TEST-TIME OPTIMIZATION OF 3D POINT CLOUD LLM VIA MANIFOLD-AWARE IN-CONTEXT GUIDANCE AND REFINEMENT

Anonymous authors
Paper under double-blind review

ABSTRACT

Multimodal Large Language Models (MLLMs) have demonstrated impressive capabilities in textual and 2D visual reasoning, yet their ability to understand and reason over 3D data remains limited. The issues become more challenging for understanding standalone 3D point cloud due to the high interclass confusion. In this work, we propose **P**oint-**G**raph **LLM** (PGLLM), a framework that enables more effective 3D point cloud understanding by integrating in-context prompting and score refinement at test-time, respecting supporting data manifold. Our method first employs a pre-trained point cloud encoder which are used to construct a graph where edges encode visual similarity. Each support point cloud sample is converted to a textual caption via pre-trained PointLLM. For a test query, the graph is used to retrieve relevant neighbors whose captions serve as contextual demonstrations for a second stage LLM for final reasoning, a process we term in-context guidance. Furthermore, we introduce a confidence score refinement mechanism based on label propagation to enhance the reliability of LLM predictions for classification and out-of-distribution (OOD) detection tasks. All above optimizations are carried out fully at test-time. Extensive experiments across diverse 3D datasets and tasks demonstrate that PGLLM consistently improves accuracy and robustness over prior baselines with very almost no additional computation cost, showcasing a promising direction toward native 3D reasoning with MLLMs.

1 INTRODUCTION

While multimodal large language models (MLLMs) have revolutionized textual and 2D visual reasoning, their ability to interpret and reason about 3D environments remains fundamentally limited. Recent efforts Tang et al. (2024); Qi et al. (2024a) have explored 3D understanding with MLLMs by equipping them with mechanisms to perceive 3D information, often through auxiliary modalities or intermediate representations. Typically, these approaches operate by either projecting 3D point clouds into 2D images Zhu et al. (2023) or by piping pre-extracted features into the language model Xu et al. (2024); Guo et al. (2023). While these pipelines demonstrate promising results, they fall short in granting MLLMs direct access to the underlying geometric structure of 3D data. Consequently, enabling MLLMs to natively process and reason over rich 3D point cloud information remains an open and challenging research question.

A notable step in this direction is PointLLM (Xu et al., 2024), which introduces a framework capable of understanding colored 3D object point clouds in response to human instructions. PointLLM fuses geometric, appearance, and linguistic information by coupling a point cloud encoder with a pre-trained LLM such as LLaMA (Touvron et al., 2023). To handle downstream tasks such as classification or captioning, PointLLM adopts a two-stage pipeline, where the PointLLM first generates a textual description, and a second stage LLM interprets this output for final task-specific predictions. While this method demonstrates competitive performance, it faces a key limitation, high inter-class visual similarity in 3D point clouds often leads to confusion, particularly when each point cloud is interpreted in isolation. As a result, models may struggle to distinguish fine-grained differences between closely related categories, leading to reduced reliability.

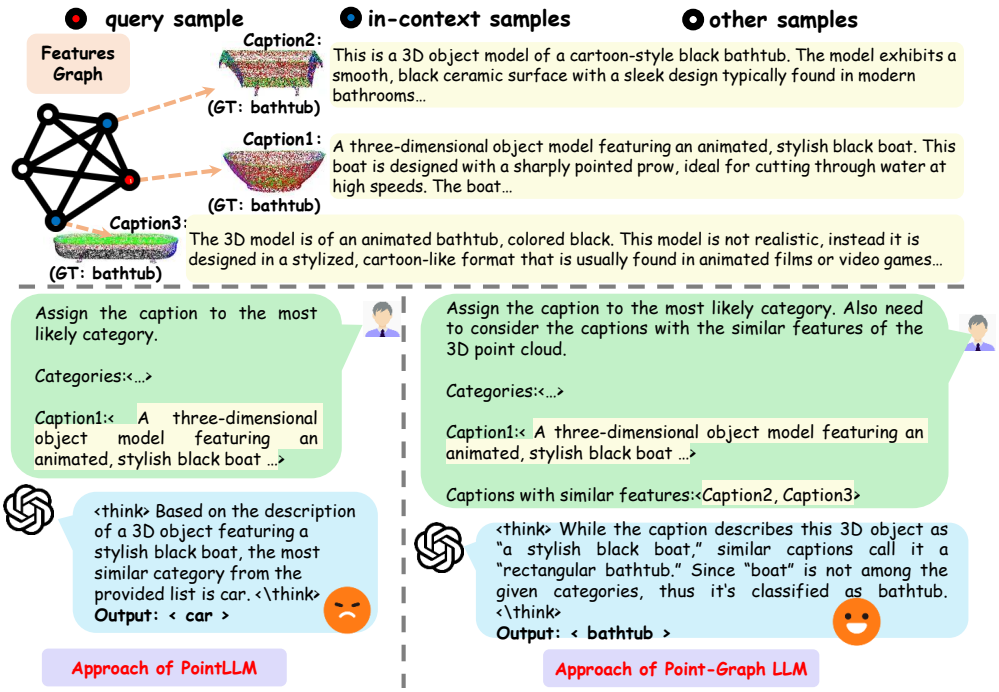


Figure 1: Manifold-aware in-context guidance leverages the 3D captions of adjacency nodes as demonstration for downstream understanding tasks.

To overcome this limitation, we draw inspiration from the recent success of In-Context Learning (ICL) (Agarwal et al., 2024; Li et al., 2023a; Brown et al., 2020), which enables LLMs to generalize to novel tasks by conditioning on a small set of demonstrations provided in the prompt. ICL has shown strong performance across various domains, including multi-modal settings (Tsimpoukelli et al., 2021; Jiang et al., 2024b; Huang et al., 2024), where LLMs are guided by examples combining visual example and text descriptions. The effectiveness of ICL crucially depends on the availability of informative and task-relevant demonstrations, e.g. image-text pairs. Selecting relevant and conducive demonstrations thus remain a open challenge.

To address this challenge, we propose an in-context learning strategy that leverages the manifold structure of unlabeled supporting data, which may be maintained independently or drawn from the test set. We represent this manifold as a graph, where each node corresponds to a 3D point cloud sample. To enrich each node, we employ PointLLM to generate a caption using the standard prompt "*What is this?*". The affinity between nodes is computed based on feature similarity in the embedding space of a pre-trained 3D encoder (Yu et al., 2021a). At inference time, for a given query point cloud, we identify its neighboring nodes in the graph and append their corresponding 3D captions to the query prompt. This augmented prompt is then passed to the second-stage LLM (e.g., ChatGPT), enabling it to perform context-aware reasoning. We refer to this mechanism as in-context guidance, which injects semantically relevant knowledge into the LLM's reasoning process at test time, without requiring model retraining.

Moreover, for classification-related downstream tasks such as recognition and out-of-distribution (OOD) detection, the confidence calibration of LLM outputs is critical for robust decision-making (Xiao et al., 2022). Relying solely on raw predictions from LLMs may be risky, especially when predictions are overconfident or miscalibrated. To address this, we further tap on the manifold build upon the supporting data and employ a refinement step, where predicted confidences are used to smooth and correct noisy labels on the graph. This is implemented via a lightweight label propagation algorithm (Zhu & Ghahramani, 2002), allowing the model to refine its predictions by considering the consistency and confidence of nearby nodes. In this way, even anecdotal or ambiguous predictions on individual samples can be improved by leveraging the collective structure of the data.

In summary, we propose a novel 3D point cloud understanding framework **P**oint-**G**raph **LLM** that integrates in-context learning, graph-based reasoning, and confidence-aware label propagation. By bridging the gap between geometric perception and LLM reasoning, our approach enables more accurate, interpretable, and robust 3D understanding. The main contributions are summarized as follows.

- We propose an in-context guidance mechanism to optimize the effectiveness of 3D point cloud LLM for downstream tasks. This approach leverages the test-time data manifold to construct helpful demonstrations to enrich the prompt.
- We introduce a score-based inference mechanism that further improves the performance of LLMs on classification related downstream tasks by refining the initial predictions on individual samples.
- We conduct extensive experiments on multiple downstream tasks and diverse 3D point cloud datasets, demonstrating consistent performance gains over existing baselines across all settings.

2 Related Work

Large Language Models for 3D Understanding. Large Language Models (LLMs) have demonstrated remarkable performance across a wide range of natural language and 2D vision tasks (Brown et al., 2020; Achiam et al., 2023; Touvron et al., 2023). Recently, there has been growing interest in extending LLMs to 3D understanding (Hong et al., 2023; Qi et al., 2024a;b; Yang et al., 2025; Xu et al., 2024; Guo et al., 2023; Yuan et al., 2025). 3D-LLM (Hong et al., 2023) introduces a family of LLMs grounded in physical 3D reasoning, laying the foundation for language-guided 3D perception. Subsequent approaches such as PointLLM (Xu et al., 2024) and Point-Bind (Guo et al., 2023) directly process colored object point clouds by combining point cloud encoders with pre-trained LLMs, enabling open-vocabulary 3D understanding. ShapeLLM (Qi et al., 2024a) pioneers the use of multi-view distillation and introduces the ReCon++ encoder, establishing a 3D multi-modal evaluation benchmark (3D MM-Vet) to unify embodied 3D interaction tasks. GPT4Point (Qi et al., 2024b) extends the capabilities of LLMs to handle point cloud captioning and visual question answering. Similarly, LiDAR-LLM (Yang et al., 2025) focuses on outdoor scene understanding by integrating LiDAR data with LLMs for large-scale 3D reasoning. While these methods demonstrate the potential of LLMs for 3D tasks, they typically treat each point cloud in isolation and do not fully exploit the structure of the data manifold.

In-Context Learning. In-context learning (ICL)(Brown et al., 2020) enables large language models to perform downstream tasks by conditioning on a set of demonstrations, without any parameter updates. Initially developed for natural language tasks(Zhang et al., 2022; Li et al., 2023b), ICL has since been extended to multi-modal domains. For instance, Flamingo (Alayrac et al., 2022) adapts ICL to vision-language tasks by incorporating cross-modal attention, while many-shot prompting strategies (Jiang et al., 2024b; Huang et al., 2024) have been shown to significantly enhance the effectiveness of ICL in image classification and question answering. These works demonstrate that with appropriate context, LLMs can generalize across diverse tasks in a flexible and label-efficient manner. However, applying ICL to 3D understanding remains underexplored, particularly in scenarios where constructing relevant demonstrations is non-trivial due to limited supervision.

Manifold Learning for Visual Tasks. Manifold learning has been widely adopted to enhance visual recognition tasks by modeling the intrinsic geometry of data distributions. Graph-based methods (Li et al., 2025; Stojnić et al., 2024; Chen et al., 2025) leverage similarities in feature space to propagate high-confidence labels or scores, improving performance in zero-shot and few-shot settings. For instance, label propagation techniques have been combined with graph neural networks (Bao et al., 2024; Stadler et al., 2021; Wu et al., 2023) to improve the separation between in-distribution (ID) and out-of-distribution (OOD) samples. Beyond classification, graph-based selection has also been applied in 3D domains, as demonstrated in GraphI2Ps (Bie et al., 2025), which filters false matches during point cloud registration via neighborhood pruning. While these methods focus on traditional backbones or vision-language models, our work is the first to integrate manifold learning into the in-context learning process of LLMs, enabling context-aware reasoning over 3D point clouds by structurally selecting and organizing prompts from the data manifold.

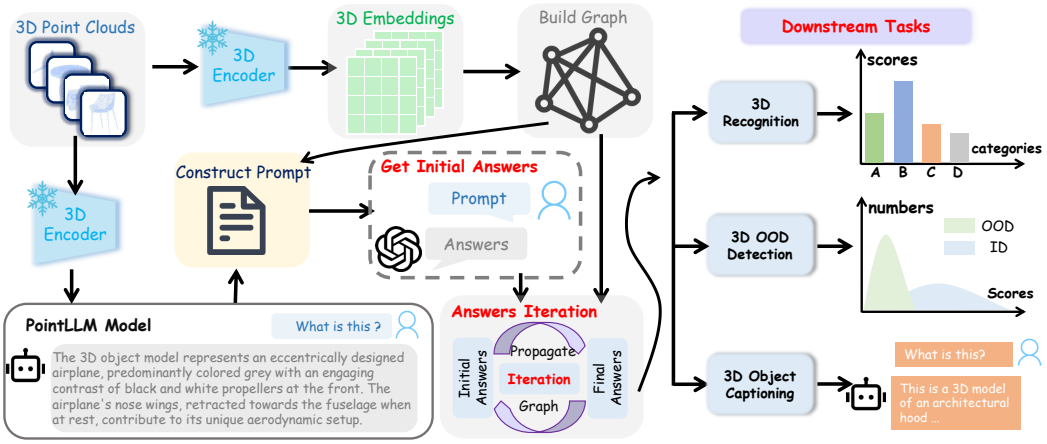


Figure 2: Overview of the proposed framework for PGLLM. After encoding the 3D test samples, the framework feeds them into PointLLM for caption generation and uses them to construct a KNN graph. Initial answers are then synthesized via LLM inference. Subsequently, leveraging relational structures within the KNN graph, we introduce an answer iteration mechanism to optimize performance on downstream tasks.

3 Methodology

3.1 Preliminaries: PointLLM and 3D Captioning

We begin by formalizing the setup. Let $\mathcal{D}_u = \{x_i\}_{i=1}^{N_u}$ denote an unlabeled 3D point cloud support dataset with l categories, $\mathcal{D}_u$ may be the testing dataset or any unlabeled reference dataset. Let f_p be a pre-trained point cloud encoder and we apply f_p to the dataset yields a sequence of point cloud features:

$$\mathcal{P} = \{p_1, p_2, \ldots, p_{N_u}\}, \quad p_i = f_p(x_i).$$

For each sample x_i, we use the default prompt "*What is this?*" with PointLLM (Xu et al., 2024) to generate a textual caption c_i, resulting in a caption set, $\mathcal{C} = \{c_1, c_2, \ldots, c_{N_u}\}$. These captions serve as high-level semantic descriptions of the point clouds and form the basis for further in-context learning and downstream tasks.

3.2 In-Context Guidance via Graph Neighbor Retrieval

In-context learning (ICL) allows LLMs to perform downstream tasks by conditioning on a set of relevant demonstrations embedded within the input prompt. To construct effective in-context guidance for each test sample, we build a graph $G = (V, E)$, where each node $v_i \in V$ represents a point cloud x_i, and edges $e_{ij} \in E$ encode pairwise similarity between samples. Specifically, we compute the cosine similarity between point cloud features p_i and p_j, and define symmetric the edge weight matrix $W \in \mathbb{R}^{N_u \times N_u}$ following a K-Nearest Neighbors (KNN) criterion,

$$W_{ij} = \begin{cases} e_{ij} & \text{if } e_{ij} \in \text{Top}_K(\{e_{ij}\}_{j=1}^{N_u}) \\ 0 & \text{otherwise} \end{cases}, \quad s.t. \quad e_{ij} = \frac{< p_i, p_j >}{||p_i|| \cdot ||p_j||} \tag{1}$$

For each query sample x_i, we retrieve its K nearest neighbors on the graph, yielding a neighbor set $\mathcal{X}_i = \{x_{i_1}, \ldots, x_{i_K}\}$ and the corresponding caption set $\mathcal{C}_i = \{c_{i_1}, \ldots, c_{i_K}\}$. These captions serve as in-context demonstrations appended to the prompt, illustrated in Fig. 1 as "*Caption2*", "*Caption3*", etc. This enables the LLM to reason about the query with reference to structurally similar samples. New query samples not attached to the supporting graph can be integrated into the graph without much computing overhead following the dynamic graph expansion scheme (Li et al., 2025).

3.3 SCORE REFINEMENT VIA LABEL PROPAGATION

Unlike conventional classification methods that directly predict class labels, we guide the LLM to output class confidence scores for each 3D caption. This score-based formulation enhances robustness and enables downstream tasks such as OOD detection.

3D Recognition: Given a 3D caption c_i, we prompt the LLM to output a per-category score, $S_l^{(i)} \in \mathbb{R}^l$, where each element reflects the confidence of respective class label. Aggregating across the dataset yields the initial score matrix, $S_0 \in \mathbb{R}^{l \times N_u}$. To refine predictions by leveraging geometric similarity among point clouds, we apply label propagation (Zhu & Ghahramani, 2002) over the graph W. Let S_t denote the refined score matrix at iteration t. The update rule is:

$$S_t = \alpha S_{t-1}\tilde{W} + (1-\alpha)S_0, \quad \tilde{W} = D^{-\frac{1}{2}}WD^{-\frac{1}{2}}, \quad D = \text{diag}(\sum_j W_{ij}),$$
$$\hat{\mathbf{y}} = \arg\max_i S_t, \tag{2}$$

where $\hat{\mathbf{y}}$ gives the final predicted class for each sample, and α controls the balance between the initial LLM output and propagated scores.

3D OOD Detection: For OOD detection, we prompt the LLM to produce a single confidence score $S(x_i) \in \mathbb{R}$ for each caption, indicating its similarity to the known in-distribution classes. A threshold δ is used to determine OOD status:

$$\hat{\mathbf{y}} = \begin{cases} \text{OOD} & \text{if } S(x_i) \leq \delta, \\ \text{ID} & \text{otherwise.} \end{cases} \tag{3}$$

This scalar score can be smoothed through the same graph-based propagation mechanism as in Eq. 2 with $l = 2$.

3D Caption Refinement with In-Context Learning: For the 3D object captioning task, we enhance the semantic quality of the initial PointLLM-generated captions using in-context refinement. We reuse the same graph-based strategy to select neighboring samples with semantically and structurally relevant captions. These are appended as demonstrations to the input prompt for each query caption. Unlike recognition tasks, the goal here is caption correction rather than classification. We guide the LLM to preserve the original semantics but improve fluency and fix any factual errors. This leverages the LLM's generative ability to produce more accurate and natural descriptions.

3.4 FRAMEWORK OVERVIEW

An overview of PGLLM is presented in Fig. 2. First, we extract 3D features from the test set using a frozen point cloud encoder. These features are passed to PointLLM to generate initial captions. A KNN graph is then constructed over the feature space to capture local geometric relationships. Based on this graph, we identify structurally similar neighbors for each sample, whose captions are used as in-context demonstrations. Depending on the downstream task we summarize the test-time optimization practices as follows. An illustration is deferred to the Appendix.

- For 3D recognition, the LLM outputs class-wise scores, which are further refined via graph-based label propagation.
- For OOD detection, scalar similarity scores are computed, refined via graph-based label propagation and eventually thresholded to determine OOD status.
- For captioning, in-context refinement is applied to enhance the quality of generated text while maintaining semantic fidelity.

This unified framework leverages LLMs not only as language generators but also as structured reasoning engines capable of adapting to multiple 3D tasks with minimal supervision.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Dataset: We evaluate our method on four well-established 3D point cloud benchmarks. **ModelNet40** (Wu et al., 2015) contains 2,468 test 3D objects across 40 categories. **ShapeNetCore** (Chang et al., 2015) s a canonical subset of the full ShapeNet repository with 5,158 unique test models from 55 object categories. Following PointLLM (Xu et al., 2024), we sample 200 objects from **Objaverse** (Deitke et al., 2023) for testing. **S3DIS** (Armeni et al., 2016) provides semantically segmented 3D point clouds from indoor environments and we follow Chen et al. (2025) to select 8,931 point clouds with rich semantic annotations for evaluation. For OOD dataset partitioning, we follow the 3DOS protocol Alliegro et al. (2022) to divide ShapeNetCore into SN1, SN2, and SN3 subsets. Similarly, we partition ModelNet40 into MN1, MN2, and MN3 subsets. Further dataset details are provided in the supplementary material.

Implementation Details: We use Point-BERT (Yu et al., 2021b) as the 3D encoder and a pre-trained PointLLM-7B (Xu et al., 2024) to generate initial captions. We evaluate DeepSeek-V3 (Liu et al., 2024), Qwen-Plus (Yang et al., 2024) and GPT-4 Achiam et al. (2023) as the second stage LLMs. As for KNN Graph construction, we set the K-value of 3. For score propagation, we set the α to 0.5, and the number of iterations T to 5 in Eq. 2.

Competing Methods: We compare against existing LLM-based 3D understanding methods: InstructBLIP (Dai et al., 2023), LLaVA (Liu et al., 2023), 3D-LLM (Hong et al., 2023), Point-Bind LLM (Guo et al., 2023), ShapeLLM (Qi et al., 2024a), PointLLM (Xu et al., 2024), and MiniGPT-3D (Tang et al., 2024). Both PointLLM and MiniGPT-3D employs ChatGPT 4 as second stage LLM for classification, thus facilitating fair comparison. To the best of our knowledge, this is the first work to explore 3D OOD detection within an LLM framework. Therefore, we compare against several VLM-based zero-shot OOD baselines: MCM (Ming et al., 2022), NegLabel (Jiang et al., 2024a), ZLap (Kalantidis et al., 2024), and GSP (Chen et al., 2025). Finally, we evaluate two variants of PGLLM with different support set $\mathcal{D}_u$. When testing data distribution is available, i.e. transductive inference, we use all testing data as the support dataset and refer to the method as PGLLM$^{\text{T}}$. Alternative, we leverage an external dataset, Objaverse, to build the support dataset and refer to the method as PGLLM$^{\text{O}}$. Specifically, we randomly selected 100K samples and their corresponding captions from the 660K training data of Objaverse to build the graph. For both PGLLM$^{\text{O}}$ and PGLLM$^{\text{T}}$, PointLLM-7B is used to generate initial captions. All competing methods use the same 3D encoder as PointLLM (Xue et al., 2024).

Evaluation Metrics We use classification accuracy (ACC) for 3D recognition, and AUROC and FPR95 for 3D OOD detection, which are standard metrics in OOD evaluation. For 3D object captioning, we assess semantic alignment using Sentence-BERT (Reimers & Gurevych, 2019), SimCSE (Gao et al., 2021), and GPT-4 as evaluators.

Method	2nd Stage LLM	ModelNet40								ShapeNetCore							
		MN1		MN2		MN3		**Average**		SN1		SN2		SN3		**Average**	
		AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓
MCM(Ming et al., 2022)	–	85.3	53.6	80.2	74.2	77.5	72.6	81.0	66.8	85.1	51.6	83.2	46.1	66.4	75.8	78.2	57.8
NegLabel(Jiang et al., 2024a)	–	74.3	77.4	65.8	86.6	61.5	81.9	67.2	82.0	60.6	87.8	80.6	78.6	88.0	48.3	76.4	71.6
ZLaP(Kalantidis et al., 2024)	–	72.8	99.8	86.1	61.1	70.8	76.0	76.6	79.0	88.2	52.8	72.3	66.4	77.4	90.0	79.3	69.7
GSP(Chen et al., 2025)	–	82.4	77.0	77.9	65.4	76.1	76.5	78.8	73.0	**90.6**	**38.9**	70.7	64.0	79.7	93.7	80.4	65.5
PointLLM-7B(Xu et al., 2024)	GPT-4	84.0	100.0	82.1	100.0	74.0	100.0	80.0	100.0	80.1	100.0	88.8	100.0	94.1	92.2	87.7	97.4
PGLLM$^{\text{O}}$ (Ours)	GPT-4	87.3	56.6	86.2	44.3	79.2	60.8	84.3	53.9	79.7	55.8	90.9	41.3	96.0	26.5	88.9	41.2
PGLLM$^{\text{T}}$ (Ours)	GPT-4	**89.6**	**53.1**	**87.2**	**43.0**	**80.8**	**60.2**	**85.9**	**52.1**	81.8	52.4	**93.9**	**26.7**	97.6	**9.7**	**91.1**	**29.6**
PGLLM$^{\text{T}}$ (Ours)	DeepSeek-V3	86.4	70.0	83.6	59.1	76.2	68.2	82.1	65.8	83.8	62.3	91.7	36.4	97.1	18.6	90.9	39.1
PGLLM$^{\text{T}}$ (Ours)	Qwen-Plus	86.5	68.4	84.2	47.3	77.9	71.7	82.9	62.5	81.4	61.8	92.8	33.0	**97.9**	11.1	90.7	35.3
PGLLM$^{\text{T}}$ (Ours)	Qwen3-VL-8B	85.8	67.4	84.4	47.5	74.2	71.3	81.5	62.0	81.2	48.0	91.9	30.3	94.4	29.7	89.2	36.0
PGLLM$^{\text{T}}$ (Ours)	Llama3.1-8B	57.1	86.3	54.9	87.2	50.4	97.9	54.1	90.4	56.3	89.7	81.9	68.3	92.7	44.6	80.0	67.5
PGLLM$^{\text{T}}$ (Ours)	GPT-oss-20B	83.0	80.4	74.5	76.8	72.6	94.0	76.7	83.7	81.9	57.1	92.4	30.3	96.7	21.2	90.3	36.2

Table 1: Evaluation of 3D OOD detection on ModelNet40 and ShapeNetCore. **Bold** and underlined numbers denote the best and second-best results, respectively. Each "MNx" or "SNx" denotes the known class split and the rest are unknown.

4.2 EXPERIMENTAL RESULTS

3D OOD Detection: Tab. 1 summarizes the results of our comprehensive experiments, highlighting the following key observations. **(i)** Our PGLLM$^{\text{T}}$ framework, when integrated with GPT-4, establishes new state-of-the-art results on the ModelNet40 benchmark. It achieves an outstanding 85.9% AUROC on average, outperforming the previous best method (MCM) by 4.9%, while also reduc-

Method	2nd Stage LLM	3D Recognition (I) ACC	3D Recognition (C) ACC	3D Recognition Average	3D Captioning GPT-4	3D Captioning S-BERT	3D Captioning SimCSE
3D-LLM(Hong et al., 2023)	-	-	-	-	33.4	44.5	43.7
Point-Blind(Guo et al., 2023)	-	51.9	39.7	45.8	-	-	-
ShapeLLM-7B(Qi et al., 2024a)	-	-	-	-	46.9	48.2	49.2
ShapeLLM-13B(Qi et al., 2024a)	-	-	-	-	49.0	48.5	50.0
InstructBLIP-7B(Dai et al., 2023)	GPT-4	19.5	31.5	25.5	45.3	47.4	48.5
InstructBLIP-13B(Dai et al., 2023)	GPT-4	26.0	31.4	28.7	45.0	45.9	48.9
LLaVA-7B(Liu et al., 2023)	GPT-4	39.7	39.7	39.7	46.7	45.6	47.1
LLaVA-13B(Liu et al., 2023)	GPT-4	37.1	36.1	36.6	38.3	46.4	45.9
MiniGPT-3D(Tang et al., 2024)	GPT-4	61.8	60.0	60.9	**57.1**	**49.5**	**51.4**
PointLLM-7B(Xu et al., 2024)	GPT-4	53.4	51.8	52.6	44.9	47.5	48.6
PointLLM-13B(Xu et al., 2024)	GPT-4	53.0	52.6	52.8	48.2	47.9	49.1
PGLLM^{O} (Ours)	GPT-4	53.1	53.0	53.1	49.1	48.4	48.9
PGLLM^{T} (Ours)	GPT-4	**63.1**	61.8	**62.5**	50.5	48.9	49.4
PGLLM^{T} (Ours)	DeepSeek-V3	62.6	**62.0**	62.3	-	-	-
PGLLM^{T} (Ours)	Qwen-Plus	43.1	41.7	42.4	-	-	-

Table 2: Comparison of results on 3D recognition (ModelNet40) and 3D captioning (Objaverse). Recognition performance is evaluated using two prompt types: an Instruction-type (I) prompt ("*What is this?*") and a Completion-type (C) prompt ("*This is an object of*").

ing the critical FPR95 metric to 52.1%. We notice that both PointLLM-7B and ours methods use ChatGPT 4 as the second stage LLM. **(ii)** On ShapeNetCore, PGLLM^{T} with GPT-4 demonstrates breakthrough performance, attaining 97.6% AUROC on SN3 and a remarkably low FPR95 of 9.7%. This reflects an 7.1% AUROC improvement over feature-based methods such as GSP. Overall, our framework achieves an average AUROC of 91.1% and an average FPR95 of 29.6%, setting new benchmarks across key metrics. These consistent gains across two datasets affirm the effectiveness of our graph-based mechanism. **(iii)** Because GPT-4 tends to assign either 0 or 100 when scoring test samples, the baseline performs very poorly on the FPR95 metric, with values almost always equal to 100.0. By introducing score propagation, our method effectively alleviates this issue and yields much smoother, more evenly distributed scores across all samples. **(iv)** Using GPT-4 as the second stage LLM yields notable improvements in AUROC with +3.8% on ModelNet40 and +0.2% on ShapeNetCore compared to DeepSeek-V3. The performance gap between GPT-4, Qwen-Plus and DeepSeek-V3 variants highlights the ability of our framework to harness stronger LLMs for enhanced 3D understanding. **(v)** With both testing data (transductive setting) and external dataset as supporting dataset, PGLLM$^{T/O}$ outperforms the baseline (PointLLM-7B). This suggest the robustness of the graph-based method.

3D Recognition: The results in Tab. 2 reveal several key insights. **i)** Our PGLLM^{T} framework achieves an average accuracy of 62.3% with DeepSeek-V3 and 62.5% with GPT-4, outperforming all existing state-of-the-art methods. In particular, it surpasses the strongest baseline, MiniGPT-3D, by +1.6% for average. **ii)** The framework shows strong robustness across different prompt types. Specifically, it outperforms MiniGPT-3D by 1.3% with the instruction prompt "What is this?" and by 1.8% with the completion prompt "This is an object of". This dual-prompt advantage highlights the adaptability of our architecture to diverse query formats. **iii)** Although all competing methods rely on GPT-4 for evaluation, our framework achieves 62.3% average accuracy even with DeepSeek-V3 (much lower per token cost than GPT-4) as second stage LLM, surpassing all GPT-4-based baselines. This suggests lower cost LLMs may achieve comparable performance. However, Qwen-Plus demonstrates notably lower performance on the 3D Recognition task, primarily due to its limited ability to generate long numerical sequences, which adversely affects its overall results.

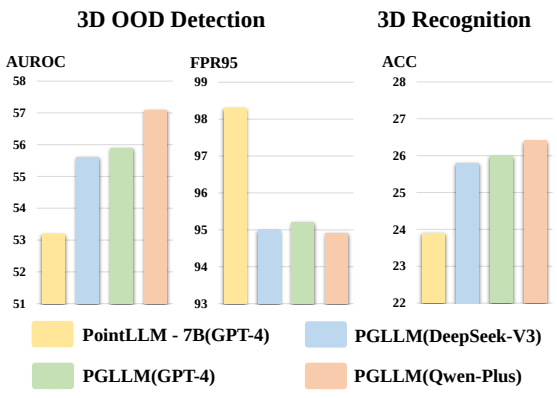


Figure 3: Results on real-world benchmark S3DIS. We report 3D OOD detection and 3D recognition tasks.

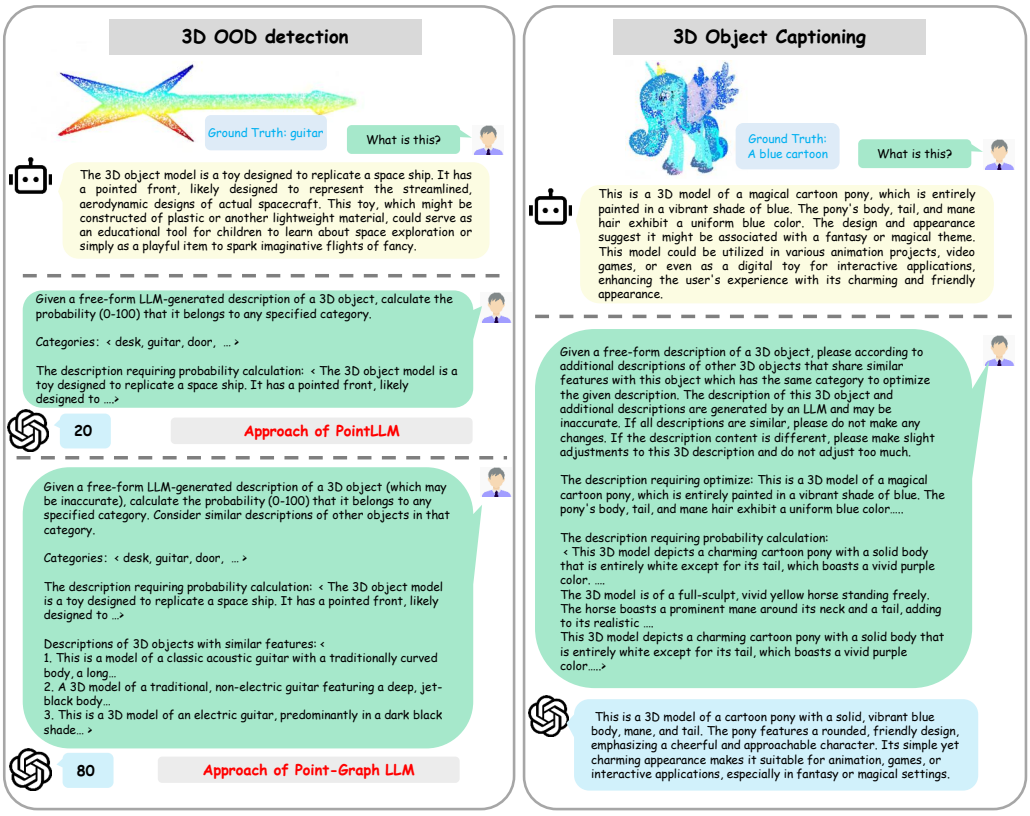
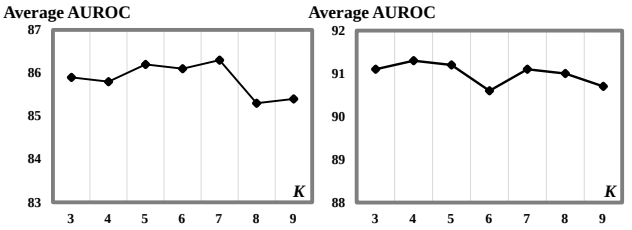


Figure 5: Qualitative examples of prompt questions and PGLLM's predictions.

3D Object Captioning: The results in Tab. 2 show that our method performs competitively on the 3D object captioning task, consistently outperforming the PointLLM-7B baseline across all three evaluation metrics. Since the evaluation relies on subjective judgments from an LLM, PGLLM uses GPT-4 exclusively for assessment to ensure a consistent evaluation standard. While our approach does not yet reach state-of-the-art performance, this gap is likely attributable to the limited size of the test dataset. In low-data regimes, constructing optimal graph structures is more challenging, which can constrain performance. Nevertheless, the observed improvements highlight the promise of our graph-based in-context guidance strategy for 3D captioning tasks. This suggests careful selection of second stage LLM is still necessary.

Experiments on Real-world Benchmark: Most existing 3D LLMs Xu et al. (2024); Guo et al. (2023); Tang et al. (2024) have been evaluated primarily on synthetic CAD datasets, leaving a validation gap in real-world scenarios. To address this, we conduct experiments on the real-world S3DIS dataset. As shown in Fig. 3, we evaluate both PointLLM and PGLLM on 3D out-of-distribution (OOD) detection and 3D recognition tasks. The results demonstrate that our method consistently outperforms baseline approaches, confirming the effectiveness of our graph-based strategy in practical settings. Notably, PGLLM remains compatible with lower-cost second-stage LLMs, achieving performance that is comparable, or even superior to higher-cost alternatives.



(a) 3D OOD detection on ModelNet40 (b) 3D OOD detection on ShapeNetCore

Figure 4: Different number of K-values for 3D OOD detection on two datasets.

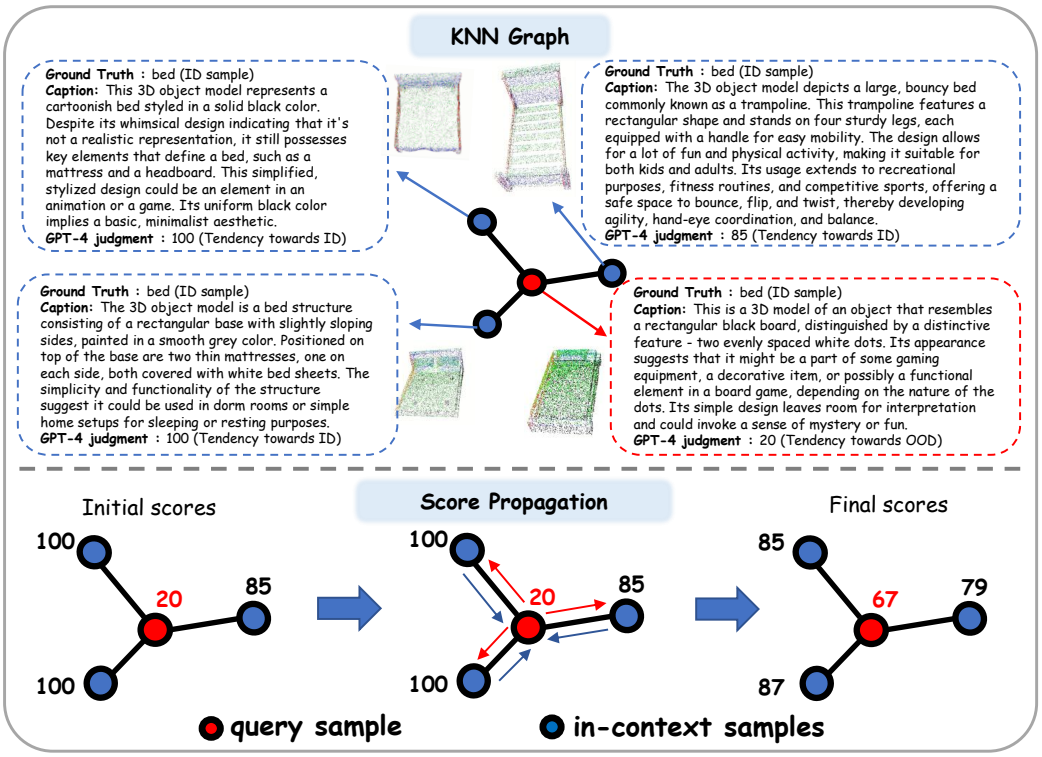


Figure 6: Qualitative example demonstrating the effectiveness of score propagation.

In-context Guidance	Score Propagation	ModelNet40 ACC↑	ModelNet40 AUROC↑	ModelNet40 FPR95↓	ShapeNetCore ACC↑	ShapeNetCore AUROC↑	ShapeNetCore FPR95↓
-	-	52.5	80.4	100.0	55.5	88.2	54.9
•	-	59.7	83.3	100.0	60.7	89.2	47.2
✓	-	60.2	83.1	100.0	61.0	89.5	46.0
-	✓	56.7	83.5	62.0	59.3	89.8	44.7
✓	✓	63.1	85.9	52.1	62.4	91.1	29.6

Table 3: Ablation study on two datasets. ACC refers to the results of 3D recognition experiments, while AUROC and FPR95 correspond to the OOD detection experiments. Both AUROC and FPR95 represent averages across all subsets of the ModelNet40 and ShapeNetCore datasets. The • denotes in-context guidance derived through direct nearest-sample retrieval without graph.

4.3 Ablation Study

We conduct an ablation study with results summarized in Tab. 3. All experiments use ChatGPT4 as the second-stage LLM and are conducted on the ModelNet40 and ShapeNetCore datasets for both 3D recognition and 3D OOD detection. We begin with a baseline that feeds the generated 3D caption of each query directly into the LLM without additional context. We also evaluate a variant where the query retrieves its nearest neighbors via KNN ($K = 3$) to form the in-context prompt, bypassing graph construction. This variant yields results comparable to the graph-based method, likely because both retrieve highly similar neighbors due to the small K. However, omitting the graph structure prevents the use of score propagation for further refinement. Introducing in-context guidance alone yields significant performance gains over the baseline, demonstrating its effectiveness in injecting relevant contextual cues into the LLM. In contrast, score propagation alone provides only modest improvements. Notably, combining in-context guidance with score propagation leads to substantial performance boosts across both tasks, underscoring their complementary and synergistic effects.

4.4 Further Analysis

Qualitative Analysis: Fig. 5 showcases qualitative examples of PGLLM on 3D OOD detection and 3D object captioning tasks. For 3D OOD detection, we observe that PointLLM's reliance on single-caption inputs leads to fragile predictions when faced with ambiguous or hard-to-interpret

Dataset	Caption Generation	Graph Construction	GPT (OOD) Inference	GPT(Recog) Inference	GPT(Caption) Inference	Score(OOD) Propagation	Score(Recog) Propagation
ModeNet40	2630ms	51μs	1951ms	2409ms	-	41μs	43μs
ShapeNetCore	2750ms	65μs	2007ms	2688ms	-	28μs	29μs
S3DIS	2510ms	81μs	1763ms	1980ms	-	31μs	34μs
Objaverse	2690ms	30μs	-	-	3016ms	-	-

Table 4: Time consumption of each step for a per-sample, where ms denotes millisecond and μs denotes microsecond

point clouds. Our in-context guidance mitigates this by providing captions from structurally similar samples, allowing the LLM to leverage contextual cues for more robust classification. For 3D object captioning, PointLLM generates captions exhibiting inaccuracies or redundancies. By incorporating in-context guidance, our method enables the LLM to refine these outputs based on contextual similarity, resulting in more accurate and semantically rich descriptions. Fig. 6 qualitatively illustrates the process of score propagation. It can be observed that when the 3D LLM generates a low-quality caption for a given sample, it directly leads to a poor score in the second-stage evaluation. However, by employing the score propagation method, this can be effectively corrected by leveraging the accurate scores from neighboring samples of the query sample.

Impact of In-Context Example Quantity: To examine the effect of in-context guidance, we vary K, the number of retrieved captions provided to the LLM, and evaluate 3D OOD detection on ModelNet40 and ShapeNetCore (Fig. 4). Performance improves as K increases, reaching a dataset-specific peak at $K = 7$ for ModelNet40 and $K = 4$ for ShapeNetCore. Beyond these values, accuracy declines, likely due to the inclusion of irrelevant or misleading examples that confuse the LLM. These results underscore the importance of selecting an appropriate number of in-context examples for optimal performance.

Inference Time: Tab. 4 presents the time consumption of each step in PGLLM across different datasets. Overall, our method introduces minimal overhead. Processing each sampe requires only 30-90μs for a per-sample during graph construction and 30-50μs for score propagation, while caption generation and GPT inference cost about several seconds the per-sample time. Notably, even with graph-based processing, the total additional time per sample remains negligible, demonstrating that PGLLM maintains high computational efficiency and is practical for real-world deployment.

5 Conclusion

In this work, we introduce PGLLM, a novel framework for 3D point cloud understanding. Our approach constructs a graph to retrieve structurally similar 3D captions for each point cloud, using them as in-context examples to guide the LLM toward more informed reasoning. In parallel, a score-based refinement mechanism leverages the intrinsic structure of the test data manifold to enhance prediction accuracy. PGLLM achieves competitive results across a range of downstream tasks and sets new state-of-the-art performance in both 3D out-of-distribution (OOD) detection and 3D recognition. Extensive experiments on diverse 3D point cloud datasets demonstrate the robustness and generalizability of our method. We believe this work offers a promising direction for advancing 3D point cloud understanding and integrating LLMs into spatial perception tasks.

6 Reproducibility Statement

We are committed to ensuring the reproducibility of our results. The full implementation code, along with scripts to reproduce all experimental results, will be released upon acceptance. All model architectures, hyperparameters, and inference details are explicitly stated in Experiments. We believe these resources are sufficient to replicate our findings.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Rishabh Agarwal, Avi Singh, Lei Zhang, Bernd Bohnet, Luis Rosias, Stephanie Chan, Biao Zhang, Ankesh Anand, Zaheer Abbas, Azade Nova, et al. Many-shot in-context learning. *Advances in Neural Information Processing Systems*, 2024.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 2022.

Antonio Alliegro, Francesco Cappio Borlino, and Tatiana Tommasi. 3dos: Towards 3d open set learning-benchmarking and understanding semantic novelty detection on point clouds. *Advances in Neural Information Processing Systems*, 2022.

Iro Armeni, Ozan Sener, Amir R. Zamir, Helen Jiang, Ioannis Brilakis, Martin Fischer, and Silvio Savarese. 3d semantic parsing of large-scale indoor spaces. In *2016 IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

Tianyi Bao, Qitian Wu, Zetian Jiang, Yiting Chen, Jiawei Sun, and Junchi Yan. Graph out-of-distribution detection goes neighborhood shaping. In *Forty-first International Conference on Machine Learning*, 2024.

Lin Bie, Shouan Pan, Siqi Li, Yining Zhao, and Yue Gao. Graphi2p: Image-to-point cloud registration with exploring pattern of correspondence via graph learning. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 2020.

Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015.

Tiankai Chen, Yushu Li, Adam Goodge, Fei Teng, Xulei Yang, Tianrui Li, and Xun Xu. Exploiting vision language model for training-free 3d point cloud ood detection via graph score propagation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. *Advances in neural information processing systems*, 2023.

Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. Simcse: Simple contrastive learning of sentence embeddings. *ArXiv*, 2021.

Ziyu Guo, Renrui Zhang, Xiangyang Zhu, Yiwen Tang, Xianzheng Ma, Jiaming Han, Kexin Chen, Peng Gao, Xianzhi Li, Hongsheng Li, et al. Point-bind & point-llm: Aligning point cloud with multi-modality for 3d understanding, generation, and instruction following. *arXiv preprint arXiv:2309.00615*, 2023.

Yining Hong, Haoyu Zhen, Peihao Chen, Shuhong Zheng, Yilun Du, Zhenfang Chen, and Chuang Gan. 3d-llm: Injecting the 3d world into large language models. *Advances in Neural Information Processing Systems*, 2023.

Brandon Huang, Chancharik Mitra, Leonid Karlinsky, Assaf Arbelle, Trevor Darrell, and Roei Herzig. Multimodal task vectors enable many-shot multimodal in-context learning. *Advances in Neural Information Processing Systems*, 2024.

Xue Jiang, Feng Liu, Zhen Fang, Hong Chen, Tongliang Liu, Feng Zheng, and Bo Han. Negative label guided OOD detection with pretrained vision-language models. In *The Twelfth International Conference on Learning Representations*, 2024a.

Yixing Jiang, Jeremy Irvin, Ji Hun Wang, Muhammad Ahmed Chaudhry, Jonathan H Chen, and Andrew Y Ng. Many-shot in-context learning in multimodal foundation models. *arXiv preprint arXiv:2405.09798*, 2024b.

Yannis Kalantidis, Giorgos Tolias, et al. Label propagation for zero-shot classification with vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Mukai Li, Shansan Gong, Jiangtao Feng, Yiheng Xu, Jun Zhang, Zhiyong Wu, and Lingpeng Kong. In-context learning with many demonstration examples. *arXiv preprint arXiv:2302.04931*, 2023a.

Xiaonan Li, Kai Lv, Hang Yan, Tianyang Lin, Wei Zhu, Yuan Ni, Guotong Xie, Xiaoling Wang, and Xipeng Qiu. Unified demonstration retriever for in-context learning. *Association for Computational Linguistics*, 2023b.

Yushu Li, Yongyi Su, Adam Goodge, Kui Jia, and Xun Xu. Efficient and context-aware label propagation for zero-/few-shot training-free adaptation of vision-language model. In *International Conference on Learning Representations*, 2025.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 2023.

Yifei Ming, Ziyang Cai, Jiuxiang Gu, Yiyou Sun, Wei Li, and Yixuan Li. Delving into out-of-distribution detection with vision-language representations. *Advances in neural information processing systems*, 2022.

Zekun Qi, Runpei Dong, Shaochen Zhang, Haoran Geng, Chunrui Han, Zheng Ge, Li Yi, and Kaisheng Ma. Shapellm: Universal 3d object understanding for embodied interaction. In *European Conference on Computer Vision*, 2024a.

Zhangyang Qi, Ye Fang, Zeyi Sun, Xiaoyang Wu, Tong Wu, Jiaqi Wang, Dahua Lin, and Hengshuang Zhao. Gpt4point: A unified framework for point-language understanding and generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024b.

Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Conference on Empirical Methods in Natural Language Processing*, 2019.

Maximilian Stadler, Bertrand Charpentier, Simon Geisler, Daniel Zügner, and Stephan Günnemann. Graph posterior network: Bayesian predictive uncertainty for node classification. *Advances in Neural Information Processing Systems*, 2021.

Vladan Stojnić, Yannis Kalantidis, and Giorgos Tolias. Label propagation for zero-shot classification with vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Yuan Tang, Xu Han, Xianzhi Li, Qiao Yu, Yixue Hao, Long Hu, and Min Chen. Minigpt-3d: Efficiently aligning 3d point clouds with large language models using 2d priors. In *Proceedings of the 32nd ACM International Conference on Multimedia*, 2024.

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

Maria Tsimpoukelli, Jacob L Menick, Serkan Cabi, SM Eslami, Oriol Vinyals, and Felix Hill. Multimodal few-shot learning with frozen language models. *Advances in Neural Information Processing Systems*, 2021.

Qitian Wu, Yiting Chen, Chenxiao Yang, and Junchi Yan. Energy-based out-of-distribution detection for graph neural networks. *arXiv preprint arXiv:2302.02914*, 2023.

Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015.

Yuxin Xiao, Paul Pu Liang, Umang Bhatt, Willie Neiswanger, Ruslan Salakhutdinov, and Louis-Philippe Morency. Uncertainty quantification with pre-trained language models: A large-scale empirical analysis. *arXiv preprint arXiv:2210.04714*, 2022.

Runsen Xu, Xiaolong Wang, Tai Wang, Yilun Chen, Jiangmiao Pang, and Dahua Lin. Pointllm: Empowering large language models to understand point clouds. In *European Conference on Computer Vision*, 2024.

Le Xue, Ning Yu, Shu Zhang, Artemis Panagopoulou, Junnan Li, Roberto Martín-Martín, Jiajun Wu, Caiming Xiong, Ran Xu, Juan Carlos Niebles, et al. Ulip-2: Towards scalable multimodal pre-training for 3d understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Qwen An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxin Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yi-Chao Zhang, Yunyang Wan, Yuqi Liu, Zeyu Cui, Zhenru Zhang, Zihan Qiu, Shanghaoran Quan, and Zekun Wang. Qwen2.5 technical report. *ArXiv*, 2024.

Senqiao Yang, Jiaming Liu, Renrui Zhang, Mingjie Pan, Ziyu Guo, Xiaoqi Li, Zehui Chen, Peng Gao, Hongsheng Li, Yandong Guo, et al. Lidar-llm: Exploring the potential of large language models for 3d lidar understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2025.

Xumin Yu, Lulu Tang, Yongming Rao, Tiejun Huang, and Jie Zhou. Pre-training 3d point cloud transformers with masked point modeling. in 2022 ieee. In *CVF Conference on Computer Vision and Pattern Recognition*, 2021a.

Xumin Yu, Lulu Tang, Yongming Rao, Tiejun Huang, Jie Zhou, and Jiwen Lu. Point-bert: Pre-training 3d point cloud transformers with masked point modeling. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021b.

Zhihao Yuan, Yibo Peng, Jinke Ren, Yinghong Liao, Yatong Han, Chun-Mei Feng, Hengshuang Zhao, Guanbin Li, Shuguang Cui, and Zhen Li. Empowering large language models with 3d situation awareness. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025.

Yiming Zhang, Shi Feng, and Chenhao Tan. Active example selection for in-context learning. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 2022.

Xiangyang Zhu, Renrui Zhang, Bowei He, Ziyu Guo, Ziyao Zeng, Zipeng Qin, Shanghang Zhang, and Peng Gao. Pointclip v2: Prompting clip and gpt for powerful 3d open-world learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, 2023.

Xiaojin Zhu and Zoubin Ghahramani. Learning from labeled and unlabeled data with label propagation. *ProQuest number: information to all users*, 2002.

A APPENDIX

A.1 OVERVIEW OF THE TASKS

We illustrate the three tasks addressed in this work in Fig. 7. Our framework integrates 3D point clouds with large language models by first constructing prompts using PointLLM and a point cloud feature graph (Step 1), then leveraging LLMs to perform downstream tasks (Step 2). PointLLM generates free-form descriptions of 3D objects (yellow boxes), which are refined and utilized by the LLM for tasks including 3D out-of-distribution (OOD) detection, 3D recognition, and 3D object captioning.

A.2 DATASET PARTITION FOR 3D OOD DETECTION

We follow the benchmark 3DOS, GSP to construct ShapeNetCore and S3DIS datasets for OOD detection. For the ModelNet40 dataset, we follow the partitioning practice established in the 3DOS work, dividing it into three subsets based on ascending index order. The specific category partitioning details are presented in Tab 7 5 6. Qualitative visualizations of some dataset samples are presented in Fig. 8 10.

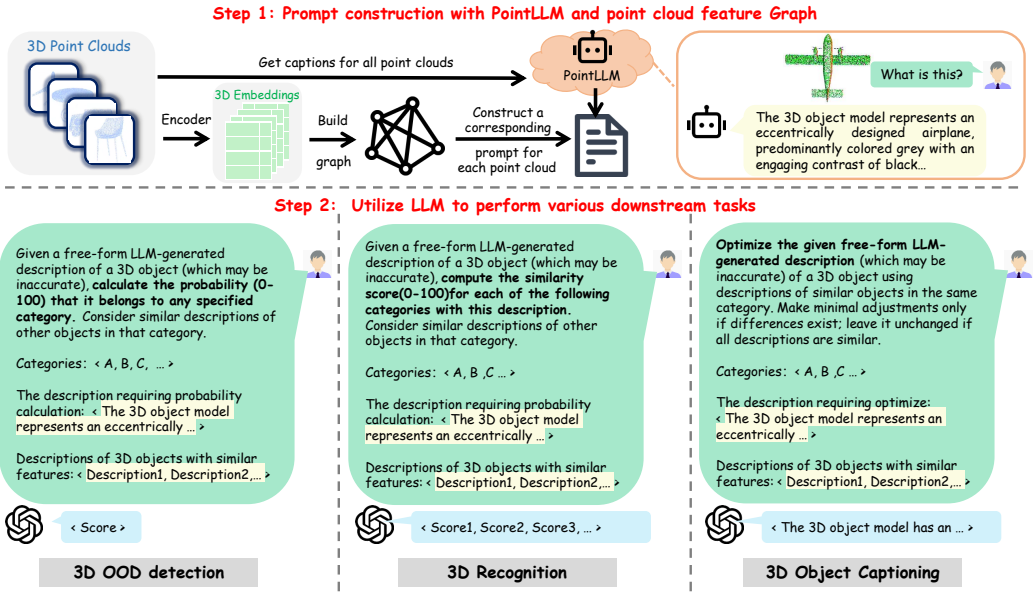


Figure 7: Demonstrations of PGLLM. We propose PGLLM, an efficient and potent framework that integrates 3D-LLMs with Large Language Models, where the text on a light yellow background indicates content generated by PointLLM. Furthermore, we demonstrate its operational mechanisms across 3D recognition, 3D OOD detection and 3D object captioning tasks.

SN1	mug, lamp, bed, washer, loudspeaker, telephone, dishwasher, camera, birdhouse, jar, bowl, bookshelf, stove, bench, display, keyboard, clock, piano
SN2	earphone, knife, chair, pillow, table, laptop, mailbox, basket, file cabinet, cabinet, sofa, printer, flowerpot, microphone, tower, bathtub, bag, trash bin
SN3	can, microwave, skateboard, faucet, train, guitar, pistol, helmet, watercraft, airplane, bottle, cap, rocket, rifle, remote, car, bus, motorbike

Table 5: For each distinct out-of-distribution (OOD) subset partition on the ShapeNetCore, the categories residing within a given subset are designated as in-distribution (ID), whereas categories from all other subsets are considered entirely OOD.

Figure 8: Visualization of the real-world benchmark S3DIS and the categories partition for OOD detection.

ID	window, door, table, chair, sofa, bookcase, clutter
OOD	beam, board, ceiling, column, floor, wall

Table 6: S3DIS dataset partitioning for OOD detection: foreground objects as ID and background objects as OOD.

A.3 Prompts and Qualitative Analysis

We provide the prompts for 3D recognition, 3D OOD detection: and 3D object captioning tasks in Tab 8. Furthermore, we provide supplementary visual analyses for 3D recognition and 3D OOD detection tasks in Fig. 11, 12, 13and 14, provided with comprehensive prompts to the LLM. As evidenced in Fig. 11, the query sample contains an erroneous caption, yet the collective accuracy of ”vase” annotations within the in-context guidance enables the LLM to yield the correct prediction.Fig. 13 illustrates a scenario where both the query sample and several in-context guidance captions contain misleading annotations. This induces the LLM to output a neutral score. Nevertheless, this outcome represents a some improvement over the erroneous predictions generated without in-context guidance. Despite the fact that the query sample's caption lacks direct information about the categories, as illustrated in Fig. 14, the captions of the in-context samples prove crucial for enabling the LLM to achieve a more accurate classification. However, as shown in Fig. 12, the LLM's output is inclined to classify the object as a ”lamp,” which contradicts the ground-truth label of ”faucet.” This discrepancy is likely attributable to the presence of the word ”lights” in the query sample's caption. Despite the in-context samples providing captions containing ”faucet,” the LLM ultimately assigned a full score of 100 to the ”lamp” category, compared to only 60 for ”faucet.” In Fig. 15, we present examples where in-context guidance exerts a negative impact. It can be observed that when the caption generated for the query sample is relatively accurate, but the captions of the in-context samples contain certain inaccuracies, the second-stage scoring becomes biased.

MN1	airplane, bathtub, bed, bench, bookshelf, bottle, bowl, car, chair, cone, cup, curtain, desk
MN2	door, dresser, flower pot, glass box, guitar, keyboard, lamp, laptop, mantel, monitor, night stand, person, piano
MN3	plant, radio, range hood, sink, sofa, stairs, stool, table, tent, toilet, tv stand, vase, wardrobe, xbox

Table 7: For each distinct out-of-distribution (OOD) subset partition on the ModelNet40, the categories residing within a given subset are designated as in-distribution (ID), whereas categories from all other subsets are considered entirely OOD.

A.4 IMPACT OF K-VALUES

To further explore the influence of K-values, we test different number of K on 3D recognition task. The results are shown in Fig. 9, it can be observed that the optimal performance is achieved at K=5 for ModelNet40 and K=6 for ShapeNet, respectively. Furthermore, performance exhibits a declining trend with increasing K-values, a finding empirically consistent with the conclusions presented in our prior submission.

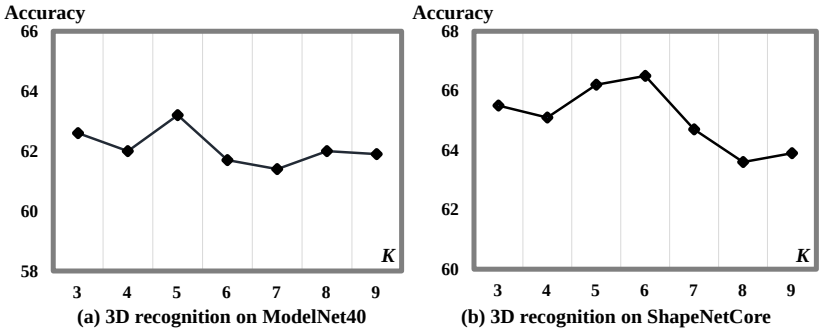


Figure 9: Different number of K-values for 3D recognition on two datasets.

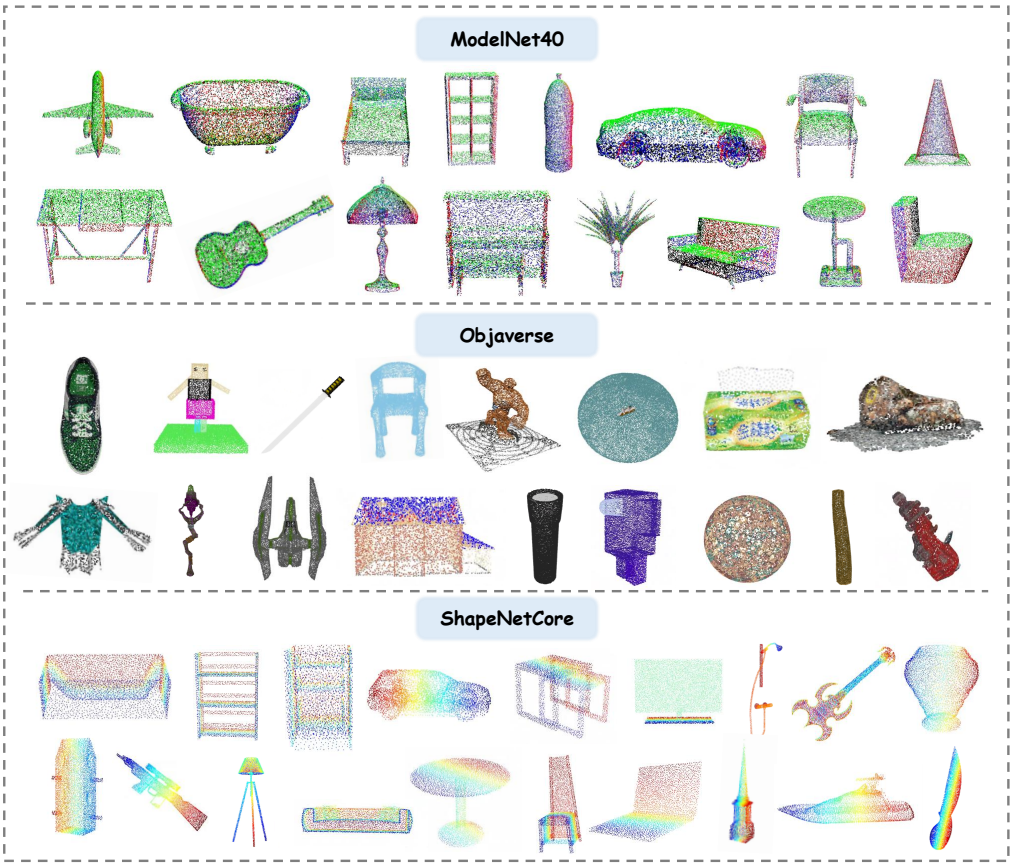


Figure 10: Visualization of the ModelNet40, Objaverse, and ShapeNetCore.

Tasks	Prompts
3D Recognition	Given a free-form description of a 3D object, the content described here belongs to one of the following 54 categories. Use this description to compute a similarity score (0-100) for each of the following 54 categories. The description of this 3D object is generated by an LLM and may be inaccurate. In addition, I will provide you with descriptions of other 3D objects that share similar features with this object of this category. 0=no relation, 100=perfect match. categories: mug, lamp, bed, washer, loudspeaker, telephone, dishwasher, camera, birdhouse, jar, bowl, bookshelf, stove, bench, display, keyboard, clock, piano, earphone, knife, chair, pillow, table, laptop, mailbox, basket, file cabinet, cabinet, sofa, printer, flowerpot, microphone, tower, bathtub, bag, trash bin, can, microwave, skateboard, faucet, train, guitar, pistol, helmet, watercraft, airplane, bottle, cap, rocket, rifle, remote, car, bus, motorbike. 3D object description: ... (# generated by PointLLM) Descriptions of 3D objects with similar features: (# generated by PointLLM and seleted by mainfold learning) 1. ... 2. ... 3. ... Please output the 54 corresponding similarity scores in the order of the above-mentioned categories, without any additional explanation.
3D OOD detection	Given a free-form description of a 3D object, please calculate the probability (0-100) that the content described in the following text pertains to any of the following categories. The description of this 3D object is generated by an LLM and may be inaccurate. In addition, I will provide you with descriptions of other 3D objects that share similar features with this object of this category. You need to take these similar 3D model descriptions into account as well. 0=no relation, 100=perfect match. categories: plant, radio, range hood, sink, sofa, stairs, stool, table, tent, toilet, tv stand, vase, wardrobe, xbox. 3D object description: ... (# generated by PointLLM) Descriptions of 3D objects with similar features: (# generated by PointLLM and seleted by mainfold learning) 1. ... 2. ... 3. ... Output only a numerical score. Do not provide additional explanations.
3D Object Captioning	Given a free-form description of a 3D object, please according to additional descriptions of other 3D objects that share similar features with this object which has the same category to optimize the given description. The description of this 3D object and additional descriptions are generated by an LLM and may be inaccurate. If all descriptions are similar, please do not make any changes. If the description content is different, please make slight adjustments to this 3D description and do not adjust too much. The description requiring optimize: ... (# generated by PointLLM) Descriptions of 3D objects with similar features: (# generated by PointLLM and seleted by mainfold learning) 1. ... 2. ... 3. ... Output only a 3D description. And don't describe too much.

Table 8: A list of prompts for 3D Recognition, 3D OOD Detection, and 3D Object Captioning tasks to LLM.

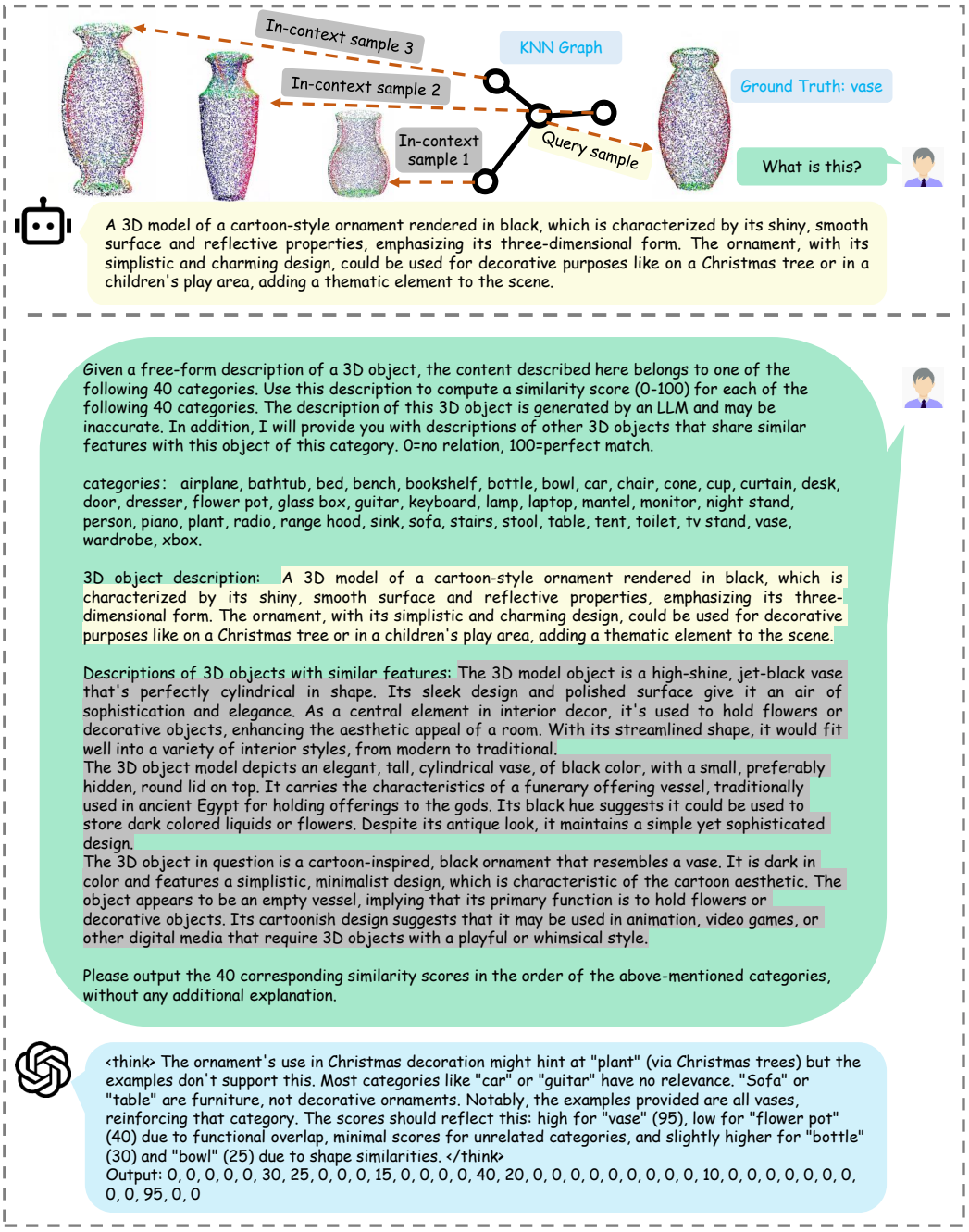


Figure 11: Qualitative example for 3D recognition task on ModelNet40.

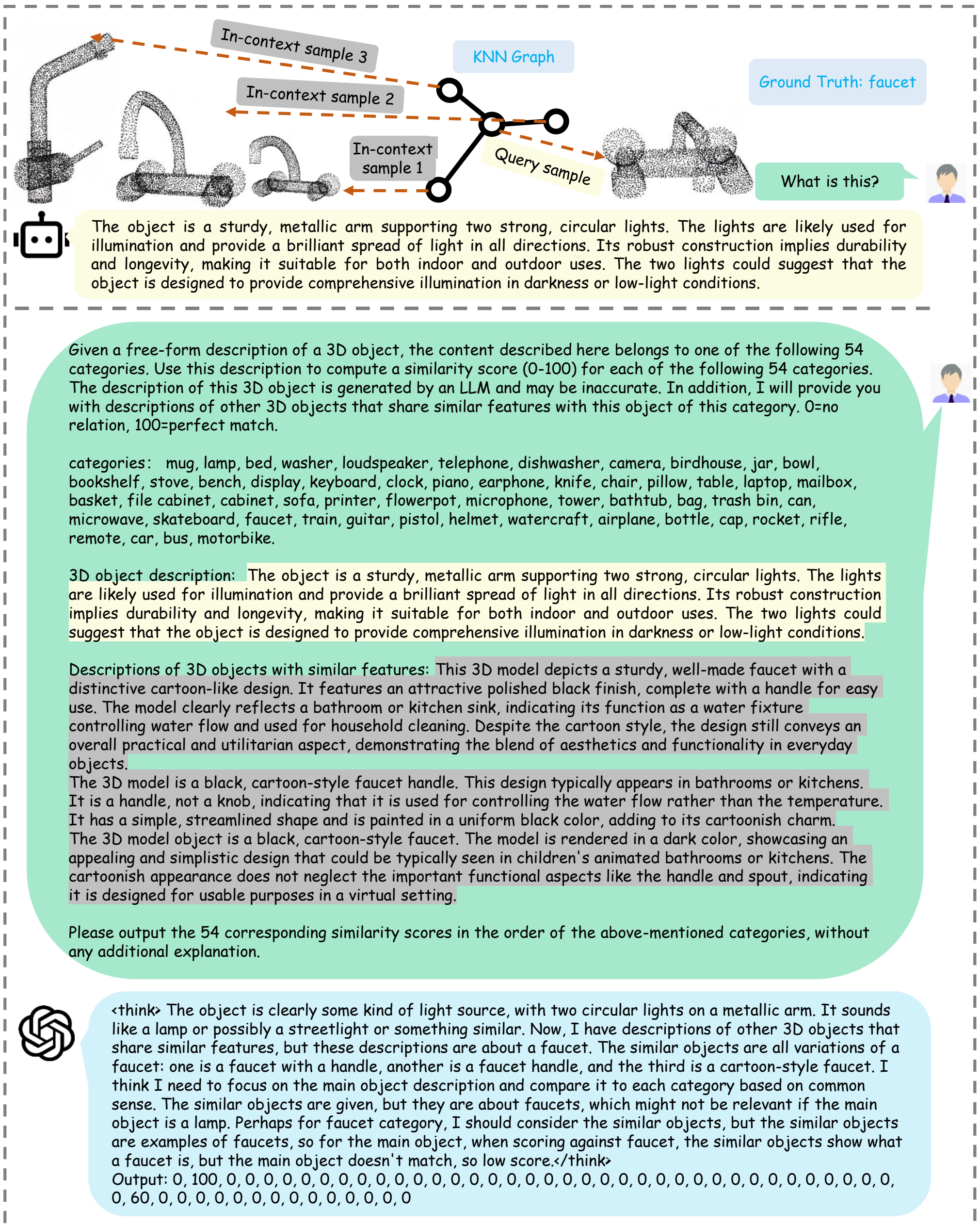


Figure 12: Qualitative example for 3D recognition task on ShapeNetCore.

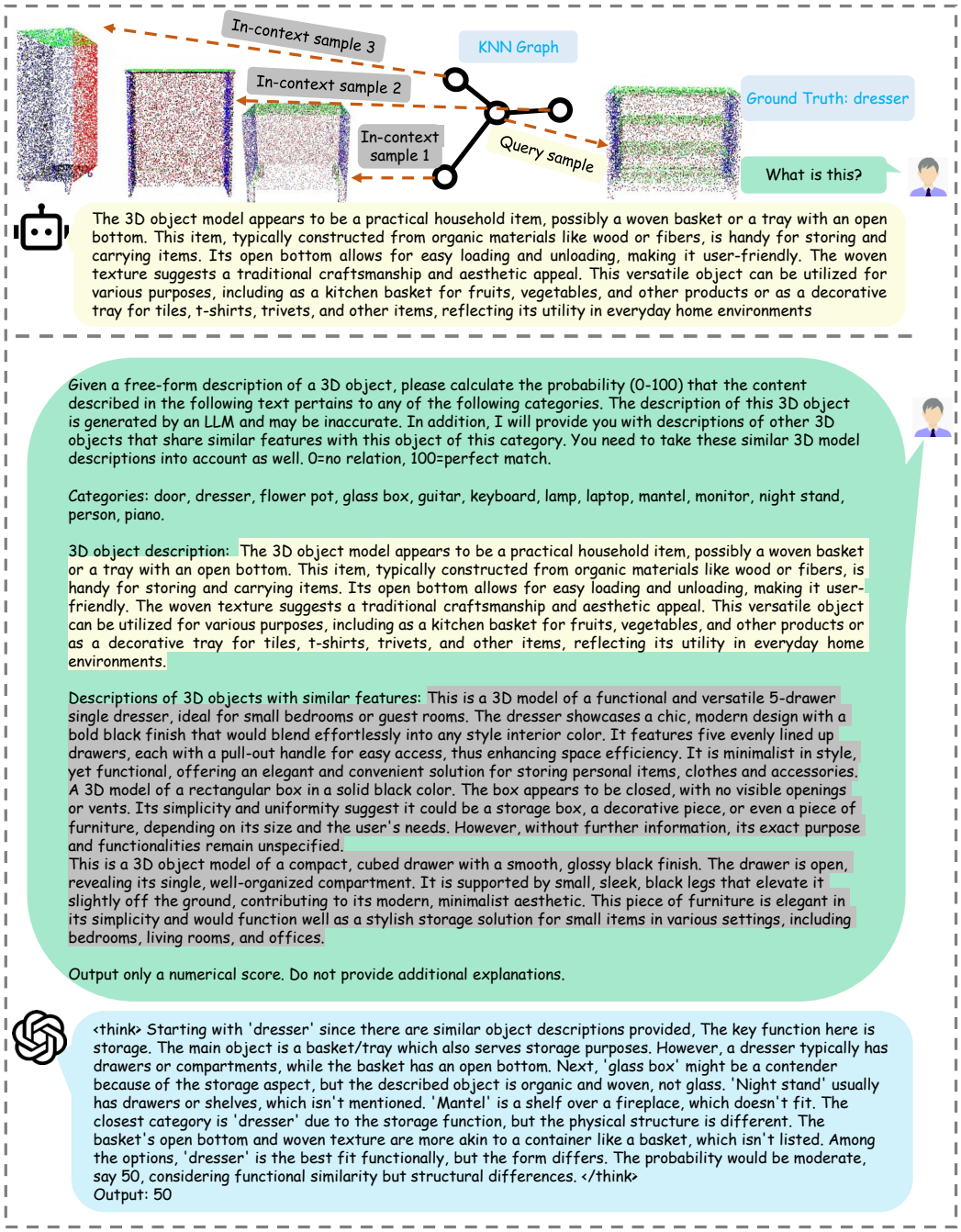


Figure 13: Qualitative example for 3D OOD detection task on ModelNet40.

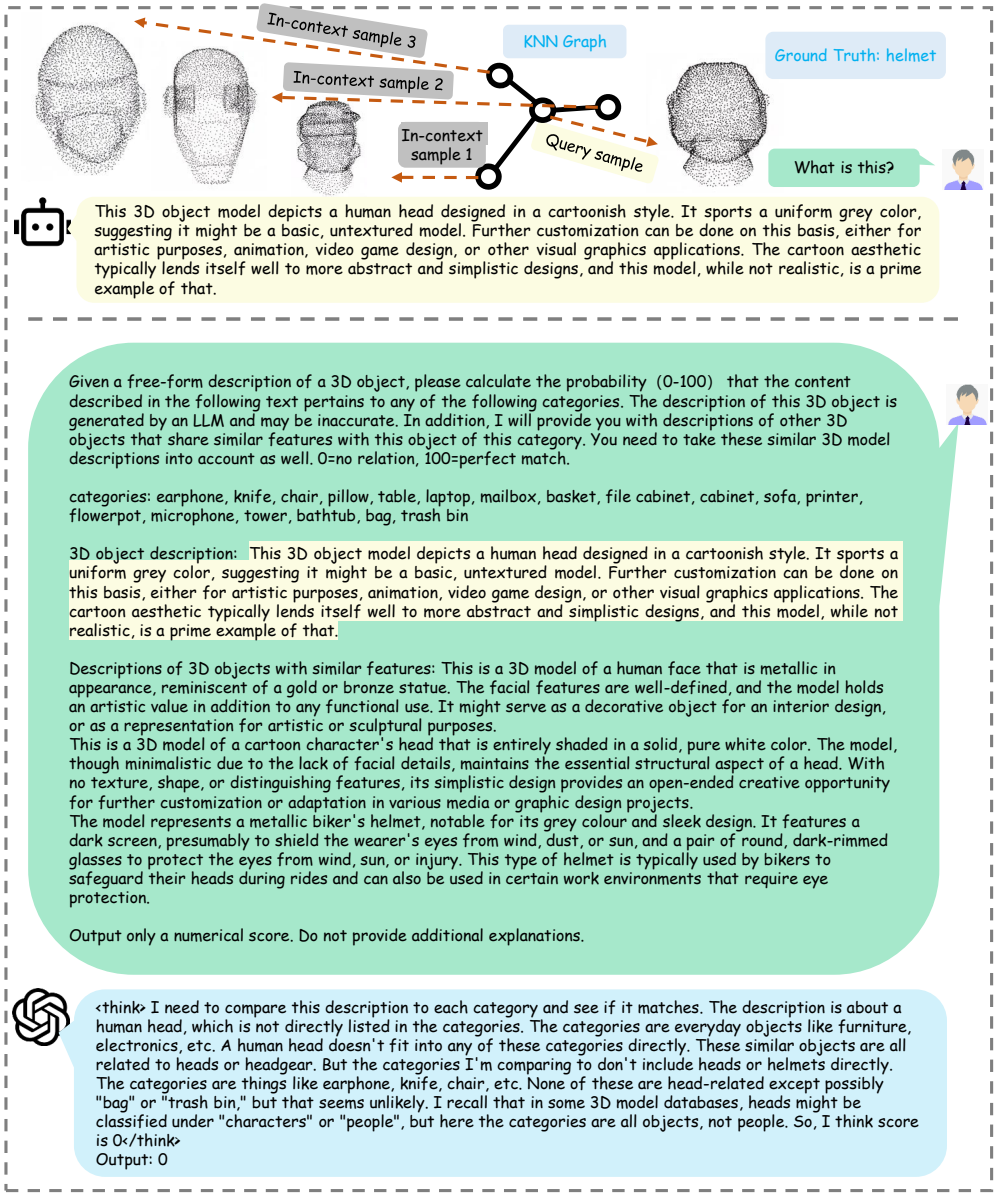


Figure 14: Qualitative example for 3D OOD detection task on ShapeNetCore.

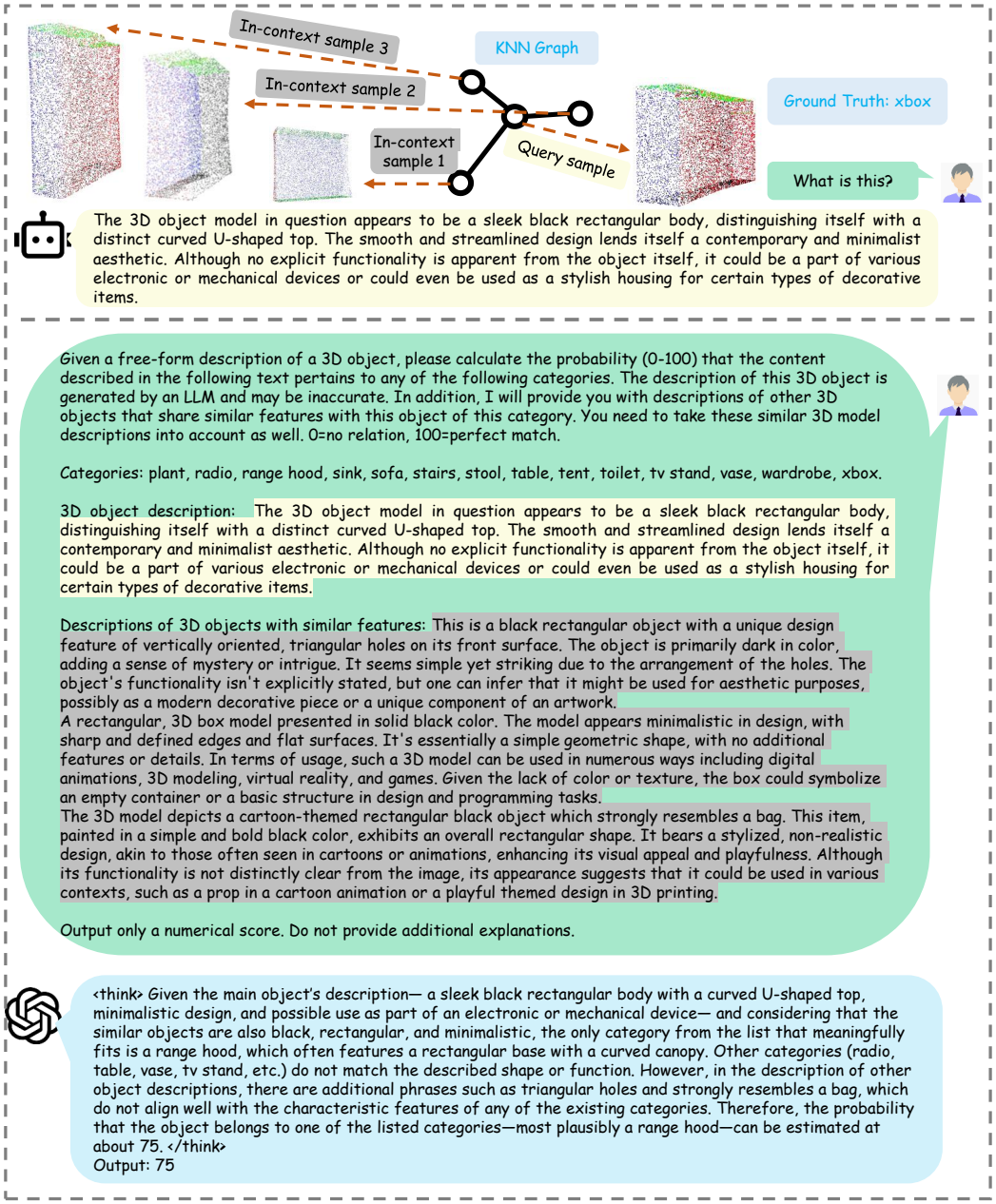


Figure 15: Bad example for 3D OOD detection task on ModelNet40.