NavGPT-2: Unleashing Navigational Reasoning Capability for Large Vision-Language Models

Gengze Zhou[1], Yicong Hong[2], Zun Wang[3], Xin Eric Wang[4], and Qi Wu[1](✉)

[1] AIML, University of Adelaide, Adelaide, Australia
{gengze.zhou, qi.wu01}@adelaide.edu.au
[2] Adobe Research, San Jose, USA
[3] University of North Carolina, Chapel Hill, USA
[4] University of California, Santa Cruz, USA

Abstract. Capitalizing on the remarkable advancements in Large Language Models (LLMs), there is a burgeoning initiative to harness LLMs for instruction following robotic navigation. Such a trend underscores the potential of LLMs to generalize navigational reasoning and diverse language understanding. However, a significant discrepancy in agent performance is observed when integrating LLMs in the Vision-and-Language navigation (VLN) tasks compared to previous downstream specialist models. Furthermore, the inherent capacity of language to interpret and facilitate communication in agent interactions is often underutilized in these integrations. In this work, we strive to bridge the divide between VLN-specialized models and LLM-based navigation paradigms, while maintaining the interpretative prowess of LLMs in generating linguistic navigational reasoning. By aligning visual content in a frozen LLM, we encompass visual observation comprehension for LLMs and exploit a way to incorporate LLMs and navigation policy networks for effective action predictions and navigational reasoning. We demonstrate the data efficiency of the proposed methods and eliminate the gap between LM-based agents and state-of-the-art VLN specialists. The source code is available at https://github.com/GengzeZhou/NavGPT-2.

Keywords: Vision-and-Language Navigation · Large Language Models · Vision-Language Models

1 Introduction

Motivating by the considerable advances in Large Language Models (LLMs), there is an emerging effort to utilize these models for instructional tasks within robotic navigation [10, 46, 53, 81, 82]. This development highlights two core capacities of LLMs: Firstly, the ability to generalize commonsense knowledge reasoning and efficiently process free-form linguistic inputs, thanks to learning enormous amounts of textual data from the web. Secondly, the interpretative of LLMs to provide navigational reasoning explicitly in a human interpretable way and the associated communicative potential during interaction with humans. Several

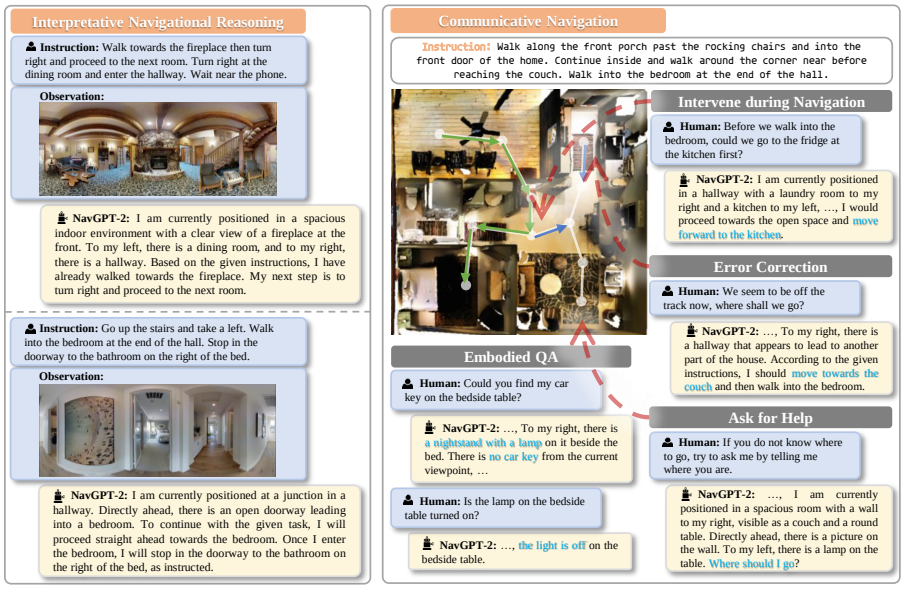

Fig. 1: Left: Besides performing effective navigation planning, NavGPT-2 is capable of generating navigational reasoning in a human-interpretable way. Right: NavGPT-2 can support multi-round interaction with the user and plan according to the user's intervention in the navigation process, actively ask for help, and answer visual questions.

studies have been initiated to integrate LLMs in the context of Vision-and-Language Navigation (VLN) [6]. Specifically, we recognize two major lines of research, including *zero-shot VLN with LLMs* and *finetune LLMs for VLN*. However, these approaches reveal a notable performance gap towards agents designed and trained tailored for solving VLN [10, 46, 82], usually lie at two extremes that carry significant limitations:

– For the zero-shot approach, previous studies prompt the LLMs with comprehensive descriptions of the navigation task and progress on the fly as the agent moves in an environment [10, 45, 46, 77, 82]. To enable the language models to "see", the agent's stepwise observations are often translated to textual descriptions generated by image captioning models such as BLIP-2 [36]. Meanwhile, the agent's past observations and decisions are summarized in pure language because the partially observable VLN process demands the agent to keep track of their experience for future planning. Eventually, the LLMs will read all the above information and infer an action that takes the agent towards the target. Despite the fact that such formulation largely exploits the generalizable knowledge of LLMs for navigation and eliminates the need to train a specialized agent on scarce embodied data, it involves complicated and fragile prompt engineering, increasingly expensive step-wise prompting, as well as noisy captioning and summarization which inevitably causes loss of information. More crucially, it still remains an open question whether LLMs can correctly understand spatial

structures and the consequence of physical motions, as evident by the huge performance gap between the LLM zero-shot approach and VLN-specialized models (∼40% success rate gap on R2R [6]). As a result, we suggest that the zero-shot approach based on existing LLMs is likely infeasible to solve VLN.

– For the fine-tuning approach, a range of research attempts to tune LLMs on instruction-trajectory pairs from VLN datasets [41, 53, 78, 81]. Specifically, the visual observations are fed to the language model either as encoded representations or as textual descriptions, and the actions are generally converted to structured textual format to adapt the auto-regressive prediction training of language models [53, 78]. Although these approaches [41, 53, 81] utilize the generalizable pretrained weights of LLMs and are much larger in scale[1], their performance still falls far behind the VLN-specialized models, likely due to the insufficient amount of training data and the discrepancy between LLMs' pretraining objectives and VLN's training target of aligning multi-view images and partial instruction. More crucially, directly tuning LLMs for VLN discards their general language capabilities, for instance, the potential to describe and explain the navigation process and communicate with humans for interactive tasks. Losing these abilities in fact against one of the most important motivations of introducing LLMs to embodied AI, yielding "black-box" uncontrollable agents.

In light of this, we propose NavGPT-2, a system that finds a balance between the two aforementioned extremes, incorporating effective navigational modules to facilitate navigational capabilities for VLM. Specifically, we built upon the VLM architecture of InstructBLIP [20], facilitating it with multi-image perception to adapt to VLN tasks. We enable VLM with navigational reasoning ability by constructing step-wise navigational reasoning data with GPT-4V and performing visual instruction tuning. We employ VLM latent as the composite visual-linguistic representation for both language decoding and action decoding with topological graphs, to enable agents to trace the long-term navigation history and effective backtracking while preserving the general language abilities of LLMs. As shown in Figure 1, NavGPT-2 could generate interpretative actions with language, and demonstrate the significant potential of building a communicative VLN agent that allows users to receive feedback and foster a connection with robots.

Our contributions are as follows: (1) We propose a pipeline to incorporate VLN specialists with VLMs free from LLM training. (2) Leveraging the robust feature enhancement afforded by pretrained VLMs, NavGPT-2 eliminates the gap between LM-based agents and SOTA VLN specialists. (3) We reserve the communicative instinct of LMs, enabling the models to explicitly explain the reason behind each navigation decision. These abilities are essential for building a practical and interactive VLN agent.

2 Related Works

Vision-and-Language Navigation (VLN) The pursuit of developing a universal navigation agent capable of following free-form linguistic directives to

[1] LLaMA2-7B [67] have 6.74 billion parameters while DUET [15] has only 0.18 billion.

navigate within an unfamiliar photorealistic environment has been a longstanding objective in the field of Vision-and-Language Navigation [7, 33, 56, 65]. These methods solve the task from two main aspects. (1) Vision-Language alignment. Some work [13, 15, 26–28, 39, 51, 57, 79] benefits from generic visual-linguistic representations [17, 37, 40, 62, 63], some exploit additional supervision from data augmentation [4, 22, 26, 34, 35, 54, 64, 69, 75], training stragies [29, 49, 72, 73, 84] to learn such cross-modality alignment. (2) Efficient action planning mechanism with historical state memorization [14, 28], self-correction [32, 50, 79], navigation map construction [2, 12, 16, 44, 70, 74, 80], and external knowledge prompts [38, 42]. In this work, we investigate the method of boosting a simplified prevalent VLN policy with visual-linguistic representations in LLM's latent space.

Large Language Models in VLN Introducing Large Language Models (LLMs) in robotics navigation is for its superior language understanding, communicative intrinsic, and commonsense reasoning ability. Previous work in VLN mainly exploits three strategies to combine LLM in solving VLN: (1) Model ensemble. Mic [58] utilizes GPT-2 [59] to provide extra guidance for downstream VLN specialists. (2) LLM as zero-shot VLN agents. NavGPT [82] reveals the potential navigational reasoning capacity of off-the-shell LLMs with a complex prompting system. DiscussNav [46] propose a multi-agent system while MapGPT [10] introduce topological mapping for zero-shot VLN agents to further improve their performance. However, a large performance gap is observed compared to supervised methods, even if the most powerful GPT-4 [52] models are used. (3) Finetune language models as VLN agents. LangNav [53] and NavCoT [41] finetune a LlaMA-7B [66] on VLN data to investigate the effectiveness of language as perceptual representation to perform navigation. NaviLLM [81] perform multi-task learning to finetune a LlaMA-7B [66] into VLN generalist.

Modality Alignment in Large Vision-Language Models (VLM) Leveraging the recent progress in Large Language Models (LLMs), there is a growing endeavor to repurpose pretrained Large Language Models for multimodal tasks, encompassing the comprehension and interpretation of visual information [1, 8, 11, 20, 23, 36, 43, 55, 71, 83]. A feasible way to connect different modalities for the pretrained LLMs is by training query-based visual resamplers, which is initially introduced by Flamingo [1] and BLIP-2 [36], and followed by subsequent implementations in MiniGPT-4 [83], InstructBLIP [20] and Qwen-VL [8]. Concurrently, another strand of research has focused on employing fully connected projection layers to directly map the output from vision encoders to the input of LLMs. This method is exemplified by LLaVA [43] and MiniGPT-v2 [11]. In this work, we adopt the Q-former designed in InsturctBLIP [20] to effectively control the length of visual content for multiple view images input at each viewpoint.

3 Method

The architecture of NavGPT-2, as depicted in Figure 2, comprises two primary components: a Large Vision-Language Model (VLM) and a navigation policy network. Within the VLM, visual observations and instructions are processed by

a component referred to as the Q-former to extract image tokens. These tokens serve as the input visual content for the LLM, enabling it to generate navigational reasoning. For action prediction, the model employs both hidden representations of image tokens and instruction text tokens that have been processed by the LLM encoder as the input features.

Problem Formulation. Given an instruction composed of L word embeddings $\mathcal{W} = \{w_i\}_{i=1}^{L}$, an agent is required to follow the instruction to navigate in a pre-defined undirected graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{V}$ denotes the navigable nodes, $\mathcal{E}$ denotes the connectivity edges. At step t, the agent perceives the surrounding environment through the observation of a set of RBG views for each connected navigable node candidate $\mathcal{O}_t \triangleq \{\langle o_i, a_i \rangle\}_{i=1}^{N}$, where N denotes the number of candidate nodes, each unique view is denoted as $o_i (i \leq N)$, with its associated angle direction with respect to the agent's heading represented as $a_i (i \leq N)$. The agent predicts the subsequent action by selecting the relative angle a_t from $\mathcal{O}_t$, the policy π parametrized by Θ that the agent is required to learn is $\pi(a_t | \mathcal{W}, \mathcal{O}_t; \Theta)$.

3.1 VLMs Latent as Visual-Linguistic Representation

In this section, we discuss the model design within the Large Vision-Language Model, how to enable frozen LLMs to generate navigational reasoning, and how to utilize VLM features in the navigation policy network.

Visual Aligning with LLMs. To effectively encode multiple-view images in the environment and construct spatial perception for navigation reasoning in a frozen LLM, we adopt the Q-former [36] design and encode each view into fixed-length visual tokens, shown in Figure 2. Specifically, for a candidate view image o_i, we incorporate a frozen ViT-g/14 from EVA-CLIP [24] as the vision encoder to extract visual feature $\mathcal{Z}_i^v = \text{ViT}(o_i)$. These visual features are later cross-attended [68] with 32 learnable queries embedding $\mathcal{Q}_i \in \mathbf{R}^{32 \times 768}$ which is self-attended with text embedding $\mathcal{W}$ of the instruction first to obtain the instruction-aware image queries $\mathcal{Q}'_i$ [20]. These queries are fed to the LLM after a linear projection W as the image tokens $\mathcal{H}_i^v = \mathcal{Q}'_i W$.

Navigation System Prompt. To inform LLM of the orientation of each candidate's view, we inject the direction information into the navigation prompt in structure input format "Candidate i, facing a_i, {direction}", shown in Figure 3. Moreover, we introduce special tokens `<IMG>`, `</IMG>`, `<INST>` and `</INST>` to insert images tokens and instructions into the prompt. Furthermore, we generate 10K navigational reasoning data from the R2R training set [6] and perform instruction-tuning to the Q-former and the projection layer on the prediction tokens, using its original auto-regressive training objective, detail is discussed in section 3.3. Up to now, a complete VLM has been built and can generate navigation reasoning with a standard LLM decoding process, shown in the bottom right of Figure 2.

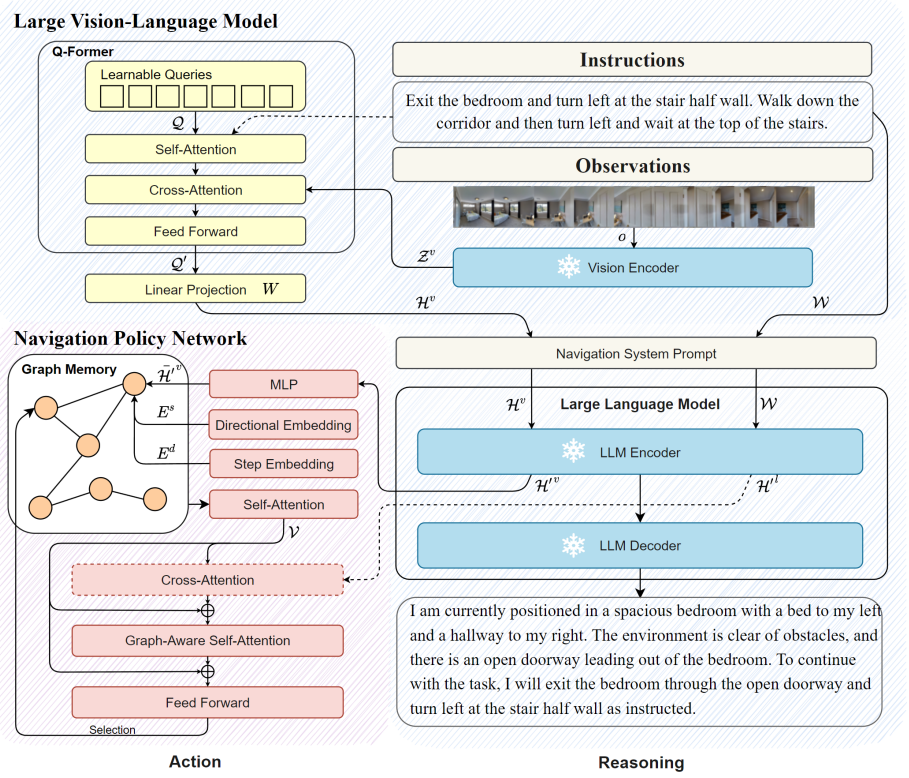

Fig. 2: Model architecture of NavGPT-2, it consists of a multimodality Large Language Model and a topological graph-based navigation policy network. The yellow blocks indicate the trainable module at stage one, the red blocks indicate the trainable module at stage two, and the blue blocks are frozen.

VLM Latents as Visual-Linguistic Representation. For an LLM $f_\phi(\cdot)$ parameterized by ϕ, we extract the LLM Latents $\mathcal{H}'^v = \{f_\phi(\mathcal{H}_i^v)\}_{i=1}^N, \mathcal{H}'^l = f_\phi(\mathcal{W})$ as Visual-Linguistic Representation for Navigation Policy. For encoder-decoder based LLMs, we retrieve the hidden representation of the image tokens and instruction tokens from the last Transformer encoder layer. For decoder-only LLMs, we obtain the latents from the last decoder layer. Specifically, the 32 image tokens for each view are merged into a single token through a multi-layer perception $\bar{\mathcal{H}}'^v = \text{MLP}(\mathcal{H}'^v)$, shown in Figure 2.

3.2 Graph Based Navigation Policy

We identify the key difficulty of fine-tuning LLMs as VLN agents lies in the LLMs' inadequate comprehension of spatial structures, coupled with their limited ability to model the agent's long-range experiences during the navigation process. Therefore, we harness a topological graph-based navigation policy [15] for effective

You are navigating in an indoor environment given the instruction: <INST>{instruction}</INST>;
The navigable locations are listed below: {
"Candidate 1, facing a_1 degree, front" : <IMG>{image_tokens}</IMG>;
"Candidate 2, facing a_2 degree, right" : <IMG>{image_tokens}</IMG>;
...};
Please choose the next direction.

Fig. 3: Navigation system prompt for NavGPT-2.

action planning. The topological graph is maintained on the fly and served as a memorization mechanism to trace the navigation experience. NavGPT-2 choose the next step from the entire constructed topological graph, enabling effective planning and back-tracing to unvisited nodes when a wrong path is taken. We introduce the graph-based policy in the following sections.

Node Embedding. The graph memory consists of visited nodes and adjacent unexplored nodes along the trajectory. All the candidate views from each visited node are average-pooled to represent this node, while each underexplored node is represented by the partial pooling of corresponding views from all its adjacent visited nodes in the trajectory. Each view is represented by the summation of its visual features $\bar{\mathcal{H}'}^v$, its directional embedding E^d representing the location of each node and step embedding E^s representing the traverse order of the agent's current episode. The step embedding of the unexplored nodes is 0 and a 'stop' node is added to the graph memory to denote a stop action. A multi-layer transformer is implemented to model the spatial relation between nodes:

$$\mathcal{V} = \text{SelfAttn}\left(\frac{1}{M}\sum_{i=1}^{M}\left(\bar{\mathcal{H}'}_i^v + E_i^d + E_i^s\right)\right), \tag{1}$$

where M is the number of views representing the node.

Cross-Modal Encoding. The navigation graph constructs at step t is denoted as $\mathcal{G}_t = \{\mathcal{V}_t, \mathcal{E}_t\}, \mathcal{G}_t \subset \mathcal{G}$. The node embeddings $\mathcal{V}_t$ are sent to a multi-layer cross-modal transformer to model the relationship between instructions and nodes. Specifically, the node embeddings are first cross-attended with the instructions encoded by the LLM, then go through a graph-aware self-attention (GASA), which considers both distances and visual similarities between nodes to enhance contextual understanding:

$$\text{GASA}(\mathcal{V}) = \text{Softmax}\left(\frac{\mathcal{V}W_q(\mathcal{V}W_k)^T}{\sqrt{d}} + A(\mathcal{E}_t)\right)\mathcal{V}W_v, \tag{2}$$

where $A(\mathcal{E}_t)$ represents the spatial affinity matrix, comprised of pairwise L2 distances among all observed nodes.

Global Action Prediction. We employ a two-layer feed-forward network to process the output node representations of the GASA to generate an action score. The agent selects the node with the highest score as the target, and follows the shortest path in the graph memory to control to the selected node. Note that we mask the scores for visited nodes to encourage agent exploration following [15].

3.3 Multi-stage Learning for Action and Reasoning

We perform a two-stage training to learn action prediction and navigation reasoning generation for LLM. In the first stage, we initialize the model from InstructBLIP [20] after visual instruction-tuning on academic-task-oriented VQA datasets. We follow the same training schema to only finetune the Q-former with a frozen LLM and vision encoder on the collected navigation reasoning data, shown as the yellow blocks in Figure 2. In the second stage, we connect the pretrained VLM with the downstream navigation policy and only finetune the policy network with frozen VLM, shown as the red blocks in Figure 2.

Data Acquisition and Curation. To train VLM with navigational reasoning ability, we propose an automatic data generation pipeline with GPT-4V. We discard historical modeling for VLM and consider the situation when spanning the agent at the intermediate steps along the ground truth trajectory. We asked GPT-4V to determine the next step toward completing the instruction based on the current observation of the surroundings and relevant landmarks. We define the single-step navigation reasoning trace as describing the immediate environment and specifying the direction or action that will be taken to proceed. Details of the prompt are in the appendix.

We randomly select 10k intermediate steps from the trajectory in the R2R [6] training set, using the equirectangular projected panoramic image centring at the agent's heading direction as the image input for GPT-4V, shown in Figure 4.

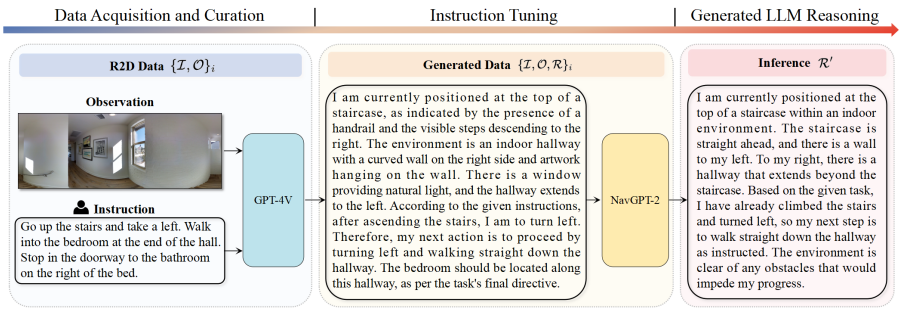

Fig. 4: Data generation pipeline and visual instruction tuning on navigation reasoning data. $\{\mathcal{I}, \mathcal{O}\}$ denotes the instruction-observation pairs on the R2R trajectories. $\mathcal{R}$ is the generated reasoning from GPT-4V, $\mathcal{R}'$ is the generated reasoning from NavGPT-2.

Policy learning with DAgger. When fine-tuning the downstream navigation policy network, we follow previous work to combine Behaviour cloning and DAgger loss [61]:

$$\mathcal{L}_{\text{BC}} = -\sum_{t=1}^{T} \log \pi(v_t^* | \mathcal{W}, \mathcal{G}_t), \quad \mathcal{L}_{\text{DAG}} = -\sum_{t=1}^{T} \log \pi(\tilde{v}_t^* | \mathcal{W}, \tilde{\mathcal{G}}_t), \tag{3}$$

where v_t^* indicates the ground truth action, $\tilde{v}_t^*$ denotes the pseudo label determined by the shortest path toward the destination from the partial graph $\tilde{\mathcal{G}}_t$ generated by the agent through on-policy action sampling. The overall loss function is given by $\mathcal{L} = \lambda \mathcal{L}_{\text{BC}} + \mathcal{L}_{\text{DAG}}$, where λ is a balancing factor.

4 Experiments

Evaluation Metrics. We adopt a comprehensive set of navigation metrics to evaluate performance [6], including Trajectory Length (TL), which measures the average path length in meters; Navigation Error (NE), the average distance between the final and target locations; Success Rate (SR), the percentage of paths with NE less than 3 meters; Oracle Success Rate (OSR), the success rate under an ideal stop policy; Success Rate penalized by Path Length (SPL) [5], which combines success with efficiency considerations; Normalized Dynamic Time Warping (NDTW) [30], assessing the fidelity between predicted and annotated paths; and Success weighted by Normalized Dynamic Time Warping (SDTW), a composite metric rewarding both navigation success and path fidelity.

4.1 Implementation Details

We build NavGPT-2 based on InstructBLIP [20] and exploit four variations of LLMs, including FlanT5-XL (3B), FlanT5-XXL (11B), Vicuna-7B and Vicuna-13B [18, 19]. All models apply the same vision encoder (ViT-g/14 [24]), and all parameters of the vision encoder and LLMs are kept frozen during the entire training process. In stage one, we initialize the model from pretrained InstructBLIP checkpoints and train the Q-former for 200K steps with a batch size of 8. The AdamW optimizer [47] is employed and configured with $\beta_1 = 0.9, \beta_2 = 0.999$ and a weight decay of 0.05. To optimize learning efficiency, a linear warmup strategy is applied to the learning rate for the first 1,000 steps, gradually increasing it from 10^{-8} to 10^{-5}, which is then followed by a cosine decay to a minimum learning rate of 0. In stage two, we freeze the pretrained VLM from stage one and finetune the downstream policy network with a batch size of 2 and a learning rate of 10^{-5}. Our best model is trained on the combination of the R2R and the synthetic data from PREVALENT [27]. All experiments are conducted on a single NVIDIA A100 GPU.

Table 1: Comparison of single-run performance on R2R dataset. NavGPT-2 surpass all pervious methods incorporating LLMs and eliminate the gap between SOTA methods on the same training scale. †: Methods that use additional visual data than 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 [6]	–	11.33	6.01	53	39	–	8.39	7.81	28	21	–	8.13	7.85	27	20	–
RCM [72]	–	10.65	3.53	75	67	–	11.46	6.09	50	43	–	11.97	6.12	50	43	38
Speaker Follower [25]	–	–	3.36	74	66	–	–	6.62	45	36	–	14.82	6.62	-	35	28
EnvDrop [64]	–	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 [27]	–	10.32	3.67	–	69	65	10.19	4.71	–	58	53	10.51	5.30	61	54	51
AirBert [26]†	–	11.09	2.68	–	75	70	11.78	4.10	–	62	56	12.41	4.13	–	62	57
VLN↻BERT [28]	–	11.13	2.90	–	72	68	12.01	3.93	–	63	57	12.35	4.09	70	63	57
MARVAL [31]†	–	10.60	2.99	–	73	69	10.15	4.06	–	65	61	10.22	4.18	67	62	58
HAMT [13]	–	11.15	2.51	–	76	72	11.46	2.29	–	66	61	12.27	3.93	72	65	60
HOP+ [57]	–	11.31	2.33	–	78	73	11.76	3.49	–	67	61	12.67	3.71	–	66	60
BEVBert [3]	–	13.56	2.17	**88**	**81**	74	14.55	2.81	84	75	64	15.87	3.13	81	73	62
DUET+ScaleVLN [75]†	–	11.90	2.16	87	80	**75**	12.40	2.34	**87**	**79**	**70**	14.27	2.73	**83**	**77**	**68**
Baseline:																
DUET [15]	–	12.32	2.28	86	79	73	13.94	3.31	81	72	60	14.73	3.65	76	69	59
w/o local branch	–	–	–	–	–	–	12.96	3.67	–	68	59	13.08	3.93	–	67	58
Language Models zero-shot VLN:																
NavGPT (GPT-4) [82]	✓	–	–	–	–	–	11.45	6.46	42	34	29	–	–	–	–	–
MapGPT (GPT-4) [10]	✓	–	–	–	–	–	–	6.92	58	39	26	–	–	–	–	–
DiscussNav (GPT-4) [46]	✓	–	–	–	–	–	9.69	5.32	61	43	40	–	–	–	–	–
Langage Models with/as VLN Specialists:																
NavCoT (LLaMA2-7B) [41]	✗	10.08	6.46	48	41	38	9.95	6.26	48	40	37	–	–	–	–	–
LangNav (LLaMA2-7B) [53]†	✗	–	–	–	56	–	–	–	–	46	–	–	–	–	–	–
NaviLLM (Vicuna-7B) [81]	✗	–	–	–	–	–	12.81	3.51	–	67	59	13.21	3.71	–	68	**60**
NavGPT-2$_{\text{FlanT5-XL}}$ (ours, 1.5B)	✓	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**
NavGPT-2$_{\text{FlanT5-XXL}}$ (ours, 5B)	✓	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**

4.2 Comparison with State of the Art

We compare the single-run performance on the R2R dataset with previous SOTA methods in Table 1. Specifically, we categorized them into distinct categories:

– ***VLN Specialists with Vision-Language-Action Pretraining*** [3, 14, 15, 26–28, 31, 57, 75]: These methods are initialized from general vision-language models [40, 63] and incorporate VLN-tailored pretraining with auxiliary tasks such as masked language modeling (MLM) [21], masked region classification (MRC) [48], and single-step action prediction (SAP) [14] before fine-tuning on downstream VLN tasks.

– ***Zero-shot Methods*** [10, 46, 82]: Methods that apply GPT-4 to zero-shot VLN using different textual inputs and prompting strategies.

– ***Methods finetuning LLMs*** [41, 53, 81]: Methods that finetune LLMs on VLN data with alternative modification on multimodality LLMs combining VLN specialized model designs.

– ***Baseline***: We construct a baseline method based on the global action branch of DUET [15], referring as DUET (w/o local branch) in Table 1. This model shares the same architecture for the action planning policy network as

NavGPT-2 and trains on the same amount of data. The only difference lies in the language branch, where we harness the LLM's latent and the baseline adopt a 12-layer transformer initialized from LXMERT [63]. Only a 2% SR drop is observed on the test when cutting off the local branch for DUET, demonstrating the global branch as the dominating branch during navigation, facilitating NavGPT-2 to simplify the navigation policy.

As shown in Table 1, we list our model with a similar size[2] as previous LLM-based methods, NavGPT-$2_{\text{FlanT5-XXL}}$ (1.5B) and NavGPT-$2_{\text{FlanT5-XXL}}$ (5B). Our 5B model outperforms NaviLLM [81] (7B) by 4% SR on test split while still maintaining the language capacity to generate self-explained navigation reasoning. NavGPT-2 far exceeding NavGPT [82] and surpasses all the zero-shot methods, since those methods rely on extensive prompt engineering and struggle to model navigation history. Compared to the baseline methods, NavGPT-2 bypass it by 5% SR and 2% SPL on the test split even if we do not incorporate with VLN pertaining. The current SOTA method [75] is achieved by scaling up the training environment for DUET with HM3D [60] and Gibson [76], besides the original 61 scenes in MP3D [9]. When considering the same training scale in MP3D, our method beats the original DUET by 3% in SR on the test unseen split and is comparable with metric map-based method [3] which incorporates an extra depth sensor to construct Bird's-Eye-View (BEV) perception.

4.3 Navigational Reasoning Generation

In Figure 1 we show the navigational reasoning generated by NavGPT-2 during navigation. NavGPT-2 can construct a comprehensive perception of the surroundings, as shown in the example on the left, NavGPT-2 recognizes the fireplace, dining room, hallway, and their relative locations. Moreover, it can reason about the navigation progress and identify associated sections of the instruction inferring the next step, and can even infer the expected observation.

Table 2: Human study on NavGPT-2 generated navigational reasoning.

Methods	Accuracy	Informativeness	Rationality
GPT-4V	2.31	2.95	2.34
NavGPT-$2_{\text{FlanT5-XL}}$	1.66	1.93	1.78

A human study is conducted in Table 2 on 30 held-out samples to evaluate the accuracy and informativeness of VLM reasoning. Specifically, we engaged 10 volunteers to evaluate the navigational reasoning generated by NavGPT-2, focusing on its accuracy, informativeness, and rationality. The accuracy assessment, with a scoring range from 0 (Entirely incorrect) to 3 (Completely accurate),

[2] Our model is smaller (1.5B and 5B) than original FlanT5 models (3B and 11B) as we only utilize the LLM encoder during navigation.

Table 3: Comparison of different scales of training data.

Methods	#	Val Seen					Val Unseen				
		TL	NE↓	OSR↑	SR↑	SPL↑	TL	NE↓	OSR↑	SR↑	SPL↑
10% R2R training data:											
DUET	1	12.75	6.18	53.97	44.47	38.43	13.08	5.78	58.19	48.19	39.93
NavGPT-2$_{\text{FlanT5-XL}}$	2	14.46	6.13	57.59	45.64	36.80	13.66	5.21	62.37	52.02	41.75
50% R2R training data:											
DUET	3	13.46	4.65	68.07	56.51	49.49	13.57	4.41	70.50	59.90	50.11
NavGPT-2$_{\text{FlanT5-XL}}$	4	12.95	3.99	68.85	61.51	52.55	14.01	3.98	71.90	63.30	51.83
100% R2R training data:											
DUET	5	12.38	3.62	73.36	66.31	60.13	13.20	4.07	71.95	63.90	54.83
NavGPT-2$_{\text{FlanT5-XL}}$	6	13.02	3.34	74.24	69.44	61.72	13.68	3.37	74.37	67.52	56.01

considered the precision in describing the surrounding environment, the correct recognition of navigation progress, and the appropriateness of the next action's planning. Informativeness was judged on a scale from 0 (Not informative) to 3 (Highly informative), based on the completeness of the environmental description provided. Rationality was also rated from 0 (Entirely irrational) to 3 (Completely rational), evaluating the correctness of the action planned by NavGPT-2.

As shown in Table 2, NavGPT-2 scored 1.66 on Accuracy, 1.93 on Informativeness, and 1.78 on Rationality, indicating the quality of generated reasonings is acceptable given the full mark is 3. Noticeably the GPT-4V scored 2.31, 2.95, and 2.34 respectively, demonstrating an effective way to generate navigational reasoning data.

4.4 The Effect of Data Amount

In Table 3 we initialize DUET from LXMERT and compare the model performance when finetuning 10%, 50%, and full R2R training data. NavGPT-2 outperforms all DUET variants in SR on the validation unseen split, and it reaches the same performance as DUET with full R2R data when feeding with 50% less data, showcasing the data efficiency of utilizing LLMs latent as the vision-language representation and the benefits for downstream navigation policy learning.

4.5 Cross Dataset Generalization Ability

We evaluate the generalization ability of NavGPT-2 in two aspects: generalize to free-form language instructions and to various unseen environments.

Generalize to Free-form Language Instructions. To assess NavGPT-2's comprehension of various forms of language instruction, we evaluate the zero-shot performance of NavGPT-2 on the RxR dataset. The RxR dataset is characterized by its instructions of finer granularity, detailing rich landmarks and encompassing longer trajectories. As shown in Table 4, NavGPT-2 outperforms DUET by 3.67%

Table 4: Comparison of zero-shot performance on RxR and HM3D.

Methods	#	RxR-EN					HM3D				
		nDTW↑	sDTW↑	OSR↑	SR↑	SPL↑	TL	NE↓	OSR↑	SR↑	SPL↑
DUET	1	37.77	17.39	44.54	25.07	19.65	20.27	6.60	42.70	25.60	13.32
NavGPT-2$_{\text{FlanT5-XXL}}$	2	38.50	19.24	48.56	**28.75**	**22.36**	17.96	4.91	69.80	**47.20**	**27.99**

in SR on RxR (English) unseen split, demonstrating our VLM-based method improves the out-of-domain language understanding capabilities.

Generalize to Unseen Environments. We highlight that generalization challenges in unseen environments stem from the biases in connectivity graphs of training environments and visual differences in new scenes. Therefore, we assess NavGPT-2's zero-shot performance in HM3D by sampling 1000 trajectories from ScaleVLN [75] using Habitat Simulator rendered images, which offer environments visually deviant and structurally distinct graphs from MP3D. As shown in Table 4, NavGPT-2 significantly outperforms DUET. We hypothesize this improvement is due to the projection of visual features into the same LLM hidden space as language, leading to a more robust alignment across environments.

4.6 Ablation Study

We ablate the core design choices applied in this paper, including the effect of incorporating a navigation-specific policy model, pretraining the Q-former with reasonings and leveraging different LLMs in NavGPT-2.

Effect of Policy Network. In Table 5, we study the necessity of applying a navigation-specific policy model in NavGPT-2. To achieve this, we remove all the visual-language cross-attention layers in the Q-former and policy network and use only a single graph-aware self-attention layer followed by a single feed-forward layer to predict the action (Model#2). By doing so, we force the LLM to take over the visual-textual based decision-making to exploit its navigational capability. It is clear from the drastic drop in performance that a frozen LLM is incapable of inferring effective representations that indicate a correct action. Although tuning the LLM should improve the results, we can see from previous work that tuning a LLaMA2-7B model on the full R2R data and GPT-4-augmented data only leads to a 43% SR in Val Unseen [53], still far behind NavGPT-2, implying that this approach might be infeasible with existing LLMs.

Effect of Pretraining with Reasonings. Following the above, we also highlighted the navigational reasoning abilities that NavGPT-2 unleashed from frozen LLMs in Figure 4. Additionally, we can see from Model#3 of Table 5 that the pretraining of Q-former on reasonings brings slight improvement to the success rates of the model. It is expected that such information, containing rich spatial descriptions and visual landmarks, facilitates the Q-former to produce better textual representations of the multi-view observations.

Effect of Different LLMs. Technically, NavGPT-2 can incorporate a wide range of different LLMs, however, their performance on VLN might not scale

Table 5: Effect of navigation policy network and pretrained Q-former for reasoning.

Methods	#	Val Seen					Val Unseen				
		TL	NE↓	OSR↑	SR↑	SPL↑	TL	NE↓	OSR↑	SR↑	SPL↑
NavGPT-2$_{\text{FlanT5-XL}}$	1	13.02	3.34	74.24	69.44	61.72	13.68	3.37	74.37	67.52	56.01
w/o policy model	2	25.57	8.03	68.85	25.27	13.92	26.70	8.03	69.60	21.46	10.23
w/o pre-trained Q-former for reasoning	3	12.29	3.67	71.79	67.58	60.62	13.04	3.75	73.82	66.75	57.10

Table 6: Comparison of different LLMs.

Methods	#	Val Seen					Val Unseen				
		TL	NE↓	OSR↑	SR↑	SPL↑	TL	NE↓	OSR↑	SR↑	SPL↑
NavGPT-2$_{\text{FlanT5-XL}}$	1	13.02	3.34	74.24	69.44	61.72	13.68	3.37	74.37	67.52	56.01
NavGPT-2$_{\text{FlanT5-XXL}}$	2	13.08	2.98	79.43	73.65	65.25	13.25	3.18	79.61	71.31	60.07
NavGPT-2$_{\text{Vicuna-7B}}$	3	11.85	4.85	63.37	53.57	46.83	12.29	4.86	65.56	53.77	45.26
NavGPT-2$_{\text{Vicuna-13B}}$	4	11.89	5.09	61.31	52.01	45.95	13.16	5.63	60.11	48.28	40.14

with the model size. We compare four variations of LLMs in NavGPT-2, shown in Table 6. For the encoder-decoder model FlanT5, a 3.79% increment in SR is observed on the Val Unseen split when the model size is increased from XL (Model#1, 1.5B) to XXL (Model#2, 5B). But for the decoder-only models Vicuna (Model#3 and Model#4), although they are larger in size than FlanT5-XXL, their performance is much worse. These results in fact align with the findings in InstructBLIP [20] which shows that the FlanT5-based model is superior at multi-choice questions while Vicuna-based models are generally better at open-ended generation tasks due to the difference in their training data. As a result, it is unsurprising that FlanT5-based NavGPT-2 performs better in the multi-choice setting of VLN. Furthermore, we hypothesize that the full attention in FlanT5's encoder is more effective in addressing the alignment task between visual and instruction sequences in VLN than the causal attention in decoder-only Vicuna. We will leave a detailed investigation of this problem for future work.

5 Conclusion

In this work, we strive to eliminate the gap between LLMs-based agents and VLN-specialised agents, while maintaining the interpretative intrinsic of LLMs to generate navigational reasonings during navigation. Through comprehensive experimentation, we highlight the critical aspects of integrating LLMs with downstream navigation policy networks. It is demonstrated that the Vision-Language Model (VLM) latent serves as a superior and more efficient visual-linguistic representation, enabling policy networks to learn better alignment between vision, language, and action. Our approach offers a scalable framework to leverage the broad language comprehension capabilities of LLMs, paving the way for the development of versatile navigation agents capable of interacting with humans and understanding free-form human intentions with greater efficacy.

Acknowledgements

We thank all the reviewers for their valuable comments and suggestions. Yicong Hong wants to thank NVIDIA for the Academic Hardware Grant that provided GPU support for this project. This project is supported by the University of Adelaide's Centre for Augmented Reasoning (CAR).

References

1. Alayrac, J.B., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., Lenc, K., Mensch, A., Millican, K., Reynolds, M., et al.: Flamingo: a visual language model for few-shot learning. Advances in Neural Information Processing Systems **35**, 23716–23736 (2022) 4
2. An, D., Qi, Y., Huang, Y., Wu, Q., Wang, L., Tan, T.: Neighbor-view enhanced model for vision and language navigation. In: Proceedings of the 29th ACM International Conference on Multimedia. pp. 5101–5109 (2021) 4
3. An, D., Qi, Y., Li, Y., Huang, Y., Wang, L., Tan, T., Shao, J.: Bevbert: Topo-metric map pre-training for language-guided navigation. arXiv preprint arXiv:2212.04385 (2022) 10, 11
4. 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) 4
5. 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) 9
6. Anderson, P., Wu, Q., Teney, D., Bruce, J., Johnson, M., Sünderhauf, N., Reid, I., Gould, S., van den Hengel, A.: Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3674–3683 (2018) 2, 3, 5, 8, 9, 10
7. Anderson, P., Wu, Q., Teney, D., Bruce, J., Johnson, M., Sünderhauf, N., Reid, I., Gould, S., Van Den Hengel, A.: Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3674–3683 (2018) 4
8. Bai, J., Bai, S., Yang, S., Wang, S., Tan, S., Wang, P., Lin, J., Zhou, C., Zhou, J.: Qwen-vl: A frontier large vision-language model with versatile abilities. arXiv preprint arXiv:2308.12966 (2023) 4
9. Chang, A., Dai, A., Funkhouser, T., Halber, M., Niebner, M., Savva, M., Song, S., Zeng, A., Zhang, Y.: Matterport3d: Learning from rgb-d data in indoor environments. In: 2017 International Conference on 3D Vision (3DV). pp. 667–676. IEEE (2017) 11
10. 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 preprint arXiv:2401.07314 (2024) 1, 2, 4, 10
11. Chen, J., Zhu, D., Shen, X., Li, X., Liu, Z., Zhang, P., Krishnamoorthi, R., Chandra, V., Xiong, Y., Elhoseiny, M.: Minigpt-v2: large language model as a unified interface for vision-language multi-task learning. arXiv preprint arXiv:2310.09478 (2023) 4

12. Chen, K., Chen, J.K., Chuang, J., Vázquez, M., Savarese, S.: Topological planning with transformers for vision-and-language navigation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11276–11286 (2021) 4
13. Chen, S., Guhur, P.L., Schmid, C., Laptev, I.: History aware multimodal transformer for vision-and-language navigation. Advances in Neural Information Processing Systems **34**, 5834–5847 (2021) 4, 10
14. Chen, S., Guhur, P.L., Schmid, C., Laptev, I.: History aware multimodal transformer for vision-and-language navigation. Advances in Neural Information Processing Systems **34**, 5834–5847 (2021) 4, 10
15. Chen, S., Guhur, P.L., Tapaswi, M., Schmid, C., Laptev, I.: 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. pp. 16537–16547 (2022) 3, 4, 6, 8, 10, 21
16. Chen, S., Guhur, P.L., Tapaswi, M., Schmid, C., Laptev, I.: 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. pp. 16537–16547 (2022) 4
17. Chen, Y.C., Li, L., Yu, L., El Kholy, A., Ahmed, F., Gan, Z., Cheng, Y., Liu, J.: Uniter: Universal image-text representation learning. In: European conference on computer vision. pp. 104–120. Springer (2020) 4
18. Chiang, W.L., Li, Z., Lin, Z., Sheng, Y., Wu, Z., Zhang, H., Zheng, L., Zhuang, S., Zhuang, Y., Gonzalez, J.E., Stoica, I., Xing, E.P.: Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality (March 2023), https://lmsys.org/blog/2023-03-30-vicuna/ 9
19. Chung, H.W., Hou, L., Longpre, S., Zoph, B., Tay, Y., Fedus, W., Li, Y., Wang, X., Dehghani, M., Brahma, S., et al.: Scaling instruction-finetuned language models. arXiv preprint arXiv:2210.11416 (2022) 9
20. 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. arXiv preprint arXiv:2305.06500 (2023) 3, 4, 5, 8, 9, 14
21. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018) 10
22. Dou, Z.Y., Peng, N.: Foam: A follower-aware speaker model for vision-and-language navigation. arXiv preprint arXiv:2206.04294 (2022) 4
23. Driess, D., Xia, F., Sajjadi, M.S., Lynch, C., Chowdhery, A., Ichter, B., Wahid, A., Tompson, J., Vuong, Q., Yu, T., et al.: Palm-e: An embodied multimodal language model. arXiv preprint arXiv:2303.03378 (2023) 4
24. Fang, Y., Wang, W., Xie, B., Sun, Q., Wu, L., Wang, X., Huang, T., Wang, X., Cao, Y.: Eva: Exploring the limits of masked visual representation learning at scale. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 19358–19369 (2023) 5, 9, 23
25. 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) 10
26. 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) 4, 10

27. 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) 4, 9, 10
28. Hong, Y., Wu, Q., Qi, Y., Rodriguez-Opazo, C., Gould, S.: A recurrent vision-and-language bert for navigation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1643–1653 (June 2021) 4, 10
29. Huang, H., Jain, V., Mehta, H., Ku, A., Magalhaes, G., Baldridge, J., Ie, E.: Transferable representation learning in vision-and-language navigation. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 7404–7413 (2019) 4
30. Ilharco, G., Jain, V., Ku, A., Ie, E., Baldridge, J.: General evaluation for instruction conditioned navigation using dynamic time warping. arXiv preprint arXiv:1907.05446 (2019) 9
31. 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. arXiv preprint arXiv:2210.03112 (2022) 10
32. 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) 4
33. Ku, A., Anderson, P., Patel, R., Ie, E., Baldridge, J.: Room-across-room: Multilingual vision-and-language navigation with dense spatiotemporal grounding. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 4392–4412 (2020) 4
34. Li, J., Bansal, M.: Panogen: Text-conditioned panoramic environment generation for vision-and-language navigation. arXiv preprint arXiv:2305.19195 (2023) 4
35. 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) 4
36. Li, J., Li, D., Savarese, S., Hoi, S.: Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. arXiv preprint arXiv:2301.12597 (2023) 2, 4, 5
37. Li, L.H., Yatskar, M., Yin, D., Hsieh, C.J., Chang, K.W.: Visualbert: A simple and performant baseline for vision and language. arXiv preprint arXiv:1908.03557 (2019) 4
38. Li, X., Wang, Z., Yang, J., Wang, Y., Jiang, S.: Kerm: Knowledge enhanced reasoning for vision-and-language navigation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2583–2592 (2023) 4
39. Li, X., Li, C., Xia, Q., Bisk, Y., Celikyilmaz, A., Gao, J., Smith, N., Choi, Y.: Robust navigation with language pretraining and stochastic sampling. arXiv preprint arXiv:1909.02244 (2019) 4
40. Li, X., Yin, X., Li, C., Zhang, P., Hu, X., Zhang, L., Wang, L., Hu, H., Dong, L., Wei, F., et al.: Oscar: Object-semantics aligned pre-training for vision-language tasks. In: European Conference on Computer Vision. pp. 121–137. Springer (2020) 4, 10
41. 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) 3, 4, 10

42. Lin, B., Zhu, Y., Chen, Z., Liang, X., Liu, J., Liang, X.: Adapt: Vision-language navigation with modality-aligned action prompts. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15396–15406 (2022) 4
43. Liu, H., Li, C., Wu, Q., Lee, Y.J.: Visual instruction tuning. Advances in neural information processing systems **36** (2024) 4
44. Liu, R., Wang, X., Wang, W., Yang, Y.: Bird's-eye-view scene graph for vision-language navigation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10968–10980 (2023) 4
45. 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) 2
46. Long, Y., Li, X., Cai, W., Dong, H.: Discuss before moving: Visual language navigation via multi-expert discussions. arXiv preprint arXiv:2309.11382 (2023) 1, 2, 4, 10
47. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: International Conference on Learning Representations (2018) 9
48. Lu, J., Batra, D., Parikh, D., Lee, S.: Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. Advances in neural information processing systems **32** (2019) 10
49. 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) 4
50. 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) 4
51. Majumdar, A., Shrivastava, A., Lee, S., Anderson, P., Parikh, D., Batra, D.: Improving vision-and-language navigation with image-text pairs from the web. In: European Conference on Computer Vision. pp. 259–274. Springer (2020) 4
52. OpenAI: Gpt-4 technical report (2023) 4
53. 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) 1, 3, 4, 10, 13
54. Parvaneh, A., Abbasnejad, E., Teney, D., Shi, J.Q., van den Hengel, A.: Counterfactual vision-and-language navigation: Unravelling the unseen. Advances in Neural Information Processing Systems **33**, 5296–5307 (2020) 4
55. Peng, Z., Wang, W., Dong, L., Hao, Y., Huang, S., Ma, S., Wei, F.: Kosmos-2: Grounding multimodal large language models to the world. arXiv preprint arXiv:2306.14824 (2023) 4
56. 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) 4
57. 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 (2023) 4, 10
58. 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) 4
59. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al.: Language models are unsupervised multitask learners. OpenAI blog **1**(8), 9 (2019) 4

60. Ramakrishnan, S.K., Gokaslan, A., Wijmans, E., Maksymets, O., Clegg, A., Turner, J.M., Undersander, E., Galuba, W., Westbury, A., Chang, A.X., et al.: Habitat-matterport 3d dataset (hm3d): 1000 large-scale 3d environments for embodied ai. In: Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2) (2021) 11
61. 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) 9
62. Su, W., Zhu, X., Cao, Y., Li, B., Lu, L., Wei, F., Dai, J.: Vl-bert: Pre-training of generic visual-linguistic representations. arXiv preprint arXiv:1908.08530 (2019) 4
63. Tan, H., Bansal, M.: Lxmert: Learning cross-modality encoder representations from transformers. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). pp. 5100–5111 (2019) 4, 10, 11, 21
64. Tan, H., Yu, L., Bansal, M.: Learning to navigate unseen environments: Back translation with environmental dropout. In: Proceedings of NAACL-HLT. pp. 2610–2621 (2019) 4, 10
65. Thomason, J., Murray, M., Cakmak, M., Zettlemoyer, L.: Vision-and-dialog navigation. In: Conference on Robot Learning. pp. 394–406 (2020) 4
66. Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al.: Llama: Open and efficient foundation language models. arXiv preprint arXiv:2302.13971 (2023) 4
67. Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., et al.: Llama 2: Open foundation and fine-tuned chat models. arXiv preprint arXiv:2307.09288 (2023) 3
68. 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) 5
69. 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) 4
70. Wang, H., Wang, W., Shu, T., Liang, W., Shen, J.: Active visual information gathering for vision-language navigation. In: European Conference on Computer Vision. pp. 307–322. Springer (2020) 4
71. Wang, W., Chen, Z., Chen, X., Wu, J., Zhu, X., Zeng, G., Luo, P., Lu, T., Zhou, J., Qiao, Y., et al.: Visionllm: Large language model is also an open-ended decoder for vision-centric tasks. Advances in Neural Information Processing Systems **36** (2024) 4
72. 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) 4, 10
73. Wang, X., Xiong, W., Wang, H., Wang, W.Y.: Look before you leap: Bridging model-free and model-based reinforcement learning for planned-ahead vision-and-language navigation. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 37–53 (2018) 4
74. 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) 4

75. 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. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 12009–12020 (2023) 4, 10, 11, 13
76. Xia, F., Zamir, A.R., He, Z., Sax, A., Malik, J., Savarese, S.: Gibson env: Real-world perception for embodied agents. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 9068–9079 (2018) 11
77. 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) 2
78. Zhang, J., Wang, K., Xu, R., Zhou, G., Hong, Y., Fang, X., Wu, Q., Zhang, Z., He, W.: Navid: Video-based vlm plans the next step for vision-and-language navigation. arXiv preprint arXiv:2402.15852 (2024) 3
79. Zhao, C., Qi, Y., Wu, Q.: Mind the gap: Improving success rate of vision-and-language navigation by revisiting oracle success routes. In: Proceedings of the 31st ACM International Conference on Multimedia. pp. 4349–4358 (2023) 4
80. Zhao, Y., Chen, J., Gao, C., Wang, W., Yang, L., Ren, H., Xia, H., Liu, S.: Target-driven structured transformer planner for vision-language navigation. In: Proceedings of the 30th ACM International Conference on Multimedia. pp. 4194–4203 (2022) 4
81. Zheng, D., Huang, S., Zhao, L., Zhong, Y., Wang, L.: Towards learning a generalist model for embodied navigation. arXiv preprint arXiv:2312.02010 (2023) 1, 3, 4, 10, 11
82. 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) 1, 2, 4, 10, 11
83. Zhu, D., Chen, J., Shen, X., Li, X., Elhoseiny, M.: Minigpt-4: Enhancing vision-language understanding with advanced large language models. arXiv preprint arXiv:2304.10592 (2023) 4
84. 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) 4

Supplementary Material for NavGPT-2

Section A provides additional details for DUET as it is the main comparison with NavGPT-2. The prompt for GPT-4V used in the data generation pipeline and additional experiment results are described in Section B and Section C. Section D illustrates the limitation of NavGPT-2 with the discussion of future directions. Finally, Section E discusses the broader impacts of our work.

A DUET Revisit

NavGPT-2 exploit the similar design adapted from Dual-scale Graph Transformer (DUET) [15] as the downstream navigation policy. It includes a text encoder to encode instructions, a global and a local branch to enable coarse-scale and fine-scale cross-modal reasoning.

A.1 Text Embedding and Visual Embedding

For the text encoder, DUET utilizes a 12-layer transformer initialized from LXMERT [63]. For visual embedding, the visual observation at each node is 36 view images from 12 horizontal directions times 3 vertical directions. To distinguish these nodes, a directional embedding E^{ang} of the absolute angle for each view is added to the visual feature $\mathcal{Z}^v$ extracted by the vision encoder. Moreover, since DUET inputs all 36 view images to construct the spatial observation for the model, the navigable adjacent nodes are only observed at a few view images, denoted as navigable views. A navigable embedding E^{nav} is added to the visual features. The final visual embedding is sent to a 2 layers transformer to encode the spatial relations between views and obtain the panoramic view embeddings:

$$\mathcal{H}^{pano} = \text{SelfAttn}\left(\mathcal{Z}^v + E^{ang} + E^{nav}\right). \tag{4}$$

On the contrary, NavGPT-2 only inputs the navigable views, thus the directional embedding E^{ang} and the navigable embedding E^{nav} are removed in the downstream policy, instead we directly add the step embedding and location embedding before sending to the 2 layers transformer.

A.2 DUET Local Branch

NavGPT-2 adopt the same navigation policy network architecture as the DUET global branch, discussed in §3.2[3], so we omit the explanation of the global branch in DUET. In this section, we introduce the local branch of DUET. This branch performs action prediction based on the current node's instruction and egocentric observation. No graph information is provided besides the local observation.

[3] Refer to section 3.2 in the main paper.

Local Visual Embedding Two types of location embedding are added to the panoramic view embedding $\mathcal{H}^{pano}$. The first type is the relative location of the current node to the starting node, to encode the long distance direction between nodes. The second type is each adjacent view to the current node, to encode egocentric directions such as "turn right".

Local Cross-model Encoding The local branch utilizes a standard cross-modal transformer of 4 layers to model vision and language relations. During action prediction, a mask is set to the unnavigable views, and the action logits are only calculated for the navigable views at the current node.

A.3 Dynamic Fusion

The final action prediction of DUET is performed by dynamically fusing the action predicted by local and global branches. The local branch predicts actions within the adjacent nodes $\mathcal{V}_t^a$. It is incongruent with the action space used by the global branch, which chooses the next action from all nodes $\mathcal{V}_t$ in the constructed graph at step t. To reconcile this discrepancy, the local action scores s_i^l are transformed encompassing options such as "stop" and $\mathcal{V}_t$, into a representation suitable for the global action space by summing up scores of visited nodes in $\mathcal{V}_t^a$ as a backtrack score s_b:

$$s_i^{l'} = \begin{cases} s_{\mathrm{b}}, & \text{if } \mathcal{V}_i \in \mathcal{V}_t - \mathcal{V}_t^a, \\ s_i^{l'}, & \text{otherwise.} \end{cases} \tag{5}$$

This adjustment facilitates navigation toward other unexplored nodes not directly linked to the current node, necessitating the agent to retrace its steps through neighboring nodes that have previously been visited. The final navigation score is given by:

$$s_i = \sigma_t s_i^g + (1 - \sigma_t) s_i^{l'}, \tag{6}$$

where s_i^g is the logits from global branch, σ_t is a learnable scalar for fusion.

B GPT-4V Prompt

The prompt used for GPT-4V to generate navigation reasoning, discussed in section §3.3 is shown in Figure 5.

C Additional Rerults

In this section, we conduct additional experiments to illustrate the choice of navigation policy network for NavGPT-2 and the effectiveness of LLM features. To align the same training schema of the navigation policy, we conduct the experiments for DUET initiating it from LXMERT without VLN specialized pretraining.

{image}
As an AI navigating an indoor environment, you're given the task {instruction}. You find yourself at a particular juncture within the execution of this command. Based on your current observation of the surroundings, including obstacles, pathways, and relevant landmarks, determine the next step toward completing this task. Your response should briefly describe your immediate environment and specify the direction or action you will take to proceed. Summarize this in a concise paragraph, integrating both your observation and decision-making process.

Fig. 5: Navigation reasoning generation prompt for GPT-4V.

Table 7: Comparison of single-run performance on R2R dataset.

Methods	#	Val Seen					Val Unseen				
		TL	NE↓	OSR↑	SR↑	SPL↑	TL	NE↓	OSR↑	SR↑	SPL↑
w/o Visual-Language-Action Pretrain:											
DUET	1	12.38	3.62	73	66	60	13.20	4.07	72	64	55
w/o local branch	2	11.43	3.50	74	67	62	12.08	4.08	71	62	54
w/ EVA-CLIP-g	3	12.64	3.73	73	66	60	14.27	4.07	72	63	54
NavGPT-2$_{\text{FlanT5-XL}}$ (ours, 1.5B)	4	13.02	3.34	74	69	62	13.68	3.37	74	68	56
NavGPT-2$_{\text{FlanT5-XXL}}$ (ours, 5B)	5	13.08	2.98	79	74	65	13.25	3.18	80	71	60

C.1 Effect of Vision Encoder

Because NavGPT-2 exploits a stronger vision encoder [24], we conduct an ablation study on the original DUET to investigate the performance gain brought by the vision encoder. As shown in Table 7, after switching the visual representation to the stronger vision feature same as NavGPT-2, little performance gain is observed for the DUET global branch (Model # 3 compared to Model # 2). We hypothesize this is due to the global branch for DUET performing vision-language alignment on a coarse scale, while the fine-grained alignment is performed in the local branch. Therefore, the main performance gain in NavGPT-2 is not contributed by the stronger vision encoder but the better representation from LLM hidden.

C.2 Effect of VLN Pretrain

We consider the same training scale and the same training schema of DUET as NavGPT-2, without pertaining auxiliary VLN tasks and directly finetuning on the VLN dataset. Under the same training schema and scale of data, NavGPT-2 performs significantly better than the original DUET, shown in Table 7. This showcases the superiors of LLM features that enable the learning of cross-modality alignment in the downstream task when the visual feature is projected to the

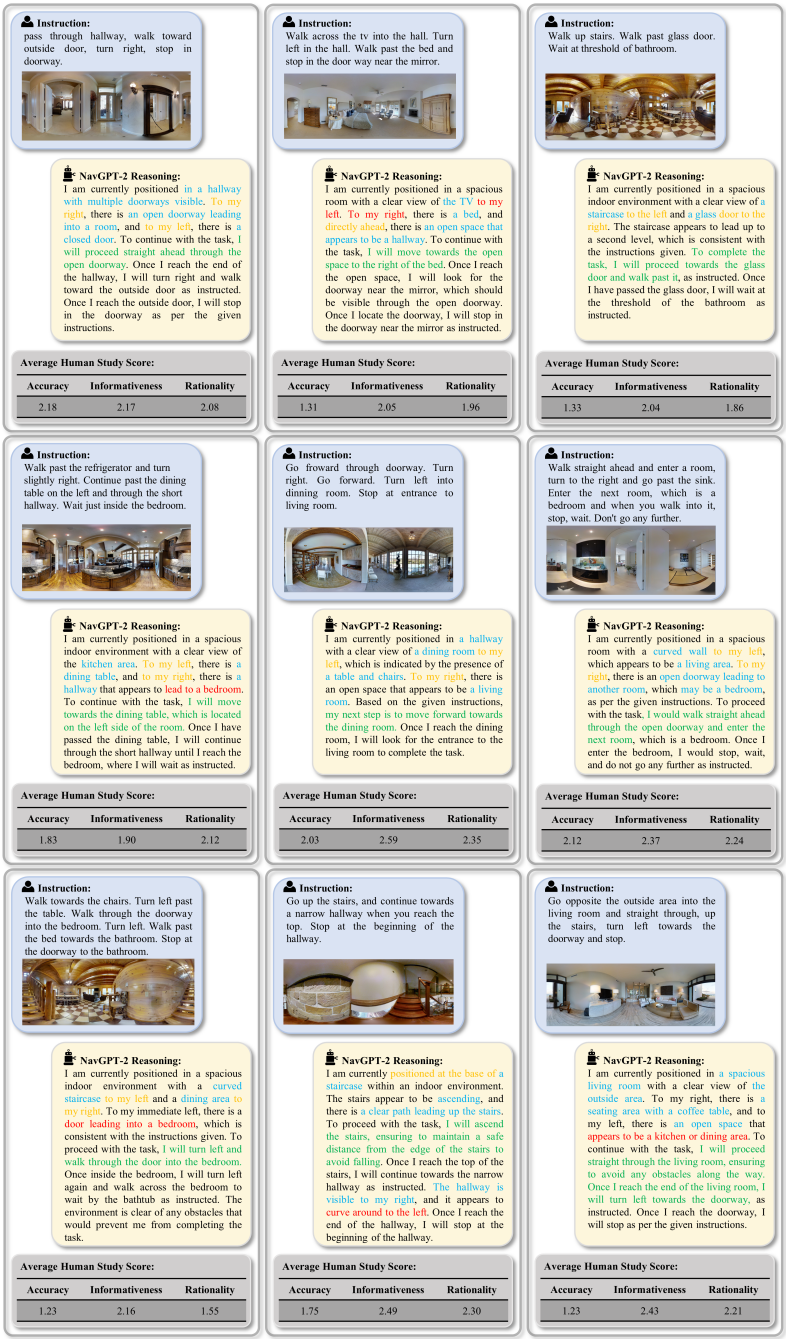

Fig. 6: Qualitive Results for NavGPT-2. It can correctly recognize object and scenes and their corresponding locations, grouding the observation to the given instruction and plan the next step. However, hallucination of the non-existent object or misjudged the direction is also observed.

LLM's language space by the Q-former. Without VLN tailored pertaining, the performance of DUET significantly drops. We leave adding the pertaining process for the downstream navigation policy in future work.

C.3 Additional Qualitive Results

In this section, we present extra qualitative results in addition to §4.3. In Figure 6, we present the navigational reasoning produced by NavGPT-2 during navigation. NavGPT-2 is capable of forming a detailed understanding of its surroundings with objects and scenes and their corresponding orientations. Furthermore, it adeptly reasons about the progress of navigation and correlates it with specific portions of the instruction. Impressively, it is also able to predict expected observations, such as "appears to lead to a bedroom," based on the current visual inputs. This demonstrates NavGPT-2's ability not only to navigate but also to anticipate and interpret complex environments intelligently.

D Limitations and Future Work

Although NavGPT-2 could generate navigation reasoning to some extent, it is hard to evaluate the effectiveness of these reasonings, since it is set as a single-step reasoning based on local observation and does not model the navigation history in the VLM. Instead, such history information is encoded in the downstream navigation policy. As a result, the consistency between navigation reasonings is underexplored. Moreover, the reasoning and action predicted by downstream navigation policy are not strictly synchronized in NavGPT-2, such synchronization could be done either explicitly by tuning LLM with the same supervision signal of action or by collaborating with the reasoning generation loss during fine-tuning the downstream policy network, we leave the synchronization to future work. Finally, the communicative capability of NavGPT-2 is not evaluated in this work, we suggest investigating the communicative ability of LM-based VLN agents and the synchronization between their reasoning and actions as a future direction.

E Broader Effect

Our research endeavors to leverage Large Vision-Language Models (VLM) to develop VLN agents, while preserving the linguistic prowess of VLMs for explaining action predictions in natural language. We posit that the inherent communicative capability, commonsense knowledge, and broad linguistic comprehension of VLM constitute the cornerstone for creating instruction-following navigation agents with generalizability. NavGPT-2 illuminates the reasonings of VLM throughout the navigation process explicitly and interpretably. Due to safety and ethical considerations, we currently conduct all experiments using the open-source Vision-and-Language Navigation dataset within a simulated environment, which ensures controlled agent behavior. Concurrently, we acknowledge that the potential practical application of this technology warrants further exploration, particularly in

terms of action and reasoning synchronization, which remains an underexplored area. Notably, we observe the propensity of VLMs to hallucinate non-existent scenes or objects and fail to identify object directions, shown in Figure 6, which is also a common issue within VLM research. Future investigations are essential to address how to harmonize VLM action and reasoning and to enhance the agent's ability to self-explain in a manner intelligible to humans, a critical consideration for ensuring safety in real-world applications.