Improving Vision-and-Language Navigation with Explicit Sub-Instruction Alignment

Mulang Shi
UNC Chapel Hill
mlshi@cs.unc.edu

Abstract

Vision-and-Language Navigation (VLN) tasks require an agent to interpret natural language instructions to navigate complex environments. However, existing methods face difficulties grounding multi-step instructions to intermediate visual observations. Typically, these approaches input the entire instruction at once, forcing the model to implicitly align visual observations with specific instruction segments, complicating the grounding process. To mitigate this issue, we introduce Sub-Aligner, a novel sub-instruction index prediction module designed to explicitly identify the most relevant instruction segment corresponding to the agent's current visual observations. Additionally, we propose a dual-stage, scene-aware description module that summarizes the agent's surroundings from both directional and panoramic perspectives, effectively bridging the semantic gap between visual context and complex, multi-step language instructions. Empirical evaluations demonstrate that integrating Sub-Aligner consistently enhances navigation performance across different VLN agents on benchmark datasets, Room-to-Room (R2R) and Room-for-Room (R4R).

1 Introduction

Vision-and-Language Navigation (VLN) requires an embodied agent to follow natural language instructions to navigate complex environments Anderson et al. [2018], Zhang et al. [2024]. This task involves grounding object references, identifying navigable paths, and aligning language with visual observations, and has real-world applications in assistive robotics and indoor navigation Hong et al. [2020a,b], Chen et al. [2021, 2022].

While recent advances have improved VLN agents through cross-modal Transformers Chen et al. [2022], fine-tuning with augmented data Li et al. [2024], Li and Bansal [2023], map-based reasoning Liu et al. [2023], and zero-shot instruction following via large multimodal language models (mLLMs) Zhou et al. [2024], Chen et al. [2024], most existing approaches lack explicit mechanisms to align intermediate visual observations with specific instruction sub-goals. This limitation introduces grounding ambiguity and reduces interpretability, making it difficult to trace how agents connect visual inputs to the corresponding textual instructions. Therefore, we argue that explicitly modeling alignment at the sub-instruction level is crucial for both accurate decision-making and interpretable grounding.

To address this, we propose a plug-and-play sub-instruction index prediction named Sub-Aligner, designed to enable fine-grained instruction grounding by explicitly identifying the most relevant sub-instruction segment aligned with the agent's current visual observation at each navigation step. Additionally, we introduce a dual-stage scene description module specifically designed to bridge the semantic gap between visual observations and lengthy instructions. This module generates detailed semantic descriptions that enrich the agent's current visual contexts with textual representations, thereby providing more accurate references for aligning visual observations with corresponding sub-instruction segments.

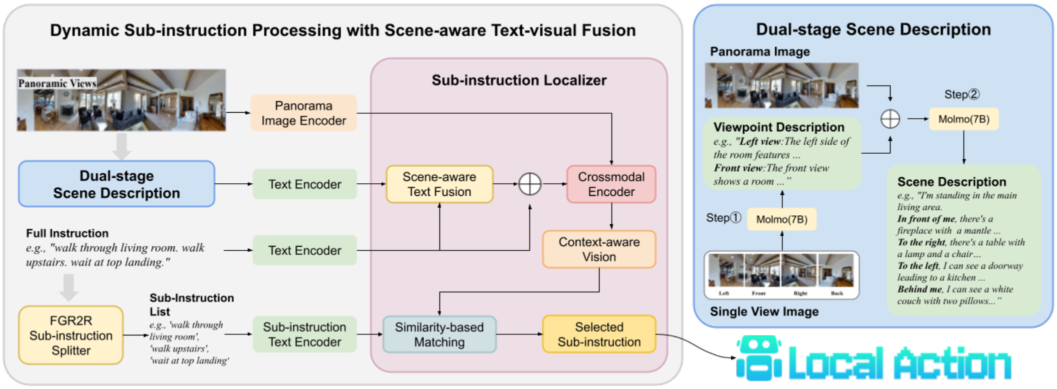
Dynamic Sub-instruction Processing with Scene-aware Text-visual Fusion
Panoramic Views
Panorama Image Encoder
Dual-stage Scene Description
Text Encoder
Full Instruction
e.g., "walk through living room. walk upstairs. wait at top landing."
Text Encoder
FGR2R Sub-instruction Splitter
Sub-Instruction List
e.g., 'walk through living room', 'walk upstairs', 'wait at top landing'
Sub-instruction Text Encoder
Sub-instruction Localizer
Scene-aware Text Fusion
Crossmodal Encoder
Context-aware Vision
Similarity-based Matching
Selected Sub-instruction
Dual-stage Scene Description
Panorama Image
Step②
Molmo(7B)
Viewpoint Description
e.g., "Left view:The left side of the room features ... Front view:The front view shows a room ..."
Step①
Molmo(7B)
Left
Front
Right
Back
Single View Image
Scene Description
e.g., "I'm standing in the main living area. In front of me, there's a fireplace with a mantle ... To the right, there's a table with a lamp and a chair ... To the left, I can see a doorway leading to a kitchen ... Behind me, I can see a white couch with two pillows..."
Local Action

Figure 1: **Overview of our sub-instruction predictor architecture** Given instructions and panoramic views, our dual-stage VLM-based module generates scene-aware descriptions from each view input. These are fused with segmented sub-instructions and visual embeddings in the sub-instruction localizer to dynamically select the most relevant sub-instruction at each step.

Specifically, our method takes as input a full navigation instruction, panoramic visual observations, and scene descriptions generated by dual-stage scene descriptor. It outputs a sub-instruction index at each step, indicating which segment of the instruction is currently relevant to the agent's visual context. To compute this, we fuse the instruction and scene descriptions via a gating mechanism and inject the result into a cross-modal encoder along with panoramic visual features. A prediction head then selects the most relevant sub-instruction. The scene descriptions guiding this process are generated by a dual-stage module, which we utilize a Vision-and-Language Model (VLM) first describes directional views and then fuses them with panoramic context to form a scene description.

We evaluate our proposed approach across two representative VLN agents: a fine-tuned agent (DUET Chen et al. [2022]) and a zero-shot agent (MapGPT Chen et al. [2024]), using the standard VLN benchmarks: R2R Anderson et al. [2018] and R4R Jain et al. [2019] datasets. Experimental results consistently demonstrate improvements in both evaluation settings, confirming the effectiveness of our sub-instruction grounding approach.

- We propose a plug-and-play sub-instruction index prediction module that dynamically identifies the most relevant instruction segment based on the agent's visual context, enabling fine-grained grounding at each navigation step.
- We introduce a dual-stage scene description module designed to narrow the semantic gap between visual observations and language instructions by generating structured textual cues from directional and panoramic views, enhancing sub-instruction selection.
- We demonstrate the effectiveness and generalizability of our method across both fine-tuned and zero-shot VLN agents, achieving consistent improvements on R2R and R4R benchmarks.

2 Related Works

Vision-and-Language Navigation (VLN) requires an agent to interpret natural language instructions and navigate through a 3D environment to reach a goal. Early VLN agents adopted sequence-to-sequence models Fried et al. [2018], Wang et al. [2019], Tan et al. [2019]. Transformer-based models enhanced cross-modal grounding Hao et al. [2020], Ma et al. [2019b], Majumdar et al. [2020], and DUET Chen et al. [2022] further introduced dual-branch representations for joint global planning and local grounding. More recently, zero-shot agents using large multimodal models (mLLMs) Zhou et al. [2024], Chen et al. [2024] treat navigation as language-conditioned action generation. While promising, these mLLM-based agents often struggle to maintain fine-grained alignment between instructions and evolving visual contexts, especially over long trajectories.

Sub-Instruction for VLN To enhance grounding granularity, FGR2R Hong et al. [2020a] augments R2R with human-annotated sub-instructions, enabling explicit step-level supervision. However, obtaining these annotations can be costly. Other approaches implicitly learn sub-instruction-level grounding Ma et al. [2019a], Zhang and Kordjamshidi [2022, 2023] by jointly training the grounding module with the navigation model. Nonetheless, these implicit methods lack explicit alignment mechanisms and remain detached from the agent's decision-making process. In contrast, our work explicitly integrates sub-instruction prediction into the navigation pipeline, allowing agents to dynamically identify sub-goals at each timestep and utilize the corresponding sub-instruction directly as input for the navigator.

3 Method

In this section, we first formalize the VLN task (§3.1), then introduce our dual-stage scene description module (§3.2) for contextual grounding, followed by our core sub-instruction localizer (§3.3), and finally describe how the predicted sub-instruction guides action selection and learning (§3.4). Our figure 1 illustrates the overall architecture, highlighting the two key components: the **Dual-stage Scene Description** and the **Sub-instruction Localizer**.

3.1 Problem Formulation

We follow standard VLN settings Anderson et al. [2018], modeling the environment as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where nodes represent viewpoints and edges denote navigable paths. At each timestep t, the agent is at a viewpoint with panoramic observation $\mathcal{R}_t$. Given a natural language instruction $\mathcal{W}$, which is pre-segmented into a sequence of sub-instructions $\{\mathcal{S}_1, \dots, \mathcal{S}_N\}$, the agent's goal is to select an action a_t that leads it towards the target location.

3.2 Dual-stage Scene Description Module

To facilitate more accurate sub-instruction prediction, we introduce a dual-stage scene description module that generates detailed semantic summaries of the environment from panoramic viewpoints. **First**, we extract four directional views $I_d = \{I_{\text{front}}, I_{\text{left}}, I_{\text{right}}, I_{\text{back}}\}$ and generate descriptions $S_d = \texttt{VLM}(I_d, P_d)$, where P_d is a direction-specific prompt. **Second**, we combine the four directional captions with panoramic input I_p and prompt the VLM again to generate a panoramic scene description from a first-person perspective by $S_{\text{scene}} = \texttt{VLM}(I_p, S_d)$. This design improves scene understanding by encouraging the VLM to first focus on salient directional details and then compose a coherent, spatially structured global summary. *Additional prompt details are in Appendix F.*

3.3 Sub-instruction Localizer

At each timestep t, we predict the index $\hat{i}_t$ of the most relevant sub-instruction $\mathcal{S}_{\hat{i}_t} \in \{\mathcal{S}_1, ..., \mathcal{S}_N\}$, given the full instruction $\mathcal{W}$, current panoramic observation $\mathcal{R}_t$, generated scene description S_{scene}, and candidate sub-instruction list $\{\mathcal{S}_i\}$. We first encode $\mathcal{W}$ and S_{scene} using a shared text encoder to obtain their [CLS] token representations, denoted $\mathbf{e}_{\mathcal{W}}$ and $\mathbf{e}_S$. These are fused via a learned gating mechanism:

$$\mathbf{g} = \sigma(\mathbf{W}_g[\mathbf{e}_{\mathcal{W}}; \mathbf{e}_S] + \mathbf{b}_g) \quad (1)$$

$$\mathbf{e}_{\text{fused}} = \mathbf{g} \odot \mathbf{e}_{\mathcal{W}} + (1 - \mathbf{g}) \odot \mathbf{e}_S \quad (2)$$

where $\mathbf{W}_g, \mathbf{b}_g$ are learnable parameters and σ is the sigmoid function. Next, we inject $\mathbf{e}_{\text{fused}}$ into a cross-modal encoder along with visual observation $\mathcal{R}_t$ to produce a scene-aware query vector $\mathbf{q}_t$. Each candidate sub-instruction $\mathcal{S}_i$ is encoded into a key vector $\mathbf{k}_i$ using the same text encoder. We then compute dot-product similarity:

$$s_i = \mathbf{q}_t^\top \mathbf{k}_i \quad \Rightarrow \quad \hat{i}_t = \arg\max_i s_i$$

This mechanism enables the agent to dynamically select the most relevant instruction segment aligned with its current visual observation.

3.4 Training

We train the model end-to-end using two objectives: (1) a navigation loss that supervises the agent's action decisions, and (2) a sub-instruction prediction loss that aligns visual context with instruction segments. The total loss is:

$$L = \lambda_{\text{nav}} L_{\text{nav}} + \lambda_{\text{sub}} L_{\text{sub}},$$

where L_{nav} is the standard cross-entropy loss over ground-truth actions using teacher forcing, and L_{sub} is our sub-instruction prediction loss. The supervision for L_{sub} is provided by annotated sub-instruction labels from the Fine-Grained R2R (FGR2R) dataset Hong et al. [2020a].

4 Experiments

4.1 Evaluation Setup and Metrics

We evaluate Sub-Aligner on the VLN task under fine-tuned and zero-shot settings, reporting results on the R2R Anderson et al. [2018] and R4R Jain et al. [2019] benchmarks, focusing primarily on the validation unseen and test unseen splits. We measure navigation performance mainly using standard VLN metrics including Success Rate (SR) and Success weighted by Path Length (SPL). For the R4R benchmark, we additionally report sDTW and nDTW metrics to assess path fidelity. *See Appendix D for more metrics and the corresponding definitions.*

4.2 Main Results

Table 1 shows the comparison between Sub-Aligner against prior VLN agents under both fine-tuning and zero-shot setups. On R2R Anderson et al. [2018], it raises DUET's SR from 72% to 74% and SPL from 60% to 64%. Plugging our sub-instruction localizer into MapGPT boosts SR from 44% to 49% and SPL by +7 without retraining, demonstrating the plug-and-play effectiveness of our module.

We further assess the generalizability of our method by evaluating it on the R4R dataset Jain et al. [2019]. As shown in Table 1, our method improves DUET's SR by +0.45 and sDTW by +1.76, when models are pretrained on R2R and fine-tuned on R4R. Sub-instruction guidance continues to improve agents' instruction following (sDTW +1.33), shows that explicit instruction-path alignment benefits generalization to longer paths.

Ablation Summary We also analyze component contributions and sub-instruction prediction accuracy. Results (Appendix C) show that (i) higher prediction accuracy directly correlates with better SR/SPL, and (ii) both the sub-instruction localizer and scene description contribute complementary gains.

Table 1: Key results on R2R and R4R (val-unseen). *Full results with more baselines are in Appendix Table 2.*

Dataset	Model	SR↑	SPL↑	sDTW↑	nDTW↑
R2R	DUET Chen et al. [2022]	72.0	60.0	–	–
	+ Sub-Aligner (ours)	**74.0**	**64.0**	–	–
	MapGPT Chen et al. [2024]	44.0	35.0	–	–
	+ Sub-Aligner (ours)	**49.0**	**42.0**	–	–
R4R	DUET Chen et al. [2022]	47.8	**43.5**	24.3	38.0
	+ Sub-Aligner (ours)	**48.3**	43.4	**26.1**	**38.8**

5 Conclusion

We present Sub-Aligner, a plug-and-play module for sub-instruction prediction in VLN. By aligning visual observations with instruction segments at each step, it improves interpretability and long-horizon decision-making. Together with a dual-stage VLM-based scene description module, Sub-Aligner enhances semantic grounding and achieves consistent gains across both finetuned and zero-shot agents on R2R and R4R. These findings support our intuition that *scene-aware sub-instruction grounding* is a missing piece for existing VLN pipelines.

References

Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton Van Den Hengel. 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, 2018.

Jiaqi Chen, Bingqian Lin, Ran Xu, Zhenhua Chai, Xiaodan Liang, and Kwan-Yee K. Wong. 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*, 2024.

Shizhe Chen, Pierre-Louis Guhur, Cordelia Schmid, and Ivan Laptev. History aware multimodal transformer for vision-and-language navigation. In *NeurIPS*, 2021.

Shizhe Chen, Pierre-Louis Guhur, Makarand Tapaswi, Cordelia Schmid, and Ivan Laptev. 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, 2022.

Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, Jiasen Lu, Taira Anderson, Erin Bransom, Kiana Ehsani, Huong Ngo, YenSung Chen, Ajay Patel, Mark Yatskar, Chris Callison-Burch, Andrew Head, Rose Hendrix, Favyen Bastani, Eli VanderBilt, Nathan Lambert, Yvonne Chou, Arnavi Chheda, Jenna Sparks, Sam Skjonsberg, Michael Schmitz, Aaron Sarnat, Byron Bischoff, Pete Walsh, Chris Newell, Piper Wolters, Tanmay Gupta, Kuo-Hao Zeng, Jon Borchardt, Dirk Groeneveld, Jen Dumas, Crystal Nam, Sophie Lebrecht, Caitlin Wittlif, Carissa Schoenick, Oscar Michel, Ranjay Krishna, Luca Weihs, Noah A. Smith, Hannaneh Hajishirzi, Ross Girshick, Ali Farhadi, and Aniruddha Kembhavi. Molmo and pixmo: Open weights and open data for state-of-the-art multimodal models. *arXiv preprint arXiv:2409.17146*, 2024.

Daniel Fried, Ronghang Hu, Volkan Cirik, Anna Rohrbach, Jacob Andreas, Louis-Philippe Morency, Taylor Berg-Kirkpatrick, Kate Saenko, Dan Klein, and Trevor Darrell. Speaker-follower models for vision-and-language navigation. In *Neural Information Processing Systems (NeurIPS)*, 2018.

Weituo Hao, Chunyuan Li, Xiujun Li, Lawrence Carin, and Jianfeng Gao. Towards learning a generic agent for vision-and-language navigation via pre-training. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13134–13143, 2020. doi: 10.1109/CVPR42600.2020.01315.

Yicong Hong, Cristian Rodriguez, Qi Wu, and Stephen Gould. Sub-instruction aware vision-and-language navigation. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3360–3376, Online, November 2020a. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.271. URL https://aclanthology.org/2020.emnlp-main.271/.

Yicong Hong, Qi Wu, Yuankai Qi, Cristian Rodriguez-Opazo, and Stephen Gould. Vlnbert: A recurrent vision-and-language bert for navigation. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1643–1653, 2020b. URL https://api.semanticscholar.org/CorpusID:227228335.

Vihan Jain, Gabriel Magalhaes, Alexander Ku, Ashish Vaswani, Eugene Ie, and Jason Baldridge. Stay on the path: Instruction fidelity in vision-and-language navigation. In Anna Korhonen, David Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1862–1872, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1181. URL https://aclanthology.org/P19-1181/.

Hongxin Li, Zeyu Wang, Xu Yang, Yuran Yang, Shuqi Mei, and Zhaoxiang Zhang. Memonav: Working memory model for visual navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17913–17922, 2024.

Jialu Li and Mohit Bansal. Panogen: Text-conditioned panoramic environment generation for vision-and-language navigation. *Advances in neural information processing systems*, 36:21878–21894, 2023.

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

Chih-Yao Ma, Jiasen Lu, Zuxuan Wu, Ghassan AlRegib, Zsolt Kira, Richard Socher, and Caiming Xiong. Self-monitoring navigation agent via auxiliary progress estimation. *arXiv preprint arXiv:1901.03035*, 2019a.

Chih-Yao Ma, Jiasen Lu, Zuxuan Wu, Ghassan AlRegib, Zsolt Kira, Richard Socher, and Caiming Xiong. Self-monitoring navigation agent via auxiliary progress estimation. *arXiv preprint arXiv:1901.03035*, 1901.03035, 2019b. URL https://arxiv.org/abs/1901.03035.

Arjun Majumdar, Ayush Shrivastava, Stefan Lee, Peter Anderson, Devi Parikh, and Dhruv Batra. Improving vision-and-language navigation with image-text pairs from the web. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

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

Xin Wang, Qiuyuan Huang, Asli Celikyilmaz, Jianfeng Gao, Dinghan Shen, Yuan-Fang Wang, William Yang Wang, and Lei Zhang. Reinforced cross-modal matching and self-supervised imitation learning for vision-language navigation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6622–6631, 2019. doi: 10.1109/CVPR.2019.00679.

Jingyi Zhang, Jiaxing Huang, Sheng Jin, and Shijian Lu. Vision-language models for vision tasks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

Yue Zhang and Parisa Kordjamshidi. Explicit object relation alignment for vision and language navigation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 322–331, 2022.

Yue Zhang and Parisa Kordjamshidi. Vln-trans: Translator for the vision and language navigation agent. *arXiv preprint arXiv:2302.09230*, 2302.09230, 2023. URL https://arxiv.org/abs/2302.09230.

Gengze Zhou, Yicong Hong, and Qi Wu. Navgpt: Explicit reasoning in vision-and-language navigation with large language models. *arXiv preprint arXiv:2305.16986*, 2023.

Gengze Zhou, Yicong Hong, and Qi Wu. 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, 2024.

A Limitation

Sub-Aligner depends on sub-instruction supervision from FGR2R, which may not generalize to new datasets without annotation. Its dual-stage VLM module also adds offline computational cost and inherits VLM-specific biases. Moreover, the model assumes clear-cut sub-instruction boundaries and does not handle ambiguous or overlapping intents.

B Additional Results

Table 2: Experimental results on the R2R val-unseen.

Model	**Val Unseen**		**Test Unseen**	
	SR↑	SPL↑	SR↑	SPL↑
Zero-shot (ZS)				
NavGPT Zhou et al. [2023]	34	29	–	–
MapGPT-GPT-4 Chen et al. [2024]	39	26	–	–
MapGPT-GPT-4V Chen et al. [2024]	44	35	–	–
+ Sub-Aligner (ours)	**49**	**42**	–	–
Pretrained + finetuned				
PREVALENT Hao et al. [2020]	58	53	54	51
RecBERT Hong et al. [2020b]	63	57	63	57
HAMT Chen et al. [2021]	66	61	65	60
DUET Chen et al. [2022]	72	60	69	59
+ GPT-4o Sub-indexer	60	42	–	–
+ Sub-Aligner (ours)	**74**	**64**	**71**	**60**

C Ablation Study

Sub-instruction Prediction To assess the impact of sub-instruction quality, we simulate varying prediction accuracies during inference by replacing the predicted sub-instruction with the ground truth sub-instruction at increasing rates, from 70% to 100% accuracy. The navigation policy remains fixed; only the sub-instruction selection is perturbed. Fig. 2 plots SR/SPL as a function of sub-instruction accuracy, confirming its impact on downstream performance. Each 10% improvement in ACC yields roughly 0.5% gain in SR and 0.2% gain in SPL. This supports our claim that progress tracking is a critical bottleneck.

Table 2 also compares our indexer to GPT-4o. Despite GPT-4o's scale, with no task-specific tuning trails, it lags behind our predictor by +12 SR with 43.09% ACC, validating the benefit of targeted sub-instruction modeling.

Component Analysis We ablate two core components of Sub-Aligner in Table 3: **S** denotes the sub-instruction localizer introduced in Section 3.3, and **V** refers to scene description with panorama. Notably, the sub-instruction localizer (S) alone yields gains over DUET. Fusing it with visual grounding (V) achieves the best balance of SR and SPL, confirming that explicit progress tracking and context-aware fusion are complementary.

V	**S**	TL↓	NE↓	SR↑	SPL↑
–	–	13.94	3.31	72.00	60.00
✓	–	14.47	3.22	72.12	60.72
–	✓	12.81	3.07	72.50	62.74
✓	✓	14.05	**2.88**	**74.37**	**64.05**

Table 3: Ablation on module components (R2R *val-unseen*). V = Scene description fused with visual panorama, S = Sub-instruction localizer.

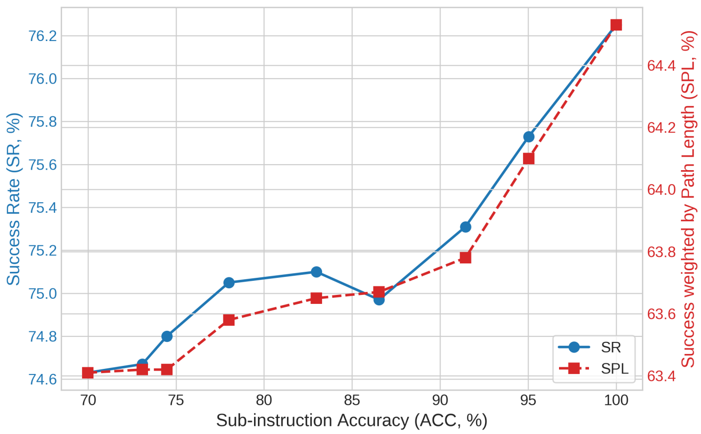

Figure 2: SR/SPL vs. sub-instruction prediction accuracy on R2R val-unseen.

D Evaluation Metrics

We evaluate R2R and R4R datasets (MIT License) using the following standard metrics to evaluate navigation and sub-instruction grounding:

- **Success Rate (SR):** Percentage of episodes where the agent ends within 3 meters of the goal.
- **Success weighted by Path Length (SPL):** Success rate weighted by the total navigation length.
- **Navigation Error (NE):** Distance between final agent location and target location.
- **Trajectory Length (TL):** The total distance taken by the agent during navigation.
- **Normalized Dynamic Time Warping (nDTW):** Measuring the path fidelity by penalizing deviations from the reference path.
- **Success rate weighted by Dynamic Time Warping (sDTW):** Considering nDTW of successful navigation.
- **Accuracy (ACC):** Accuracy of sub-instruction index prediction against ground-truth FGR2R annotations.

E Implementation Details

Our method builds upon DUET Chen et al. [2022], a dual-scale transformer that performs both global planning and local grounding. In our Dual-Stage Scene Description, we utilize MOLMO-7B-D Deitke et al. [2024] as the LLM to generate scene descriptions. All experiments are tested on a server with 4×NVIDIA A100 GPUs.

F Prompt Design for Scene Descriptions

We adopt a dual-stage prompt-based generation process using a vision-language model (MOLMO-7B-D Deitke et al. [2024]) to synthesize semantic descriptions of the environment:

F.1 Stage 1: Directional View Descriptions

Each directional image (`left`, `front`, `right`, `back`) is paired with a view-specific prompt to guide the model toward generating short, navigation-relevant descriptions (typically 1–2 sentences).

Directional Prompt Example (Left View)

You are an **advanced navigation expert**. You see one image, which is the **left view** of a room. Please provide a **concise (1-2 sentences)** description focusing on **key landmarks or objects relevant to navigation** (e.g., doorways, tables, chairs, paintings). **Pay special attention to the left side**... (truncated)

F.2 Stage 2: Global Fusion Prompt

Once all four directional captions are generated, we compose them into a single coherent viewpoint-level description using the panoramic image and the following global prompt:

Global Prompt for Fusion

You have four partial descriptions of a room from different views: **Left view**, **Front view**, **Right view**, **Back view**. Please combine all these details into a single cohesive first-person narrative...

Follow these specific rules:

- Begin with “I’m standing in the main living area. In front of me, ...”
- Follow the order: front → left → right → back
- Do not invent or speculate beyond the sub-descriptions
- Output in valid JSON format with the key `overall_description`