Cross from Left to Right Brain: Adaptive Text Dreamer for Vision-and-Language Navigation

Pingrui Zhang[1,2], **Yifei Su**[3,4], **Pengyuan Wu**[2], **Dong An**[5], **Li Zhang**[6], **Zhigang Wang**[2],
Dong Wang[2], **Yan Ding**[2], **Bin Zhao**[2], **Xuelong Li**[7†]

[1]Fudan University [2]Shanghai AI Laboratory
[3]School of Artificial Intelligence, University of Chinese Academy of Sciences
[4]MAIS, Institute of Automation of Chinese Academy of Sciences
[5]Mohamed bin Zayed University of Artificial Intelligence
[6]University of Science and Technology of China
[7]TeleAI, China Telecom Corp Ltd
zhangpingrui@pjlab.org.cn

Abstract

Vision-and-Language Navigation (VLN) requires the agent to navigate by following natural instructions under partial observability, making it difficult to align perception with language. Recent methods mitigate this by imagining future scenes, yet they rely on vision-based synthesis, leading to high computational cost and redundant details. To this end, we propose to adaptively imagine key environmental semantics via *language* form, enabling a more reliable and efficient strategy. Specifically, we introduce a novel Adaptive Text Dreamer (ATD), a dual-branch self-guided imagination policy built upon a large language model (LLM). ATD is designed with a human-like left-right brain architecture, where the left brain focuses on logical integration, and the right brain is responsible for imaginative prediction of future scenes. To achieve this, we fine-tune only the Q-former within both brains to efficiently activate domain-specific knowledge in the LLM, enabling dynamic updates of logical reasoning and imagination during navigation. Furthermore, we introduce a cross-interaction mechanism to regularize the imagined outputs and inject them into a navigation expert module, allowing ATD to jointly exploit both the reasoning capacity of the LLM and the expertise of the navigation model. We conduct extensive experiments on the R2R benchmark, where ATD achieves state-of-the-art performance with fewer parameters. The code is here.

1 Introduction

Vision-and-Language Navigation (VLN) [4] is a popular Embodied AI task where an agent follows natural language instructions to reach a goal in previously unseen 3D environments. A fundamental challenge of VLN lies in its partially observable nature—agents perceive only a limited field of view at each step, while relevant landmarks or goals may lie far beyond. This restricted observability introduces substantial uncertainty in grounding language to the environment, requiring the agent to reason beyond immediate perception and infer unobserved semantic cues for successful navigation.

To address this uncertainty, early approaches introduced memory-based mechanisms [17, 18, 32, 89, 92] to help agents aggregate past observations or build spatial representations of traversed regions. While beneficial, memory alone is insufficient for anticipating critical, yet-unseen information, especially when reasoning about instruction-relevant targets outside the current or historical view. To overcome this, recent works have proposed imagination-based strategies [51, 77] to extend the agent's

Figure 1: Given the task "Walk down the stairs and walk towards the red sofa. Turn left and enter the bathroom", and assuming 3 candidate viewpoints are encountered during the current navigation step. Left: Existing methods generate the visual imagination for each candidate. While effective, the rendered images often contain blurry from GT or redundant regions, increasing both imagination time and alignment difficulty. Right: Our ATD predicts the key future semantics (red color) in textual form, thereby facilitating the construction of compact yet semantically rich future representations with improved efficiency. ●: Visited Node. ●: Candidate Node. □: Key Environmental Semantic.

perceptual horizon by simulating observations of unvisited locations. These approaches enable agents to proactively infer what might lie ahead, reducing uncertainty and improving long-horizon planning.

While promising, existing imagination-based methods primarily rely on visually grounded generation, which presents several limitations. For example, DREAMWALKER [83] generates pixel-level images along candidate trajectories, incurring high computational cost and suffering from noise accumulation. HNR [91] improves efficiency through feature-level synthesis, yet still depends on encoding full panoramic views, often introducing redundant or distracting information. In contrast, we argue that effective navigation hinges not on full-scene reconstruction, but on identifying task-relevant semantics. For example, given the instruction "Walk down the stairs and walk towards the red sofa.", the core challenge lies in inferring where the `red sofa` might appear—while other visual content remains largely irrelevant (Fig. 1). This insight motivates a shift toward more abstract, selective imagination. Language, with its compositional and abstract representation, is naturally suited to support such targeted imagination [54].

Motivated by this, we propose to imagine the environment through language and introduce **A**daptive **T**ext **D**reamer (**ATD**), a dual-branch LLM self-guided dynamic imagination policy for VLN. The model consists of two branches of LLMs: one estimates the current navigation state (**Left brain: State Estimation LLM**), while the other generates imaginative descriptions for the future pathway (**Right brain: Imagination LLM**). Similar to the human brain, the left hemisphere is responsible for logical integration, while the right hemisphere excels in divergent thinking. For both the left and right brains, we use Q-Former [50] to perform visual instruction tuning on the frozen InstructBLIP [22], using the respective text-predicted ground truth collected for each brain's function, thereby activating the corresponding domain's commonsense knowledge within the LLM. During navigation, the left brain constrains the imagination of the right brain. This architecture allows the imagination of the text dreamer to be continuously updated throughout the navigation process, preventing overemphasis on the completed parts of the navigation instruction, which could otherwise negatively impact the imagination. To bridge the LLM with the policy, we use the latent adaptive cross-interactive imagination embedding of ATD to inject into the node embedding of the action expert policy. This latent alignment structure allows us to preserve the rich linguistic reasoning capabilities of the LLM while fully leveraging the specialized skills of the navigation action expert [8, 9, 95].

We evaluate our proposed ATD on the R2R [4] benchmark and observe clear improvements over prior approaches with fewer parameters. Specifically, our method achieves gains in Success Rate (SR) of 8.0% and 12.0%, and in Success weighted by Path Length (SPL) of 5.0% and 11.0%, on the `val seen` and `val unseen`, respectively. These results highlight the efficiency and effectiveness of our proposed **Adaptive Text Dreamer** guided navigation. Our contributions are summarized as follows:

(1) We highlight the advantage of using language for future imagination and propose a human-like left-right brain structure to enable an adaptive text dreamer. This approach utilizes one LLM to guide another, resulting in enhanced and more accurate imaginative capabilities. (2) We design a scheme to train the State Estimation LLM and Imagination LLM to predict the key environmental semantics during navigation. (3) Our method achieves state-of-the-art performance on both SR and SPL metrics, while maintaining fewer parameters.

2   Related work

Vision-and-language Navigation (VLN).   VLN requires agents to navigate complex environments via visual clues to follow various human instructions, e.g., step-by-step commands [4, 12, 44], high-level instructions [45, 69, 78], and chat dialog [26, 63, 64, 81]. Impressive progress has been achieved mainly by three factors: (1) Data Augmentation. Some work [47, 48, 86] benefit from augmenting the scenes and observations, while others [25, 94] resort to enriching the instructions. Besides, the speaker-follower paradigm [27, 80, 84, 88, 93] effectively scales VLN performance via abundant instruction-path pairs. (2) Planning Strategy. Beyond classical imitation learning [4, 18, 37, 74] and reinforcement learning [32, 85, 89] strategies, various auxiliary tasks [58, 59, 112], explicit external knowledge [28, 51, 72], and rollback mechanisms [38, 59] further advance the VLN performance. (3) Representation Alignment Learning. Well-designed historical observation representation, e.g., recurrent vectors [32], trajectory views [17, 68], topological map [18, 24], grid maps [2, 92], and feature volume [55], offer solid foundation for representation alignment in VLN. Building on this, various pertaining based on proxy-tasks [2, 18, 31, 67, 70, 71, 102] and vision-language models [35, 108] have significantly boosted VLN performance.

Imagination in VLN.   Given the partially observable Markov property of VLN, recent works introduce future scene imagination to enhance multimodal alignment [21, 65, 83, 91]. Initially, MIND [49] generates future sub-goal images prior to decision-making to facilitate embodied QA [23, 105]. PathDreamer [41] built a world model to predict future trajectory images and integrate a text-path compatibility model [106] for VLN. Similarly, VLN-SIG [46] and ImagineNav [107] adopt visual cookbook techniques and generative models, respectively, to imagine future candidate observations and make decisions via similarity measures or LLM reasoning. DreamWalker [83] further extends this paradigm to continuous environments [43] and leverages Monte Carlo Tree search [30, 40] for decision-making. NWM [6] first scales the navigation world model to the 1B level, enabling long-horizon imagination in complex real environments. HNR [91] and UnitedVLN [21] leverage the advanced volumetric rendering [7, 61, 62] and 3DGS [11, 19, 39] to generate future observations, respectively, boosting VLN-CE performance. Beyond predicting RGB, some works assist navigation by imagining semantic maps [104], geometric maps [77], and hybrid imagination-reality memory [67] of scenes. This work avoids costly future generations by encouraging the model's right brain to imagine candidate node semantics during training, implicitly encoding future information without accessing future poses.

VLN with LLM.   LM-Nav [76] uses GPT-3 [60] to extract landmarks from instructions to assist navigation. MiCs generate detailed plans from static and dynamic perspectives as knowledge. LLM-Planner [79] and SayNav [73] utilize LLMs [1, 60] as core planners to dynamically plan high-level actions. As for VLN, NavGPT [111] builds an LLM agent that converts tasks, observations, and actions into text for zero-shot decision-making. Subsequent works further integrate this paradigm with map-guided prompting [14], multi-expert discussions [57], memory maps [98], and hierarchical reasoning [103]. Other works explore applying LLMs in continuous environments. A^{2}Nav [16] uses GPT-3 [60] to decompose instructions into a few sub-tasks and trains separate navigators for execution. InstructNav [56] and CA-Nav [15] extend the decomposition paradigm with versatile value maps to guide navigation. AO-Planner [13] and Select2Plan [10] design navigable points within images as prompts to guide VLM to navigate. Benefiting from the continuous development of parameter-efficient fine-tuning techniques [33, 34, 36, 87], some work fine-tune VLMs with in-domain VLN data for better performance. Zheng *et al.* [108] trains a general navigator via casting various tasks into a unified QA form. Subsequent efforts [20, 99, 100] extend it to continuous environments by predicting meta-actions [43] or waypoints [42] in text form. MapNav [101] and P3Nav [109] further explore the utility of semantic maps and adaptive historical sampling as inputs.
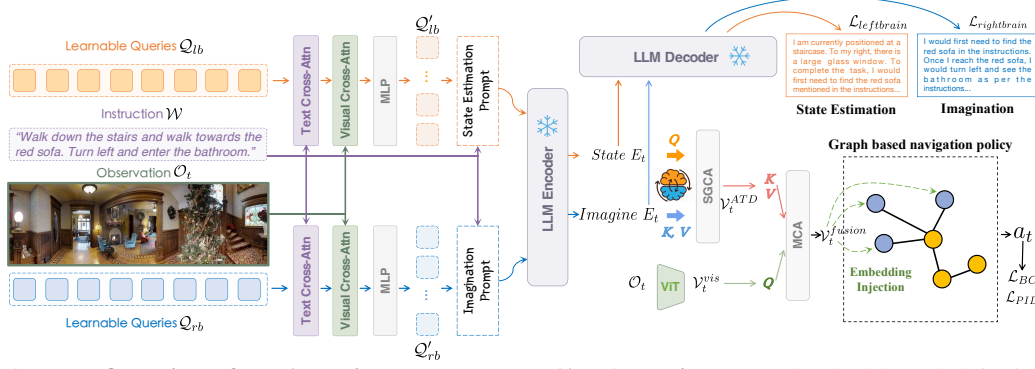
Figure 2: **Overview of the Adaptive Text Dreamer (ATD) architecture.** The dual-branch left–right vision-language model (VLM) structure consists of a left-brain branch that performs state estimation using a large language model (LLM), and a right-brain branch that generates future candidate imaginations constrained by latent embeddings derived from the state estimation. The imagined latent representations are then integrated into a graph-based navigation expert to guide action decisions.

Beyond predicting actions, introducing intermediate reasoning tasks during navigation [96, 110] can enhance interpretability.

3   Method

As illustrated in Fig. 2, the proposed Adaptive Text Dreamer (ATD) architecture consists of a dual-branch left–right reasoning structure and a graph-based navigation expert. We design the left-right VLM branches to support the VLN task in a complementary manner. The left-brain VLM employs a Q-Former module to extract image tokens, which are then fed into a large language model (LLM) to generate state estimation (Sec.3.1). The right-brain VLM shares the same architecture as the left brain but is responsible for generating imagination of future candidates (Sec.3.1). To ensure the imagination within the current context, we constrain it via latent embeddings derived from the state estimation. The attended imagination latent is then injected into the graph-based navigation policy to guide the action decision process (Sec. 3.2).

Task Setup.   In Vision-and-Language Navigation (VLN), the environment is represented as an undirected graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{V} = \{V_i\}_{i=1}^{K}$ denotes the set of K navigable nodes, and $\mathcal{E}$ represents the connectivity edges between nodes. The agent is initialized at a starting node $V_0 \in \mathcal{V}$ and is provided with a natural language instruction $\mathcal{W} = \{w_i\}_{i=1}^{L}$ consisting of L words. The goal of the agent is to navigate through the graph based on the instruction and reach the target location or object specified by the instruction. At each time step t, the agent perceives its surrounding environment by observing a set of RGB views $\mathcal{O}_t \triangleq \langle o_i, a_i \rangle_{i=1}^{N}$ for each connected navigable node candidate. Here, N denotes the number of candidate nodes, where each view o_i (for $i \leq N$) is associated with a direction angle a_i relative to the agent's heading. The agent selects the next action by predicting the relative angle a_t from $\mathcal{O}_t$. The agent's policy π, parameterized by Θ, is learned to predict the action a_t based on the given instruction $\mathcal{W}$ and the observed set of views $\mathcal{O}_t$, i.e., $\pi(a_t|\mathcal{W}, \mathcal{O}_t; \Theta)$.

3.1   State Estimation LLM and Imagination LLM

In this section, we introduce the process of collecting instruction data and fine-tuning frozen Instruct-BLIP [22] with Q-former [50] to obtain the two "brains" of our ATD model. The training data for these two LLMs is collected in the discrete simulation of VLN, as it has a predefined graph that allows us to pre-extract visual features for captioning or reasoning.

State Estimation LLM.   State estimation is always an important and challenging problem in VLN, as determining the current stage of navigation within the instruction helps to exclude interference from completed parts of the text information. Some previous work [53] uses object detection to update the landmark list as a representation of state updates. However, this approach is rigid and

lacks flexibility, as it does not involve true reasoning to assess the progress of navigation. Simply determining whether an object appears in the field of view may overlook whether the object actually matches the spatial relationship specified for the landmark in the instruction, thus failing to ensure that it is indeed the object described in the instruction.

Thus, the flexible reasoning ability of the LLM is leveraged for state estimation. Following [110], the state description is collected by feeding observations and instructions into GPT-4V [1]. In the prompt, GPT-4V is tasked with reasoning about the agent's current position in the environment based on a single visual input and the navigation instructions, while determining the current state of the navigation. The detailed prompt can be found in the appendix. The reasoning output $\mathcal{R}_t$ of GPT-4V is then used to fine-tune an LLM that with fewer parameters, distilling the state estimation capability into the model. During the training process of state estimation, the model learns to predict the current state based on the observations and instructions, i.e., the left brain. Specifically, the process of training the State Estimation LLM, including its inputs and outputs, can be formulated as follows:

$$\mathcal{Q}'_{lb} = \text{Q-former}^{lb}(\mathcal{W}, \mathcal{O}_t, \mathcal{Q}_{lb}), \tag{1}$$

$$\langle \hat{\mathcal{R}}_t, State\ E_t \rangle = \text{LLM}_{frozen}(\text{StateEsimationPrompt}(\mathcal{W}, \mathcal{Q}'_{lb})). \tag{2}$$

Imagination LLM. As mentioned below in Sec. 1, in order to represent the prediction of future key environmental semantics using highly abstract language, it is essential to train an LLM as a text dreamer, i.e., the right brain. To achieve this, we formulate the process of generating imaginative representations as follows:

$$\mathcal{Q}'_{rb} = \text{Q-former}^{rb}(\mathcal{W}, \mathcal{O}_t, \mathcal{Q}_{rb}), \tag{3}$$

$$\langle \mathcal{I}_t, Imagine\ E_t \rangle = \text{LLM}_{frozen}(\text{ImaginationPrompt}(\mathcal{W}, \mathcal{Q}'_{rb})), \tag{4}$$

where $\mathcal{Q}_{lb}, \mathcal{Q}_{rb} \in \mathbb{R}^{n \times d_1}$ represents the learnable query tokens for Q-former, $\mathcal{Q}'_{lb}, \mathcal{Q}'_{rb} \in \mathbb{R}^{n \times d_1}$ denotes the text-visual fusion output from the left-right brain branch, $\hat{\mathcal{R}}_t, \mathcal{I}_t$ is the reasoning and imagination output generated by the LLM at time step t, and $State\ E_t, Imagine\ E_t \in \mathbb{R}^{m \times d_2}$ represents the hidden state and imagine embedding feature of the LLM. Both $\text{StateEstimationPrompt}$ and ImaginationPrompt are designed to integrate visual and textual information for input to subsequent LLMs. The specific details of the prompts can be found in the appendix.

To obtain the Imagination LLM, long and detailed captions $\{\mathcal{C}^i_{candidate\_t}\}^N_{i=1}$ for N candidate nodes at each sampled location in the trajectory are collected using Qwen2.5-VL-7B [5] as the ground truth for the text dreamer. Then, the ImaginationPrompt directs the LLM to imagine what scenes or objects might appear in different directions based on the current node's observation and the given instruction for the current episode.

Loss. The loss is computed at each sampled point in the trajectory to train the two branches of the LLM. Cross-entropy loss is used during training of the brains, as shown in the following equations:

$$\mathcal{L}_{leftbrain} = -\sum_{t=1}^{T} \mathcal{R}_t \log(\hat{\mathcal{R}}_t), \quad \mathcal{L}_{rightbrain} = -\sum_{t=1}^{T}\sum_{i=1}^{N} \{\mathcal{C}^i_{candidate\_t}\} \log(\mathcal{I}_t), \tag{5}$$

After this training stage, we obtain the left and right brain LLMs, which are used to guide the subsequent navigation policy and further enhance the model's decision-making.

3.2 Adaptive Text Dreamer

After the aforementioned training, the State Estimation LLM and Imagination LLM are obtained. In this section, these two LLMs are integrated as the two brains into the navigation policy for training, resulting in the final navigation policy—Adaptive Text Dreamer (ATD).

State Grounded Imagination. The imagination of the right brain is unconstrained, allowing it full freedom to envision future scenarios. At the same time, to prevent irrelevant details from the imagination from excessively interfering with navigation, the imagination is constrained using the navigation state estimation information. Thus, state information is employed to ground the important and relevant details from the imagination through State Grounded Cross-Attention (SGCA) layers. We serve state estimation embedding $State\ E_t$ as queries and the imagination embedding $Imagine\ E_t$

as keys and values, and then compute their cross-attention weight matrix $\mathbf{A} \in \mathbb{R}^{n \times N \times M}$ [82]. The process is formulated as follows:

$$\mathbf{Q}_S = State\ E_t \mathbf{W}^Q, \mathbf{K}_I = Imagine\ E_t \mathbf{W}^K, \mathbf{V}_I = Imagine\ E_t \mathbf{W}^V \tag{6}$$

$$\mathbf{A} = \mathrm{SoftMax}(\mathrm{Sim}_{\cos}(\mathbf{Q}_S, \mathbf{K}_I)), \tag{7}$$

where $\mathbf{Q}_S \in \mathbb{R}^{N_q \times d}, \mathbf{K}_I, \mathbf{V}_I \in \mathbb{R}^{n \times M_{kv} \times d}$. $\mathrm{Sim}_{\cos}$ denotes the cosine similarity function. The attention matrix $\mathbf{A} \in \mathbb{R}^{n \times N_q \times M_{kv}}$ represents the constraint weights from state estimation to imagination, guiding the refinement of the information in the imagination. Thus, when the state changes, $\mathbf{A}$ adjusts accordingly to filter the imagination information. This adaptive refinement process, known as SGCA, can be mathematically expressed as:

$$\mathrm{SGCA}(\mathbf{Q}_S, \mathbf{K}_I, \mathbf{V}_I) = \mathbf{A} \cdot \mathbf{V}_I. \tag{8}$$

The grounded feature is then fed into the node embedding of the subsequent navigation policy.

Graph-based Navigation Policy. To ensure efficient action planning, a navigation policy based on a dynamic topological graph [18] is employed, which updates in real-time to encode navigation history through nodes and edges. Edges record traversable paths, allowing the graph to explicitly represent the environmental topology and spatial relationships. Nodes are categorized into visited locations, navigable neighbors, and the current position, with new nodes (e.g., unvisited adjacent states) added incrementally as the agent explores. The policy utilizes a multi-layer transformer to update each node's visual embeddings, denoted as $\mathcal{V}_t^{vis}$, which represents the average pooling of the observed view representations surrounding each node.

Latent Embedding Injection. The output of SGCA is the ATD embedding $\mathcal{V}_t^{ATD}$ of the node. The adaptive imagination information is then integrated into the node embedding of the policy graph. Multi-head Cross-Attention (MCA) is used to fuse these two node embeddings, with $\mathcal{V}^{vis}t$ as the query and $\mathcal{V}_t^{ATD}$ as both the key and value. This process is formulated as follows:

$$\mathcal{V}_t^{fusion} = \mathrm{MCA}(\mathcal{V}_t^{ATD}, \mathcal{V}_t^{vis}), \tag{9}$$

where the $\mathcal{V}_t^{fusion}$ represents the fused embedding of the graph node. At step t, a navigation graph $\mathcal{G}_t = \mathcal{V}_t, \mathcal{E}_t$ is constructed, where $\mathcal{G}_t \subset \mathcal{G}$. The node embeddings $\mathcal{V}_t$ are passed through a multi-layer cross-modal transformer to capture the relationship between the instructions and the nodes. Initially, the node embeddings undergo cross-attention with the instructions encoded by the LLM. Afterward, a graph-aware self-attention (GASA) layer is applied, which enhances the contextual understanding by considering both the spatial distances and visual similarities between nodes. The GASA operation is formulated as follows:

$$\mathrm{GASA}(\mathcal{V}) = \mathrm{Softmax}\left(\frac{\mathcal{V}W_q(\mathcal{V}W_k)^T}{\sqrt{d}} + EW_e\right)\mathcal{V}W_v, \tag{10}$$

E is the pairwise distance matrix derived from the graph edges $\mathcal{E}_t$. The matrices W_q, W_k, W_e, and W_v are learnable parameters. $\mathcal{V}$ represents the node embedding, which, in this case, is $\mathcal{V}^{fusion}$. The output of GASA is then fed into a Multi-Layer Perceptron (MLP) to predict the next goal action.

Loss. In the loss computation process, we follow previous imitation learning methods [18, 75], using both Behaviour Cloning (BC) loss and Pseudo Interactive Demonstrator (PID) loss to supervise the model. The loss formulas are as follows:

$$\mathcal{L}_{\mathrm{BC}} = -\sum_{t=1}^{T} \log \pi(a_t^* | \mathcal{W}, \mathcal{G}_t), \quad \mathcal{L}_{\mathrm{PID}} = -\sum_{t=1}^{T} \log \pi(\tilde{a}_t^* | \mathcal{W}, \tilde{\mathcal{G}}_t), \tag{11}$$

where a_t^* represents the ground truth action, and $\tilde{a}_t^*$ denotes the pseudo label, which is determined by the shortest path to the destination from the partial graph $\tilde{\mathcal{G}}_t$ generated by the agent through on-policy action sampling. This pseudo action closely resembles the action of the optimal policy π, which selects the next target node based on the shortest distance to the final destination using the environment graph G. Finally, The total loss function is given by $\mathcal{L} = \lambda \mathcal{L}_{\mathrm{BC}} + \mathcal{L}_{\mathrm{PID}}$, where λ acts as a trade-off factor.

Table 1: **Performance comparison on the R2R dataset.** ATD ooutperforms all previous methods that utilize LLMs and closes the performance gap among state-of-the-art methods trained at a small scale. †: Indicates methods that leverage additional visual data beyond MP3D.

Methods	Freeze LLM	Val Seen					Val Unseen					Test Unseen				
		TL	NE↓	OSR↑	SR↑	SPL↑	TL	NE↓	OSR↑	SR↑	SPL↑	TL	NE↓	OSR↑	SR↑	SPL↑
Human	–	–	–	–	–	–	–	–	–	–	–	11.85	1.61	90	86	76
Seq2Seq [4]	–	11.33	6.01	53	39	–	8.39	7.81	28	21	–	8.13	7.85	27	20	–
RCM [90]	–	10.65	3.53	75	67	–	11.46	6.09	50	43	–	11.97	6.12	50	43	38
Speaker Follower [27]	–	–	3.36	74	66	–	–	6.62	45	36	–	14.82	6.62	-	35	28
EnvDrop [80]	–	11.00	3.99	–	62	59	10.70	5.22	–	52	48	11.66	5.23	59	51	47
VLN Specialists with Vision-Language-Action Pretraining:																
PREVALENT [31]	–	10.32	3.67	–	69	65	10.19	4.71	–	58	53	10.51	5.30	61	54	51
AirBert [29]†	–	11.09	2.68	–	75	70	11.78	4.10	–	62	56	12.41	4.13	–	62	57
VLNBert [32]	–	11.13	2.90	–	72	68	12.01	3.93	–	63	57	12.35	4.09	70	63	57
MARVAL [37]†	–	10.60	2.99	–	73	69	10.15	4.06	–	65	61	10.22	4.18	67	62	58
HAMT [17]	–	11.15	2.51	–	76	72	11.46	2.29	–	66	61	12.27	3.93	72	65	60
HOP+ [71]	–	11.31	2.33	–	78	73	11.76	3.49	–	67	61	12.67	3.71	–	66	60
DUET [18]	–	12.32	2.28	86	79	73	13.94	3.31	81	72	60	14.73	3.65	76	69	59
BEVBert [2]	–	13.56	2.17	88	**81**	**74**	14.55	2.81	84	**75**	**64**	15.87	3.13	81	**73**	**62**
Language Models zero-shot VLN:																
NavGPT (GPT-4) [111]	✓	–	–	–	–	–	11.45	6.46	42	34	29	–	–	–	–	–
MapGPT (GPT-4) [14]	✓	–	–	–	–	–	–	6.92	58	39	26	–	–	–	–	–
DiscussNav (GPT-4) [57]	✓	–	–	–	–	–	9.69	5.32	61	43	40	–	–	–	–	–
Langage Models with/as VLN Specialists:																
NavCoT (LLaMA2-7B) [52]	✗	10.08	6.46	48	41	38	9.95	6.26	48	40	37	–	–	–	–	–
LangNav (LLaMA2-7B) [66]†	✗	–	–	–	56	–	–	–	–	46	–	–	–	–	–	–
NaviLLM (Vicuna-7B) [108]	✗	–	–	–	–	–	12.81	3.51	–	67	59	13.21	3.71	–	68	60
NavGPT2 (FlanT5-XL-1.5B) [110]	✓	13.02	3.34	74	69	62	13.68	3.37	74	68	56	–	–	–	–	–
w/ PREVALENT	✓	12.44	2.97	80	73	65	12.81	3.33	79	70	59	13.51	3.40	77	71	60
NavGPT2 (FlanT5-XXL-5B) [110]	✓	13.08	2.98	79	74	65	13.25	3.18	80	71	60	–	–	–	–	–
w/ PREVALENT	✓	14.13	2.84	83	74	63	14.01	2.98	84	74	61	14.74	3.33	80	72	60
ATD (FlanT5-XL-1.5B)	✓	12.02	3.21	79	73	65	12.84	3.19	80	72	61	–	–	–	–	–
w/ PREVALENT	✓	12.26	2.67	82	**76**	**67**	13.33	2.81	83	**75**	**63**	14.00	3.03	80	**74**	**63**

4 Experiment

4.1 Dataset and Evaluation Metrics

We conduct systematic evaluations of the proposed model on the widely used R2R [4] benchmark in discrete environments. The R2R dataset provides step-by-step navigation instructions, with each instruction averaging 32 words and covering approximately 6 navigation steps. We evaluate the performance via a comprehensive suite of standard navigation metrics [4, 43, 69], including: Trajectory Length (TL), measuring the average path length in meters; Navigation Error (NE), denoting the average distance between the final position and destination; Success Rate (SR) the percentage of trajectories with NE less than 3 meters; Oracle Success Rate (OSR), the success rate under an ideal stopping policy; Success weighted by Path Length (SPL) [3], combining the success rate and the trajectory efficiency to comprehensively assess navigation performance.

4.2 Implementation Details.

The proposed ATD framework is built on InstructBLIP [22] with Flan-T5-xl as LLM decoder. For training the State Estimation and Imagination LLMs, we only fine-tune the Q-former module and keep the LLM and vision backbone frozen with a batch size of 1. All fine-tuning are conducted on a single GPU. The AdamW optimizer is used with a learning rate of $1e$-5 and weight decay of 0.05. For training the ATD navigation policy, we again use a single GPU, but with a batch size of 2 and the AdamW optimizer, keeping the learning rate at $1e$-5. More detailed experiment settings can be found in the supplementary material.

4.3 Quantitative Analysis

Compared Baselines. We have categorized the previous SOTA methods and listed them in a Tab. 1. Seq2Seq [4], RCM [89], Speaker Follower [27], and EnvDrop [80] are early methods that typically use architectures like recurrent neural networks and were not trained beyond the navigation policy in MP3D. Later, some methods [2, 17, 18, 29, 31, 32, 37, 71] employed a two-stage training process: first, pretraining an initialized vision-language model in the MP3D simulation, followed by fine-tuning the navigation policy with the vision-language model. These VLN specialists, utilizing their Vision-Language-Action pretraining method, achieve significant improvements over earlier approaches. Subsequently, methods using LLMs emerged. NavGPT [111], MapGPT [14] and DiscussNav [57] attempt to complete the VLN task in a training-free manner, leveraging the zero-shot
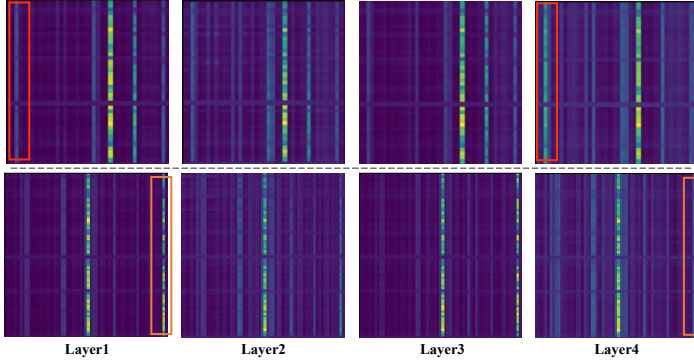
Figure 3: **Visualization of attention matrices across four SGCA layers. Row 1**: The red box shows the increasing emphasis on important information, while **Row 2**: The orange box illustrates the suppression of completed navigation steps.
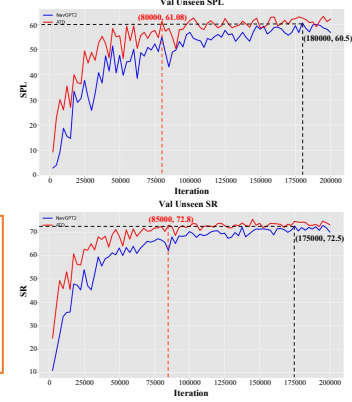
Figure 4: **Comparison of convergence speeds.** The red vertical lines highlight the points where ATD surpasses NavGPT2's maximum SPL and SR values.

capability of GPT-4 and directly prompting the LLM to make decisions for the next action. Later, some methods [52, 66, 108] fine-tune LLMs as VLN experts, while NavGPT2 [110] connects the LLM and VLN specialist through a latent space.

Main Results.   Table 1 presents the metrics of **ATD** compared to the methods mentioned above on the R2R benchmark. Compared to other LLM-leveraged methods, our approach achieves the best performance with the fewest parameters—1.5B. NaviLLM [108] is the first fully fine-tuned 7B LLM used as an action generator, and ATD outperforms it by 6% SR and 3% SPL on the test split. This indicates that maintaining the text generation capability of the LLM is crucial for navigation. Compared to NavGPT2, ATD outperforms its 1.5B model by 3% in SR and 3% in SPL on the test split. This suggests that our human-like left-right brain architecture is superior to the single-brain bridged architecture. BEVBert [2] utilizes additional depth information to construct a bird-eye-view map. ATD achieves comparable performance on the validation unseen split and outperforms it on the test split. This indicates that our ATD using an LLM's imagination ability enhances the generalization of the navigation policy.

Zero-shot Cross Environments evaluation.   We compared the cross-environment capability of ATD with that of traditional navigation policies, as shown in the Tab. 2. Both models were trained on R2R and evaluated on REVERIE [69]. It was observed that ATD outperformed by 4% in OSR, 3% in SR, and 4% in SPL. This demonstrates that our model has better generalization ability.

4.4 Qualitative Results

Attention Visualization.   As shown in Fig. 3, we visualize the attention matrices **A** produced by the SGCA at each layer. For this visualization, the number of SGCA layers is set to 4. As shown in **Row 1** of Fig. 3, a region within the red box gradually gains emphasis during the information interaction between state estimation and imagination. By the final layer (Layer 4), its attention weight becomes significantly higher compared to earlier layers. This suggests that SGCA may indeed play a role in filtering and highlighting the more important information within the imagination. In **Row 2**, the region within the orange box shows a noticeable decrease in attention weight as it propagates through the layers, becoming minimal in the final output. This may indicate that, under the supervision of state estimation, SGCA learns to suppress certain parts of the information from imagination—potentially those related to navigation steps that have already been completed.

Comparison of Convergence Speed.   Line charts of the SR and SPL metrics on the validation unseen split over training iterations are shown in Fig. 4. This figure allows us to compare the convergence speeds of NavGPT2 and ATD. The chart above in Fig. 4 shows the variation in the

Table 2: Compare zero-shot result on REVERIE.

Methods	REVERIE		
	OSR↑	SR↑	SPL↑
DUET	29	24	19
ATD$_{\text{FlanT5-XXL}}$	**33**	**27**	**23**

Table 3: **Comparison of different LLMs.**

Methods	Val Seen					Val Unseen				
	TL	NE↓	OSR↑	SR↑	SPL↑	TL	NE↓	OSR↑	SR↑	SPL↑
ATD$_{\text{FlanT5-XL}}$	12.26	2.67	81.78	**75.61**	67.49	13.33	2.81	82.76	**74.63**	63.08
ATD$_{\text{FlanT5-XXL}}$	11.38	2.47	80.51	75.51	**70.02**	11.91	2.82	81.01	73.82	**65.35**

Table 4: **Ablation of SELLM and IMLLM.** We investigate the improvement of SELLM and IMLLM on the model performance based on the baseline.

	SEM	IM	TL	NE↓	OSR↑	SR↑	SPL↑
Val Seen	✓		12.14	3.45	76	68	62
		✓	13.55	3.12	79	71	63
		✓	13.40	3.15	77	71	62
	✓	✓	12.26	**2.67**	**82**	**76**	**67**
Val Unseen	✓		13.70	4.20	72	63	52
		✓	13.23	3.05	80	72	61
		✓	13.87	2.97	80	73	60
	✓	✓	13.33	**2.81**	**83**	**75**	**63**

Table 5: **Ablation of SGCA layer numbers.**

	Metrics	Number of Layers			
		1	2	3	4
Val Seen	NE ↓	3.05	2.63	2.88	2.67
	OSR ↑	78.35	82.47	80.8	81.78
	SR ↑	71.69	75.91	**76.3**	75.61
	SPL ↑	63.15	67.46	67.13	**67.49**
Val Unseen	NE ↓	2.82	2.9	2.88	2.81
	OSR ↑	82.08	82.12	82.12	82.76
	SR ↑	74.54	73.95	**74.93**	74.63
	SPL ↑	63.15	63.05	**63.91**	63.08

SPL curve. The black line highlights the point $(180000, 60.5)$, which represents the maximum SPL value for NavGPT2, indicating its corresponding coordinates. On the left side, observing the red vertical line, we can see that ATD surpasses the maximum SPL value of NavGPT2 at around $80,000$ iterations, reaching this point $100,000$ iterations faster. Similarly, in the line chart below of Fig. 4, by observing the position of the red vertical line corresponding to the black line, we can see that ATD reaches NavGPT2's best SR metric $90,000$ iterations faster.

4.5 Ablation Study

Effect of the LLM model. To demonstrate the transferability of our method, we also implemented the ATD approach on a different LLM, as shown in the Tab. 3. FlanT5-XL has 1.5 billion parameters, while FlanT5-XXL has 5 billion. It can be observed that ATD$_{\text{FlanT5-XL}}$ achieves slightly higher SR on both `val seen` and `val unseen` sets, whereas ATD$_{\text{FlanT5-XLL}}$ shows a significant advantage in SPL. This might be because ATD$_{\text{FlanT5-XXL}}$ has higher confidence in reaching the destination and thus stops early, resulting in a higher SPL. However, if the decision is incorrect, it could lead to a lower SR.

Effectiveness of State Estimation LLM and Imagination LLM. As shown in Tab. 4, we investigate the effectiveness of two key modules in our method. When both State Estimation LLM (SELLM) and Imagination LLM (IMLLM) are removed, it corresponds to our baseline model, which is DUET without the local branch and without BERT pretraining in the MP3D simulation. Specifically, including either the SELLM or IMLLM module leads to a significant improvement over the baseline. However, the best performance is achieved when both modules interact, as seen in the full human-like left-right brain ATD. This demonstrates the effectiveness of both our approach and its structural design. Our results show that the full model outperforms others on both the `val seen` and `val unseen` datasets. Compared to the baseline, ATD shows an 8% increase in SR and a 5% increase in SPL, demonstrating strong performance in key navigation metrics. These improvements highlight the power of combining SELLM and IMLLM, underscoring the importance of both modules working together for optimal task performance.

Effect of different SGCA Layer numbers for features fusion. Tab. 5 shows the performance of ATD with SGCA layer from 1 to 4. For both **SR** and **SPL** metrics, the best performance is achieved when the number of layers is either 3 or 4. In `val seen`, increasing the number of SGCA layers significantly improves performance: SR from 71.69% to 76.3% and SPL from 63.14% to 67.49%. However, in `val unseen`, the improvement from increasing the number of layers is marginal. Surprisingly, when the number of layers is set to 1, the model performs much better on `val unseen` than on `val seen`. This suggests that ATD has a relatively high lower bound for generalization.

5 Conclusion

In this work, we have presented the Adaptive Text Dreamer (ATD), a novel dual-branch vision-language navigation framework that leverages large language models to perform dynamic and adaptive imagination through language-based reasoning. By integrating a left-brain State Estimation LLM and a right-brain Imagination LLM, our approach effectively addresses the challenge of partial observability in Vision-and-Language Navigation, enabling agents to anticipate critical and complex semantic cues beyond their immediate perception. The latent adaptive cross-interactive imagination embedding bridges the language-based imagination with a graph-based navigation policy, resulting in improved decision-making. Experimental results on the R2R benchmark demonstrate that ATD not only outperforms previous state-of-the-art methods but also achieves these gains with fewer parameters and lower computational costs. Our study highlights the potential of leveraging advanced linguistic abstractions for targeted imagination in embodied AI, paving the way for more efficient and semantically grounded navigation agents.

Cross from Left to Right Brain: Adaptive Text Dreamer for Vision-and-Language Navigation

Supplementary Material

Abstract

This supplementary material expands upon our main study by providing additional details and data to enhance the reproducibility of our proposed method ATD.

▷ **Sec. 6**: Configuration of Data Collection Progress for State Estimation LLM and Imagination LLM. This includes the design of prompts, collection of candidate node captions, and data visualization.

▷ **Sec. 7**: This section offers a detailed elucidation of the metrics, delves into the intricacies of the training process, and presents a comprehensive introduction to the comparison methods.

▷ **Sec. 8**: Additional experiments are designed and presented to investigate the effectiveness of the model.

▷ **Sec. 9**: Discusses the limitations of our work and explores prospects for future research, also discusses the broader effect of our work.

6 Instruction Data Collection

To train the State Estimation LLM and Imagination LLM, we separately collect instruction data for Q-Former fine-tuning of each LLM.

State Estimation Intruction Collection. We employ a more capable LLM [1] to perform text-based state estimation of the navigation process and use its outputs as ground truth to train our left brain. This process is more akin to distilling the state estimation capability into the Q-Former that we fine-tune. The prompt is shown in Fig. 5.

Imagination Intruction Collection. To train the Imagination LLM, we gather the captions of candidate nodes associated with each node to serve as the training ground truth. An illustration of this collection process is shown in Fig. 6. As depicted, for each current node, we first collect images from all candidate nodes and stitch them together into a panoramic view. We then employ the Qwen2.5-VL-7B-Instruct [97] model to generate a caption for the panoramic image. By combining these captions, we produce the imagination ground truth for the current node.

System Prompt. As described in our method section, both the Left Brain and Right Brain use prompts to integrate the visual information generated by the Q-Former with the instruction input before passing it to the LLM. The system prompts for the State Estimation LLM and the Imagination LLM are illustrated in the Fig. 7.
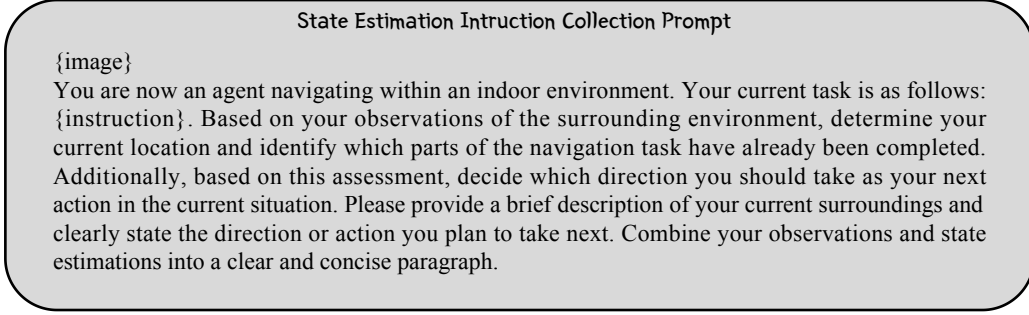
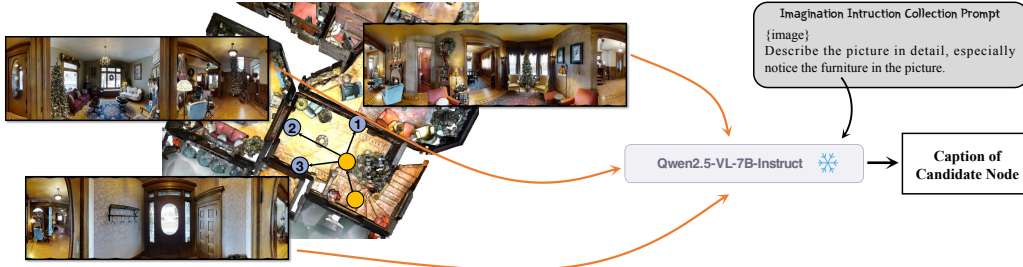Figure 5: **State Estimation generation prompt.**

Figure 6: **Imagination Ground Truth Collection.**

7 Additional Implementation Details

7.1 Evaluation Metrics

In this section, we present a comprehensive overview of the evaluation metrics employed in our study. Consistent with prior research, we utilize five key metrics: Trajectory Length (**TL**), Navigation Error (**NE**), Success Rate (**SR**), Oracle Success Rate (**OSR**), and Success Weighted by Path Length (**SPL**). The following subsections provide a detailed description of each metric along with its corresponding mathematical formulation:

- **TL.** The total length of the predicted trajectory, calculated as the sum of the shortest path distances between consecutive viewpoints along the trajectory.

$$L_{\text{traj}} = \sum_{i=1}^{N-1} d(v_i, v_{i+1}), \tag{12}$$

  where v_i is the i-th viewpoint in the predicted path, $d(a, b)$ is the shortest path distance between viewpoints a and b, and N is the number of viewpoints.

- **NE.** The shortest path distance between the final viewpoint of the predicted trajectory and the goal viewpoint.

$$E_{\text{nav}} = d(v_N, v^*), \tag{13}$$

  where v_N is the last viewpoint in the predicted trajectory and v^* is the goal viewpoint.

- **SR.** Indicates whether the agent successfully reached the goal. For tasks with a set of goal viewpoints, success is achieved if the final viewpoint of the predicted path is within the goal set. Otherwise, success is defined by whether the navigation error is below a certain threshold ϵ.

$$S = \begin{cases} 1 & \text{if } v_N \in G \\ 0 & \text{otherwise,} \end{cases} \tag{14}$$

  or, if no goal set exists:

$$S = \begin{cases} 1 & \text{if } d(v_N, v^*) < \epsilon \\ 0 & \text{otherwise,} \end{cases} \tag{15}$$

  where v_N is the final viewpoint of the predicted path, G is the goal viewpoint set, v^* is the goal viewpoint, and ϵ is the success threshold.

Figure 7: **State Estimation System Prompt and Imagination System Prompt.**

- **OSR.** Measures whether any viewpoint along the predicted path falls within the goal set, reflecting the best possible success if the agent stopped at the closest point to the goal.

$$S_{\text{oracle}} = \begin{cases} 1 & \text{if } \exists v_i \in G \\ 0 & \text{otherwise,} \end{cases} \tag{16}$$

or, if no goal set:

$$S_{\text{oracle}} = \begin{cases} 1 & \text{if } \min_i d(v_i, v^*) < \epsilon \\ 0 & \text{otherwise,} \end{cases} \tag{17}$$

where v_i is the i-th viewpoint along the path.

- **SPL.** A metric combining success and efficiency, penalizing longer trajectories relative to the shortest possible path length.

$$\text{SPL} = S \times \frac{L_{\text{gt}}}{\max(L_{\text{traj}}, L_{\text{gt}})}, \tag{18}$$

where L_{gt} is the ground-truth path length, L_{traj} is the predicted path length.

7.2 Training Details

Following NavGPT2 [110], our best-performing model is trained with additional synthetic data from PREVALENT [31]. We conducted an ablation study by excluding the PREVALENT data and observed that this synthetic data is vital in preventing our method from overfitting. Without incorporating the synthetic data, the validation loss plateaus prematurely during the early stages of training. The parameter size of our model is 1.5B because only the encoder was used during policy training, resulting in half of the parameters of the Flant5-XL model.

7.3 Compared Baselines

In this section, we provide a detailed overview of comparative methods, elaborating on their distinct methodological designs.

- **Seq2Seq [4].** The study introduces the Room-to-Room (R2R) dataset, the first benchmark for visually-grounded natural language navigation. It consists of real building layouts and natural-language instructions that guide agents between rooms, providing a realistic setting to evaluate models on understanding spatial relationships and visual contexts. And this paper presenting a framework for robots to interpret natural-language navigation instructions using visual inputs. The core method leverages sequence-to-sequence translation techniques, treating the task as a visually grounded translation problem akin to Visual Question Answering. To enable research, the authors develop the Matterport3D Simulator, a large-scale reinforcement learning environment built on real-world imagery, allowing robots to learn navigation policies in simulated 3D environments.

- **RCM [89].** Reinforced Cross-Modal Matching employs reinforcement learning to enforce both local (via a reasoning navigator for visual-scene grounding) and global (via a matching critic providing intrinsic rewards for instruction-trajectory alignment) cross-modal alignment.

- **Speaker Follower [27].** The approach addresses data scarcity and reasoning challenges via: (1) it synthesizes new natural language instructions for data augmentation and (2) implements pragmatic reasoning to evaluate how well candidate action sequences explain given instructions.

- **EnvDrop [80].** This paper presents a two-stage training framework. In the first stage, the agent is trained via a combination of imitation learning and reinforcement learning, integrating the advantages of off-policy and on-policy optimization. The second stage involves fine-tuning with "unseen" triplets (environment, path, instruction), generated using an "environmental dropout" method that mimics unseen environments by randomly removing objects to enhance variability.

- **PREVALENT [31].** This paper presents a pre-training and fine-tuning paradigm for VLN tasks. The proposed agent, Prevalent, is pre-trained via self-supervised learning on a large dataset of image-text-action triplets to learn generic representations of visual environments and language instructions. It can be integrated into existing VLN frameworks as a plug-and-play module. And we use it in our main experiments.

- **AirBert [29].** The core methods include: 1) constructing BnB, a large-scale dataset by collecting image-caption pairs (IC pairs) from hundreds of thousands of online rental listings and generating millions of path-instruction pairs (PI pairs) via automatic strategies; 2) proposing a shuffling loss to enhance learning of temporal order in PI pairs.

- **VLNBert [32].** This paper proposes a recurrent Vision-Language BERT model, equipping BERT with a recurrent function to maintain cross-modal state information and address the history-dependent decision-making challenge in the partially observable Markov decision process.

- **MARVAL [37].** This paper presents a novel approach to scaling VLN by leveraging synthetic instructions and imitation learning. The authors construct navigation trajectories across 500+ indoor environments with densely-sampled 360° panoramas, then generate 4.2 million visually-grounded instruction-trajectory pairs using Marky, a multilingual instruction generator, and synthesize novel viewpoint images via image-to-image GAN. This work highlights the power of synthetic data augmentation and imitation learning for advancing instruction-following agents in VLN.

- **HAMT [17].** HAMT employs a hierarchical ViT to encode past panoramic observations by modeling image-level features, spatial relations within panoramas, and temporal dynamics across historical panoramas. It fuses text instructions, historical context, and current observations to predict actions. The model is first pre-trained on proxy tasks (e.g., single-step action prediction, spatial relation prediction) and then optimized via reinforcement learning.

- **HOP+ [71].** The method introduces three VLN-specific pre-training tasks: Action Prediction with History (APH) to leverage historical visual trajectories for action prediction, Trajectory Order Modeling (TOM) and Group Order Modeling (GOM) to enhance temporal order reasoning. Additionally, a memory network is designed for fine-tuning to dynamically select and summarize historical information, addressing representation inconsistency between pre-training and fine-tuning stages. The experiment validating the effectiveness of history-enhanced and order-aware learning.

- **DUET [18].** The method dynamically combines fine-scale encoding of local visual observations with coarse-scale encoding on a globally built topological map via graph transformers, enabling joint long-term action planning and fine-grained cross-modal understanding.

- **BEVBert [2].** The method constructs a hybrid map system, including a local metric map to explicitly aggregate incomplete observations and remove duplicates, and a global topological map to model navigation dependencies, balancing short-term reasoning and long-term planning. It pre-trains a multimodal map representation to enhance spatial-aware cross-modal reasoning.

- **NavGPT [111].** The method enables NavGPT to perform zero-shot sequential action prediction by taking textual descriptions of visual observations, navigation history, and future explorable directions as inputs to reason about the agent's status and decide actions. It demonstrates abilities in decomposing instructions into sub-goals, integrating commonsense knowledge, identifying landmarks, tracking progress, and adjusting plans. The work leverages LLMs without specific training on navigation datasets, showcasing their potential for explicit reasoning in embodied scenes, though performance on zero-shot R2R tasks lags behind trained models.

- **MapGPT [14].** It introduces an online linguistic-formed map incorporating node information and topological relationships into prompts to help GPT understand the spatial environment, and proposes an adaptive planning mechanism for multi-step path planning based on the map.

- **DiscussNav [57].** It employs large models with distinct capabilities as domain experts (e.g., for instruction understanding, environment perception, and completion estimation). The navigation agent actively discusses with these experts at each step to gather critical information before moving, correcting errors and filtering inconsistent decisions.

- **NavCoT [52].** This paper employs Chain-of-Thought to enhance LLM-based VLN. At each timestep, the LLM acts as a world model to imagine the next observation based on instructions, selects the most aligned candidate observation, and determines actions through disentangled reasoning. Formalized training labels are constructed to guide the LLM in generating reasonable chain-of-thought outputs for improved action decisions.

- **LangNav [66].** It employs off-the-shelf vision systems to convert egocentric panoramic views into text descriptions and fine-tunes a pretrained language model to select actions based on these descriptions and trajectory history.
- **NaviLLM [108].** This paper presents NaviLLM, the first generalist model for embodied navigation, which adapts large language models (LLMs) to this field via schema-based instructions. This approach flexibly transforms various tasks into generation problems to unify diverse tasks and integrates multi-dataset sources (e.g., CVDN, SOON, ScanQA) during training.
- **NavGPT2 [110].** The method integrates a frozen LLM with a topological graph-based navigation policy network. Visual observations are encoded into image tokens via a Q-former and fused with instruction text tokens to generate navigational reasoning. The navigation policy uses graph memory to model spatial structures, enabling long-term history tracking and backtracking. Trained in two stages, NavGPT-2 first undergoes visual instruction tuning on 10K navigation reasoning data generated by GPT-4V, then fine-tunes the policy network with R2R and PREVALENT datasets.

8 Additional Experiment Results

More Cross Environment Experiment. To validate the generalization capability of our model, we conduct zero-shot evaluation on additional datasets REVERIE and R4R, and compare our model with the DUET method, please see Tab. 6. Our ATD model outperforms on both the R4R and REVERIE datasets, demonstrating its strong generalizability.

Table 6: Comparison of zero-shot performance on R4R and REVERIE `val unseen` split.

Methods	R4R					REVERIE		
	nDTW↑	sDTW↑	OSR↑	**SR↑**	**SPL↑**	OSR↑	**SR↑**	**SPL↑**
DUET	40.98	11.16	53.69	17.73	15.76	28.94	24.42	19.09
ATD$_{\text{FlanT5-XXL}}$	**44.35**	**13.58**	**55.36**	**19.95**	**18.31**	**32.52**	**27.04**	**22.51**

Ablation of Left-right Brain Structure. In Tab. 7, we perform an ablation study on the left-right brain structure of *Cross from Left to Right*. In *Raw 2*, *Cross from Right to Left* designates State Estimation as both the key and value and Imagination as the query, altering their roles in **SGCA** layers compared to ATD. In *Raw 3*, the *Parallel Structure* refers to the exclusion of the **SGCA** layer, where State Estimation and Imagination are summed in parallel and then fed into the subsequent policy for training. As can be seen from the results in the Tab. 7, the **SGCA** structure in ATD that uses State Estimation to ground Imagination achieves the best performance (*Raw 1*), which demonstrates the effectiveness of our **SGCA** layer design.

Table 7: Ablation of the left-right brain structure to validate the effectiveness of the SGCA layer.

Methods	Val Seen					Val Unseen				
	TL	NE↓	OSR↑	SR↑	SPL↑	TL	NE↓	OSR↑	SR↑	SPL↑
ATD	12.26	2.67	81.78	**75.61**	**67.49**	13.33	2.81	82.76	**74.63**	**65.35**
Change to *Cross from Right to Left*	12.38	3.17	76.79	71.79	63.97	12.14	2.92	79.65	73.95	63.58
Change to *Parallel Structure*	13.86	2.84	81.88	73.75	65.25	13.94	2.83	82.46	74.41	61.89

More Visualization of Attention Matrix. As illustrated in Fig. 8, we visualized additional attention matrices.

9 Discussion

Limitation and Future Work. Currently, the data collected for training the imagination LLM is limited to candidate nodes one step ahead of the current node. This may restrict the model's ability to perform long-horizon imagination. Future work could explore incorporating long-horizon imagination capabilities to more fully leverage the potential of the LLM's imaginative reasoning.

Broader Impact. This work contributes to the field of vision-language navigation (VLN) by proposing an adaptive, language-based imagination framework. Leveraging large vision-language

models in a dual-branch architecture, our approach achieves improved navigation performance with greater parameter efficiency, demonstrating potential applicability in embodied AI domains such as autonomous robotics, human-robot interaction, and assistive systems. Notwithstanding these advancements, several ethical and safety considerations remain paramount. Our experiments are conducted exclusively within controlled simulated environments to mitigate risks associated with unpredictable agent behavior. The propensity of large vision-language models to hallucinate or misinterpret environmental semantics highlights the necessity for rigorous evaluation and validation before real-world deployment. Additionally, as our method builds upon pretrained language models, it inherits challenges related to bias, fairness, and transparency, necessitating continuous efforts to identify and mitigate these risks.
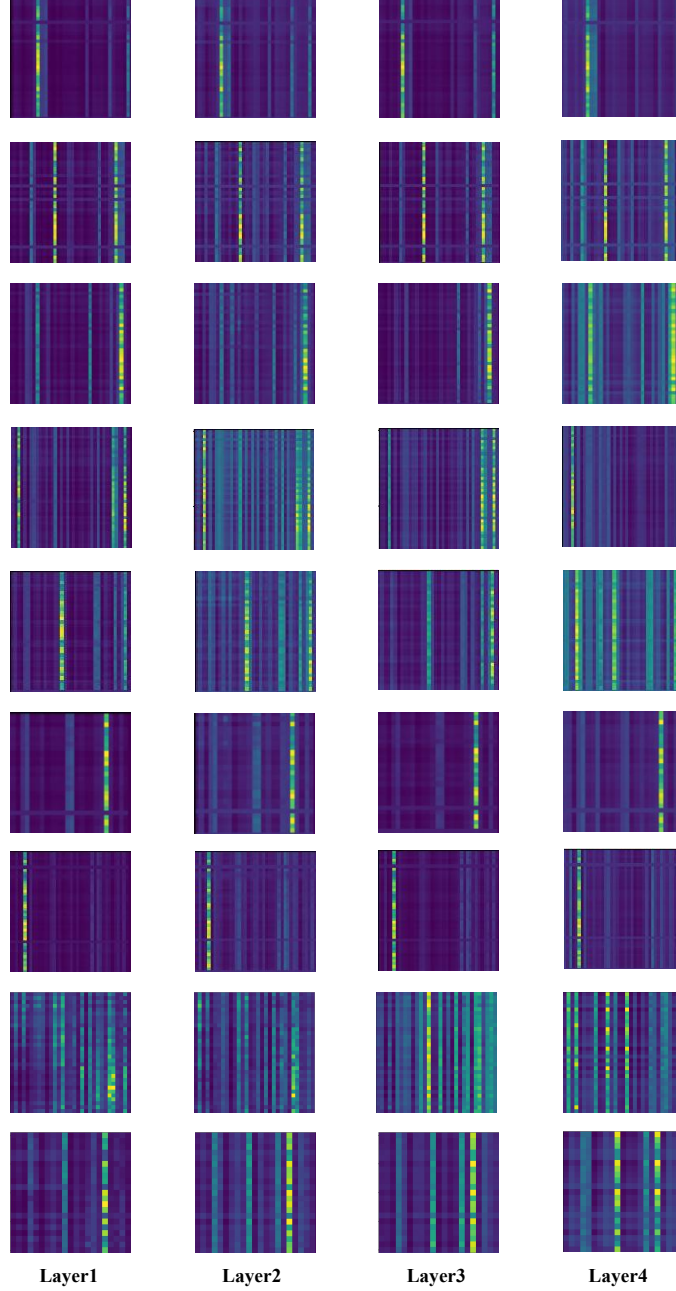

Figure 8: **Visualization of attention matrix for every layer.**

References

[1] Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F.L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al.: Gpt-4 technical report. arXiv preprint arXiv:2303.08774 (2023) 3, 5, 10

[2] An, D., Qi, Y., Li, Y., Huang, Y., Wang, L., Tan, T., Shao, J.: Bevbert: Multimodal map pre-training for language-guided navigation. arXiv preprint arXiv:2212.04385 (2022) 3, 7, 8, 13

[3] Anderson, P., Chang, A., Chaplot, D.S., Dosovitskiy, A., Gupta, S., Koltun, V., Kosecka, J., Malik, J., Mottaghi, R., Savva, M., et al.: On evaluation of embodied navigation agents. arXiv preprint arXiv:1807.06757 (2018) 7

[4] Anderson, P., Wu, Q., Teney, D., Bruce, J., Johnson, M., Sünderhauf, N., Reid, I.D., Gould, S., van den Hengel, A.: Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition pp. 3674–3683 (2017) 1, 2, 3, 7, 12

[5] Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., et al.: Qwen2. 5-vl technical report. arXiv preprint arXiv:2502.13923 (2025) 5

[6] Bar, A., Zhou, G., Tran, D., Darrell, T., LeCun, Y.: Navigation world models. arXiv preprint arXiv:2412.03572 (2024) 3

[7] Barron, J.T., Mildenhall, B., Tancik, M., Hedman, P., Martin-Brualla, R., Srinivasan, P.P.: Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 5855–5864 (2021) 3

[8] Black, K., Brown, N., Driess, D., Esmail, A., Equi, M., Finn, C., Fusai, N., Groom, L., Hausman, K., Ichter, B., Jakubczak, S., Jones, T., Ke, L., Levine, S., Li-Bell, A., Mothukuri, M., Nair, S., Pertsch, K., Shi, L.X., Tanner, J., Vuong, Q., Walling, A., Wang, H., Zhilinsky, U.: π_0: A vision-language-action flow model for general robot control (2024) 2

[9] Bu, Q., Cai, J., Chen, L., Cui, X., Ding, Y., Feng, S., Gao, S., He, X., Huang, X., Jiang, S., et al.: Agibot world colosseo: A large-scale manipulation platform for scalable and intelligent embodied systems. arXiv preprint arXiv:2503.06669 (2025) 2

[10] Buoso, D., Robinson, L., Averta, G., Torr, P., Franzmeyer, T., De Martini, D.: Select2plan: Training-free icl-based planning through vqa and memory retrieval. arXiv preprint arXiv:2411.04006 (2024) 3

[11] Charatan, D., Li, S.L., Tagliasacchi, A., Sitzmann, V.: pixelsplat: 3d gaussian splats from image pairs for scalable generalizable 3d reconstruction. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 19457–19467 (2024) 3

[12] Chen, H., Suhr, A., Misra, D., Snavely, N., Artzi, Y.: Touchdown: Natural language navigation and spatial reasoning in visual street environments. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12538–12547 (2019) 3

[13] Chen, J., Lin, B., Liu, X., Ma, L., Liang, X., Wong, K.Y.K.: Affordances-oriented planning using foundation models for continuous vision-language navigation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 23568–23576 (2025) 3

[14] Chen, J., Lin, B., Xu, R., Chai, Z., Liang, X., Wong, K.Y.K.: Mapgpt: Map-guided prompting for unified vision-and-language navigation. arXiv e-prints pp. arXiv–2401 (2024) 3, 7, 13

[15] Chen, K., An, D., Huang, Y., Xu, R., Su, Y., Ling, Y., Reid, I., Wang, L.: Constraint-aware zero-shot vision-language navigation in continuous environments. arXiv preprint arXiv:2412.10137 (2024) 3

[16] Chen, P., Sun, X., Zhi, H., Zeng, R., Li, T.H., Liu, G., Tan, M., Gan, C.: a^2 nav: Action-aware zero-shot robot navigation by exploiting vision-and-language ability of foundation models. arXiv preprint arXiv:2308.07997 (2023) 3

[17] Chen, S., Guhur, P.L., Schmid, C., Laptev, I.: History aware multimodal transformer for vision-and-language navigation. ArXiv **abs/2110.13309** (2021) 1, 3, 7, 13

[18] Chen, S., Guhur, P.L., Tapaswi, M., Schmid, C., Laptev, I.: Think global, act local: Dual-scale graph transformer for vision-and-language navigation. 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 16516–16526 (2022) 1, 3, 6, 7, 13

[19] Chen, Y., Xu, H., Zheng, C., Zhuang, B., Pollefeys, M., Geiger, A., Cham, T.J., Cai, J.: Mvsplat: Efficient 3d gaussian splatting from sparse multi-view images. In: European Conference on Computer Vision. pp. 370–386. Springer (2024) 3

[20] Cheng, A.C., Ji, Y., Yang, Z., Gongye, Z., Zou, X., Kautz, J., Bıyık, E., Yin, H., Liu, S., Wang, X.: Navila: Legged robot vision-language-action model for navigation. arXiv preprint arXiv:2412.04453 (2024) 3

[21] Dai, G., Zhao, J., Chen, Y., Qin, Y., Zhao, H., Xie, G., Yao, Y., Shu, X., Li, X.: Unitedvln: Generalizable gaussian splatting for continuous vision-language navigation. arXiv preprint arXiv:2411.16053 (2024) 3

[22] Dai, W., Li, J., Li, D., Tiong, A.M.H., Zhao, J., Wang, W., Li, B., Fung, P., Hoi, S.: Instructblip: Towards general-purpose vision-language models with instruction tuning (2023) 2, 4, 7

[23] Das, A., Datta, S., Gkioxari, G., Lee, S., Parikh, D., Batra, D.: Embodied question answering. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1–10 (2018) 3

[24] Deng, Z., Narasimhan, K., Russakovsky, O.: Evolving graphical planner: Contextual global planning for vision-and-language navigation. ArXiv **abs/2007.05655** (2020) 3

[25] Fan, S., Liu, R., Wang, W., Yang, Y.: Navigation instruction generation with bev perception and large language models. In: European Conference on Computer Vision. pp. 368–387. Springer (2024) 3

[26] Fan, Y., Chen, W., Jiang, T., Zhou, C., Zhang, Y., Wang, X.E.: Aerial vision-and-dialog navigation. arXiv preprint arXiv:2205.12219 (2022) 3

[27] Fried, D., Hu, R., Cirik, V., Rohrbach, A., Andreas, J., Morency, L.P., Berg-Kirkpatrick, T., Saenko, K., Klein, D., Darrell, T.: Speaker-follower models for vision-and-language navigation. Advances in neural information processing systems **31** (2018) 3, 7, 12

[28] Gao, C., Chen, J., Liu, S., Wang, L., Zhang, Q., Wu, Q.: Room-and-object aware knowledge reasoning for remote embodied referring expression. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3064–3073 (2021) 3

[29] Guhur, P.L., Tapaswi, M., Chen, S., Laptev, I., Schmid, C.: Airbert: In-domain pretraining for vision-and-language navigation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1634–1643 (2021) 7, 13

[30] Guo, X., Singh, S., Lee, H., Lewis, R.L., Wang, X.: Deep learning for real-time atari game play using offline monte-carlo tree search planning. Advances in neural information processing systems **27** (2014) 3

[31] Hao, W., Li, C., Li, X., Carin, L., Gao, J.: 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. pp. 13137–13146 (2020) 3, 7, 12, 13

[32] Hong, Y., Wu, Q., Qi, Y., Rodriguez-Opazo, C., Gould, S.: Vln bert: A recurrent vision-and-language bert for navigation. In: Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition. pp. 1643–1653 (2021) 1, 3, 7, 13

[33] Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., De Laroussilhe, Q., Gesmundo, A., Attariyan, M., Gelly, S.: Parameter-efficient transfer learning for nlp. In: International conference on machine learning. pp. 2790–2799. PMLR (2019) 3

[34] Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., et al.: Lora: Low-rank adaptation of large language models. ICLR **1**(2), 3 (2022) 3

[35] Huang, J., Yong, S., Ma, X., Linghu, X., Li, P., Wang, Y., Li, Q., Zhu, S.C., Jia, B., Huang, S.: An embodied generalist agent in 3d world. arXiv preprint arXiv:2311.12871 (2023) 3

[36] Jia, M., Tang, L., Chen, B.C., Cardie, C., Belongie, S., Hariharan, B., Lim, S.N.: Visual prompt tuning. In: European conference on computer vision. pp. 709–727. Springer (2022) 3

[37] Kamath, A., Anderson, P., Wang, S., Koh, J.Y., Ku, A., Waters, A., Yang, Y., Baldridge, J., Parekh, Z.: 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. pp. 10813–10823 (2023) 3, 7, 13

[38] Ke, L., Li, X., Bisk, Y., Holtzman, A., Gan, Z., Liu, J., Gao, J., Choi, Y., Srinivasa, S.: Tactical rewind: Self-correction via backtracking in vision-and-language navigation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 6741–6749 (2019) 3

[39] Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3d gaussian splatting for real-time radiance field rendering. ACM Trans. Graph. **42**(4), 139–1 (2023) 3

[40] Kocsis, L., Szepesvári, C.: Bandit based monte-carlo planning. In: European conference on machine learning. pp. 282–293. Springer (2006) 3

[41] Koh, J.Y., Lee, H., Yang, Y., Baldridge, J., Anderson, P.: Pathdreamer: A world model for indoor navigation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 14738–14748 (2021) 3

[42] Krantz, J., Gokaslan, A., Batra, D., Lee, S., Maksymets, O.: Waypoint models for instruction-guided navigation in continuous environments. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15162–15171 (2021) 3

[43] Krantz, J., Wijmans, E., Majumdar, A., Batra, D., Lee, S.: 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. pp. 104–120. Springer (2020) 3, 7

[44] Ku, A., Anderson, P., Patel, R., Ie, E., Baldridge, J.: Room-across-room: Multilingual vision-and-language navigation with dense spatiotemporal grounding. arXiv preprint arXiv:2010.07954 (2020) 3

[45] Lee, J., Miyanishi, T., Kurita, S., Sakamoto, K., Azuma, D., Matsuo, Y., Inoue, N.: Citynav: Language-goal aerial navigation dataset with geographic information. arXiv preprint arXiv:2406.14240 (2024) 3

[46] Li, J., Bansal, M.: Improving vision-and-language navigation by generating future-view image semantics. 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 10803–10812 (2023) 3

[47] Li, J., Bansal, M.: Panogen: Text-conditioned panoramic environment generation for vision-and-language navigation. In: Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., Levine, S. (eds.) Advances in Neural Information Processing Systems. vol. 36, pp. 21878–21894. Curran Associates, Inc. (2023) 3

[48] Li, J., Tan, H., Bansal, M.: Envedit: Environment editing for vision-and-language navigation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15407–15417 (2022) 3

[49] Li, J., Tang, S., Wu, F., Zhuang, Y.: Walking with mind: Mental imagery enhanced embodied qa. In: Proceedings of the 27th ACM International Conference on Multimedia. pp. 1211–1219 (2019) 3

[50] Li, J., Li, D., Savarese, S., Hoi, S.: Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In: International conference on machine learning. pp. 19730–19742. PMLR (2023) 2, 4

[51] Li, X., Wang, Z., Yang, J., Wang, Y., Jiang, S.: Kerm: Knowledge enhanced reasoning for vision-and-language navigation. 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 2583–2592 (2023) 1, 3

[52] Lin, B., Nie, Y., Wei, Z., Chen, J., Ma, S., Han, J., Xu, H., Chang, X., Liang, X.: Navcot: Boosting llm-based vision-and-language navigation via learning disentangled reasoning. arXiv preprint arXiv:2403.07376 (2024) 7, 8, 13

[53] Lin, B., Nie, Y., Wei, Z., Zhu, Y., Xu, H., Ma, S., Liu, J., Liang, X.: Correctable landmark discovery via large models for vision-language navigation. IEEE Transactions on Pattern Analysis and Machine Intelligence (2024) 4

[54] Lin, J., Du, Y., Watkins, O., Hafner, D., Abbeel, P., Klein, D., Dragan, A.: Learning to model the world with language. arXiv preprint arXiv:2308.01399 (2023) 2

[55] Liu, R., Wang, W., Yang, Y.: Volumetric environment representation for vision-language navigation. 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 16317–16328 (2024) 3

[56] Long, Y., Cai, W., Wang, H., Zhan, G., Dong, H.: Instructnav: Zero-shot system for generic instruction navigation in unexplored environment. arXiv preprint arXiv:2406.04882 (2024) 3

[57] Long, Y., Li, X., Cai, W., Dong, H.: Discuss before moving: Visual language navigation via multi-expert discussions. In: 2024 IEEE International Conference on Robotics and Automation (ICRA). pp. 17380–17387. IEEE (2024) 3, 7, 13

[58] Ma, C.Y., Lu, J., Wu, Z., AlRegib, G., Kira, Z., Socher, R., Xiong, C.: Self-monitoring navigation agent via auxiliary progress estimation. arXiv preprint arXiv:1901.03035 (2019) 3

[59] Ma, C.Y., Wu, Z., AlRegib, G., Xiong, C., Kira, Z.: The regretful agent: Heuristic-aided navigation through progress estimation. In: Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition. pp. 6732–6740 (2019) 3

[60] Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., et al.: Language models are few-shot learners. arXiv preprint arXiv:2005.14165 **1**, 3 (2020) 3

[61] Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. Communications of the ACM **65**(1), 99–106 (2021) 3

[62] Müller, T., Evans, A., Schied, C., Keller, A.: Instant neural graphics primitives with a multiresolution hash encoding. ACM transactions on graphics (TOG) **41**(4), 1–15 (2022) 3

[63] Nguyen, K., Daumé III, H.: Help, anna! visual navigation with natural multimodal assistance via retrospective curiosity-encouraging imitation learning. arXiv preprint arXiv:1909.01871 (2019) 3

[64] Nguyen, K., Daumé III, H.: Help, anna! visual navigation with natural multimodal assistance via retrospective curiosity-encouraging imitation learning. arXiv preprint arXiv:1909.01871 (2019) 3

[65] Nie, D., Guo, X., Duan, Y., Zhang, R., Chen, L.: Wmnav: Integrating vision-language models into world models for object goal navigation. arXiv preprint arXiv:2503.02247 (2025) 3

[66] Pan, B., Panda, R., Jin, S., Feris, R., Oliva, A., Isola, P., Kim, Y.: Langnav: Language as a perceptual representation for navigation. arXiv preprint arXiv:2310.07889 (2023) 7, 8, 14

[67] Pan, Y., Xu, Y., Liu, Z., Wang, H.: Planning from imagination: Episodic simulation and episodic memory for vision-and-language navigation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 6345–6353 (2025) 3

[68] Pashevich, A., Schmid, C., Sun, C.: Episodic transformer for vision-and-language navigation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15942–15952 (2021) 3

[69] Qi, Y., Wu, Q., Anderson, P., Wang, X., Wang, W.Y., Shen, C., Hengel, A.v.d.: Reverie: Remote embodied visual referring expression in real indoor environments. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9982–9991 (2020) 3, 7, 8

[70] Qiao, Y., Qi, Y., Hong, Y., Yu, Z., Wang, P., Wu, Q.: Hop: History-and-order aware pre-training for vision-and-language navigation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15418–15427 (2022) 3

[71] Qiao, Y., Qi, Y., Hong, Y., Yu, Z., Wang, P., Wu, Q.: Hop+: History-enhanced and order-aware pre-training for vision-and-language navigation. IEEE Transactions on Pattern Analysis and Machine Intelligence **45**(7), 8524–8537 (2023) 3, 7, 13

[72] Qiao, Y., Qi, Y., Yu, Z., Liu, J., Wu, Q.: March in chat: Interactive prompting for remote embodied referring expression. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15758–15767 (2023) 3

[73] Rajvanshi, A., Sikka, K., Lin, X., Lee, B., Chiu, H.P., Velasquez, A.: Saynav: Grounding large language models for dynamic planning to navigation in new environments. In: Proceedings of the International Conference on Automated Planning and Scheduling. vol. 34, pp. 464–474 (2024) 3

[74] Ross, S., Gordon, G., Bagnell, D.: A reduction of imitation learning and structured prediction to no-regret online learning. In: Proceedings of the fourteenth international conference on artificial intelligence and statistics. pp. 627–635. JMLR Workshop and Conference Proceedings (2011) 3

[75] Ross, S., Gordon, G., Bagnell, D.: A reduction of imitation learning and structured prediction to no-regret online learning. In: Proceedings of the fourteenth international conference on artificial intelligence and statistics. pp. 627–635. JMLR Workshop and Conference Proceedings (2011) 6

[76] Shah, D., Osiński, B., Levine, S., et al.: Lm-nav: Robotic navigation with large pre-trained models of language, vision, and action. In: Conference on robot learning. pp. 492–504. PMLR (2023) 3

[77] Shah, H., Xing, J., Messikommer, N., Sun, B., Pollefeys, M., Scaramuzza, D.: Foresightnav: Learning scene imagination for efficient exploration. arXiv preprint arXiv:2504.16062 (2025) 1, 3

[78] Shridhar, M., Thomason, J., Gordon, D., Bisk, Y., Han, W., Mottaghi, R., Zettlemoyer, L., Fox, D.: Alfred: A benchmark for interpreting grounded instructions for everyday tasks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10740–10749 (2020) 3

[79] Song, C.H., Wu, J., Washington, C., Sadler, B.M., Chao, W.L., Su, Y.: Llm-planner: Few-shot grounded planning for embodied agents with large language models. 2023 IEEE/CVF International Conference on Computer Vision (ICCV) pp. 2986–2997 (2022) 3

[80] Tan, H., Yu, L., Bansal, M.: Learning to navigate unseen environments: Back translation with environmental dropout. arXiv preprint arXiv:1904.04195 (2019) 3, 7, 12

[81] Thomason, J., Murray, M., Cakmak, M., Zettlemoyer, L.: Vision-and-dialog navigation. In: Conference on Robot Learning. pp. 394–406. PMLR (2020) 3

[82] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017) 6

[83] Wang, H., Liang, W., Gool, L.V., Wang, W.: Dreamwalker: Mental planning for continuous vision-language navigation. 2023 IEEE/CVF International Conference on Computer Vision (ICCV) pp. 10839–10849 (2023) 2, 3

[84] Wang, H., Liang, W., Shen, J., Van Gool, L., Wang, W.: Counterfactual cycle-consistent learning for instruction following and generation in vision-language navigation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 15471–15481 (2022) 3

[85] Wang, H., Wu, Q., Shen, C.: Soft expert reward learning for vision-and-language navigation. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IX 16. pp. 126–141. Springer (2020) 3

[86] Wang, S., Zhou, D., Xie, L., Xu, C., Yan, Y., Yin, E.: Panogen++: Domain-adapted text-guided panoramic environment generation for vision-and-language navigation. Neural Networks **187**, 107320 (2025) 3

[87] Wang, S., Chen, L., Chen, P., Dong, J., Xue, B., Jiang, J., Kong, L., Wu, C.: Mos: Unleashing parameter efficiency of low-rank adaptation with mixture of shards. arXiv preprint arXiv:2410.00938 (2024) 3

[88] Wang, X., Wang, W., Shao, J., Yang, Y.: Lana: A language-capable navigator for instruction following and generation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 19048–19058 (2023) 3

[89] Wang, X., Huang, Q., Celikyilmaz, A., Gao, J., Shen, D., Wang, Y.F., Wang, W.Y., Zhang, L.: Reinforced cross-modal matching and self-supervised imitation learning for vision-language navigation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 6629–6638 (2019) 1, 3, 7, 12

[90] Wang, X., Huang, Q., Celikyilmaz, A., Gao, J., Shen, D., Wang, Y.F., Wang, W.Y., Zhang, L.: Reinforced cross-modal matching and self-supervised imitation learning for vision-language navigation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6629–6638 (2019) 7

[91] Wang, Z., Li, X., Yang, J., Liu, Y., Hu, J., Jiang, M., Jiang, S.: Lookahead exploration with neural radiance representation for continuous vision-language navigation. 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 13753–13762 (2024) 2, 3

[92] Wang, Z., Li, X., Yang, J., Liu, Y., Jiang, S.: Gridmm: Grid memory map for vision-and-language navigation. In: Proceedings of the IEEE/CVF International conference on computer vision. pp. 15625–15636 (2023) 1, 3

[93] Wang, Z., Li, J., Hong, Y., Wang, Y., Wu, Q., Bansal, M., Gould, S., Tan, H., Qiao, Y.: Scaling data generation in vision-and-language navigation. 2023 IEEE/CVF International Conference on Computer Vision (ICCV) pp. 11975–11986 (2023) 3

[94] Wei, Z., Lin, B., Nie, Y., Chen, J., Ma, S., Xu, H., Liang, X.: Unseen from seen: Rewriting observation-instruction using foundation models for augmenting vision-language navigation. arXiv preprint arXiv:2503.18065 (2025) 3

[95] Wu, Y., Tian, R., Swamy, G., Bajcsy, A.: From foresight to forethought: Vlm-in-the-loop policy steering via latent alignment. arXiv preprint arXiv:2502.01828 (2025) 2

[96] Xu, Y., Pan, Y., Liu, Z., Wang, H.: Flame: Learning to navigate with multimodal llm in urban environments. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 9005–9013 (2025) 4

[97] Yang, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Li, C., Liu, D., Huang, F., Wei, H., et al.: Qwen2. 5 technical report. arXiv preprint arXiv:2412.15115 (2024) 10

[98] Zhan, Z., Yu, L., Yu, S., Tan, G.: Mc-gpt: Empowering vision-and-language navigation with memory map and reasoning chains. arXiv preprint arXiv:2405.10620 (2024) 3

[99] Zhang, J., Wang, K., Wang, S., Li, M., Liu, H., Wei, S., Wang, Z., Zhang, Z., Wang, H.: Uni-navid: A video-based vision-language-action model for unifying embodied navigation tasks. arXiv preprint arXiv:2412.06224 (2024) 3

[100] Zhang, J., Wang, K., Xu, R., Zhou, G., Hong, Y., Fang, X., Wu, Q., Zhang, Z., Wang, H.: Navid: Video-based vlm plans the next step for vision-and-language navigation. arXiv preprint arXiv:2402.15852 (2024) 3

[101] Zhang, L., Hao, X., Xu, Q., Zhang, Q., Zhang, X., Wang, P., Zhang, J., Wang, Z., Zhang, S., Xu, R.: Mapnav: A novel memory representation via annotated semantic maps for vlm-based vision-and-language navigation. arXiv preprint arXiv:2502.13451 (2025) 3

[102] Zhang, P., Gao, X., Wu, Y., Liu, K., Wang, D., Wang, Z., Zhao, B., Ding, Y., Li, X.: Moma-kitchen: A 100k+ benchmark for affordance-grounded last-mile navigation in mobile manipulation (2025) 3

[103] Zhang, S., Qiao, Y., Wang, Q., Guo, L., Wei, Z., Liu, J.: Flexvln: Flexible adaptation for diverse vision-and-language navigation tasks. arXiv preprint arXiv:2503.13966 (2025) 3

[104] Zhang, S., Yu, X., Song, X., Wang, X., Jiang, S.: Imagine before go: Self-supervised generative map for object goal navigation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16414–16425 (2024) 3

[105] Zhang, Y., Chen, K., Bai, X., Guo, Q., Zhang, M., et al.: Question-guided knowledge graph re-scoring and injection for knowledge graph question answering. arXiv preprint arXiv:2410.01401 (2024) 3

[106] Zhao, M., Anderson, P., Jain, V., Wang, S., Ku, A., Baldridge, J., Ie, E.: On the evaluation of vision-and-language navigation instructions. arXiv preprint arXiv:2101.10504 (2021) 3

[107] Zhao, X., Cai, W., Tang, L., Wang, T.: Imaginenav: Prompting vision-language models as embodied navigator through scene imagination. arXiv preprint arXiv:2410.09874 (2024) 3

[108] Zheng, D., Huang, S., Zhao, L., Zhong, Y., Wang, L.: Towards learning a generalist model for embodied navigation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13624–13634 (2024) 3, 7, 8, 14

[109] Zhong, Y., Feng, C., Yan, F., Liu, F., Zheng, L., Ma, L.: P3nav: A unified framework for embodied navigation integrating perception, planning, and prediction. arXiv preprint arXiv:2503.18525 (2025) 3

[110] Zhou, G., Hong, Y., Wang, Z., Wang, X.E., Wu, Q.: Navgpt-2: Unleashing navigational reasoning capability for large vision-language models. In: European Conference on Computer Vision (2024) 4, 5, 7, 8, 12, 14

[111] Zhou, G., Hong, Y., Wu, Q.: Navgpt: Explicit reasoning in vision-and-language navigation with large language models. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 38, pp. 7641–7649 (2024) 3, 7, 13

[112] Zhu, F., Zhu, Y., Chang, X., Liang, X.: Vision-language navigation with self-supervised auxiliary reasoning tasks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10012–10022 (2020) 3