NavGPT: Explicit Reasoning in Vision-and-Language Navigation with Large Language Models

Gengze Zhou[1], Yicong Hong[2], Qi Wu[1]*

[1]The University of Adelaide
[2]The Australian National University
{gengze.zhou, qi.wu01}@adelaide.edu.au, mr.yiconghong@gmail.com

Abstract

Trained with an unprecedented scale of data, large language models (LLMs) like ChatGPT and GPT-4 exhibit the emergence of significant reasoning abilities from model scaling. Such a trend underscored the potential of training LLMs with unlimited language data, advancing the development of a universal embodied agent. In this work, we introduce the NavGPT, a purely LLM-based instruction-following navigation agent, to reveal the reasoning capability of GPT models in complex embodied scenes by performing zero-shot sequential action prediction for vision-and-language navigation (VLN). At each step, NavGPT takes the textual descriptions of visual observations, navigation history, and future explorable directions as inputs to reason the agent's current status, and makes the decision to approach the target. Through comprehensive experiments, we demonstrate NavGPT can explicitly perform high-level planning for navigation, including decomposing instruction into sub-goals, integrating commonsense knowledge relevant to navigation task resolution, identifying landmarks from observed scenes, tracking navigation progress, and adapting to exceptions with plan adjustment. Furthermore, we show that LLMs is capable of generating high-quality navigational instructions from observations and actions along a path, as well as drawing accurate top-down metric trajectory given the agent's navigation history. Despite the performance of using NavGPT to zero-shot R2R tasks still falling short of trained models, we suggest adapting multi-modality inputs for LLMs to use as visual navigation agents and applying the explicit reasoning of LLMs to benefit learning-based models. Code is available at: https://github.com/GengzeZhou/NavGPT.

Introduction

Amid the remarkable advances in large language model (LLM) training (Touvron et al. 2023; Brown et al. 2020; Chowdhery et al. 2022; Zhang et al. 2022; Wei et al. 2021; Bubeck et al. 2023; OpenAI 2023), we note a shift towards integrating LLMs into embodied robotics tasks such as SayCan (Ahn et al. 2022) and PaLM-E (Driess et al. 2023). This trend stems from two primary considerations: the scale of training data and the scale of models. First, the development of techniques for processing textual information provides an abundant source of natural language training data for learning interdisciplinary and generalizable knowledge. Furthermore, by accessing unlimited language data, significant emergent abilities (Wei et al. 2022a) are observed when scaling up the model, resulting in a remarkable enhancement in the reasoning capabilities when solving problems across wide domains. Consequently, training an LLM with unlimited language data is seen as a viable pathway toward realizing a universal embodied agent.

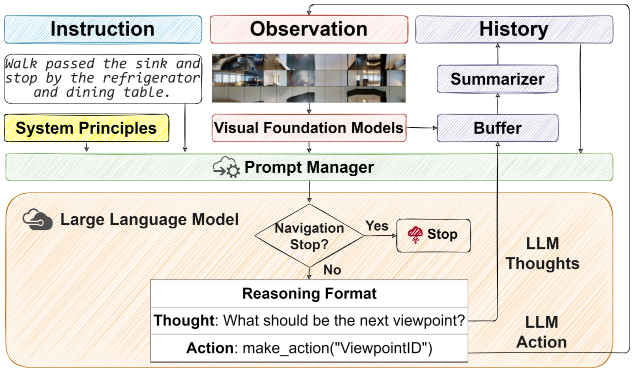


Figure 1: The architecture of NavGPT. NavGPT synergizes reasoning and actions in LLMs to perform zero-shot Vision-and-Language Navigation following navigation system principles. It interacts with different visual foundation models to adapt multi-modality inputs, handle the length of history with a history buffer and a summarizer, and aggregate various sources of information through a prompt manager. NavGPT parses the generated results from LLMs (LLM *Thoughts* and LLM *Action*) to move to the next viewpoint.

This insight has spurred the integration of LLMs into vision-and-language navigation (VLN) (Anderson et al. 2018), an exploratory task toward achieving real-world instruction-following embodied agents. The latest research attempt to leverage GPT models (OpenAI 2023; Brown et al. 2020) to benefit navigation. For example, using LLMs as a parser for diverse language input (Shah et al. 2023) — extracting landmarks from instruction to support visual matching and planning, or leveraging LLMs' commonsense reasoning abilities (Zhou et al. 2023; Dorbala, Mullen Jr, and Manocha 2023) to incorporate prior knowledge of inter-object correlations to extend agents' perception and facilitate the decision making. However, we notice that the rea-

*Corresponding author.

soning ability of LLMs in navigation is still under-explored, *i.e.* , can LLMs understand the interactive world, the actions, and consequences in text form, and use all the information to solve a navigation task?

In light of this, we introduce NavGPT, a fully automatic LLM-based system designed for language-guided visual navigation, with the capability to handle multi-modality inputs, unconstrained language guidance, interaction with an open-world environment, and progress tracking with navigation history. NavGPT perceives the visual world by reading descriptions of observations generated by visual foundation models (VFMs), and synergizing *Thoughts* (reasoning) and *Actions* (decision making) in an explicit text form. To an extreme extent, we use NavGPT to perform zero-shot VLN[1] to clearly reveal the reasoning process of LLMs during navigation.

Through comprehensive experiments, we found that LLMs possess the capability to execute complex navigational planning. This includes the deconstruction of instructions into distinct sub-goals, assimilation of commonsense knowledge pertinent to navigational tasks, identification of landmarks within the context of observed environments, continuous monitoring of navigational progression, and responding to anomalies by modifying their initial plan. The aforementioned phenomenon reflects an astonishing reasoning ability in understanding and solving navigation problems. Furthermore, we show that LLMs have the ability to draw navigation trajectories in a metric map and regenerate navigation instruction based on navigation history, revealing the historical and spatial awareness of LLMs for navigation tasks. However, there remains a significant gap between the zero-shot performance of current open-sourced LLMs in VLN compared to the fine-tuned models, where the bottleneck of NavGPT lies in the information loss while translating visual signals into natural language and summarizing observations into history. As a result, we suggest the future direction of building general VLN agents to be LLMs with multi-modality inputs or a navigation system making use of high-level navigation planning, historical and spatial awareness from LLMs.

Our contributions can be summarized as follow: (1) We introduce a novel instruction-following LLMs agent for visual navigation with a supportive system to interact with the environment and track navigation history. (2) We investigate the capabilities and limitations of current LLMs' reasoning for making navigation decisions. (3) We reveal the capability of LLMs in high-level planning for navigation, by observing the thoughts of LLMs, making the planning process of navigation agents accessible and explainable.

Related Work

Vision-and-Language Navigation Language-driven vision navigation is demanded by widely applicable embodied navigation agents. Previous study shows the essentials of modules to achieve such a goal (Anderson et al. 2018; Qi et al. 2020b; Krantz et al. 2020; Ku et al. 2020; He et al. 2021; Gu et al. 2022; Zhu et al. 2022; Hong et al. 2020a, 2022; Zhao, Qi, and Wu 2023; Qiao et al. 2023b), whereas a large number of research reveal the crucial effect of training strategies (Wang et al. 2019; Tan, Yu, and Bansal 2019). Importantly, the main problem in VLN is the generalizability of agents in unseen environments. Data augmentation (Wang et al. 2022; Li, Tan, and Bansal 2022; Tan, Yu, and Bansal 2019; Parvaneh et al. 2020; Li and Bansal 2023), memory mechanism (Chen et al. 2021b; Pashevich, Schmid, and Sun 2021; Hong et al. 2023), pre-training (Hao et al. 2020; Chen et al. 2022a; Qiao et al. 2023a; Wang et al. 2023) have been adopted to alleviate data scarcity. However, those augmentations and pre-training are limited to the sampled data from a fixed number of scenes, which is not enough to reflect a realistic application scene where objects could be out of the domains and language instructions are more diverse. In our work, we utilize the reasoning and knowledge storage of LLMs and perform VLN in a zero-shot manner as an initial attempt to reveal the potential usage of LLMs for VLN in the wild. A number of studies (Chen et al. 2021a; Deng, Narasimhan, and Russakovsky 2020; Chen et al. 2022b) have presented compelling methodologies that underscore the significance of topological maps in facilitating long-term planning, specifically in the aspect of backtracking to prior locations. In addition, Dorbala *et al.* (Dorbala et al. 2022) use CLIP (Radford et al. 2021) to perform zero-shot VLN by chunking instructions into keyphrases and completely rely on the text-image matching capability from CLIP to navigate. However, the planning and decision-making processes of the agents above are implicit and not accessible. On the contrary, benefiting from the intrinsic of LLMs, we are able to access the reasoning process of agents, making it explainable and controllable.

Large Language Models With the massive success in large-scale language model training (Touvron et al. 2023; Brown et al. 2020; Chowdhery et al. 2022; Zhang et al. 2022; Wei et al. 2021), a new cohort of Large Language Models (LLMs) has shown evolutionary progress toward achieving Artificial General Intelligence (AGI) (Bubeck et al. 2023; OpenAI 2023). This burgeoning class of LLMs, underpinned by increasingly sophisticated architectures and training methodologies (Scao et al. 2022), has the potential to revolutionize various domains by offering unprecedented capabilities in natural language understanding and generation. The main concern for LLMs is that their knowledge is limited and confined after training is finished. The latest works study how to utilize LLMs interacting with tools to expand their knowledge as a plugin, including extending LLM to process multimodality content (Wu et al. 2023; Yongliang et al. 2023), teaching LLMs to access the internet with correct API calls (Schick et al. 2023), and expanding their knowledge with local databases to accomplish QA tasks (Peng et al. 2023). Another stream of works studies how to prompt LLMs in a hierarchical system to facilitate the alignment of reasoning and corresponding actions (Yao et al. 2022; Karpas et al. 2022) beyond the Chain of Thought

[1] Our NavGPT is solely powered by off-the-shelf LLMs, without any learnable module or any prior experience in solving interactive navigation. Hence, all navigation tasks defined in this paper are novel to NavGPT.

(CoT) (Wei et al. 2022b). These works set up the preliminaries for building an embodied agent directly using LLMs.

LLMs in Robotics Navigation The employment of Large Language Models (LLMs) in the field of robotics remains in the primary stage (Vemprala et al. 2023; Bubeck et al. 2023). A handful of contemporary studies, however, have begun to explore the utilization of generative models for navigation. Shah *et al.* (Shah et al. 2023) employs GPT-3 (Brown et al. 2020) in an attempt to identify "landmarks" or subgoals, while Huang *et al.* (Huang et al. 2022) concentrates its efforts on the application of an LLM for the generation of code. Zhou *et al.* (Zhou et al. 2023) use LLM to extract the commonsense knowledge of the relations between targets and objects in observations to perform zero-shot object navigation (ZSON) (Gadre et al. 2022; Majumdar et al. 2022). Despite these recent advancements, our study diverges in its concentration on converting visual scene semantics into input prompts for the LLM, directly performing VLN based on the commonsense knowledge and reasoning ability of LLMs. The work closest to ours is LGX (Dorbala, Mullen Jr, and Manocha 2023), but they are doing object navigation where agents are not required to follow the instruction and in their method, they use the GLIP (Li et al. 2022a) model to decide the stop probability and did not consider memorization of navigation history, action, and reasoning between LLM.

Method

VLN Problem Formulation

We formulate the VLN problem as follows: Given a natural language instruction $\mathcal{W}$, composed of a series of words $\{w_1, w_2, w_3, \dots, w_n\}$, at every step s_t, the agent interprets the current location via the simulator to obtain an observation $\mathcal{O}$. This observation comprises N alternative viewpoints, representing the egocentric perspectives of agents in varying orientations.

Each unique view observation is denoted as $o_i(i \leqslant N)$, with its associated angle direction represented as $a_i(i \leqslant N)$. The observation can thus be defined as $\mathcal{O}_t \triangleq [\langle o_1, a_1\rangle, \langle o_2, a_2\rangle, \dots, \langle o_N, a_N\rangle]$. Throughout the navigation process, the agents' action space is confined to the navigation graph G. The agent must select from the $M = |C_{t+1}|$ navigable viewpoints, where C_{t+1} indicates the set of candidate viewpoints, by aligning the observation $\mathcal{O}_t^C \triangleq [\langle o_1^C, a_1^C\rangle, \langle o_2^C, a_2^C\rangle, \dots, \langle o_M^C, a_M^C\rangle]$ with the oracle $\mathcal{W}$. The agent prognosticates the subsequent action by selecting the relative angle a_i^C from $\mathcal{O}_t^C$, then enacts this action through interaction with the simulator to transition from the current state $s_t = \langle v_t, \theta_t, \phi_t\rangle$ to $s_{t+1} = \langle v_{t+1}, \theta_{t+1}, \phi_{t+1}\rangle$, where v denotes the current viewpoint location, θ denotes the current heading angle, and ϕ denotes the current elevation angle of the agent. The agent also maintains a record of the state history h_t and adjusts the conditional transition probability between states $\mathcal{S}_t = T(s_{t+1}|a_i^C, s_t, h_t)$, where function T denotes the conditional transition probability distribution.

In summary, the policy π parametrized by Θ that the agent is required to learn is based on the oracle $\mathcal{W}$ and the current observation $\mathcal{O}_t^C$, which is $\pi(a_t|\mathcal{W}, \mathcal{O}_t, \mathcal{O}_t^C, \mathcal{S}_t; \Theta)$. In this study, NavGPT conducts the VLN task in a zero-shot manner, where the Θ is not learned from the VLN datasets, but from the language corpus that the LLMs are trained on.

NavGPT

NavGPT is a system that interacts with environments, language guidance, and navigation history to perform action prediction. Let $\mathcal{H}_{<t+1} \triangleq [\langle \mathcal{O}_1, \mathcal{R}_1, \mathcal{A}_1\rangle, \langle \mathcal{O}_2, \mathcal{R}_2, \mathcal{A}_2\rangle, \dots, \langle \mathcal{O}_t, \mathcal{R}_t, \mathcal{A}_t\rangle]$ be the navigation history of observation $\mathcal{O}$, LLM reasoning $\mathcal{R}$ and action $\mathcal{A}$ triplets for the previous t steps. To obtain the navigation decision $\mathcal{A}_{t+1}$, NavGPT needs to synergize the visual perception from VFMs $\mathcal{F}$, language instruction $\mathcal{W}$, history $\mathcal{H}$ and navigation system principle $\mathcal{P}$ with the help of prompt manager $\mathcal{M}$, define as follow:

$$\begin{aligned}&\langle \mathcal{R}_{t+1}, \mathcal{A}_{t+1}\rangle \\ =&LLM(\mathcal{M}(\mathcal{P}), \mathcal{M}(\mathcal{W}), \mathcal{M}(\mathcal{F}(\mathcal{O}_t)), \mathcal{M}(\mathcal{H}_{<t+1}))\end{aligned} \tag{1}$$

Navigation System Principle $\mathcal{P}$. The Navigation System Principle formulates the behavior of LLM as a VLN agent. It clearly defines the VLN task and the basic reasoning format and rules for NavGPT at each navigation step. For example, NavGPT should move among the static viewpoints (positions) of a pre-defined graph of the environment by identifying the unique viewpoint ID. NavGPT should not fabricate nonexistent IDs.

Visual Foundation Models $\mathcal{F}$. NavGPT as an LLM agent requires visual perception and expression ability from VFMs to translate the current environment's visual observation into natural language description. The VFMs here play the role of translator, to translate visual observations using their own language, *e.g.* natural language, objects' bounding boxes, and objects' depth. Through the process of prompt management, the visual perception results will be reformated and translated into pure natural language for LLMs to understand.

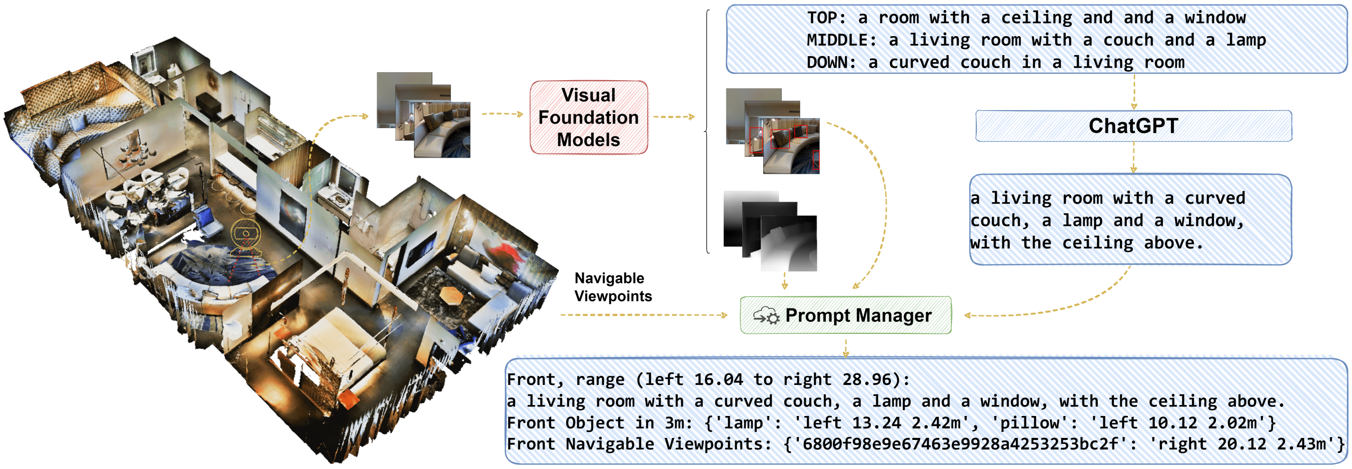
Navigation History $\mathcal{H}_{<t+1}$. The navigation history is essential for NavGPT to evaluate the progress of the completion of the instruction, to update the current state, and make the following decisions. The history is composed of summarized descriptions of previous observations $\mathcal{O}_{<t+1}$ and actions $\mathcal{A}_{<t+1}$, along with the reasoning thoughts $\mathcal{R}_{<t+1}$ from LLM.

Prompt Manager $\mathcal{M}$. The key to using LLM as a VLN agent is to convert all the above content into a natural language that LLM can understand. This process is done by the prompt manager, which collects the results from different components and parses them into a single prompt for LLM to make navigation decisions.

Visual Perceptron for NavGPT

In this section, we introduce the visual perception process of NavGPT. We take visual signals as a foreign language and handle the visual input using different visual foundation models to translate them into natural language, shown in figure 2.

For an agent standing at any viewpoint in the environment, the observation is composed of egocentric views from



```
TOP: a room with a ceiling and and a window
MIDDLE: a living room with a couch and a lamp
DOWN: a curved couch in a living room
```

```
a living room with a curved
couch, a lamp and a window,
with the ceiling above.
```

```
Front, range (left 16.04 to right 28.96):
a living room with a curved couch, a lamp and a window, with the ceiling above.
Front Object in 3m: {'lamp': 'left 13.24 2.42m', 'pillow': 'left 10.12 2.02m'}
Front Navigable Viewpoints: {'6800f98e9e67463e9928a4253253bc2f': 'right 20.12 2.43m'}
```

Figure 2: The process of forming natural language description from visual input. We used 8 directions to represent a viewpoint and show the process of forming the descriptions for one of the directions.

different orientations. The number of total views is defined by the field of view of each view image and the relative angle of each view. In our work, we set the field of view of each view as $45°$, and turn the heading angle θ $45°$ per view from $0°$ to $360°$, 8 directions in total. Besides, we turn the elevation angle ϕ $30°$ per view from $30°$ above the horizontal level to $30°$ below, 3 levels in total. As a result, we obtain $3 \times 8 = 24$ egocentric views for each viewpoint.

To translate visual observation into natural language, we first utilize the BLIP-2 (Li et al. 2023a) model as the translator. With the strong text generation capability of LLMs, BLIP-2 can achieve stunning zero-shot image-to-text generation quality. By carefully setting the granularity of visual observation (field of views and the total view number in each observation), we prompt BILP-2 to generate a decent language description of each view with a detailed depiction of the shapes and color of objects and the scenes they are in while avoiding useless caption of views from a smaller FoV, from which partial observation is available and it is hard to recognize even for humans. See appendix for details.

Notice that for the heading direction, the rotation interval is equal to the field of view, therefore there is no overlapping between each orientation. For the elevations, there is a $15°$'s overlapping between the top, middle, and down views. In NavGPT we mainly focus on the heading angle of agents during navigation, therefore, we prompt GPT-3.5 to summarize the scenes from the top, middle, and down views for each orientation into a sentence of description.

Besides natural language descriptions of the scene from BLIP-2, we also excavate the lower-level feature extracted by other vision models. These vision models serve as auxiliary translators, translating visual input into their own "language" like the class of objects and corresponding bounding boxes. The detection results will be aggregated by the prompt manager into prompts for LLMs. In this work, we utilize Fast-RCNN (Girshick 2015) to extract the bounding boxes of objects in each egocentric view. After locating the objects, we calculate the relative heading angle for each object and the agent. We also extract the depth information of the center pixel of the object provided by the Matterport3D simulator (Anderson et al. 2018). With the depth, objects' relative orientation, and class, we filter the detection results by leaving the object within 3 meters from the current viewpoint. The results from VFMs will be processed by the prompt manager into observation for the current viewpoint in natural language.

Synergizing Reasoning and Actions in LLMs

In the VLN task, the agent needs to learn the policy $\pi(a_t|\mathcal{W}, \mathcal{O}_t, \mathcal{O}_t^C, \mathcal{S}_t; \Theta)$, which is difficult because the implicit connection between actions and observations and demain intensive computation. In order to explicitly access and enhance the agent's comprehension of the current state during navigation, we follow the ReAct paper (Yao et al. 2022) to expand the agent's action space to $\tilde{\mathcal{A}} = \mathcal{A} \cup \mathcal{R}$, where $\mathcal{R} \in \mathcal{L}$ is in the entire language space $\mathcal{L}$, denoting the thought or reasoning trace of the agent.

The reasoning traces $\mathcal{R}$ of the agent will not trigger any interaction with the external environment, therefore no observation will be returned when the agent is outputting the reasoning during each navigation step. We synergize the NavGPT's actions and thoughts by prompting it to make navigation decisions after outputting the reasoning trace at each step. Introducing the reasoning traces aims to bootstrap the LLMs in two aspects:

Firstly, prompting the LLMs to think before choosing an action, enables LLMs to perform complex reasoning in planning and creating strategies to follow the instructions under the new observations. For example, as shown in figure 3, NavGPT can generate a long-term navigation plan by analyzing the current observation and the instruction, performing higher-level planning such as decomposing instruction and planning to reach the sub-goal, which is never seen explicitly in previous works.

Secondly, including reasoning traces $\mathcal{R}$ in the navigation history $\mathcal{H}_{<t}$ enhances the problem-solving ability of

NavGPT. By injecting reasoning traces into navigation history, NavGPT inherits from the previous reasoning traces, to reach a sub-goal with high-level planning consistently through steps, and can track the navigation progress with exception-handling abilities like adjusting the plan.

NavGPT Prompt Manager

With the Navigation System Principle $\mathcal{P}$, translated results from VFMs, and the History of Navigation $\mathcal{H}_{<t}$, the prompt manager parses and reformates them into prompts for LLMs. Details of the prompt are presented in the appendix.

Specifically, for Navigation System Principle $\mathcal{P}$, NavGPT prompt manager will create a prompt to convey LLMs with the rules, declaring the VLN task definition, defining the simulation environment for NavGPT, and restricting LLMs' behavior in the given reasoning format.

For perception results from VFMs $\mathcal{F}$, the prompt manager gathers the results from each direction and orders the language description by taking the current orientation of NavGPT as the front, shown in figure 2, arranging the description from 8 directions into prompt by concatenating them clockwise.

For navigation history $\mathcal{H}_{<t+1}$, the observation, reasoning, and actions triples $\langle \mathcal{O}_i, \mathcal{R}_i, \mathcal{A}_i \rangle$ are stored in a history buffer, shown in figure 1. Directly extracting all triples in the buffer will create too long a prompt for LLMs to accept. To handle the length of history, the prompt manager utilizes GPT-3.5 to summarize the observations from viewpoints in the trajectory, inserting the summarized observations into the observation, reasoning, and actions triples in the prompt.

Experiment

Implementation Details. We evaluate NavGPT based on GPT-4 (OpenAI 2023) and GPT-3.5 on the R2R dataset (Anderson et al. 2018). The R2R dataset is composed of 7189 trajectories, each corresponding to three fine-grained instructions. The dataset is separated into the train, val seen, val unseen, and test unseen splits, with 61, 56, 11, and 18 indoor scenes, respectively. We apply the 783 trajectories in the 11 val unseen environments in all our experiments and for comparison to previous supervised approaches. We utilize BLIP-2 ViT-G FlanT5$_{\text{XL}}$ (Li et al. 2023a) as images translator and Fast-RCNN (Girshick 2015) as object detector. The depth information of objects is extracted from the Mattport3D simulator (Anderson et al. 2018) by taking the depth of the center pixel in the bounding box.

Evaluation Metrics. The evaluation of NavGPT utilizes standardized metrics from the R2R dataset. These include Trajectory Length (TL), denoting the average distance traveled by the agent; Navigation Error (NE), representing the mean distance from the agent's final location to the destination; Success Rate (SR), indicating the proportion of navigation episodes where the agent successfully reaches the target location within a 3-meter margin of error; Oracle Success Rate (OSR), the success rate of agent stopped at the closest point to the goal on its trajectory; and Success Rate weighted by the normalized inverse of Path Length (SPL), which balances navigation precision and efficiency by adjusting the success rate based on the ratio of the optimal path length to the agent's predicted path length.

Qualitive Results

We elaborately study the qualitative results of the reason trace from NavGPT. We reveal the potential high-level planning capability of GPT-4 under embodied navigation tasks.

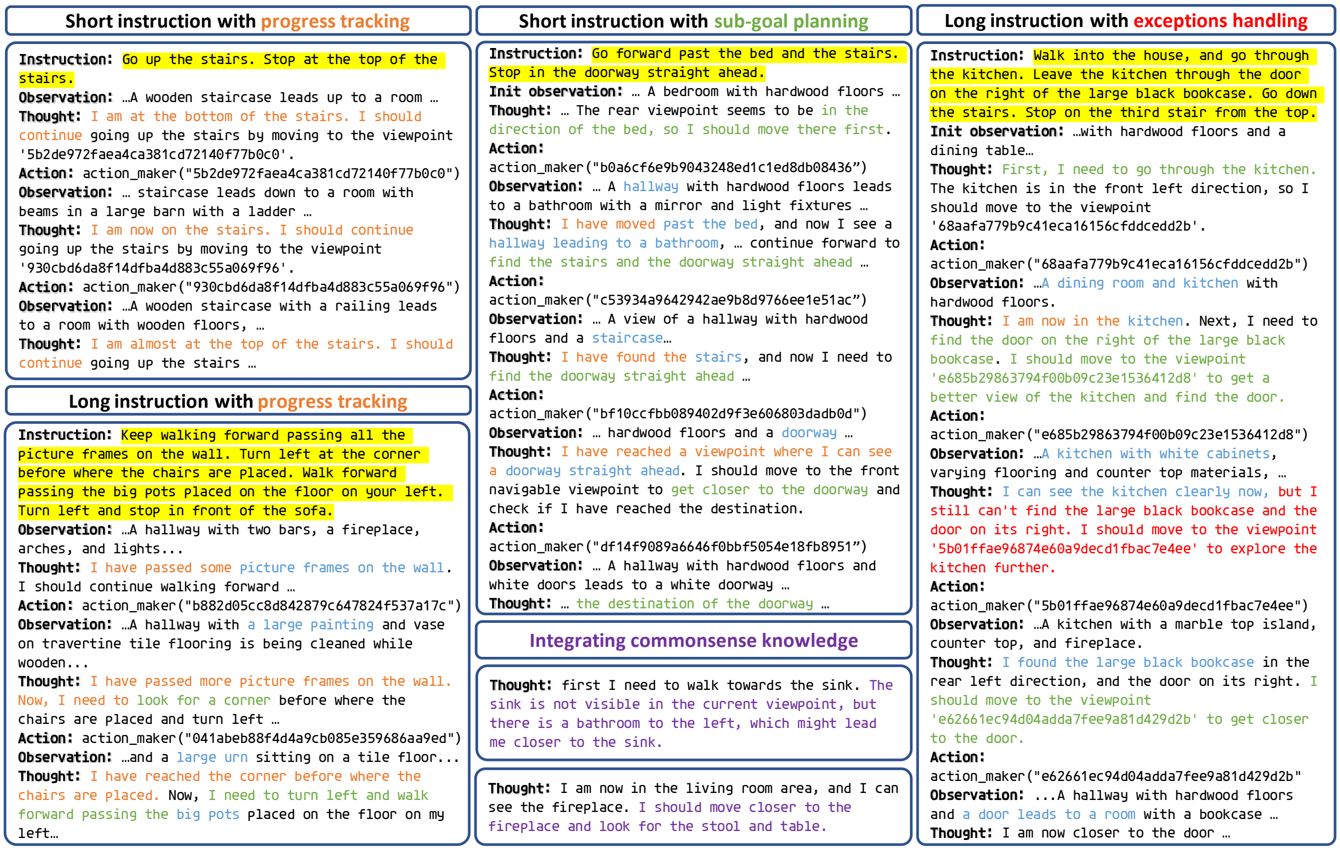
Reasoning Capability of GPT-4 for Language-guide Navigation As shown in figure 3, with GPT-4, NavGPT can perform various types of reasoning and high-level planning during navigation. For short instructions, NavGPT can track the navigation progress through steps to accomplish a single action described in the instructions, similar to the self-monitoring VLN agents (Ma et al. 2019; Zhu et al. 2020; Gao et al. 2023). For long instructions, NavGPT can break it down with sub-goals, similar to previous works on fine-graining R2R data (Hong et al. 2020b; He et al. 2021; Zhao et al. 2022), and plan to reach the destination by effectively identifying landmarks from observations, similar to works on utilizing objects information to perform cross-modality matching in VLN (Gao et al. 2021; Qi et al. 2020a, 2021). When navigating to a viewpoint with unexpected observation, NavGPT can plan to explore the environment and use commonsense knowledge to assist decision-making, similar to VLN methods incorporate external knowledge(Li et al. 2022b; Gao et al. 2021; Li et al. 2023b).

History and Spatial Relative Relation Awareness for LLMs During Navigation We examined NavGPT's awareness of historical and spatial relations by employing GPT-4 to delineate the trajectory in navigational history and to construct a map of visited viewpoints utilizing pyplot. The process involved extracting exclusively the actions $\mathcal{A}_{t+1}$, observations $\mathcal{O}_{t+1}$, and the entire navigation history $\mathcal{H}_{t+1}$. The specifics of the prompt are presented in the appendix.

As shown in figure 4, we observed that GPT-4 could effectively extract landmarks from the redundant observation descriptions and generate navigation history descriptions with actions. This could be a potential way of generating new trajectory instructions for VLN. Besides, the result shows GPT-4 can comprehensively understand the history of navigation, and thus can perform the essential progress tracking during navigation. Moreover, shown in figure 4, GPT-4 can successfully catch the relative position relations between viewpoints and draw a top-down view of the trajectory for visited viewpoints. By providing language descriptions of actions taken by the agents, including the turning angle and relative distances between viewpoints, GPT-4 shows a stunning awareness of spatial relations. Such impressive reasoning ability supports NavGPT to perform high-level planning shown in figure 3, underlines the significant potential LLMs hold for embodied navigation tasks.

Comparison with Supervised Methods

We compare the results of using NavGPT to zero-shot the sequential navigation tasks with previous models trained on the R2R dataset. As shown in table 1, NavGPT surpasses some supervised baselines and is compatible with some supervised agents. However, a significant discrepancy can be



Short instruction with progress tracking

Instruction: Go up the stairs. Stop at the top of the stairs.
Observation: …A wooden staircase leads up to a room …
Thought: I am at the bottom of the stairs. I should continue going up the stairs by moving to the viewpoint '5b2de972faea4ca381cd72140f77b0c0'.
Action: action_maker("5b2de972faea4ca381cd72140f77b0c0")
Observation: … staircase leads down to a room with beams in a large barn with a ladder …
Thought: I am now on the stairs. I should continue going up the stairs by moving to the viewpoint '930cbd6da8f14dfba4d883c55a069f96'.
Action: action_maker("930cbd6da8f14dfba4d883c55a069f96")
Observation: …A wooden staircase with a railing leads to a room with wooden floors, …
Thought: I am almost at the top of the stairs. I should continue going up the stairs …

Long instruction with progress tracking

Instruction: Keep walking forward passing all the picture frames on the wall. Turn left at the corner before where the chairs are placed. Walk forward passing the big pots placed on the floor on your left. Turn left and stop in front of the sofa.
Observation: …A hallway with two bars, a fireplace, arches, and lights...
Thought: I have passed some picture frames on the wall. I should continue walking forward …
Action: action_maker("b882d05cc8d842879c647824f537a17c")
Observation: …A hallway with a large painting and vase on travertine tile flooring is being cleaned while wooden...
Thought: I have passed more picture frames on the wall. Now, I need to look for a corner before where the chairs are placed and turn left …
Action: action_maker("041abeb88f4d4a9cb085e359686aa9ed")
Observation: …and a large urn sitting on a tile floor...
Thought: I have reached the corner before where the chairs are placed. Now, I need to turn left and walk forward passing the big pots placed on the floor on my left…

Short instruction with sub-goal planning

Instruction: Go forward past the bed and the stairs. Stop in the doorway straight ahead.
Init observation: … A bedroom with hardwood floors …
Thought: … The rear viewpoint seems to be in the direction of the bed, so I should move there first.
Action:
action_maker("b0a6cf6e9b9043248ed1c1ed8db08436”)
Observation: … A hallway with hardwood floors leads to a bathroom with a mirror and light fixtures …
Thought: I have moved past the bed, and now I see a hallway leading to a bathroom, … continue forward to find the stairs and the doorway straight ahead …
Action:
action_maker("c53934a9642942ae9b8d9766ee1e51ac”)
Observation: … A view of a hallway with hardwood floors and a staircase…
Thought: I have found the stairs, and now I need to find the doorway straight ahead …
Action:
action_maker("bf10ccfbb089402d9f3e606803dadb0d")
Observation: … hardwood floors and a doorway …
Thought: I have reached a viewpoint where I can see a doorway straight ahead. I should move to the front navigable viewpoint to get closer to the doorway and check if I have reached the destination.
Action:
action_maker("df14f9089a6646f0bbf5054e18fb8951”)
Observation: … A hallway with hardwood floors and white doors leads to a white doorway …
Thought: … the destination of the doorway …

Integrating commonsense knowledge

Thought: first I need to walk towards the sink. The sink is not visible in the current viewpoint, but there is a bathroom to the left, which might lead me closer to the sink.

Thought: I am now in the living room area, and I can see the fireplace. I should move closer to the fireplace and look for the stool and table.

Long instruction with exceptions handling

Instruction: Walk into the house, and go through the kitchen. Leave the kitchen through the door on the right of the large black bookcase. Go down the stairs. Stop on the third stair from the top.
Init observation: …with hardwood floors and a dining table…
Thought: First, I need to go through the kitchen. The kitchen is in the front left direction, so I should move to the viewpoint '68aafa779b9c41eca16156cfddcedd2b'.
Action:
action_maker("68aafa779b9c41eca16156cfddcedd2b")
Observation: …A dining room and kitchen with hardwood floors.
Thought: I am now in the kitchen. Next, I need to find the door on the right of the large black bookcase. I should move to the viewpoint 'e685b29863794f00b09c23e1536412d8' to get a better view of the kitchen and find the door.
Action:
action_maker("e685b29863794f00b09c23e1536412d8")
Observation: …A kitchen with white cabinets, varying flooring and counter top materials, …
Thought: I can see the kitchen clearly now, but I still can't find the large black bookcase and the door on its right. I should move to the viewpoint '5b01ffae96874e60a9decd1fbac7e4ee' to explore the kitchen further.
Action:
action_maker("5b01ffae96874e60a9decd1fbac7e4ee")
Observation: …A kitchen with a marble top island, counter top, and fireplace.
Thought: I found the large black bookcase in the rear left direction, and the door on its right. I should move to the viewpoint 'e62661ec94d04adda7fee9a81d429d2b' to get closer to the door.
Action:
action_maker("e62661ec94d04adda7fee9a81d429d2b"
Observation: ...A hallway with hardwood floors and a door leads to a room with a bookcase …
Thought: I am now closer to the door …

Figure 3: The qualitative of NavGPT. NavGPT can explicitly perform high-level planning for sequential action prediction, including decomposing instruction into sub-goals, integrating commonsense knowledge, identifying landmarks from observed scenes, tracking navigation progress, exceptions handling with plan adjustment.

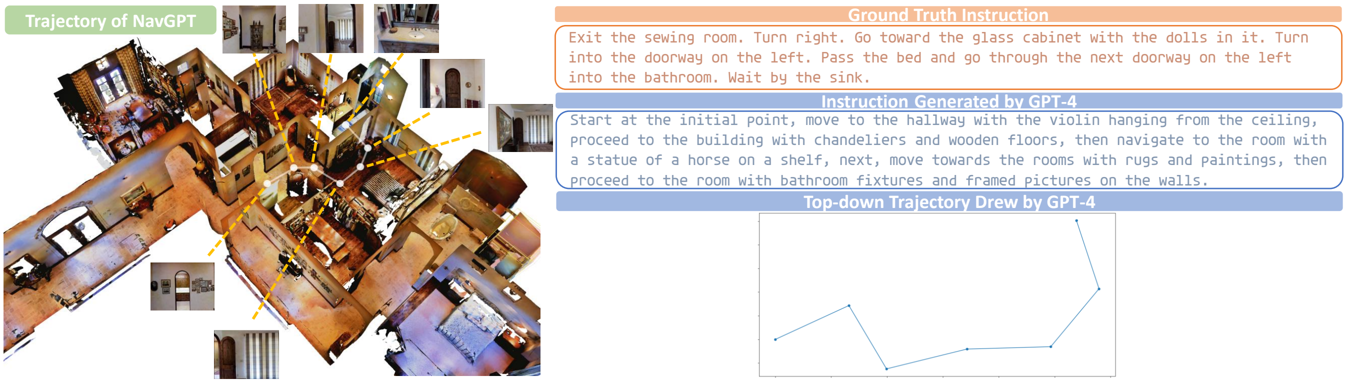
discerned. We suggest the limitations inhibiting the performance of LLMs in solving VLN can be primarily attributed to two factors: the precision of language-based depiction of visual scenes and the tracking capabilities regarding objects.

NavGPT's functionality is heavily reliant on the quality of captions generated from VFMs. If the target object delineated in the instruction is absent in the observation description, NavGPT is compelled to explore the environment. The ideal circumstance entails all target objects being visible pursuant to the instruction. However, the inherent granularity of language description inevitably incurs a loss of information. Moreover, NavGPT must manage the length of the navigation history to prevent excessively verbose descriptions as the steps accrue. To this end, a summarizer is implemented, albeit at the cost of further information loss. This diminishes NavGPT's tracking ability, impeding the formation of seamless perceptions of the entire environment as the trajectory lengthens.

Effect of Visual Components

We perform additional experiments to investigate the effectiveness of visual components in NavGPT, we construct a baseline with GPT-3.5 for its easier access and budget-friendly costs. To evaluate the zero-shot ability in various environments, we construct a new validation split sampling both from the original training set and the validation unseen set. The scenes from the training and validation unseen set are 61 and 11 respectively, 72 scenes in total. We randomly picked 1 trajectory from the 72 environment, each is associated with 3 instructions. In total, we sample 216 samples to conduct the ablation study.

Effect of Granularity in Visual Observation Descriptions. The Field of View (FoV) of an image critically influences BILP-2's captioning ability, with an overly large FoV leading to generalized room descriptions and an extremely small FoV hindering object recognition due to limited content. As shown in table 2, we investigate 3 granularity of visual representation from a viewpoint. Specifically, variant #1 utilizes an image with 60 FoV, turn heading angle 30 degrees clock-wise to obtain 12 views from a viewpoint, while variant #2 and #3 utilize an image with 30, 45 FoV, turn elevation angle 30 degrees from top to down, and turn heading angle 30, 45 degrees clockwise to form 36 views, 24 views respectively. From the results, we found that using FoV 45 a viewpoint generates the most suitable natural language description for navigation, surpassing variant #1 and #2 by 6.48% and 2.78% respectively.


Trajectory of NavGPT
Ground Truth Instruction
Instruction Generated by GPT-4
Top-down Trajectory Drew by GPT-4

Exit the sewing room. Turn right. Go toward the glass cabinet with the dolls in it. Turn into the doorway on the left. Pass the bed and go through the next doorway on the left into the bathroom. Wait by the sink.

Start at the initial point, move to the hallway with the violin hanging from the ceiling, proceed to the building with chandeliers and wooden floors, then navigate to the room with a statue of a horse on a shelf, next, move towards the rooms with rugs and paintings, then proceed to the room with bathroom fixtures and framed pictures on the walls.

Figure 4: We evaluate GPT-4 on a case where NavGPT successfully follows the ground truth path, using only the historical actions $\mathcal{A}_{<t+1}$ and observations $\mathcal{O}_{<t+1}$ to generate an instruction (without reasoning trace $\mathcal{R}_{<t+1}$ to avoid information leaking), and using the entire navigation history $\mathcal{H}_{<t+1}$ to draw a top-down trajectory.

Training Schema	Method	TL	NE↓	OSR↑	SR↑	SPL↑
Train Only	Seq2Seq (Anderson et al. 2018)	8.39	7.81	28	21	-
	Speaker Follower (Fried et al. 2018)	-	6.62	45	35	-
	EnvDrop (Tan, Yu, and Bansal 2019)	10.70	5.22	-	52	48
Pretrain + Finetune	PREVALENT (Hao et al. 2020)	10.19	4.71	-	58	53
	VLN↻BERT (Hong et al. 2021)	12.01	3.93	69	63	57
	HAMT (Chen et al. 2021b)	11.46	2.29	73	66	61
	DuET (Chen et al. 2022b)	13.94	3.31	81	72	60
No Train	DuET (Init. LXMERT (Tan and Bansal 2019))	22.03	9.74	7	1	0
	NavGPT (Ours)	11.45	6.46	42	34	29

Table 1: Comparison with previous methods on R2R validation unseen split.

Granularity	#	TL	NE↓	OSR↑	SR↑	SPL↑
FoV@60	1	12.38	9.07	14.35	10.19	6.52
FoV@30	2	12.67	8.92	15.28	13.89	9.12
FoV@45	3	12.18	8.02	26.39	16.67	13.00

Table 2: The effect of granularity in visual observation.

Observation	#	TL	NE↓	OSR↑	SR↑	SPL↑
Baseline	1	16.11	9.83	15.28	11.11	6.92
+ Obj	2	11.07	8.88	23.34	15.97	11.71
+ Obj + Dis	3	12.18	8.02	26.39	16.67	13.00

Table 3: The effect of additional information.

Effect of Semantic Scene Understanding and Depth Estimation. NavGPT also collaborates with other visual foundation models to enhance the perception of the environment. We investigate the effectiveness of adding the object information and the relative distance between the agent and the detected objects. We constructed a baseline method based on the caption results from BILP-2 and powered by GPT-3.5. As shown in table 3, by adding object information, the SR increases by 4.86% compared with the baseline, for the additional object information emphasizes the salient object in the scenes. Moreover, we observed a phenomenon in which agents failed to reach the destination because they do not know how close they are to the destination. Once the target viewpoint is visible in sight, they tend to stop immediately. Therefore by adding depth information, the agent has a better understanding of the current position and further raises the SR by 0.7% and SPL by 1.29%.

Conclusion

In this work, we explore the potential of utilizing LLMs in embodied navigation tasks. We present NavGPT, an autonomous LLM system specifically engineered for language-guided navigation, possessing the ability to process multi-modal inputs and unrestricted language guidance, engage with open-world environments, and maintain the navigation history. Limited by the quality of language description of visual scenes and the tracking abilities of objects, NavGPT's zero-shot performance on VLN is still not compatible with trained methods. However, the reasoning trace of GPT-4 illuminates the latent potential of LLMs in embodied navigation planning. Interaction of LLMs with downstream specialized models or the development of multi-modal LLMs for navigation, heralding the future of versatile VLN agents.

References

Ahn, M.; Brohan, A.; Brown, N.; Chebotar, Y.; et al. 2022. Do As I Can and Not As I Say: Grounding Language in Robotic Affordances. In *arXiv preprint arXiv:2204.01691*.

Anderson, P.; Wu, Q.; Teney, D.; Bruce, J.; Johnson, M.; Sünderhauf, N.; Reid, I.; Gould, S.; and Van Den Hengel, A. 2018. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *CVPR*.

Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; et al. 2020. Language models are few-shot learners. In *NeurIPS*.

Bubeck, S.; Chandrasekaran, V.; Eldan, R.; Gehrke, J.; et al. 2023. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*.

Chen, K.; Chen, J. K.; Chuang, J.; Vázquez, M.; and Savarese, S. 2021a. Topological planning with transformers for vision-and-language navigation. In *CVPR*.

Chen, S.; Guhur, P.-L.; Schmid, C.; and Laptev, I. 2021b. History aware multimodal transformer for vision-and-language navigation. In *NeurIPS*.

Chen, S.; Guhur, P.-L.; Tapaswi, M.; Schmid, C.; et al. 2022a. Learning from unlabeled 3d environments for vision-and-language navigation. In *ECCV*.

Chen, S.; Guhur, P.-L.; Tapaswi, M.; Schmid, C.; et al. 2022b. Think Global, Act Local: Dual-scale Graph Transformer for Vision-and-Language Navigation. In *CVPR*.

Chowdhery, A.; Narang, S.; Devlin, J.; Bosma, M.; et al. 2022. PaLM: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*.

Deng, Z.; Narasimhan, K.; and Russakovsky, O. 2020. Evolving graphical planner: Contextual global planning for vision-and-language navigation. In *NeurIPS*.

Dorbala, V. S.; Mullen Jr, J. F.; and Manocha, D. 2023. Can an Embodied Agent Find Your" Cat-shaped Mug"? LLM-Based Zero-Shot Object Navigation. *arXiv preprint arXiv:2303.03480*.

Dorbala, V. S.; Sigurdsson, G.; Piramuthu, R.; et al. 2022. CLIP-Nav: Using CLIP for Zero-Shot Vision-and-Language Navigation. *arXiv preprint arXiv:2211.16649*.

Driess, D.; Xia, F.; Sajjadi, M. S.; Lynch, C.; et al. 2023. PaLM-E: An Embodied Multimodal Language Model. *arXiv preprint arXiv:2303.03378*.

Fried, D.; Hu, R.; Cirik, V.; Rohrbach, A.; et al. 2018. Speaker-follower models for vision-and-language navigation. In *NeurIPS*.

Gadre, S. Y.; Wortsman, M.; Ilharco, G.; Schmidt, L.; and Song, S. 2022. CLIP on Wheels: Open-Vocabulary Models are (Almost) Zero-Shot Object Navigators. *arXiv preprint arXiv:2203.10421v1*.

Gao, C.; Chen, J.; Liu, S.; Wang, L.; Zhang, Q.; and Wu, Q. 2021. Room-and-object aware knowledge reasoning for remote embodied referring expression. In *CVPR*.

Gao, C.; Peng, X.; Yan, M.; Wang, H.; et al. 2023. Adaptive Zone-Aware Hierarchical Planner for Vision-Language Navigation. In *CVPR*.

Girshick, R. 2015. Fast R-CNN. In *ICCV*.

Gu, J.; Stefani, E.; Wu, Q.; Thomason, J.; and Wang, X. 2022. Vision-and-Language Navigation: A Survey of Tasks, Methods, and Future Directions. In *ACL*.

Hao, W.; Li, C.; Li, X.; Carin, L.; and Gao, J. 2020. Towards learning a generic agent for vision-and-language navigation via pre-training. In *CVPR*.

He, K.; Huang, Y.; Wu, Q.; Yang, J.; et al. 2021. Landmark-RxR: Solving Vision-and-Language Navigation with Fine-Grained Alignment Supervision. In *NeurIPS*.

Hong, Y.; Rodriguez, C.; Qi, Y.; Wu, Q.; and Gould, S. 2020a. Language and visual entity relationship graph for agent navigation. In *NeurIPS*.

Hong, Y.; Rodriguez-Opazo, C.; Wu, Q.; and Gould, S. 2020b. Sub-Instruction Aware Vision-and-Language Navigation. In *NeurIPS*.

Hong, Y.; Wang, Z.; Wu, Q.; and Gould, S. 2022. Bridging the Gap Between Learning in Discrete and Continuous Environments for Vision-and-Language Navigation. In *CVPR*.

Hong, Y.; Wu, Q.; Qi, Y.; Rodriguez-Opazo, C.; and Gould, S. 2021. VLN↻BERT: A recurrent vision-and-language bert for navigation. In *CVPR*.

Hong, Y.; Zhou, Y.; Zhang, R.; Dernoncourt, F.; Bui, T.; Gould, S.; and Tan, H. 2023. Learning navigational visual representations with semantic map supervision. In *ICCV*.

Huang, C.; Mees, O.; Zeng, A.; and Burgard, W. 2022. Visual Language Maps for Robot Navigation. *arXiv preprint arXiv:2210.05714*.

Karpas, E.; Abend, O.; Belinkov, Y.; Lenz, B.; et al. 2022. MRKL Systems: A modular, neuro-symbolic architecture that combines large language models, external knowledge sources and discrete reasoning. *arXiv preprint arXiv:2205.00445*.

Krantz, J.; Wijmans, E.; Majumdar, A.; Batra, D.; and Lee, S. 2020. Beyond the nav-graph: Vision-and-language navigation in continuous environments. In *ECCV*.

Ku, A.; Anderson, P.; Patel, R.; Ie, E.; et al. 2020. Room-Across-Room: Multilingual Vision-and-Language Navigation with Dense Spatiotemporal Grounding. In *EMNLP*.

Li, J.; and Bansal, M. 2023. PanoGen: Text-Conditioned Panoramic Environment Generation for Vision-and-Language Navigation. In *NeurIPS*.

Li, J.; Li, D.; Savarese, S.; and Hoi, S. 2023a. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*.

Li, J.; Tan, H.; and Bansal, M. 2022. EnvEdit: Environment Editing for Vision-and-Language Navigation. In *CVPR*.

Li, L. H.; Zhang, P.; Zhang, H.; Yang, J.; Li, C.; Zhong, Y.; Wang, L.; Yuan, L.; Zhang, L.; Hwang, J.-N.; et al. 2022a. Grounded language-image pre-training. In *CVPR*.

Li, M.; Wang, Z.; Tuytelaars, T.; and Moens, M.-F. 2023b. Layout-aware Dreamer for Embodied Referring Expression Grounding. In *AAAI*.

Li, X.; Zhang, Y.; Yuan, W.; et al. 2022b. Incorporating External Knowledge Reasoning for Vision-and-Language Navigation with Assistant's Help. *Applied Sciences*.

Ma, C.-Y.; Lu, J.; Wu, Z.; AlRegib, G.; et al. 2019. Self-monitoring navigation agent via auxiliary progress estimation. *arXiv preprint arXiv:1901.03035*.

Majumdar, A.; Aggarwal, G.; Devnani, B.; Hoffman, J.; and Batra, D. 2022. Zson: Zero-shot object-goal navigation using multimodal goal embeddings. *arXiv preprint arXiv:2206.12403*.

OpenAI. 2023. GPT-4 Technical Report. arXiv:2303.08774.

Parvaneh, A.; Abbasnejad, E.; Teney, D.; Shi, J. Q.; et al. 2020. Counterfactual vision-and-language navigation: Unravelling the unseen. In *NeurIPS*.

Pashevich, A.; Schmid, C.; and Sun, C. 2021. Episodic transformer for vision-and-language navigation. In *ICCV*.

Peng, B.; Galley, M.; He, P.; Cheng, H.; et al. 2023. Check your facts and try again: Improving large language models with external knowledge and automated feedback. *arXiv preprint arXiv:2302.12813*.

Qi, Y.; Pan, Z.; Hong, Y.; Yang, M.-H.; van den Hengel, A.; and Wu, Q. 2021. The road to know-where: An object-and-room informed sequential bert for indoor vision-language navigation. In *ICCV*.

Qi, Y.; Pan, Z.; Zhang, S.; Hengel, A. v. d.; and Wu, Q. 2020a. Object-and-action aware model for visual language navigation. In *ECCV*.

Qi, Y.; Wu, Q.; Anderson, P.; Wang, X.; et al. 2020b. Reverie: Remote embodied visual referring expression in real indoor environments. In *CVPR*.

Qiao, Y.; Qi, Y.; Hong, Y.; Yu, Z.; Wang, P.; and Wu, Q. 2023a. HOP+: History-enhanced and Order-aware Pre-training for Vision-and-Language Navigation. *IEEE TPAMI*.

Qiao, Y.; Qi, Y.; Yu, Z.; Liu, J.; and Wu, Q. 2023b. March in chat: Interactive prompting for remote embodied referring expression. In *ICCV*.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; et al. 2021. Learning transferable visual models from natural language supervision. In *ICML*.

Scao, T. L.; Wang, T.; Hesslow, D.; Saulnier, L.; et al. 2022. What Language Model to Train if You Have One Million GPU Hours? *arXiv preprint arXiv:2210.15424*.

Schick, T.; Dwivedi-Yu, J.; Dessì, R.; Raileanu, R.; et al. 2023. Toolformer: Language models can teach themselves to use tools. *arXiv preprint arXiv:2302.04761*.

Shah, D.; Osiński, B.; Levine, S.; et al. 2023. Lm-nav: Robotic navigation with large pre-trained models of language, vision, and action. In *CoRL*.

Tan, H.; and Bansal, M. 2019. LXMERT: Learning Cross-Modality Encoder Representations from Transformers. In *EMNLP*.

Tan, H.; Yu, L.; and Bansal, M. 2019. Learning to Navigate Unseen Environments: Back Translation with Environmental Dropout. In *NAACL*.

Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Vemprala, S.; Bonatti, R.; Bucker, A.; and Kapoor, A. 2023. Chatgpt for robotics: Design principles and model abilities. *arXiv preprint arXiv:2306.17582*.

Wang, S.; Montgomery, C.; Orbay, J.; Birodkar, V.; et al. 2022. Less is More: Generating Grounded Navigation Instructions from Landmarks. In *CVPR*.

Wang, X.; Huang, Q.; Celikyilmaz, A.; Gao, J.; et al. 2019. Reinforced cross-modal matching and self-supervised imitation learning for vision-language navigation. In *CVPR*.

Wang, Z.; Li, J.; Hong, Y.; et al. 2023. Scaling data generation in vision-and-language navigation. In *ICCV*.

Wei, J.; Bosma, M.; Zhao, V. Y.; Guu, K.; Yu, A. W.; Lester, B.; Du, N.; Dai, A. M.; and Le, Q. V. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.

Wei, J.; Tay, Y.; Bommasani, R.; Raffel, C.; Zoph, B.; Borgeaud, S.; Yogatama, D.; Bosma, M.; Zhou, D.; Metzler, D.; et al. 2022a. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Chi, E.; Le, Q.; and Zhou, D. 2022b. Chain of thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*.

Wu, C.; Yin, S.; Qi, W.; Wang, X.; Tang, Z.; and Duan, N. 2023. Visual chatgpt: Talking, drawing and editing with visual foundation models. *arXiv preprint arXiv:2303.04671*.

Yao, S.; Zhao, J.; Yu, D.; Du, N.; Shafran, I.; Narasimhan, K.; and Cao, Y. 2022. React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*.

Yongliang, S.; Kaitao, S.; Xu, T.; Dongsheng, L.; et al. 2023. HuggingGPT: Solving AI Tasks with ChatGPT and its Friends in HuggingFace. arXiv:2303.17580.

Zhang, S.; Roller, S.; Goyal, N.; Artetxe, M.; Chen, M.; Chen, S.; Dewan, C.; Diab, M.; Li, X.; Lin, X. V.; et al. 2022. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.

Zhao, C.; Qi, Y.; and Wu, Q. 2023. Mind the Gap: Improving Success Rate of Vision-and-Language Navigation by Revisiting Oracle Success Routes. In *ACM MM*.

Zhao, Y.; Chen, J.; Gao, C.; Wang, W.; Yang, L.; Ren, H.; Xia, H.; and Liu, S. 2022. Target-Driven Structured Transformer Planner for Vision-Language Navigation. In *ACM MM*.

Zhou, K.; Zheng, K.; Pryor, C.; Shen, Y.; Jin, H.; Getoor, L.; and Wang, X. E. 2023. ESC: Exploration with Soft Commonsense Constraints for Zero-shot Object Navigation. *arXiv preprint arXiv:2301.13166*.

Zhu, F.; Zhu, Y.; Chang, X.; and Liang, X. 2020. Vision-language navigation with self-supervised auxiliary reasoning tasks. In *CVPR*.

Zhu, W.; Qi, Y.; Narayana, P.; Sone, K.; Basu, S.; Wang, X. E.; Wu, Q.; Eckstein, M. P.; and Wang, W. Y. 2022. Diagnosing Vision-and-Language Navigation: What Really Matters. In *NAACL*.