \log\left(1 + \sum_{Z' \in \mathcal{Z}'^{+}_{\theta_q}} \exp\Big(-\alpha\big(\|h_\theta(Z') - \theta_q\|_2 - \epsilon\big)\Big)\right) + \frac{1}{|\Theta|} \sum_{\theta_q \in \Theta} \log\left(1 + \sum_{Z' \in \mathcal{Z}'^{-}_{\theta_q}} \exp\Big(\alpha\big(\|h_\theta(Z') - \theta_q\|_2 - \epsilon\big)\Big)\right) \tag{1}$$

Let Θ represent the collection of all proxy vectors, where each proxy $\theta_q \in \Theta$ functions as a class-level representative in the embedding space. From this set, $\Theta_+ \subseteq \Theta$ denotes the subset of proxies that are associated with at least one positive sample within the current mini-batch B. For any selected proxy θ_q, the latent embeddings $\mathcal{Z}'$ contained in B (with $Z' \in \mathcal{Z}'$) are divided into two disjoint groups: the positive set $\mathcal{Z}'{\theta_q}^+$, consisting of embeddings from the same class as θ_q, and the negative set $\mathcal{Z}'{\theta_q}^- = \mathcal{Z}' \setminus \mathcal{Z}'^{+}_{\theta_q}$, which contains all remaining embeddings. The temperature parameter α regulates the concentration of the optimization process: larger values emphasize hard positive and negative pairs by sharpening gradients, whereas smaller values yield smoother updates by distributing influence more uniformly across pairs. The margin parameter ϵ introduces an explicit separation constraint, ensuring that embeddings from the same class are pulled closer to their corresponding proxies while embeddings from different classes are pushed beyond a minimum distance threshold.

Manifold Point-to-Point Loss: This loss helps in estimating the point to point similarities preserving the geometric structure:

$$\mathcal{L}_{\text{manifold}} = \sum_{i,j} \left(\delta \cdot (1 - s(Z'_i, Z'_j)) - \|h_\theta(Z'_i) - h_\theta(Z'_j)\|_2\right)^2 \tag{2}$$

where $s(Z'_i, Z'_j)$ is the manifold similarity computed as:

$$s(Z'_i, Z'_j) = \frac{s'(Z'_i, Z'_j) + s'(Z'_j, Z'_i)}{2}$$

with $s'(Z'_i, Z'_j) = \alpha(Z'_i, Z'_j) \cdot \beta(Z'_i, Z'_j)$, where:

$$\alpha(Z'_i, Z'_j) = \frac{1}{\left(1 + o(Z'_i, Z'_j)^2\right)^{N_\alpha}}$$
$$\beta(Z'_i, Z'_j) = \frac{1}{\left(1 + p(Z'_i, Z'_j)\right)^{N_\beta}}$$

In (2), h_θ denotes a lightweight neural transformation whose architecture is detailed in Table 2, and δ is a scaling constant that sets an upper bound on the separation between dissimilar samples. This objective aligns Euclidean distances in the transformed embedding space with manifold-induced dissimilarities $1 - s(Z'_i, Z'_j)$, thereby encouraging the learned metric to conform to the intrinsic geometry of the data manifold. The term $o(Z'_i, Z'_j)$ measures the orthogonal distance from point z_i to the manifold associated with Z'_j, while $p(Z'_i, Z'_j)$ captures the distance between Z'_j and the projection of z_i onto that manifold. The hyperparameters N_α and N_β regulate the rate at which similarity attenuates with increasing distance, with the constraint $N_\alpha > N_\beta$ ensuring that off-manifold points experience a steeper decay in similarity than points residing on the same manifold.

Distance Calculation. For each embedding pair (Z'_i, Z'_j), the distances $o(Z'_i, Z'_j)$ and $p(Z'_i, Z'_j)$ are computed using the local manifold basis P_j associated with point Z'_j. The projection of Z'_i onto the manifold spanned by P_j is defined as $\text{proj}_{P_j}(Z'_i) = Z'_j + \sum_k \langle Z'_i - Z'_j, v_k \rangle v_k$, where $\{v_k\}$ denotes the basis vectors of P_j. The orthogonal distance is then given by $o(Z'_i, Z'_j) = \|Z'_i - \text{proj}_{P_j}(Z'_i)\|_2$, while the projected distance along the manifold is defined as $p(Z'_i, Z'_j) = \|\text{proj}_{P_j}(Z'_i) - Z'_j\|_2$. This procedure is applied to all point pairs, enabling the loss to capture the local geometric structure of the data manifold.

4 Experiments

We focus on hallucination detection, where the model predicts factual consistency labels, rather than modifying generation to reduce hallucinations. We conduct experiments across six large language models on hallucination detection (HaluEval) and factual verification (FEVER) tasks to evaluate our sampling approach. We follow prior work in in-context learning and report single run accuracy since demonstration selection is deterministic given fixed representations and prompts. Demonstration selection, prompt construction, and decoding are all fixed across runs, and no stochastic sampling or retriever randomness is introduced. Our method (**MB-ICL**) attains higher accuracy relative to baseline methods in most of the scenarios. Additionally, lower perplexity scores compared to BM25 and self adaptive ICL sampling suggest reduced predictive uncertainty. To assess robustness to decoding stochasticity, we examine accuracy across different sampling temperatures. For the Qwen3-4B and Falcon3-3B models, our method shows limited sensitivity to temperature changes, in contrast to

BM25 and self adaptive ICL based sampling methods, indicating more concentrated predictive distributions under typical inference settings. Our ablation study across all model parameters indicates that the prototype size Z' has the largest impact on performance. In particular, when the original latent dimension Z is substantially reduced to Z', we observe consistent changes in performance, suggesting the presence of manifold learning effects induced by the reduced representation capacity.

4.1 Dataset

We conduct experiments across four distinct tasks using six different large language models. The FEVER (Thorne et al., 2018) dataset is a widely used benchmark for evaluating a model's ability to assess the factual correctness of a given sentence. It consists of two splits (train and test), where each instance contains a claim paired with a label indicating whether the statement is factual or hallucinated. The HaluEval (Li et al., 2023b) dataset is comprised of 5,000 general user queries paired with ChatGPT-generated responses, along with 30,000 task specific examples spanning three tasks: question answering, knowledge grounded dialogue conversation, and summarization. In the HaluEval question answering setting, the model is provided with a question, a supporting knowledge snippet, and a candidate answer, and is required to determine whether the answer is hallucinated. The dialogue and summarization subsets follow a similar format.

4.2 Evaluation details & Results

All sampling strategies, including our proposed method, were evaluated under a two shot in context learning (ICL) setting, where each method selected exactly two demonstrations per input across all models and tasks. This setup ensures that performance differences arise from the quality of the selected demonstrations rather than the quantity of contextual information provided to the model.

For the FEVER dataset, which includes predefined training and test splits, the training split was used exclusively for sampling ICL demonstrations, while evaluation was conducted on the held out test split. For the HaluEval tasks summarization, question answering, and dialogue the dataset does not provide explicit splits. In these cases, demonstrations were sampled directly from the dataset and subsequently removed prior to evaluation to prevent data leakage. All methods followed the same procedure to maintain consistency across comparisons.

Table 3 summarizes the performance of all methods across six language models and four tasks. Across the majority of settings, our manifold based ICL sampling approach (MB-ICL) achieves higher accuracy than baseline retrieval and selection strategies, including BM25, perplexity based selection, clustering, and self-adaptive ICL (SA-ICL). MB-ICL improves hallucination detection accuracy across most models and tasks, with particularly strong gains in dialogue and summarization settings.

MB-ICL improves performance across models and tasks in the majority of settings. On Mistral-7B, the method achieves large gains on dialogue and summarization hallucination detection, with absolute improvements of up to 9.9 accuracy points over standard retrieval baselines (Table 3). For Falcon3-3B, MB-ICL similarly yields clear improvements on the summarization task, where baseline selection strategies show limited effectiveness. In contrast, recent adaptive ICL selection methods generally report more modest gains, typically on the order of 1–2 percentage points, in comparable settings, highlighting the impact of effective demonstration selection for hallucination detection tasks. We note that in certain question-answering settings, lexical retrieval methods such as BM25 outperform MB-ICL, suggesting that surface level overlap can remain a strong signal when factual correctness is tightly coupled to explicit evidence matching.

To further examine the effect of the number of demonstrations, we conducted an ablation study varying the ICL shot count for the Qwen3-4B and Falcon3-3B models. As shown in Figures 4a and 4b, performance improves as the number of demonstrations increases up to approximately ten shots, after which gains begin to saturate. This trend is consistent across both dialogue and question answering tasks, indicating diminishing returns from additional demonstrations beyond this point.

During the hyperparameter tuning, we found that the impact of the prototype dimensionality Z' on LLM performance was more influential than that of the others (Fig 8). Figures 5b and 5a show accuracy as a function of prototype size for the Qwen3-4B and Mistral-7B models. Across both models, reducing the prototype dimensionality leads to consistent performance improvements. This trend suggests that projecting representations into a more compact space improves the quality of selected

demonstrations, likely by suppressing task irrelevant variation while preserving the information necessary for hallucination detection.

Finally, we note that these trends are consistent across different model families and tasks, indicating that the observed improvements are not tied to a specific architecture or dataset. Collectively, these results demonstrate that the proposed manifold based sampling strategy provides a reliable and effective mechanism for selecting ICL demonstrations in hallucination detection settings. All reported results are obtained with the decoding temperature fixed at 0, making both demonstration selection and generation fully deterministic. Following prior work on deterministic in context learning, we therefore report single run accuracy. Robustness is instead assessed via temperature perturbation analysis, as reported in Section 4.4.

4.3 Comparison methods

We compare our approach against five baseline sampling methods. **KNN** selects the most similar example(s) from the dataset for each query based on cosine similarity. **Clustering** chooses examples that are closest to the class centroid within each group. We note that clustering-based selection often performs near chance level in hallucination detection settings, reflecting the limited alignment between global semantic clusters and factual consistency labels. The **Perplexity** (Gonen et al., 2023) based method follows prior work on perplexity driven selection, where all input fields except the label are concatenated into a text prompt and passed through the corresponding LLM to compute perplexity scores. Prompts with the lowest perplexity values are then selected as in context learning (ICL) demonstrations. **BM25** is a lexical similarity based retrieval method that ranks candidate examples using term frequency inverse document frequency (TF–IDF) statistics and selects the top scoring examples as demonstrations. **SA-ICL** (Self-Adaptive ICL) (Wu et al., 2023) is based on prior self-adaptive ICL work, where the top-k relevant demonstrations for each test query are first retrieved using similarity measures and subsequently reordered to obtain an arrangement that minimizes the code length of the label Y, following the Minimum Description Length (MDL) principle.

4.4 Effect of Temperature and Perplexity based evaluation

As an additional sanity check, we examined the effect of generation temperature on model behavior by varying the temperature from 0 to 1 in increments of 0.1. We compared the resulting output variability of our method with that of **BM25** and **SA-ICL** on the HaluEval question answering and dialogue tasks Fig 7a, 7b, 6a, 6b. Across this range, our approach exhibited more stable behavior relative to the comparison methods. Specifically, as the temperature increased from $T = 0$ to $T = 1$, the variation in model outputs remained limited, suggesting that the sampling procedure is comparatively less sensitive to temperature changes.

This pattern is consistent with a more concentrated token probability distribution during generation. In addition to accuracy, we report average perplexity scores under fixed decoding conditions (temperature = 0) as a complementary diagnostic signal. Perplexity here is not interpreted as a calibrated measure of model uncertainty, but rather as an indicator of how well the selected in-context demonstrations align with the model's internal likelihood structure. Lower perplexity suggests that the resulting prompts induce outputs that are more consistent with the model's learned distribution, which empirically correlates with improved hallucination detection performance in our setting. Importantly, we treat perplexity as an observational metric and do not claim a causal relationship between perplexity reduction and factual correctness.

5 Future Works and Limitations

While our results demonstrate majority improvements across multiple tasks and model families, the current evaluation is limited to language models with up to 8B parameters, and it remains an open question how manifold based prototype sampling scales to substantially larger models (e.g., 70B+). Additionally, our approach introduces a one time training cost for a lightweight projection head and requires local manifold construction via PCA, which adds computational overhead compared to purely heuristic retrieval methods, though this cost remains significantly lower than fine tuning. Finally, while demonstration selection is deterministic and robust under temperature perturbations, our evaluation does not explore example level uncertainty estimates or adaptive prototype updates at inference time, which could further improve ro-

Model	Task	KNN	Clustering	BM25	Perplexity	SA-ICL	MB-ICL (Ours)
Qwen3-4B	HaluEval Dialogue	14.7	49.9	70.0	70.0	69.0	**72.0**
	HaluEval Summ.	53.4	54.9	56.4	52.0	56.0	**57.2**
	HaluEval QA	12.7	50.0	69.0	**74.0**	71.0	70.5
	FEVER	72.2	70.5	72.1	72.0	–	**73.7**
Llama3-8B	HaluEval Dialogue	11.1	50.0	47.0	50.0	49.0	**51.0**
	HaluEval Summ.	51.7	50.1	51.1	50.0	50.7	**52.0**
	HaluEval QA	7.0	50.0	63.0	62.0	65.0	**69.0**
	FEVER	65.5	62.8	64.0	63.0	–	**66.4**
Falcon-3-3B	HaluEval Dialogue	5.9	50.0	50.0	49.0	49.0	**51.0**
	HaluEval Summ.	50.4	51.8	50.8	49.1	50.2	**57.5**
	HaluEval QA	5.8	50.0	49.0	50.0	50.0	**51.0**
	FEVER	73.8	71.8	72.0	64.0	–	**74.7**
GPT-Neo-2.7B	HaluEval Dialogue	–	49.9	50.0	48.0	50.0	**52.0**
	HaluEval QA	–	50.0	49.0	45.0	49.0	**52.0**
	FEVER	43.0	50.0	49.0	49.0	–	**50.2**
Mistral-7B	HaluEval Dialogue	59.7	49.9	57.0	52.0	58.0	**69.6**
	HaluEval Summ.	39.9	57.2	56.0	50.0	51.0	**63.2**
	HaluEval QA	62.5	50.0	**66.0**	52.0	62.0	53.0
	FEVER	64.1	64.7	59.0	61.0	–	**66.7**
Vicuna-7B	HaluEval Dialogue	16.4	49.9	49.0	50.0	50.0	**51.0**
	HaluEval Summ.	50.0	50.7	26.0	0.0	39.0	**56.3**
	HaluEval QA	26.8	50.0	50.0	50.0	50.0	**51.0**
	FEVER	65.5	61.3	67.0	66.3	–	**71.1**

Table 3: Performance comparison of MB-ICL against baselines. Missing values (–) indicate tasks where the baseline was not applicable or data was unavailable for that specific model configuration.

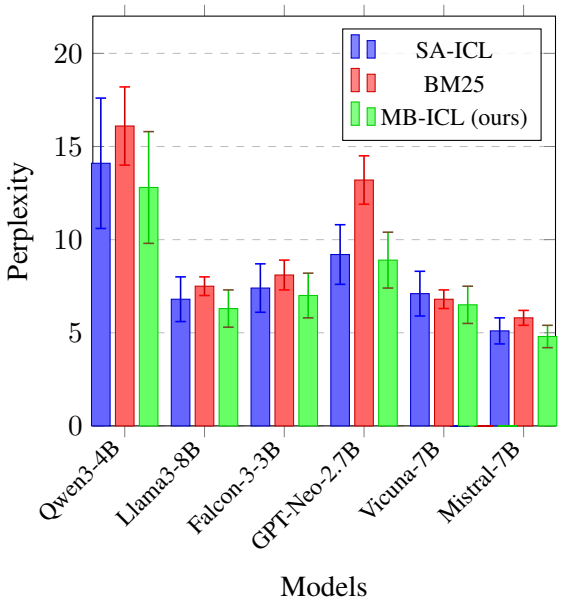


Figure 2: Perplexity comparison across models on HaluEval dialogue task. Lower perplexity indicates better performance.

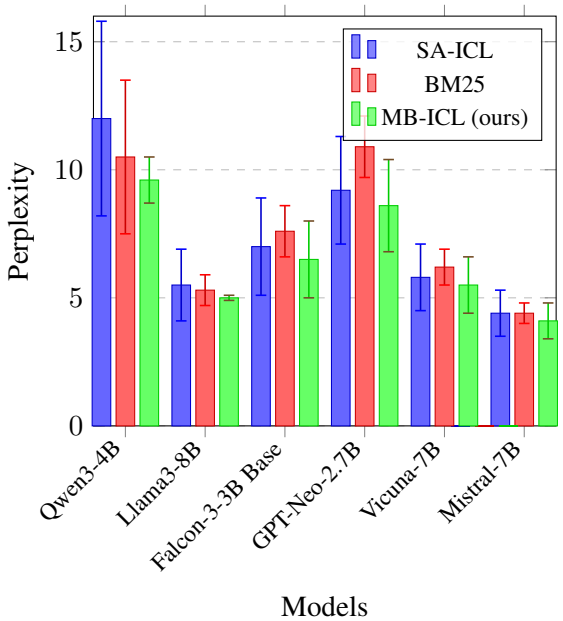


Figure 3: Perplexity comparison across models on HaluEval QA task. Lower perplexity indicates better performance.

bustness in dynamic or streaming scenarios.

References

Shubhang Bhatnagar and Narendra Ahuja. 2024. Piecewise-linear manifolds for deep metric learning.

Preprint, arXiv:2403.14977.

Krishna Vamshi Bodla and Haizhao Yang. 2025. Protocode: Prototype-driven interpretability for code generation in llms. *Preprint*, arXiv:2509.25247.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. *Preprint*, arXiv:2005.14165.

Lawrence Cayton. 2005. Algorithms for manifold learning.

Anthony Chen, Panupong Pasupat, Sameer Singh, Hongrae Lee, and Kelvin Guu. 2023. Purr: Efficiently editing language model hallucinations by denoising language model corruptions. *Preprint*, arXiv:2305.14908.

I-Chun Chern, Steffi Chern, Shiqi Chen, Weizhe Yuan, Kehua Feng, Chunting Zhou, Junxian He, Graham Neubig, and Pengfei Liu. 2023. Factool: Factuality detection in generative ai – a tool augmented framework for multi-task and multi-domain scenarios. *Preprint*, arXiv:2307.13528.

Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James Glass, and Pengcheng He. 2024. Dola: Decoding by contrasting layers improves factuality in large language models. *Preprint*, arXiv:2309.03883.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Tianyu Liu, Baobao Chang, Xu Sun, Lei Li, and Zhifang Sui. 2024. A survey on in-context learning. *Preprint*, arXiv:2301.00234.

Mohamed Elaraby, Mengyin Lu, Jacob Dunn, Xueying Zhang, Yu Wang, Shizhu Liu, Pingchuan Tian, Yuping Wang, and Yuxuan Wang. 2023. Halo: Estimation and reduction of hallucinations in open-source weak large language models. *Preprint*, arXiv:2308.11764.

Hila Gonen, Srini Iyer, Terra Blevins, Noah Smith, and Luke Zettlemoyer. 2023. Demystifying prompts in language models via perplexity estimation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10136–10148, Singapore. Association for Computational Linguistics.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning. *Preprint*, arXiv:1911.05722.

Alexander Holiday, Mahdi Kooshkbaghi, Juan M. Bello-Rivas, C. William Gear, Antonios Zagaris, and Ioannis G. Kevrekidis. 2019. Manifold learning for parameter reduction. *Journal of Computational Physics*, 392:419–431.

Adam Tauman Kalai, Ofir Nachum, Santosh S. Vempala, and Edwin Zhang. 2025. Why language models hallucinate. *Preprint*, arXiv:2509.04664.

Sungyeon Kim, Dongwon Kim, Minsu Cho, and Suha Kwak. 2020. Proxy anchor loss for deep metric learning. *Preprint*, arXiv:2003.13911.

Eleni D. Koronaki, Nikolaos Evangelou, Cristina P. Martin-Linares, Edriss S. Titi, and Ioannis G. Kevrekidis. 2023. Nonlinear dimensionality reduction then and now: Aims for dissipative pdes in the ml era. *Preprint*, arXiv:2310.15816.

Jia Li, Ge Li, Chongyang Tao, Jia Li, Huangzhao Zhang, Fang Liu, and Zhi Jin. 2023a. Large language model-aware in-context learning for code generation. *Preprint*, arXiv:2310.09748.

Junyi Li, Xiaoxue Cheng, Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2023b. HaluEval: A large-scale hallucination evaluation benchmark for large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6449–6464, Singapore. Association for Computational Linguistics.

Xiaonan Li, Kai Lv, Hang Yan, Tianyang Lin, Wei Zhu, Yuan Ni, Guotong Xie, Xiaoling Wang, and Xipeng Qiu. 2023c. Unified demonstration retriever for in-context learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4644–4668, Toronto, Canada. Association for Computational Linguistics.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2022. What makes good in-context examples for GPT-3? In *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pages 100–114, Dublin, Ireland and Online. Association for Computational Linguistics.

Weihao Liu, Fangyu Lei, Tongxu Luo, Jiahe Lei, Shizhu He, Jun Zhao, and Kang Liu. 2023. Mmhqa-icl: Multimodal in-context learning for hybrid question answering over text, tables and images. *Preprint*, arXiv:2309.04790.

Junyu Luo, Cao Xiao, and Fenglong Ma. 2023. Zero-resource hallucination prevention for large language models. *Preprint*, arXiv:2309.02654.

Potsawee Manakul, Adian Liusie, and Mark J. F. Gales. 2023. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. *Preprint*, arXiv:2303.08896.

Costas Mavromatis, Balasubramaniam Srinivasan, Zhengyuan Shen, Jiani Zhang, Huzefa Rangwala, Christos Faloutsos, and George Karypis. 2023. Which examples to annotate for in-context learning? towards effective and efficient selection. *Preprint*, arXiv:2310.20046.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. *Preprint*, arXiv:2305.14251.

Dor Muhlgay, Ori Ram, Inbal Magar, Yoav Levine, Nir Ratner, Yonatan Belinkov, Omri Abend, Kevin Leyton-Brown, Amnon Shashua, and Yoav Shoham. 2024. Generating benchmarks for factuality evaluation of language models. *Preprint*, arXiv:2307.06908.

Baolin Peng, Michel Galley, Pengcheng He, Hao Cheng, Yujia Xie, Yu Hu, Qiuyuan Huang, Lars Liden, Zhou Yu, Weizhu Chen, and Jianfeng Gao. 2023. Check your facts and try again: Improving large language models with external knowledge and automated feedback. *Preprint*, arXiv:2302.12813.

Chengwei Qin, Aston Zhang, Chen Chen, Anirudh Dagar, and Wenming Ye. 2024. In-context learning with iterative demonstration selection. *Preprint*, arXiv:2310.09881.

Vipula Rawte, S. M Towhidul Islam Tonmoy, Krishnav Rajbangshi, Shravani Nag, Aman Chadha, Amit P. Sheth, and Amitava Das. 2024. Factoid: Factual entailment for hallucination detection. *Preprint*, arXiv:2403.19113.

Ohad Rubin, Jonathan Herzig, and Jonathan Berant. 2022. Learning to retrieve prompts for in-context learning. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2655–2671, Seattle, United States. Association for Computational Linguistics.

Pranab Sahoo, Prabhash Meharia, Akash Ghosh, Sriparna Saha, Vinija Jain, and Aman Chadha. 2024. A comprehensive survey of hallucination in large language, image, video and audio foundation models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 11709–11724, Miami, Florida, USA. Association for Computational Linguistics.

Benjamin E. Sonday, Amit Singer, C. William Gear, and Ioannis G. Kevrekidis. 2010. Manifold learning techniques and model reduction applied to dissipative pdes. *Preprint*, arXiv:1011.5197.

Taylor Sorensen, Joshua Robinson, Christopher Rytting, Alexander Shaw, Kyle Rogers, Alexia Delorey, Mahmoud Khalil, Nancy Fulda, and David Wingate. 2022. An information-theoretic approach to prompt engineering without ground truth labels. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 819–862, Dublin, Ireland. Association for Computational Linguistics.

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. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 14379–14391, Bangkok, Thailand. Association for Computational Linguistics.

Eshaan Tanwar, Subhabrata Dutta, Manish Borthakur, and Tanmoy Chakraborty. 2023. Multilingual LLMs are better cross-lingual in-context learners with alignment. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6292–6307, Toronto, Canada. Association for Computational Linguistics.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and VERification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 809–819, New Orleans, Louisiana. Association for Computational Linguistics.

Neeraj Varshney, Wenlin Yao, Hongming Zhang, Jianshu Chen, and Dong Yu. 2023. A stitch in time saves nine: Detecting and mitigating hallucinations of llms by validating low-confidence generation. *Preprint*, arXiv:2307.03987.

Xinyi Wang, Wanrong Zhu, Michael Saxon, Mark Steyvers, and William Yang Wang. 2024. Large language models are latent variable models: Explaining and finding good demonstrations for in-context learning. *Preprint*, arXiv:2301.11916.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022. Emergent abilities of large language models. *Preprint*, arXiv:2206.07682.

Patrick H. Winston. 1980. Learning and reasoning by analogy. *Commun. ACM*, 23(12):689–703.

Jinyang Wu, Mingkuan Feng, Shuai Zhang, Feihu Che, Zengqi Wen, Chonghua Liao, and Jianhua Tao. 2025. Beyond examples: High-level automated reasoning paradigm in in-context learning via mcts. *Preprint*, arXiv:2411.18478.

Zhiyong Wu, Yaoxiang Wang, Jiacheng Ye, and Lingpeng Kong. 2023. Self-adaptive in-context learning: An information compression perspective for in-context example selection and ordering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1423–1436, Toronto, Canada. Association for Computational Linguistics.

Jiacheng Ye, Zhiyong Wu, Jiangtao Feng, Tao Yu, and Lingpeng Kong. 2023. Compositional exemplars for

in-context learning. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 39818–39833. PMLR.

Shuo Zhang, Liangming Pan, Junzhou Zhao, and William Yang Wang. 2024. The knowledge alignment problem: Bridging human and external knowledge for large language models. *Preprint*, arXiv:2305.13669.

Yiming Zhang, Shi Feng, and Chenhao Tan. 2022. Active example selection for in-context learning. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9134–9148, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

A Prompt Templates

The following is the prompt template that we used in our experiments to combine the sampled ICL demonstrations with the test query, all the models and baseline methods, including ours, follow the same prompt template.

Listing 1: ICL prompt construction for Halueval QA task

```
def build_icl_prompt(knowledges,
    questions, answers, few_shots=None):
    prompt = (
        "You are an unbiased document-
            grounded fact checker. "
        "You are provided a Knowledge, a
             question based on the
            knowledge and a answer based
             on the question. "
        "Based on the provided knowledge
             for the question identify
            if the corresponding answer
            is hallucinated or not.\n"
        "Hallucination response: 'yes' =
             hallucinated, 'no' = not
            hallucinated.\n\n"
    )

    if few_shots:
        for know, ques, ans, label in
            few_shots:
            prompt += (
                f"Knowledge: {know}\n"
                f"Question: {ques}\n"
                f"Answer: {ans}\n"
                f"Hallucination response
                    : [BEGIN]{label}[
                    DONE]\n\n"
            )

    prompt += (
        f"Knowledge: {knowledges}\n"
        f"question: {questions}\n"
        f"Answer: {answers}\n"
        "Hallucination response: [BEGIN]
            "
    )

    return prompt
```

Listing 2: ICL prompt construction for HaluEval Dialogue task

```
def build_icl_prompt(knowledges,
    diag_hists, responses, few_shots=
    None):
    prompt = (
        "You are an unbiased document-
            grounded fact checker. "
        "You are provided a Knowledge, a
             dialogue history based on
            the knowledge and a response
             "
        "based on the Knowledge and the
            dialogue history. "
        "Based on the provided knowledge
             and dialogue history
            identify if the
            corresponding "
        "response is hallucinated or not
            .\n"
        "Hallucination response: 'yes' =
             hallucinated, 'no' = not
            hallucinated.\n\n"
    )

    if few_shots:
        for know, diag_hist, res, label
            in few_shots:
            prompt += (
                f"Knowledge: {know}\n"
                f"Dialogue history: {
                    diag_hist}\n"
                f"Response: {res}\n"
                f"Hallucination response
                    : [BEGIN]{label}[
                    DONE]\n\n"
            )

    prompt += (
        f"Knowledge: {knowledges}\n"
        f"Dialogue history: {diag_hists
            }\n"
        f"Response: {responses}\n"
        "Hallucination response: [BEGIN]
            "
    )

    return prompt
```

Listing 3: ICL prompt construction for HaluEval Summarization task

```
def build_icl_prompt(documents,
    summaries, few_shots=None):
    prompt = (
        "You are an unbiased document-
            grounded fact checker. "
        "Identify if the corresponding
            summary for the given
            document "
        "is hallucinated or not.\n"
        "Hallucination response: 'yes' =
             hallucinated, "
        "'no' = not hallucinated.\n\n"
    )

    if few_shots:
        for doc, summary, label in
            few_shots:
```

```
            prompt += (
                f"Document: {doc}\n"
                f"Summary: {summary}\n"
                f"Hallucination response
                    : [BEGIN]{label}[
                    DONE]\n\n"
            )

    prompt += (
        f"Document: {documents}\n"
        f"Summary: {summaries}\n"
        "Hallucination response: [BEGIN]
            "
    )

    return prompt
```

Listing 4: ICL prompt construction for FEVER dataset

```
def build_icl_prompt(test_problem,
    few_shots=None):
    prompt = ""
    if few_shots:
        for prob, sol in few_shots:
            prompt += (
                "You are an unbiased
                    fact checker. "
                "Classify the following
                    claim as supported
                    if it is valid "
                "or refuted if it is
                    invalid:\n"
                f"Claim: {prob}\n"
                "[BEGIN]\n"
                f"{sol}\n"
                "[DONE]\n\n"
            )

    prompt += (
        "You are an unbiased fact
            checker. "
        "Classify the following claim as
             supported if it is valid "
        "or refuted if it is invalid:\n"
        f"Claim: {test_problem}\n"
        "[BEGIN]\n"
    )

    return prompt
```

B Performance Considerations

A popular approach to mitigating hallucinations in LLMs is fine-tuning, which modifies the underlying model weights to align with factual distributions. While effective, fine-tuning requires significant GPU memory and multiple training iterations. In contrast, our proposed manifold based sampling framework operates as a lightweight retrieval mechanism that achieves comparable performance gains using a fraction of the resources. By training only a thin projection head h_θ (see Table 2) and keeping the base model frozen, our method reduces the training memory footprint by several orders of magnitude, making MB-ICL suitable for domain-specific hallucination where fine-tuning is impractical.

Method	Memory (GB)	Time (hrs)	Acc
LoRA-SFT, 4-bit	24	2.6	0.71
LoRA-SFT, 8-bit	40	3.6	0.74
LoRA-SFT, 16-bit	72	4.1	0.73
MB-ICL (ours)	0.64	3.4	0.71

Table 4: Comparison of computational cost and performance between supervised fine-tuning (SFT) and MB-ICL for Vicuna-7B model. MB-ICL achieves comparable performance with orders-of-magnitude lower GPU memory usage.

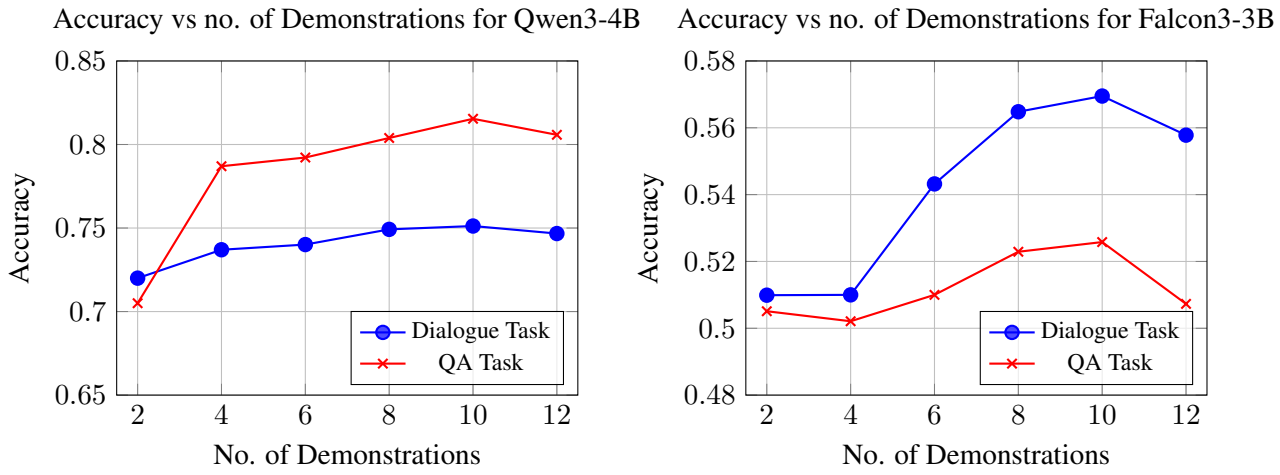
We evaluate the performance of MB-ICL against a fine-tuned baseline. All experiments are conducted on Vicuna-7B using A6000 RTX GPUs with 48 GB of memory. Fine-tuning the 16-bit model requires two GPUs, while all other fine-tuning configurations are run on a single GPU. For each model, we attached LoRA adapters to the last 3 to 5 layers, including the highest performing configuration in Table 4. A summary of experiment configurations and computational results is provided in Table 4.

C Implementation details

The network h_θ, which is crucial in our method to sample the ICL demonstrations, is trained for 200 epochs on the training dataset D_{F_t}. Training utilizes two independent Adam optimizers: one for the network parameters and another for the proxy parameters. Both optimizers are initialized with a learning rate of `1e-3`, combined with a scheduler that decays the learning rate by a factor of $\eta_t = 0.97$. The dimensionality of the encoded vector Z is determined by the underlying Large Language Model (M). A mini-batch size of 128 samples is maintained throughout training.

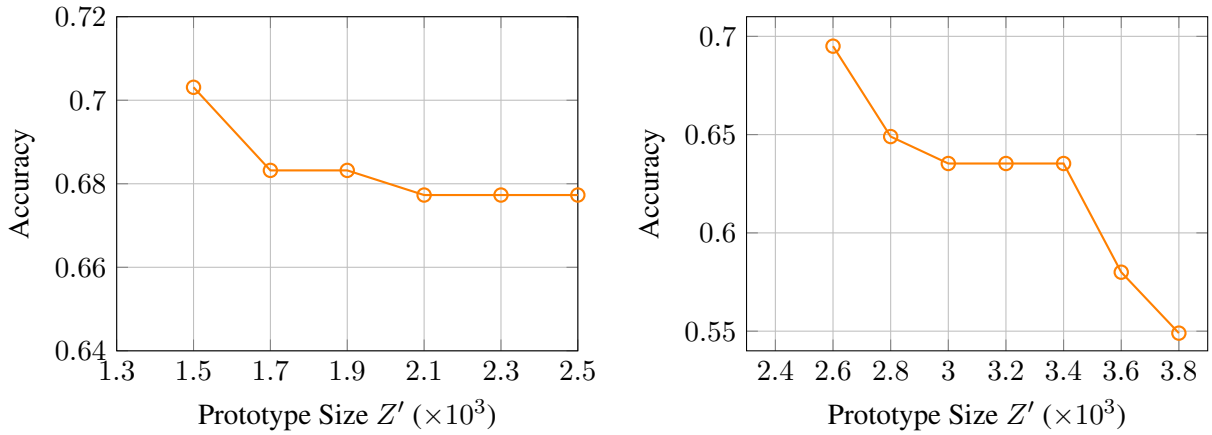
For the initial set of experiments, the hyperparameters for manifold construction and manifold point-to-point loss estimation are configured as follows: $T = 90\%$, $\delta = 2$, $m = 3$, $N_\alpha = 4$, and $N_\beta = 0.5$. The momentum constant for updating θ_m is set to $\mu = 0.99$. For Proxy Anchor loss, we employ $\alpha = 32$ and $\epsilon = 0.1$. These settings serve as the baseline configuration; subsequently, an ablation study is conducted on the above parameters for LLMs that exhibited comparatively lower performance than competing methods.

All experiments were conducted on an NVIDIA RTX A6000 GPU. Training the lightweight neu-



(a) Qwen3-4B (b) Falcon3-3B

Figure 4: Performance of MB-ICL under varying numbers of in-context demonstrations



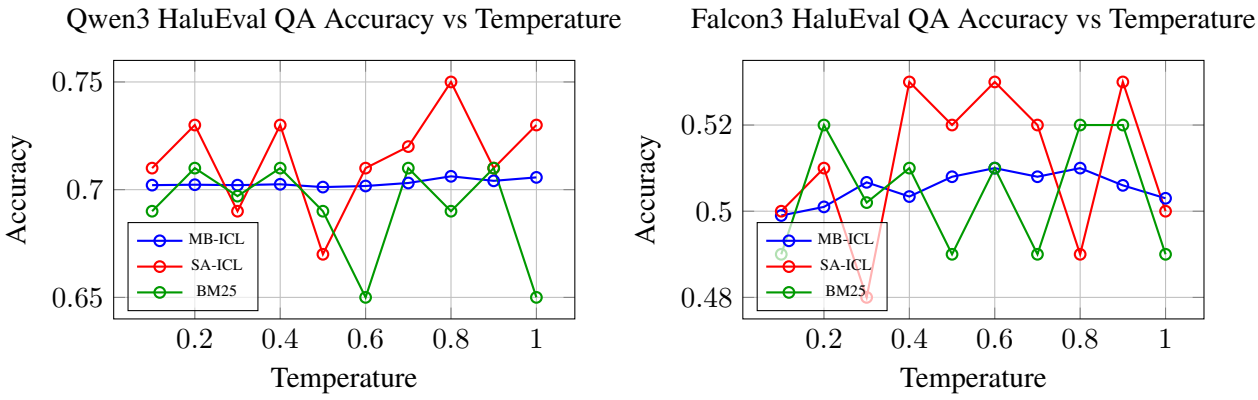
(a) Qwen3-4B QA task (b) Mistral-7B Dialogue task

Figure 5: Accuracy vs Prototype Size Z'

ral network h_θ requires approximately 640 MB of GPU memory and about 3-4 hours of training time.

D AI usage

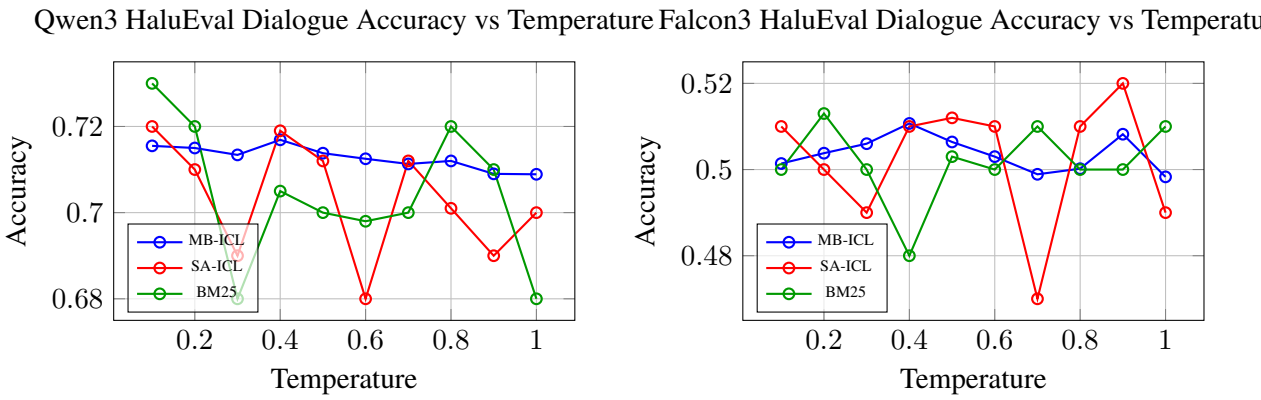
We have used AI tools(Chatgpt) for writing assistance. We did not use it during any of the ideation or experimentation phase.



(a) Qwen3 HaluEval QA

(b) Falcon3 HaluEval QA

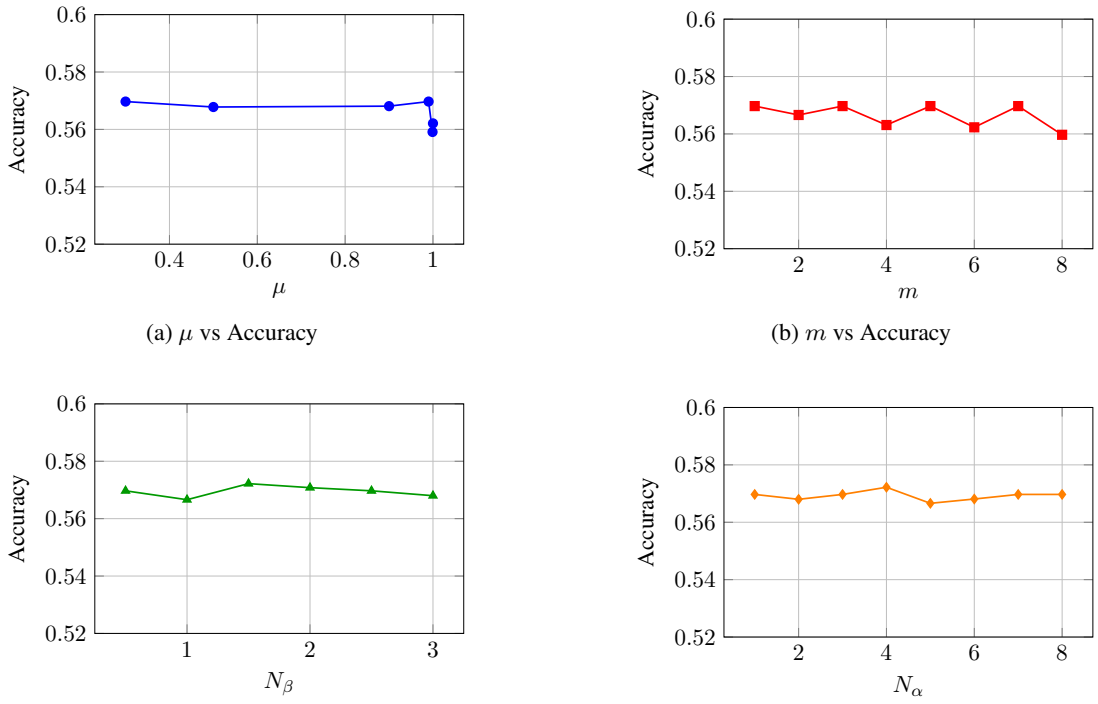
Figure 6: Accuracy vs Temperature comparison for Qwen3 and Falcon3 models on HaluEval QA dataset



(a) Qwen3 HaluEval Dialogue task

(b) Falcon3 HaluEval Dialogue task

Figure 7: Accuracy vs Temperature comparisons for Qwen3 and Falcon3 models on HaluEval Dialogue dataset



(a) μ vs Accuracy

(b) m vs Accuracy

(c) N_β vs Accuracy

(d) N_α vs Accuracy

Figure 8: Hyperparameter tuning for Qwen3-4B on Summarization task