NavRAG: Generating User Demand Instructions for Embodied Navigation through Retrieval-Augmented LLM

Anonymous ACL submission

Abstract

Vision-and-Language Navigation (VLN) is an essential skill for embodied agents, allowing them to navigate in 3D environments following natural language instructions. High-performance navigation models require a large amount of training data, the high cost of manually annotating data has seriously hindered this field. Therefore, some previous methods translate trajectory videos into step-by-step instructions for expanding data, but such instructions do not match well with users' communication styles that briefly describe destinations or state specific needs. Moreover, local navigation trajectories overlook global context and high-level task planning. To address these issues, we propose NavRAG, a retrieval-augmented generation (RAG) framework that generates user demand instructions for VLN. NavRAG leverages LLM to build a hierarchical scene description tree for 3D scene understanding from global layout to local details, then simulates various user roles with specific demands to retrieve from the scene tree, generating diverse instructions with LLM. We annotate over 2 million navigation instructions across 861 scenes and evaluate the data quality and navigation performance of trained models. The model trained on our NavRAG dataset achieves SOTA performance on the REVERIE benchmark.

1 Introduction

Vision-and-Language Navigation (VLN) (Anderson et al., 2018; Krantz et al., 2020; Qi et al., 2020; Zhu et al., 2021) requires the agent to understand natural language instructions and navigate to the described destination in 3D environments. The immense semantic space and diverse forms of language instructions require massive data to train a VLN agent capable of generalizing across different scenarios. However, the high cost of manual annotation has seriously hindered this field, driving efforts to develop instruction generators for automating data generation.

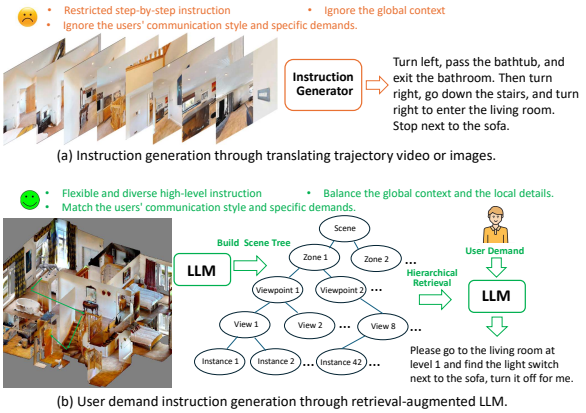

Figure 1: The comparison of previous navigation instruction generation methods (a) and NavRAG (b).

As shown in Figure 1 (a), many previous approaches train a navigation instruction generator that takes video or images from Web or simulators as input and produces step-by-step instructions. Leveraging large-scale generated navigation data, this strategy has delivered outstanding results in some navigation tasks using trajectory-based instructions, such as R2R (Anderson et al., 2018) and REVERIE (Qi et al., 2020). However, such instruction generators still remain some shortcomings: 1) These instruction generators are trained on small-scale and domain-specific datasets, the generated instructions lack diversity; 2) Such step-by-step instructions are limited to local navigation trajectories overlooking the global context and high-level task planning; 3) These instructions don't match well with users' natural expressions that describe destinations or state specific needs.

To tackle these challenges, this work proposes NavRAG, an instruction generation method leveraging retrieval-augmented LLM, as illustrated in Figure 1(b). Specifically, for each 3D scene, NavRAG constructs a scene description tree in a bottom-up manner for hierarchical scene representations. This hierarchical tree comprises multiple layers of language descriptions: the instance layer captures descriptions, attributes, and functionalities

of individual instances; the view layer summarizes spatial relationships within a view; the viewpoint layer integrates multiple views into a panoramic environmental description; the zone layer clusters viewpoints within the same functional area (*e.g.*, a bedroom or kitchen); and finally, the scene-level description provides an overview of all zones and their connectivity.

After establishing the environmental context with the scene tree, the generated navigation instructions are expected to *meet the user demands*. Therefore, unlike previous instruction generators that were only used to describe navigation trajectories, NavRAG set up several different user roles (with varying ages, genders, occupations, lifestyles and demands to navigation agent) to simulate and record the instructions sent to navigation agent during one day of this role. Meanwhile, to balance generation quality and cost, our framework initially generates the *coarse* instruction only through the overview of the scene, then uses retrieval-augmented LLM to perform top-down, layer-by-layer retrieval of the best destination and relevant texts from the scene tree, and finally refines the coarse instruction into a more detailed and accurate *refined* instruction using retrieval-augmented LLM.

In summary, our contributions are as follows:

- This work proposes an approach for automatically constructing scene description trees and generating user demand navigation instructions using retrieval-augmented LLM.
- We annotate over 2 million high-quality navigation instructions across 861 3D scenes for training and evaluation.
- The VLN models trained on our NavRAG dataset achieve superior performance on VLN benchmarks, validating the effectiveness of the proposed method.

2 Related Work

Vision-and-Language Navigation (VLN) (Anderson et al., 2018; Krantz et al., 2020; Qi et al., 2020; Zhu et al., 2021) enables embodied agents to navigate to the destination described by the language instructions. Early VLN researches focus on discrete environments within 90 scenes of Matterport3D (Chang et al., 2017), which uses a predefined navigation graph, the agent observes panoramic RGB and depth images, teleporting between graph nodes to follow natural language instructions. Under this setting, the datasets include the step-by-step instruction dataset R2R (Anderson et al., 2018), the multilingual instruction dataset RxR (Ku et al., 2020) with longer trajectories, the Remote Embodied Visual Referring Expression (REVERIE) (Qi et al., 2020) dataset, and the Scenario Oriented Object Navigation (SOON) (Zhu et al., 2021) task. Although efficient for training in discrete environments, these datasets lack real-world applicability. To address this, R2R-CE (Krantz et al., 2020) introduce continuous environments (Savva et al., 2019) with instructions from the R2R dataset, where agents navigate freely in 3D spaces using low-level actions (e.g., turn 15°, move 0.25m) in the Habitat simulator (Savva et al., 2019). In this work, we focus on generating large-scale, high-quality navigation instructions, for simplicity and efficiency, our NavRAG is currently validated in the discrete environments, while the annotated data remains easily transferable to continuous settings.

Navigation Instruction Generation is an effective approach to addressing the scarcity of training data for VLN. Speaker-follower (Fried et al., 2018) and Env-Drop (Tan et al., 2019) use the LSTM-based instruction generator to generate the offline augmented instructions. VLN-Trans (Zhang and Kordjamshidi, 2023) propose a translator module that enables the navigation agent to generate more concise sub-instructions, leveraging recognizable and distinctive landmarks. AutoVLN (Chen et al., 2022a), MARVAL (Kamath et al., 2023) and ScaleVLN (Wang et al., 2023c) leverage multiple foundation models (Cheng et al., 2022; Radford et al., 2019; Zhao et al.; Koh et al., 2023) and use more 3D scenes to annotate instructions, such as HM3D (Ramakrishnan et al.) and Gibson (Xia et al., 2018). Recently, more works focus on designing more powerful instruction generator, such as a joint structure for instruction following and generation (Wang et al., 2023a), Knowledge enhanced speaker (Zeng et al., 2023), LLM instruction generator with chain of thought prompting (Kong et al., 2025), and LLM instruction generator with BEV perception (Fan et al., 2025). However, these methods are limited to identifying landmarks in navigation trajectories and generating low-level instructions, making it difficult to integrate global context, match user demands, and plan high-level tasks. NavRAG will generate navigation instructions better tailored to the application scenario by considering the global context and user demands through

scene description trees and retrieval-augmented LLM.

Retrieval-Augmented Generation (RAG) (Lewis et al., 2020) was initially introduced to enhance LLMs by retrieving relevant document chunks, thereby providing domain-specific knowledge for better answer. Over time, several innovations have expanded on this idea, including techniques like iterative knowledge retrieval (Shao et al., 2023), and the incorporation of knowledge graphs (Edge et al., 2024). Furthermore, adapting RAG to the field of robotics, some works (Xie et al., 2024; Booker et al., 2024) attempt constructing non-parametric memory or scene graphs for 3D scenes, and utilize retrieval-augmented LLM for question answering or navigation. However, traditional RAG methods for scene graph retrieval struggle to balance global context with local details and interpret the environment layout. NavRAG leverages the scene description tree and hierarchical retrieval strategy, achieve better scene understanding.

3 Method

3.1 Navigation Setups

In the vision-and-language navigation (VLN) setting, the navigation connectivity graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ is provided by the Matterport3D simulator (Chang et al., 2017), where $\mathcal{V}$ represents navigable nodes and $\mathcal{E}$ denotes the edges connecting them. The agent is equipped with RGB cameras and a GPS sensor. Starting from a starting node and following natural language instructions, the agent must explore the navigation connectivity graph $\mathcal{G}$ and move to the destination node. The instruction is represented by a sequence of word embeddings $\mathcal{W} = \{w_l\}_{l=1}^{L}$, where L is the number of words. At each time step t, the agent can perceive a panoramic RGB observation $\mathcal{R}_t = \{r_{t,k}\}_{k=1}^{K}$ at current node $\mathcal{V}_t$, consisting of K view images. The RGB observation of nodes can be obtained through the Matterport3D simulator, so each annotated navigation sample only needs a navigation instruction and an optimal path from the starting node to the destination node for training or evaluation.

3.2 Constructing the Scene Description Tree

Before generating instructions, it is essential to first represent and understand the environment. As illustrated in Figure 2, we propose a bottom-up, hierarchical approach for constructing a scene description tree. At the view and object levels, each object is described with fine-grained details, including its category, attributes, functionality. The spatial relations among objects is summarized in view-level description. The viewpoint level aggregates multiple views surrounding each navigable viewpoint and summarize the spatial layout around this viewpoint. The zone level integrates multiple viewpoints to define large functional areas (*e.g.*, a bedroom) within the 3D scene. Finally, the house level encompasses multiple zones, offering a high-level abstraction of the overall spatial layout and functional partitioning of the whole scene.

Navigation Graph. We introduce 800 training scenes from HM3D (Ramakrishnan et al.) and 61 training scenes along with 11 validation scenes from Matterport3D (Chang et al., 2017) for scene tree construction. Obtaining the navigation graphs of these scenes is the first step. Although MP3D already has manually annotated navigation graphs, the navigation graphs of HM3D still remains to construct. Following ScaleVLN (Wang et al., 2023c), we use a heuristic method to build high-quality navigation graphs for HM3D scenes, ensuring high space coverage, fully traversable edges, and well-positioned nodes, which samples dense viewpoints using Habitat Simulator (Savva et al., 2019)'s navigable position function, ensuring over 0.4m geodesic separation. The Agglomerative Clustering (1.0m threshold) is utilized to centralize nodes and form an initial graph by randomly connecting viewpoints within 5.0m, capping node edges at five. Finally, the graph is refined for full connectivity and traversal, producing graphs for 800 scenes.

View and Object-level Annotation. To capture detailed information about objects within a specific viewpoint of the navigation graph, we utilize the Habitat simulator (Savva et al., 2019) to uniformly sample six views (each with an image resolution of 480×480) from every viewpoint in the navigation graph. These views are then input into a multimodal LLM (*i.e.*, GPT-4o-mini (Hurst et al., 2024)) to generate descriptions of each view, objects, their attributes, and functionalities.

Viewpoint-level Annotation. Integrating descriptions and object information from multiple views, the LLM generates a comprehensive description of the environment surrounding the viewpoint. This description encompasses the area type, spatial layout, and relationships among objects, providing a holistic understanding of panorama.

Zone Partitioning and Annotation. To enhance the comprehension of the scene's spatial layout

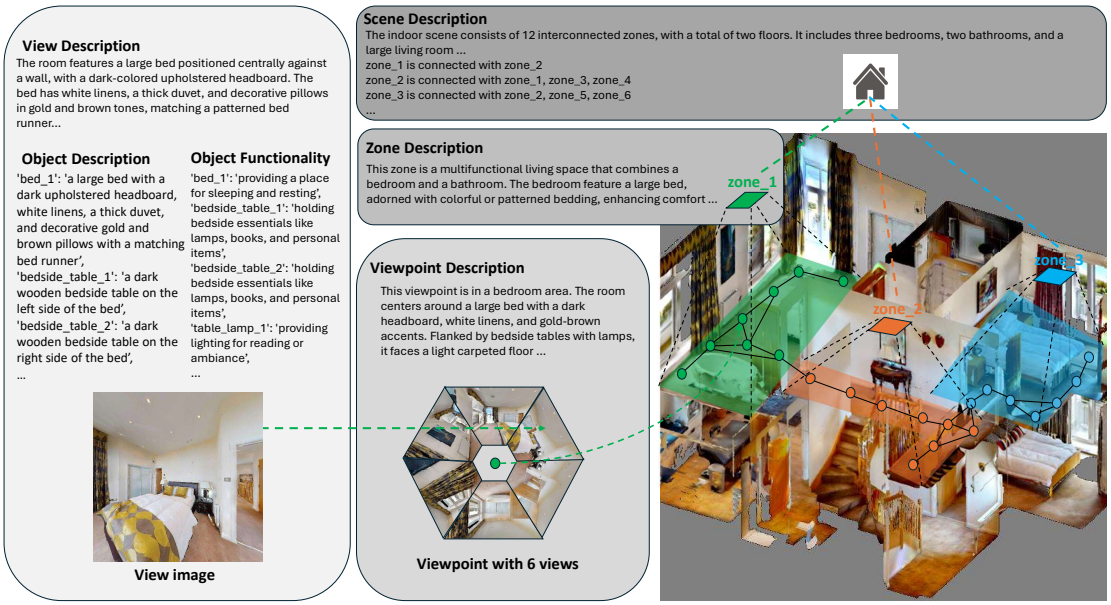

Figure 2: Demonstration of the Scene Description Tree. Based on LLM, NavRAG constructs the scene description tree in a bottom-up manner, progressively constructing from objects to views, viewpoints, zones, and the overall scene. This hierarchical structure describes environmental semantics and spatial relationships at different levels, facilitating LLM in understanding 3D environments and retrieving information for instruction generation.

(e.g., room count and connectivity) meanwhile decreasing retrieval cost from numerous viewpoints, we construct zones by merging multiple viewpoints, as shown in Figure 2. Unlike previous methods (Xie et al., 2024) using hierarchical clustering based on spatial positions to construct scene trees, we propose a new algorithm that incorporates viewpoint connectivity and environmental semantics for scene partitioning as shown in Figure 3. Hierarchical clustering based on spatial positions has two important drawbacks: 1) It overlooks viewpoint connectivity, potentially grouping nearby but wall-separated viewpoints into the same zone. 2) It ignores environmental semantics, relying solely on spatial positions cannot accurately recognize different functional areas of the scene.

To address these issues, our algorithm first selects the viewpoint with the highest connectivity to initialize a zone and uses LLM to generate its description. Then, by searching the adjacent viewpoints in descending order of connectivity, the algorithm inputs the zone description and the description of adjacent viewpoint into LLM to determine if the viewpoint belongs to the zone, if yes, this viewpoint will be added to the zone, and the zone description is updated. Once all viewpoints for this zone are identified, all nodes within the zone are removed from the navigation graph, then the next zone construction begins.

Scene-level Annotation. To provide an overview

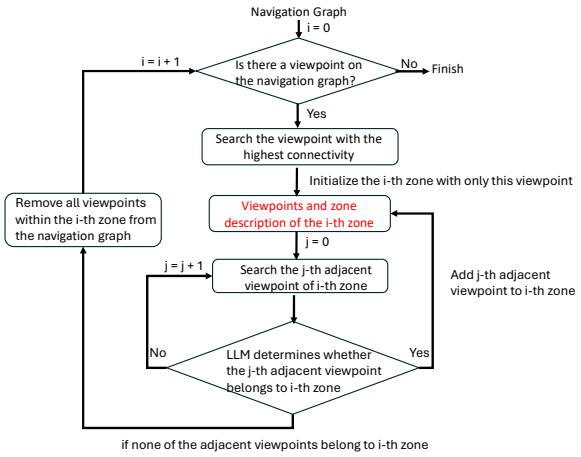

Figure 3: Framework of the zone partitioning algorithm based on connectivity relations and environmental semantics.

of the spatial layout of the entire scene, the scene-level description primarily includes the connectivity between various zones (similar to MapGPT (Chen et al., 2024)), the types of each zone, a concise summary, and the functionality.

3.3 User Demand Instruction Generation

As shown in Figure 4 and Figure 5, after constructing the scene description tree, NavRAG leverages the scene-level description, user information, and demands to generate a rough instruction for the navigation agent, such as "Walk to the warm hall and set the wooden table for lunch". Subsequently, NavRAG performs a top-down, hierarchical re-

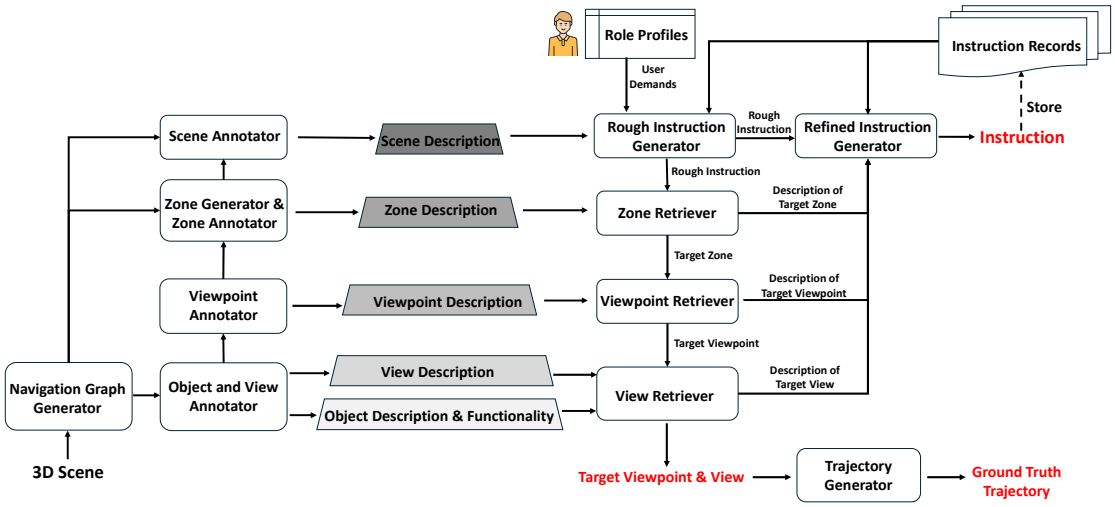

Figure 4: Framework of NavRAG for scene tree construction and navigation instruction generation through Retrieval-Augmented LLM.

trieval of potential destinations from the scene tree and integrates the retrieved environmental descriptions at different levels into the LLM, to refine rough instruction into precise and comprehensive instruction, such as "Walk to the warm hall featuring elegant wooden accents and set the large wooden table with candles and napkins for a lovely dinner ambiance".

User Demands Simulation. To further improve the diversity of generated instructions and meet the user demands, NavRAG integrates texts of user information and demands, enabling the instruction generator to simulate specific roles to generate tailored instructions. A sample of user profile and demands is as follows:

{
"Age": 33,
"Gender": "Female",
"Occupation": "Lawyer",
"Lifestyle Description": "You maintain the good habit of going to bed early and waking up early. Besides working in the study, you often do yoga and other exercises in the living room and enjoy cooking your own meals."
}

We manually annotate 20 user profiles for different roles. For each role, the prompt guides the LLM in simulating the role's behavior with a given scene description tree, generating the records of 50 navigation instructions sent to the agent during one day of this role.

Retrieval-Augmented Generation. As illustrated in Figure 4, Figure 5 and Figure 6, NavRAG performs layer-by-layer retrieval of texts at different levels based on the scene description tree, progressively localizes the navigation destination. Initially, the LLM generates a rough instruction based on scene-level descriptions, user information, and historical instruction records. It then identifies the most probable zone containing the navigation destination from the zone-level descriptions. Based on the viewpoint descriptions within that zone, the LLM selects the target viewpoint and locates the view containing the navigation target. By integrating texts from all different levels, the LLM ultimately refines rough instruction and outputs the precise and comprehensive instruction.

4 Experiments

Dataset	Generated	#Scene	#Instr.	Instr. length
REVERIE (Qi et al., 2020)	×	60	10,466	18.64
R2R (Anderson et al., 2018)	×	61	14,039	26.33
RxR-en (Ku et al., 2020)	×	60	26,464	102.13
SOON (Zhu et al., 2021)	×	34	2,780	44.09
Prevalent (Hao et al., 2020)	✓	60	1,069,620	24.23
Marky (Wang et al., 2022)	✓	60	333,777	99.45
AutoVLN (Chen et al., 2022a)	✓	900	217,703	20.52
ScaleVLN (Wang et al., 2023c)	✓	1289	4,941,710	21.61
NavRAG (Ours)	✓	861	2,115,019	29.11

Table 1: Statistics of training data on different VLN datasets.

4.1 Datasets and Evaluation Metrics

Datasets. Table 1 summarizes the main VLN datasets, including human-annotated data and model-generated data. The high cost of manual annotation limits the scale of manual training data, severely restricting the generalization ability of VLN models. An effective approach to enhancing navigation performance is to automatically generate large-scale navigation data for VLN pretraining, then fine-tune on manual data. Our NavRAG annotates over 2 million navigation instructions across 861 training scenes, each corresponding to a navigation destination (*i.e.*, target viewpoint). Using

Figure 5: Prompt, input and output of the Rough Instruction Generator and Hierarchical Retrieval.

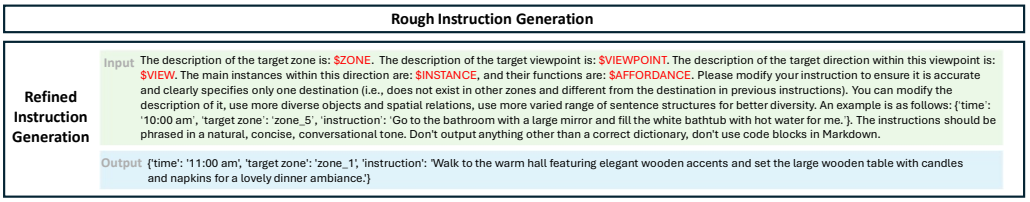

Figure 6: Prompt, input and output of the Refined Instruction Generator. $ZONE, $VIEWPOINT, $VIEW, $INSTANCE and $AFFORDANCE denote retrieved environmental descriptions at different levels.

a trajectory generator which samples the starting viewpoint and calculate the shortest path to the destination, we randomly sample 5 trajectories per instruction, yielding over 10 million navigation trajectories in total. To evaluate model performance, we also annotate 7,396 instruction-trajectory pairs across 11 unseen scenes, forming the *NavRAG Val Unseen* benchmark for performance evaluation.

Evaluation Metrics. Four main metrics are used for navigation: 1) Navigation Error (NE): the mean of the shortest path distance between the agent's final position and the destination. 2) Oracle Success Rate (OSR): the percentage that the agent has reached a position within 3 meters of the destination. 3) Success Rate (SR): the percentage of the predicted stop position being within 3 meters from the destination. (3) Success rate weighted Path Length (SPL) that normalizes the success rate with trajectory length.

4.2 VLN Models

To evaluate our NavRAG dataset, multiple VLN models are used in the experiments, as shown in Table 2 and Table 3.

DUET (Dual-scale Graph Transformer) (Chen et al., 2022b) is a VLN model that dynamically builds a topological map for efficient global exploration while integrating fine-grained local observations and coarse-grained global encoding through graph transformers.

HAMT (History Aware Multimodal Transformer) (Chen et al., 2021) is a VLN model that

Models	LLM	Training Data	NavRAG Val Unseen				REVERIE Val Unseen			
			NE↓	OSR↑	SR↑	SPL↑	NE↓	OSR↑	SR↑	SPL↑
DUET	×	AutoVLN (REVERIE-style)	13.2	30.2	16.2	10.7	6.9	49.7	42.3	26.4
DUET	×	ScaleVLN (REVERIE-style)	11.3	41.9	17.4	11.9	6.7	50.2	44.6	28.2
DUET	×	ScaleVLN (R2R-style)	12.6	31.1	10.3	4.2	9.0	41.4	27.9	11.6
NavGPT	GPT-4o-mini	-	**7.7**	43.1	28.2	11.6	9.2	25.8	20.2	13.1
MapGPT	GPT-4o-mini	-	7.8	**47.7**	**30.9**	**15.3**	8.2	37.4	30.2	21.6
MapGPT	Llama-3.1-8B (Dubey et al., 2024)	-	8.1	44.2	25.5	12.4	8.4	35.8	24.4	16.2
HAMT	×	NavRAG (Ours)	8.3	42.5	25.1	20.4	8.1	40.3	32.8	21.7
DUET	×	NavRAG (Ours)	7.7	50.0	30.7	25.4	**7.6**	**45.9**	**36.1**	**24.9**

Table 2: Zero-shot performance comparison on NavRAG and REVERIE datasets, reflecting the model's generalization ability. Gray values do not strictly follow the zero-shot setting.

integrates long-horizon history using a hierarchical vision transformer, which efficiently encodes past panoramic observations and combines text, history, and current views to predict navigation actions.

NavGPT (Zhou et al., 2024) is a purely LLM-based instruction-following navigation agent, which performs zero-shot sequential action prediction, demonstrating abilities such as high-level planning, sub-goal decomposition, commonsense integration, and navigation progress tracking.

MapGPT (Chen et al., 2024) is a LLM-based VLN agent that integrates an online linguistic-formed map to enable global exploration. By incorporating node information and topological relationships into prompts, MapGPT understands spatial environments and features an adaptive planning mechanism for multi-step path planning.

4.3 Limitations of the Existing Training Data.

Table 2 evaluates the zero-shot performance of multiple VLN methods on NavRAG and REVERIE benchmarks, and also shows the performance of models trained on NavRAG datasets. As shown in rows 1-3 of Table 2, models trained on previously generated large-scale datasets (*i.e.*, AutoVLN and ScaleVLN) perform poorly on the NavRAG benchmark, whereas LLM-based methods (rows 4-6) demonstrate relatively strong performance.

NavRAG leverages the scene description tree and retrieval-augmented LLM, resulting in a larger semantic space of instructions with more diverse sentence structures, meanwhile, better aligned with human expression. LLM-based models effectively comprehend these instructions. In contrast, instructions in ScaleVLN and AutoVLN are generated by a pre-trained instruction generator trained on a small-scale manually annotated dataset (*i.e.* REVERIE and R2R), restricting the semantic space and diversity, and further hindering the generation ability. Thus, models trained on them struggle with NavRAG benchmark and real-world applications.

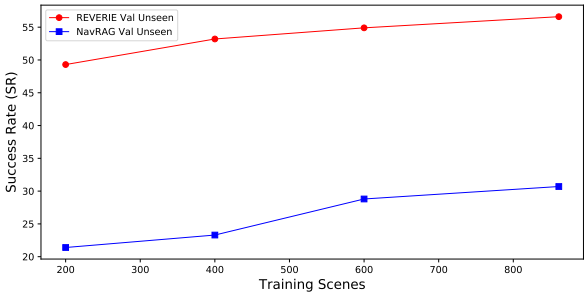

Figure 7: Navigation performance with respect to the number of pre-training scenes in NavRAG dataset.

Notably, the performance of the LLM-based method on the NavRAG benchmark surpasses the human-annotated REVERIE benchmark (NE, OSR and SR metrics), due to NavRAG's longer, more detailed, and accurate instructions (shown in Table 1). This finding further validates the quality of instructions generated by our NavRAG.

4.4 Generalization Ability of NavRAG

As shown in the last two rows of Table 2, the models trained on the NavRAG dataset achieves competitive performance on both NavRAG Val Unseen and REVERIE Val Unseen benchmarks, and even outperforms LLM-based methods (*i.e.*, NavGPT and MapGPT), showing the ability of NavRAG dataset to enhance model generalization.

Furthermore, Figure 7 illustrates that NavRAG consistently improves the performance of the VLN model as the pre-training data scale increases, underscoring the potential and value of large-scale generated navigation data.

4.5 Comparison with SOTA Methods

The last row of Table 3 presents the performance of DUET pre-trained on the NavRAG dataset and fine-tuned on the REVERIE dataset, which is comparable to the SOTA approaches with LLM.

Previous methods use manually annotated object bounding boxes of REVERIE datasets to extract visual features for model inputs. However, this strategy restricts the model's applicability in real-

Methods	LLM	Objects	REVERIE Val Unseen			
			NE↓	OSR↑	SR↑	SPL↑
HAMT (Chen et al., 2021)	×	✓	-	36.8	33.0	30.2
DUET (Chen et al., 2022b)	×	✓	-	51.1	47.0	33.7
Lily (Lin et al., 2023)	×	✓	-	53.7	48.1	34.4
KERM (Li et al., 2023)	×	✓	-	55.2	50.4	35.4
BEVBert (An et al., 2023)	×	✓	-	56.4	51.8	36.4
BSG (Liu et al., 2023)	×	✓	-	58.1	52.1	35.6
GridMM (Wang et al., 2023b)	×	✓	-	57.5	51.4	36.5
ENP-DUET (Liu et al., 2024a)	×	✓	-	54.7	48.9	33.8
AutoVLN (Chen et al., 2022a)	×	✓	-	62.1	55.9	40.9
ScaleVLN (Wang et al., 2023c)	×	✓	-	**63.9**	57.0	41.8
VER (Liu et al., 2024b)	×	✓	-	61.1	56.0	39.7
GOAT (Wang et al., 2024)	×	✓	-	-	53.4	36.7
NaviLLM (Zheng et al., 2024)	✓	✓	-	51.5	28.1	21.0
MiC (Qiao et al., 2023)	✓	✓	-	62.4	57.0	43.6
VLN-Copilot (Qiao et al., 2024)	✓	✓	-	62.6	**57.4**	**43.6**
DUET	×	×	6.0	50.0	45.8	32.5
AutoVLN	×	×	5.7	61.8	54.3	39.1
ScaleVLN	×	×	5.7	62.7	55.9	40.6
NavRAG (Ours)	×	×	**5.5**	**70.7**	**57.3**	**42.0**

Table 3: Fine-tuning performance comparison on REVERIE dataset. "Objects" indicates whether visual features of annotated object bounding boxes are utilized for training.

world deployment, since the real world does not have ground-truth object information. NavRAG removes the reliance on annotated object bounding boxes, making it more suitable for real-world deployment. For a fair comparison, we also evaluate the performance of other generated datasets after removing the object bounding box information from REVERIE, in this setting, NavRAG shows superior performance. This suggests that, despite NavRAG having a larger domain gap with the REVERIE dataset compared to AutoVLN and ScaleVLN, pretraining on more diverse instructions of the NavRAG dataset enables the model to achieve strong generalization, even leading to better fine-tuning performance surpasses domain-specific generated data.

4.6 Ablation Study

Training Data	Validation Data	NavRAG Val Unseen			
		NE↓	OSR↑	SR↑	SPL↑
GraphRAG	GraphRAG	14.1	41.4	12.1	8.7
Zone Clustering	Zone Clustering	9.8	48.9	16.4	11.6
w/o User.	w/ User.	9.4	45.6	18.6	13.7
w/ User.	w/o User.	9.1	48.1	20.8	15.7
NavRAG	NavRAG	8.9	46.8	21.5	15.4

Table 4: The ablation study of NavRAG, evaluating the effectiveness of the components. To reduce costs, only 100 scenes are annotated for DUET training.

Retrieval-Augmented Generation: NavRAG vs. GraphRAG. To validate the superiority of our scene description tree-based retrieval over traditional RAG methods (*e.g.*, GraphRAG (Edge et al., 2024)), we also annotate 100 scenes through GraphRAG to evaluate instruction quality. Specifically, GraphRAG replaces the scene description tree with a knowledge graph built from view-level descriptions. During instruction generation, it retrieves relevant text fragments from the knowledge graph, integrates them into a prompt, and feeds them to the LLM to generate instructions and navigation destinations. Comparing the first and last rows of Table 4 shows that the model trained with GraphRAG-annotated data performs poorly on its validation set, indicating low annotation quality.

Zone Partitioning Algorithm. Row 2 of Table 4 evaluates the instruction quality using zones from hierarchical clustering (Xie et al., 2024). Compared to our proposed zone partitioning algorithm, hierarchical clustering relies solely on the distance between different viewpoints, disregarding the spatial layout of the environment (*e.g.*, wall partitions) and lacking environmental semantic understanding.

Role Simulation and User Demands. To enhance the diversity of instructions and better match user demands, we design prompts that guide the LLM to simulate a user with a specific role profile and generate instructions to the agent in everyday scenarios. As shown in rows 3 and 4 of Table 5, we analyze the impact of role simulation and user demands on the quality of NavRAG-generated instructions. When user demands are not utilized for training data generation, performance significantly decreases in validation data with diverse user demands (Table 5, row 3). However, if user demands are included in the training data but removed from the validation data, the model still maintains strong performance. The experimental results indicate that enhancing the diversity of generated instructions by simulating user roles and incorporating user demands is feasible. Moreover, more diverse instructions can provide the model with stronger generalizability and performance.

5 Conclusion

In this work, we propose NavRAG, a user demand-oriented navigation data generation method through retrieval-augmented LLM. Unlike previous works that use trajectory-based instruction generators to translate navigation videos into step-by-step instructions, our NavRAG utilizes the environmental representations from a hierarchical scene description tree. By retrieving descriptions of different levels in a top-down manner and introducing the user demands, NavRAG effectively enhances the quality of instructions generated by the LLM.

6 Limitations

1) Although the strong navigation performance shows the quality of the NavRAG dataset, no effective method exists to evaluate the correctness of generated instructions. Previous approaches evaluate instruction generators by comparing generated instructions with human-annotated instructions (*e.g.*, using metrics like Bleu, SPICE, and CIDEr). However, our experiments show that small-scale human annotations lack diversity and are insufficient for accurately evaluating dataset quality. 2) The navigation targets annotated by NavRAG are limited to the viewpoint-level, failing to precisely locate specific target objects and their positions, which restricts its applicability in object-centered tasks such as mobile manipulation.

References

Dong An, Yuankai Qi, Yangguang Li, Yan Huang, Liang Wang, Tieniu Tan, and Jing Shao. 2023. Bevbert: Multimodal map pre-training for language-guided navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2737–2748.

Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton Van Den Hengel. 2018. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3674–3683.

Meghan Booker, Grayson Byrd, Bethany Kemp, Aurora Schmidt, and Corban Rivera. 2024. Embodiedrag: Dynamic 3d scene graph retrieval for efficient and scalable robot task planning. *arXiv preprint arXiv:2410.23968*.

Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niebner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. 2017. Matterport3d: Learning from rgb-d data in indoor environments. In *International Conference on 3D Vision (3DV)*.

Jiaqi Chen, Bingqian Lin, Ran Xu, Zhenhua Chai, Xiaodan Liang, and Kwan-Yee K. Wong. 2024. Mapgpt: Map-guided prompting with adaptive path planning for vision-and-language navigation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*.

Shizhe Chen, Pierre-Louis Guhur, Cordelia Schmid, and Ivan Laptev. 2021. History aware multimodal transformer for vision-and-language navigation. *Advances in neural information processing systems*, 34:5834–5847.

Shizhe Chen, Pierre-Louis Guhur, Makarand Tapaswi, Cordelia Schmid, and Ivan Laptev. 2022a. Learning from unlabeled 3d environments for vision-and-language navigation. In *European Conference on Computer Vision*, pages 638–655. Springer.

Shizhe Chen, Pierre-Louis Guhur, Makarand Tapaswi, Cordelia Schmid, and Ivan Laptev. 2022b. Think global, act local: Dual-scale graph transformer for vision-and-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16537–16547.

Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. 2022. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1290–1299.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, and Jonathan Larson. 2024. From local to global: A graph rag approach to query-focused summarization. *arXiv preprint arXiv:2404.16130*.

Sheng Fan, Rui Liu, Wenguan Wang, and Yi Yang. 2025. Navigation instruction generation with bev perception and large language models. In *European Conference on Computer Vision*, pages 368–387. Springer.

Daniel Fried, Ronghang Hu, Volkan Cirik, Anna Rohrbach, Jacob Andreas, Louis-Philippe Morency, Taylor Berg-Kirkpatrick, Kate Saenko, Dan Klein, and Trevor Darrell. 2018. Speaker-follower models for vision-and-language navigation. *Advances in neural information processing systems*, 31.

Weituo Hao, Chunyuan Li, Xiujun Li, Lawrence Carin, and Jianfeng Gao. 2020. Towards learning a generic agent for vision-and-language navigation via pre-training. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13137–13146.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Aishwarya Kamath, Peter Anderson, Su Wang, Jing Yu Koh, Alexander Ku, Austin Waters, Yinfei Yang, Jason Baldridge, and Zarana Parekh. 2023. A new path: Scaling vision-and-language navigation with synthetic instructions and imitation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10813–10823.

Jing Yu Koh, Harsh Agrawal, Dhruv Batra, Richard Tucker, Austin Waters, Honglak Lee, Yinfei Yang, Jason Baldridge, and Peter Anderson. 2023. Simple and effective synthesis of indoor 3d scenes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 1169–1178.

Xianghao Kong, Jinyu Chen, Wenguan Wang, Hang Su, Xiaolin Hu, Yi Yang, and Si Liu. 2025. Controllable navigation instruction generation with chain of thought prompting. In *European Conference on Computer Vision*, pages 37–54. Springer.

Jacob Krantz, Erik Wijmans, Arjun Majumdar, Dhruv Batra, and Stefan Lee. 2020. Beyond the nav-graph: Vision-and-language navigation in continuous environments. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVIII 16*, pages 104–120. Springer.

Alexander Ku, Peter Anderson, Roma Patel, Eugene Ie, and Jason Baldridge. 2020. Room-across-room: Multilingual vision-and-language navigation with dense spatiotemporal grounding. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4392–4412.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Xiangyang Li, Zihan Wang, Jiahao Yang, Yaowei Wang, and Shuqiang Jiang. 2023. Kerm: Knowledge enhanced reasoning for vision-and-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2583–2592.

Kunyang Lin, Peihao Chen, Diwei Huang, Thomas H Li, Mingkui Tan, and Chuang Gan. 2023. Learning vision-and-language navigation from youtube videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8317–8326.

Rui Liu, Wenguan Wang, and Yi Yang. 2024a. Vision-language navigation with energy-based policy. In *Advances in Neural Information Processing Systems*.

Rui Liu, Wenguan Wang, and Yi Yang. 2024b. Volumetric environment representation for vision-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16317–16328.

Rui Liu, Xiaohan Wang, Wenguan Wang, and Yi Yang. 2023. Bird's-eye-view scene graph for vision-language navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10968–10980.

Yuankai Qi, Qi Wu, Peter Anderson, Xin Wang, William Yang Wang, Chunhua Shen, and Anton van den Hengel. 2020. Reverie: Remote embodied visual referring expression in real indoor environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9982–9991.

Yanyuan Qiao, Qianyi Liu, Jiajun Liu, Jing Liu, and Qi Wu. 2024. Llm as copilot for coarse-grained vision-and-language navigation. In *European Conference on Computer Vision*, pages 459–476. Springer.

Yanyuan Qiao, Yuankai Qi, Zheng Yu, Jing Liu, and Qi Wu. 2023. March in chat: Interactive prompting for remote embodied referring expression. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15758–15767.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Santhosh Kumar Ramakrishnan, Aaron Gokaslan, Erik Wijmans, Oleksandr Maksymets, Alexander Clegg, John M Turner, Eric Undersander, Wojciech Galuba, Andrew Westbury, Angel X Chang, et al. Habitat-matterport 3d dataset (hm3d): 1000 large-scale 3d environments for embodied ai. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. 2019. Habitat: A platform for embodied ai research. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9339–9347.

Zhihong Shao, Yeyun Gong, Yelong Shen, Minlie Huang, Nan Duan, and Weizhu Chen. 2023. Enhancing retrieval-augmented large language models with iterative retrieval-generation synergy. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 9248–9274.

Hao Tan, Licheng Yu, and Mohit Bansal. 2019. Learning to navigate unseen environments: Back translation with environmental dropout. *arXiv preprint arXiv:1904.04195*.

Liuyi Wang, Zongtao He, Ronghao Dang, Mengjiao Shen, Chengju Liu, and Qijun Chen. 2024. Vision-and-language navigation via causal learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13139–13150.

Su Wang, Ceslee Montgomery, Jordi Orbay, Vighnesh Birodkar, Aleksandra Faust, Izzeddin Gur, Natasha Jaques, Austin Waters, Jason Baldridge, and Peter Anderson. 2022. Less is more: Generating grounded navigation instructions from landmarks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15428–15438.

Xiaohan Wang, Wenguan Wang, Jiayi Shao, and Yi Yang. 2023a. Lana: A language-capable navigator for instruction following and generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19048–19058.

Zihan Wang, Xiangyang Li, Jiahao Yang, Yeqi Liu, and Shuqiang Jiang. 2023b. Gridmm: Grid memory map for vision-and-language navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15625–15636.

Zun Wang, Jialu Li, Yicong Hong, Yi Wang, Qi Wu, Mohit Bansal, Stephen Gould, Hao Tan, and Yu Qiao. 2023c. Scaling data generation in vision-and-language navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12009–12020.

Fei Xia, Amir R Zamir, Zhiyang He, Alexander Sax, Jitendra Malik, and Silvio Savarese. 2018. Gibson env: Real-world perception for embodied agents. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9068–9079.

Quanting Xie, So Yeon Min, Tianyi Zhang, Kedi Xu, Aarav Bajaj, Ruslan Salakhutdinov, Matthew Johnson-Roberson, and Yonatan Bisk. 2024. Embodied-rag: General non-parametric embodied memory for retrieval and generation. *arXiv preprint arXiv:2409.18313*.

Haitian Zeng, Xiaohan Wang, Wenguan Wang, and Yi Yang. 2023. Kefa: A knowledge enhanced and fine-grained aligned speaker for navigation instruction generation. *arXiv preprint arXiv:2307.13368*.

Yue Zhang and Parisa Kordjamshidi. 2023. Vln-trans: Translator for the vision and language navigation agent. *arXiv preprint arXiv:2302.09230*.

Shengyu Zhao, Jonathan Cui, Yilun Sheng, Yue Dong, Xiao Liang, I Eric, Chao Chang, and Yan Xu. Large scale image completion via co-modulated generative adversarial networks. In *International Conference on Learning Representations*.

Duo Zheng, Shijia Huang, Lin Zhao, Yiwu Zhong, and Liwei Wang. 2024. Towards learning a generalist model for embodied navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13624–13634.

Gengze Zhou, Yicong Hong, and Qi Wu. 2024. Navgpt: Explicit reasoning in vision-and-language navigation with large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 7641–7649.

Fengda Zhu, Xiwen Liang, Yi Zhu, Qizhi Yu, Xiaojun Chang, and Xiaodan Liang. 2021. Soon: Scenario oriented object navigation with graph-based exploration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12689–12699.