Supplementary Materials for 3DGraphLLM: Combining Semantic Graphs and Large Language Models for 3D Scene Understanding

Tatiana Zemskova[1,2] Dmitry Yudin[1,2]
[1]AIRI, [2]MIPT

1. Ablation Study. Number of Nearest Neighbors

We investigate how increasing the number of nearest neighbors affects the quality of scene description and question answering tasks. The number of nearest neighbors varies from 0 to 4.

When no nearest neighbors are used, the model operates as a baseline Chat-Scene approach, representing the scene as a list of objects without semantic relationships. The maximum number of nearest neighbors that can be added for each object is four, constrained by GPU memory limitations during model training. In these experiments, we use GT instance segmentation to eliminate errors in graph construction that could otherwise impact performance on 3D vision-language tasks. The base LLM used in this study is LLAMA3-8B-Instruct.

As shown in Fig. 1, increasing the number of nearest neighbors improves the quality of object descriptions in the scene, while causing only a slight increase in generation time. Notably, in dense scene captioning tasks, inference speed depends not only on the number of tokens used to represent the scene but also on the length of the generated description. Therefore, we observe only a small increase in generation time when comparing graphs with two and four nearest neighbors.

Fig. 2 and Fig. 3 show that semantic relationships between objects enhance performance in question answering tasks on the ScanQA and SQA3D datasets. However, for question answering tasks, the optimal number of nearest neighbors is 2, as increasing it to 4 leads to a drop in performance. The impact of semantic edges is harder to assess in question answering tasks than in visual grounding or object description tasks, since some question types do not require knowledge of object spatial relationships.

For further experiments, we select 2 nearest neighbors, as it provides the best trade-off between performance gains and computational complexity across all three tasks.

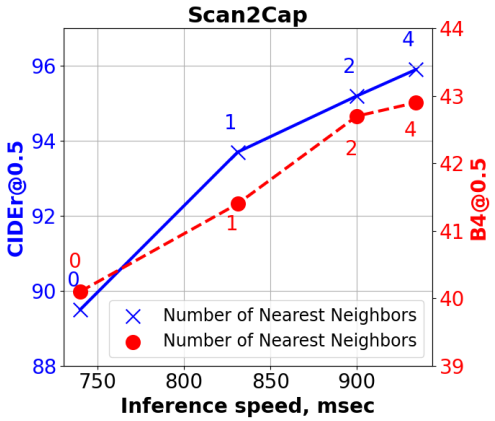

Figure 1. Dependence of inference speed and dense scene captioning quality on the number of nearest neighbors in the object subgraph. This experiment utilizes the GT instance segmentation.

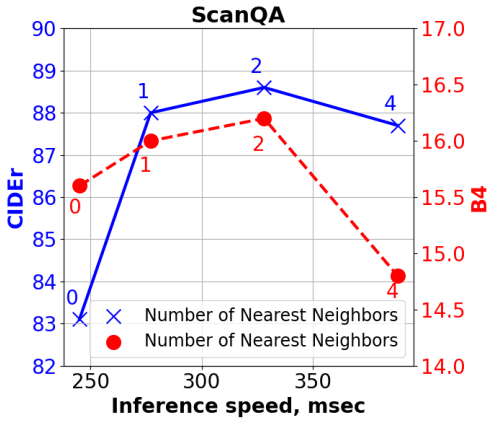

Figure 2. Dependence of inference speed and question answering quality on ScanQA dataset on the number of nearest neighbors in the object subgraph. This experiment utilizes the GT instance segmentation.

Methods	Instance segmentation	Number of edges	Minimal distance, cm	ScanRefer Acc@0.5↑	Multi3DRefer F1@0.5↑	Scan2Cap C@0.5↑	Scan2Cap B-4@0.5↑	ScanQA C↑	ScanQA B-4↑	Sqa3D EM↑
3DGraphLLM-0	GT	0	-	61.5	64.4	89.5	40.1	83.1	15.6	55.2
3DGraphLLM-2	GT	2	0	**66.9**	**69.9**	**95.2**	**42.7**	**88.6**	**16.2**	**56.3**
3DGraphLLM-0	Mask3D	0	-	52.0	55.1	80.0	37.5	84.0	15.8	53.8
3DGraphLLM-2	Mask3D	2	0	55.6	58.2	80.8	36.4	85.7	15.1	56.0
3DGraphLLM-2	Mask3D (+ NMS)	2	0	55.7	58.6	82.3	36.8	**86.2**	**16.0**	**56.2**
3DGraphLLM-2	Mask3D (+ NMS)	2	1	**56.2**	**58.7**	**82.9**	**37.3**	85.4	15.1	55.6
3DGraphLLM-0	OneFormer3D	0	-	50.0	52.8	**73.5**	**34.3**	**87.3**	**16.5**	53.8
3DGraphLLM-2	OneFormer3D	2	0	52.8	55.8	70.2	32.7	83.3	15.0	**55.0**
3DGraphLLM-2	OneFormer3D (+NMS)	2	1	**54.6**	**57.2**	72.4	33.0	81.3	12.9	**55.0**

Table 1. Ablation study on semantic edge role depending on quality of instance segmentation.

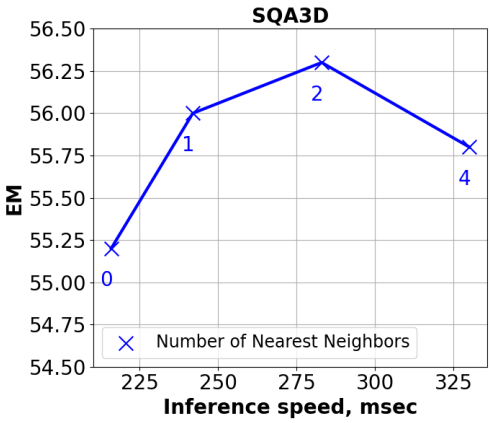

Figure 3. Dependence of inference speed and question answering quality on SQA3D dataset on the number of nearest neighbors in the object subgraph. This experiment utilizes the GT instance segmentation.

2. Ablation Study. Quality of Instance Segmentation.

For the graph, where the vertices contain objects obtained from 3D instance segmentation, we observe a consistent improvement in the performance of all three 3D Vision-Language tasks. When transitioning from GT segmentation to a noisy graph composed of vertices obtained using Mask3D, we also observe improvements in metrics for all three tasks (see Tab. 1). However, this improvement is less pronounced compared to GT instance segmentation.

We compare the graphs of nearest neighbors obtained from GT instance segmentation and Mask3D instance segmentation. The analysis shows that Mask3D instance segmentation contains a large number of object duplicates, as the scene is always divided into N=100 segments. The presence of object duplicates among the neighbors leads to a reduction in useful information about the object's environment in its subgraph. To address the duplicates in the vertices of the subgraphs, we use two filters. The NMS filter with an IoU threshold of 0.99 removes duplicate objects from the neighbors. The minimum distance filter between the centers of the object point clouds excludes the object's own duplicates from its neighbors.

Tab. 1 shows that adding these filters consistently improves the performance of visual grounding and object description tasks for the graph obtained through Mask3D instance segmentation. Since we expect an effect from adding semantic edges specifically for these tasks, we keep this filter in further experiments.

We also experiment with different methods for instance segmentation to create scene graph vertices. We use another method for instance segmentation, OneFormer3D, filtering out vertices with confidence < 0.1. We observe that for scene graphs with such vertices, semantic edges improve the performance of the visual grounding task. At the same time, the combination of nearest-neighbor filters proves effective for this type of scene segmentation, increasing the performance of object grounding and scene captioning tasks. However, since OneFormer3D showed worse results with these hyperparameters compared to Mask3D, and other baseline methods use Mask3D for object proposals, we chose Mask3D for the final version of the pipeline.

3. Ablation Study. Subgraph Representation.

We explore the possibility of further flattening the graph by replacing relationship triplets with a sequence of semantic edges. As a result, object N is described by the following sequence: $\{< OBJN > F_N^{2d}, F_N^v, F_{Nk_1}^e, F_{Nk_2}^e\}$.

Tab. 2 shows that representing relationships as triplets improves the performance of the object grounding task. The object-centered representation of relationships improves question answering performance on ScanQA according to the CIDEr metric, while the BLEU-4 metric remains on par with the flat graph representation as a sequence of relationships. For the SQA3D dataset, both approaches yield comparable results. In the dense scene captioning task, we observe a decrease in the CIDEr metric but an improvement in the Scan2Cap metric. However, as shown in Sec. 5, n-gram-based metrics may produce unreliable results when evaluating text generated by LLMs. Considering this, and the fact that the impact of semantic edges is most interpretable in the object grounding task, we represent relationships as triplets in subsequent experiments.

Methods	Instance segmentation	Relations as triplets	Number of edges	ScanRefer Acc@0.5↑	Multi3DRef F1@0.5↑	Scan2Cap C@0.5↑	Scan2Cap B-4@0.5↑	ScanQA C↑	ScanQA B-4↑	Sqa3D EM↑
3DGraphLLM-0	Mask3D	✗	0	52.0	55.1	80.0	37.5	84.0	15.8	53.8
3DGraphLLM-2	Mask3D	✗	2	54.2	56.3	**87.2**	39.3	85.6	15.1	**54.6**
3DGraphLLM-2	Mask3D	✓	2	**54.3**	**57.3**	85.6	**39.6**	**87.4**	14.9	54.5

Table 2. Ablation study on subgraph representation.

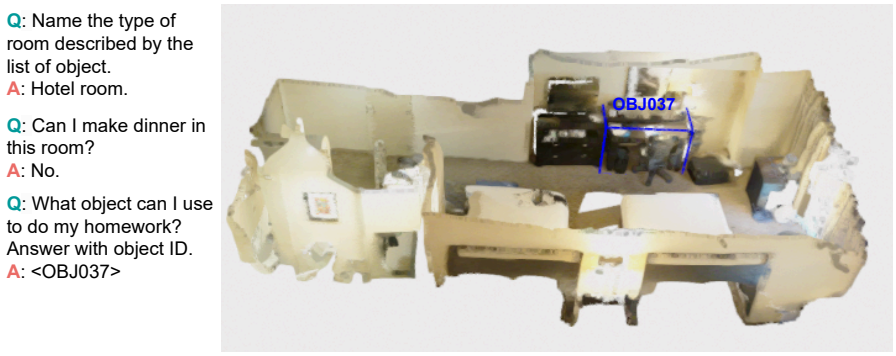

Figure 4. Functional queries about the room and objects to the 3DGraphLLM. 3DGraphLLM is capable of answering questions about the functional properties of the room and its room type as well as discerning the functional properties of objects in a room.

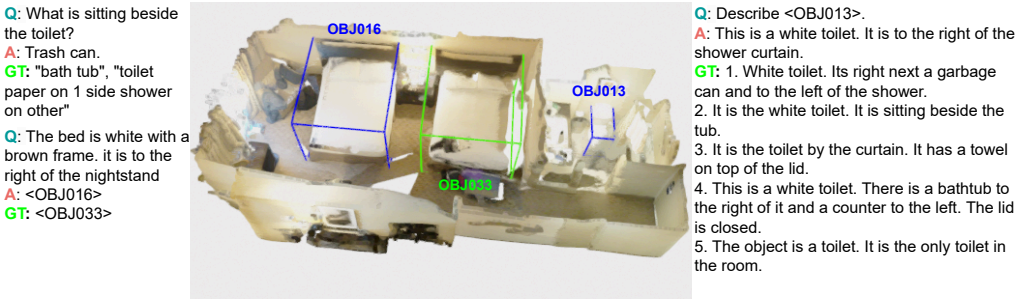

Figure 5. Common failure cases of 3DGraphLLM related to spatial relationships. In the question answering task, 3DGraphLLM incorrectly identifies the front/back and left/right directions relative to the observer. In the visual grounding task, 3DGraphLLM confuses left and right. The GT object is highlighted in green, and the 3DGraphLLM prediction is highlighted in red. In the object captioning task, 3DGraphLLM uses a spatial reference not mentioned in the GT descriptions. However, the description is correct qualitatively.

4. Functional Queries

We illustrate the ability of 3DGraphLLM to leverage common sense knowledge in its responses to question types not present in the training dataset in Fig. 4.

5. Common Failure Cases

We illustrate the most common failure cases of 3DGraphLLM related to spatial relationships in Fig. 5.

It is important to note that the quality metrics in the Scan2Cap, ScanQA, and SQA3D benchmarks are based on n-gram-based metrics comparing generated answers with reference ones, such as BLEU-1, BLEU-2, BLEU-3, BLEU-4, CIDEr, ROUGE-L, METEOR. The Exact Matching (EM) metric compares the exact match of the answer with the GT answer. The drawback of these metrics is that if an object description or answer to a question contains spatial relationships not present in the reference descriptions, it leads to a decrease in the score. Additionally, these metrics are unable to adequately evaluate LLM responses, considering the richness of formulations and the freedom to choose visual and spatial properties of an object that may be mentioned by the model. These responses represent a special type of "failure cases," illustrated in Fig. 5 on the right. For this type, the object description or answer to the question is correct from a qualitative point of view but shows zero value according to the metric CIDEr.

6. Scalability with Number of Scene Objects

To evaluate scalability, we analyze how memory usage and inference time vary with the number of objects in a scene. The maximum number of objects considered corresponds

to the highest counts observed in the ScanNet and 3RScan datasets. As shown in Tab. 3, both memory usage and inference time increase gradually as the number of objects in the scene grows. This demonstrates that while resource consumption scales with object count, the growth remains manageable and does not compromise the method's practical applicability within real-world scene configurations.

Number of Objects	Memory Usage (Gb)	Inference Time for 1 token (s)
10	23	0.08
50	28	0.14
100	35	0.23

Table 3. Performance metrics for varying number of objects in a 3D scene.