NavQ: Learning a Q-Model for Foresighted Vision-and-Language Navigation

Peiran Xu   Xicheng Gong   Yadong Mu*
Peking University
Beijing, China
xpr820@pku.edu.cn gongxicheng@stu.pku.edu.cn myd@pku.edu.cn

Abstract

In this work we concentrate on the task of goal-oriented Vision-and-Language Navigation (VLN). Existing methods often make decisions based on historical information, overlooking the future implications and long-term outcomes of the actions. In contrast, we aim to develop a foresighted agent. Specifically, we draw upon Q-learning to train a Q-model using large-scale unlabeled trajectory data, in order to learn the general knowledge regarding the layout and object relations within indoor scenes. This model can generate a Q-feature, analogous to the Q-value in traditional Q-network, for each candidate action, which describes the potential future information that may be observed after taking the specific action. Subsequently, a cross-modal future encoder integrates the task-agnostic Q-feature with navigation instructions to produce a set of action scores reflecting future prospects. These scores, when combined with the original scores based on history, facilitate an A-style searching strategy to effectively explore the regions that are more likely to lead to the destination. Extensive experiments conducted on widely used goal-oriented VLN datasets validate the effectiveness of the proposed method.*

1. Introduction

The task of Vision-and-Language Navigation (VLN) requires an agent to reach the target location in a photo-realistic environment following language instructions. As a crucial step towards embodied intelligence, this topic has recently attracted significant attention, and many related benchmarks has been published [3, 48, 54, 96, 107, 143]. In particular, REVERIE [96] concentrates on goal-oriented VLN, in which the instruction contains only the description of the target object, instead of step-by-step guidance. This setup is well-suited for the development of practical home assistants, where humans only need to provide intent-level cues rather than detailed navigation steps.

From a high-level perspective, goal-oriented VLN can be viewed as a searching problem in the scene. Despite significant progress, existing methods [13, 14, 78, 80, 117] often rely solely on the information from the visited areas to make a single-step decision, without considering the potential consequences of the action. As suggested by the A* algorithm [33], integrating a heuristic metric that evaluates the future outcome when selecting frontiers to explore may greatly improve the efficiency of searching. Thus, we hope to devise a foresighted navigation agent that explicitly reason about the future prospects, in addition to the observation along the partial trajectory. A motivating example is illustrated in Figure 1.

Currently, several research efforts have already incorporated future information into the decision-making process, and most of them focus on predicting single-step outcomes [19, 58, 125, 137]. By leveraging the overlapping fields of view between adjacent viewpoints, these methods can predict the scenario of the area reached after an action is taken. However, they focus on imagining the visual observations of neighboring nodes and only consider local hints, failing to capture long-range, semantic-level future information. On the other hand, [51, 115, 119] learn a world model to predict future information in a more principled way. Though these methods can anticipate future states for any number of steps ahead, they require multi-rounds, multi-steps expansions through the world model for each decision. This rollout process is highly time-consuming and prone to distortions and overfitting, particularly when predictions are made in the RGB space [51, 115].

To address this dilemma between the horizon and efficiency, we propose NavQ, an agent that predicts the long-horizon future information within a single forward pass. At its core is a Q-model capable of anticipating the aggregated future outcomes in the latent space. Traditionally, Q-learning will formulate a Q-function that evaluates the cumulative reward value of a state-action pair. Here, our Q-model instead outputs a Q-feature, which encapsulates the cumulation of future observations following the execution of an action. Free from reward computation, our Q-model can be pre-trained on abundant unsupervised trajectory data to enhance generalizability. Following the Q-model, an ad-

*Corresponding author.

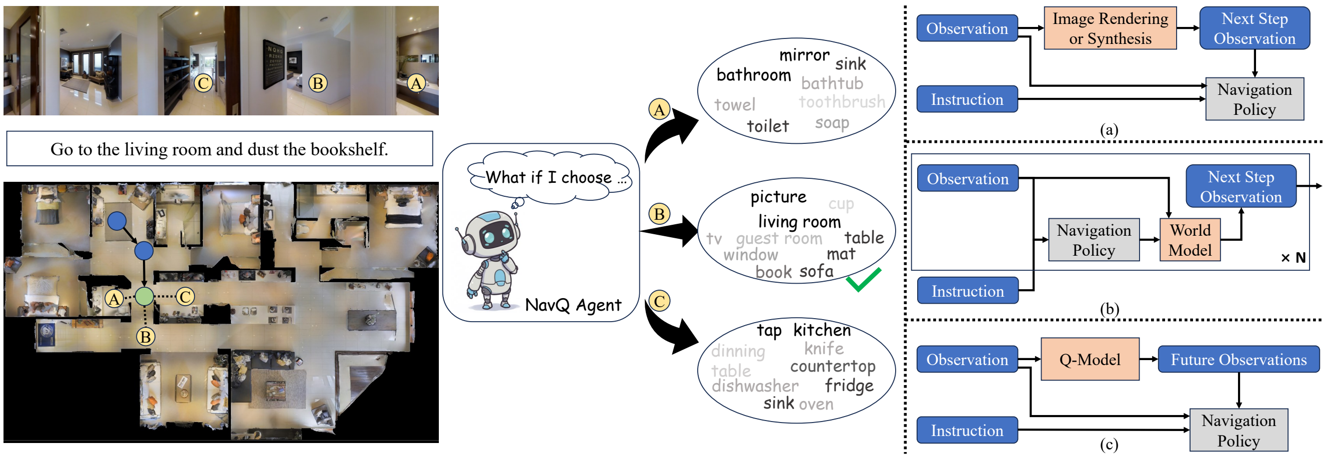

Figure 1. Left: the motivation of the proposed method. We generate cumulative Q-feature for each candidate action, which represents the future outcomes of choosing the action and enables foresighted navigation decisions. Right: a high-level comparison among the decision making processes of (a) methods based on imagining neighborhood observations, (b) methods based on a world model, and (c) our proposed method. Our Q-model is capable of forecasting the long-horizon future without time-consuming rollouts.

ditional cross-modal encoder is introduced to interact the Q-feature of each possible action with the text instruction, producing future-sensitive scores to complement the original decision making process based on history and current observation.

To sum up, our contributions are as follows:

- We devise a Q-model that learns to predict long-range future semantics in the form of an aggregated Q-feature. We put forward a self-supervised learning pipeline to train this model on randomly-sampled trajectories without instruction annotation.
- We build a cross-modal future encoder that translates the Q-feature into goal-oriented heuristics. Integrating this module into a baseline model, we achieve an A*-inspired agent that makes a balance between historical progress and future prospects.
- Extensive experiments are conducted to demonstrate the effectiveness of the proposed method.

2. Related Works

2.1. Vision-and-Language Navigation

Since its introduction in [3], vision-and-language navigation [34, 48, 54, 56, 96, 143] has received significant attention in recent years. Existing works address this task through various approaches, including (1) the exploration of different learning strategies such as imitation learning [63], reinforcement learning [22, 111, 119, 120], adversarial training [26, 68, 134], generative modeling [18], curriculum learning [131], cycle-consistent learning [114], and energy-based optimization [79]; (2) the design of offline pretraining [20, 31, 32, 42, 45, 75, 84, 98], auxiliary tasks [62, 66, 82, 121, 142], and regularizations [93, 117, 122] for a more stable and less biased training process; (3) the development of more informative history representations and scene representations [2, 4, 10, 12, 13, 21, 41, 73, 78, 80, 112, 116, 124]; (4) the design of action space for efficient exploration and backtracking [14, 28, 47, 50, 83, 110]; (5) the extraction of finer-grained visual [43, 46, 71, 88, 97, 136] and textual features [1, 16, 39, 40, 59, 69, 95, 135, 144] or the incorporation of external knowledge from large language models (LLM) [11, 72, 81, 92, 99, 101, 132, 139–141], vision-language models (VLM) [62], and knowledge bases [27, 64, 87]; (6) the implementation of data augmentation techniques, including observation perturbation [36, 52, 61, 70], automatic trajectory annotation [23, 25, 44, 53, 65, 91, 106, 118, 126] and creating new scenes [15, 49, 57, 74, 77, 123]; and (7) the introduction of diverse related tasks [6, 17, 89, 90, 104, 107, 113, 145] and practical settings [5, 29, 38, 55, 100].

In particular, a line of works focusing on leveraging future information offers inspirations for our method. Existing attempts can be roughly classified into three paradigms. (1) Some of them [51, 115, 119] train a generative world model that outputs the next observation given current observation and an action. With this model, candidate actions can be mentally expanded for multiple steps (using beam search or MCTS), and the consistency of the resulting paths with the text instruction is used to evaluate the corresponding action. (2) Other works employ future-related information to augment the visual features. [19, 58, 125] leverage various techniques like dVAE, volume rendering, NeRF, or diffusion to synthesize the resulting observation of an action. Upon the synthesized images, [137] further consults a VLM to reason about them. (3) Also, there is a series of attempts [30, 67, 103, 130, 133, 138], mainly in Object Navigation (ObjNav), working on completing the unobserved area or predicting a possible sub-goal in a top-down map. Different from the works above, our method directly predicts the Q-feature of each candidate action, thus it does not involve the time-consuming step-by-step rollout of a world model (in contrast to (1)) nor the explicit construction of a metric map (in contrast to (3)). On the other hand, we focus on the long-horizon, high-level, heuristic future semantics rather than he immediate, localized, reconstruction-based neighborhood information (in contrast to (2)).

2.2. Q-Learning and Q* Agent

As a classic algorithm in reinforcement learning (RL), Q-learning [127] and its deep learning-based variants [37, 85, 108] have achieved breakthroughs in game playing and beyond. Later, there has been a growing body of research exploring the integration of Q-learning with the powerful representational capabilities of Transformers [9, 24, 129], leading to notable advancements in the field of embodied intelligence. More recently, the concept of the Q* algorithm has garnered remarkable attention, especially in the realm of LLM-based reasoning and planning. [109] combines Q-learning with A* search [33] to improve the multi-step reasoning capability of the LLM. It proposes to learn a Q-value model on sampled reasoning trajectories, and the output of this model is added with a process-based reward to determine the best action at each step. [94] and [76] also estimate the Q-value of the agent's actions, which then serves as feedbacks and enables the self-improvement of LLM. In this work, we also aim to employ a combination of Q-learning and A* search. However, instead of building a general inference pipeline for LLMs, we design a grounded agent in the specific context of navigation. We borrow the idea of A* to implement a foresighted embodied agent, while leveraging Q-learning to efficiently equip the agent with knowledge on future outcomes.

An ObjNav method VLV [8] also involves Q-learning for navigation. It learns a value function from YouTube videos that outputs the Q-value for an image-action pair, representing the closeness to certain object classes. It should be noted that this method cannot be trivially adapted to our task, as the target in VLN is not specified by a closed-set object category. Instead, by advancing from Q-value to Q-feature, we manage to capture general-purpose, target-agnostic, future-centric knowledge from unlabeled paths. Thus, the model design and training process of our Q-model diverge significantly from that of VLV.

3. Method

3.1. Task Setting and Base Model

The target of goal-oriented VLN, or Remote Embodied Visual referring Expression, is to navigate to an object referred by text instruction. The reachable places in the scene are abstracted as a graph. At each time step, the agent perceives a panoramic image at its current node, and selects a neighboring node as its action. The panoramic observation is usually divided into 36 discrete views. We use DUET [14] as the base agent. As shown in Figure 2, it maintains a graph of the visited nodes and candidate nodes (the nodes that have been observed but not visited) during the navigation process. When determining actions, it interacts the textual feature with the coarse-grained node features on the graph and the fine-grained view features around the current position, using a global encoder (GE) and a local encoder (LE), respectively. The resulting dual-scale features are fused together to predict the action scores for all the candidate nodes on the graph. Formally, the major computation process of DUET can be summarized as:

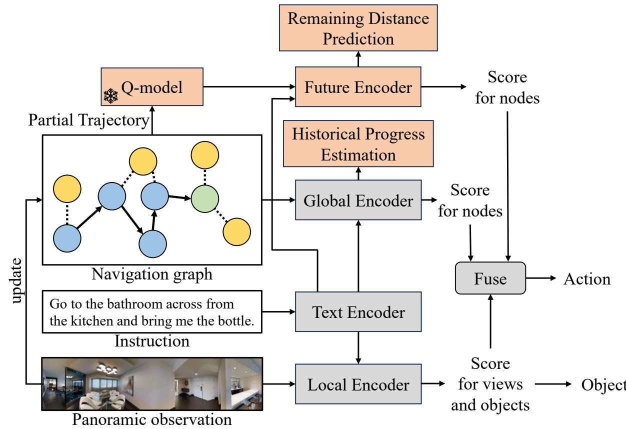

Figure 2. An overview of the proposed model. The gray modules are inherited from the baseline model [14], while the orange ones are introduced by this work.

$$G^t = \text{Update}(\{r_i^t\}_{i=1}^N, G^{t-1}), \quad (1)$$

$$\{\hat{v}_i^t\}_{i=0}^{|G^t|} = \text{GE}(G^t, w), \quad (2)$$

$$\{\hat{r}_i^t\}_{i=1}^N, \{\hat{o}_i^t\}_{i=1}^{M^t} = \text{LE}(\{r_i^t\}_{i=1}^N, \{o_i^t\}_{i=1}^{M^t}, w), \quad (3)$$

$$p^{a,t} = \text{Fuse}(\{\hat{v}_i^t\}_{i=0}^{|G^t|}, \{\hat{r}_i\}_{i=1}^N), \quad (4)$$

$$p^{o,t} = \text{Pred\_Obj}(\{\hat{o}_i^t\}_{i=1}^{M^t}). \quad (5)$$

At timestep t, $\{r_i^t\}$ are the image features of the $N = 36$ views at current location, G^t is the maintained graph, w is the feature of the text instruction, $\{o_i^t\}$ are the features of M^t possible objects at current location. The output $p^{a,t}$ and $p^{o,t}$ are probability distributions over the candidate nodes and the possible objects, respectively.

3.2. Overview

Since the action scores produced by DUET are purely based on history information in the explored area, we propose to introduce an additional future-related branch into the pipeline, running in parallel with GE. As illustrated in Figure 2, the added branch comprises a Q-model and a future encoder. The former generates Q-features for each navigation candidate, aggregating potential future observations

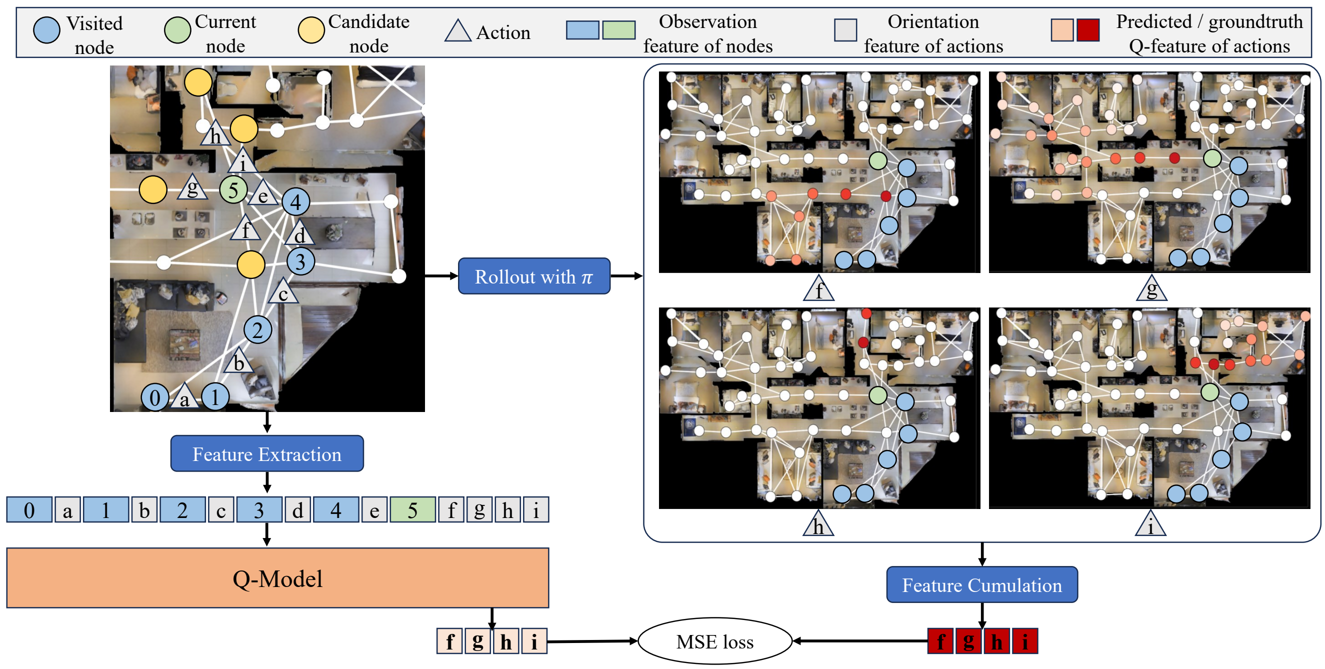

Figure 3. The design of our Q-model. Given a randomly sampled partial trajectory, the Q-model takes the observation and action features along the way as input, and predicts the Q-features for the candidate actions (f, g, h, i) at current node. The ground-truth Q-feature is the cumulated feature of all possible future nodes. We present a visualization of the cumulated nodes for each candidate action. The intensity of the red color on the node reflects the magnitude of the decay factor (γ^t) for cumulation.

along that direction into a latent vector. The latter further utilizes the text instruction to transform the task-agnostic Q-features into scores that are helpful for the navigation problem. Intuitively, by integrating the anticipation of long-horizon outcomes into the decision-making process, the model is expected to select more foresighted and efficient navigation actions. In the following two subsections, we will detail the design and training of these two modules.

3.3. Q-Model

In RL, the Q-value of a state-action pair is defined as the expected cumulative reward that an agent can attain by taking the action. A high-quality value function will help the agent execute prescient decision-making and select the optimal action. To train such a value function, we first need to define an appropriate reward function. For VLN, rewards are naturally related to the destination described by natural language instructions (e.g., distance to the destination) [111, 119, 120]. However, the scarcity of instruction-annotated data stands as a notorious issue in this field, prompting a series of endeavors [23, 25, 44, 53, 65, 91, 106, 118] trying to alleviate it. In light of this, we hope to decouple the reward computation from the training of our Q-model, by making the Q-model estimate future observations rather than future rewards. In particular, we define the Q-function as follows:

$$\boldsymbol{Q}(\mathbf{T}, a) = \boldsymbol{R}(\mathcal{A}) + \gamma \mathbb{E}_{a' \sim \pi(a' | \mathbf{T} \cup \{\mathcal{A}\})} [\boldsymbol{Q}(\mathbf{T} \cup \{\mathcal{A}\}, a')] \tag{6}$$

In the context of VLN, the state is a partial trajectory $\mathbf{T}$. The action a is to choose a local candidate node $\mathcal{A}$, which will deterministically lead to a new state $\mathbf{T} \cup \{\mathcal{A}\}$. γ is the decay ratio. $\boldsymbol{R}$ is a feature extractor. π is a navigation policy. It is clear that the formulation of Eq (6) is irrelevant with the navigation destination and the trajectory description, so $\boldsymbol{Q}$ can be learned on annotation-free scenes.

3.3.1. Data Gathering and Supervision

Based on the Bellman equation, classical Deep Q-Learning (DQN) [86] updates the Q-model using the gradients of temporal difference errors. In VLN, since there are finite nodes on the graph and revisiting is prohibited, the recursion in Eq (6) will end at some point where the current node has no valid candidate. Actually, the probability of reaching a particular node from a state-action pair under policy π, as well as the distribution of the number of steps required to reach it, can be calculated by enumerating all feasible episodes (i.e., terminated trajectories) on the graph. To be specific,

$$\mathrm{P}_\pi(\mathcal{N}, t | \mathbf{T}, a) = \sum_{\tilde{\mathbf{T}} \in \mathbb{T}(\mathbf{T}, \mathcal{A}, \mathcal{N}, t)} \mathrm{P}_\pi(\mathbf{T} \cup \{\mathcal{A}\} \rightarrow \tilde{\mathbf{T}}). \tag{7}$$

Here, $\mathbb{T}(\mathbf{T}, \mathcal{A}, \mathcal{N}, t)$ is the set of terminated trajectories that satisfy: (i) each trajectory starts with $\mathbf{T} \cup \{\mathcal{A}\}$, (ii) the portion of the trajectory after $\mathcal{A}$ contains node $\mathcal{N}$, and (iii) the number of steps from $\mathcal{A}$ to $\mathcal{N}$ is t. $\mathrm{P}_\pi(\cdot \rightarrow \cdot)$ is the probability of expanding a partial trajectory into a complete trajectory under the policy π. With this distribution of nodes and steps, Eq (6) can be transformed into a more straightforward equation:

$$\boldsymbol{Q}(\mathbf{T}, a) = \sum_{\mathcal{N},t} \mathrm{P}_{\pi}(\mathcal{N}, t|\mathbf{T}, a)\gamma^t \boldsymbol{R}(\mathcal{N}). \tag{8}$$

This formulation provides the ground-truth Q-feature for any state-action pair, enabling us to train our Q-model without RL techniques.

The rollout policy π determines the characteristic of the learned Q-function. A naive idea is to set it as a random policy that uniformly chooses a candidate as action. However, such a design can lead to a lack of discrimination between different candidates. In the graphs of navigation scenes, there are often numerous loops, which means that for an unexplored node $\mathcal{N}$, multiple candidate actions from the current node may potentially lead to it (i.e., $\mathbb{T}(\mathbf{T}, \mathcal{A}, \mathcal{N}, t)$ is non-empty for multiple candidate nodes $\mathcal{A}$). As a result, $\boldsymbol{R}(\mathcal{N})$ will be accumulated into the Q-features of multiple candidate actions, making them less informative. Put in another way, random exploration is highly inconsistent with the actual strategy adopted by a normal VLN model. To handle this problem, we note that goal-oriented VLN aims at finding the most efficient way to reach a target object. The best trajectory for any instruction is always the shortest path between two nodes. Based on this, we aim to incorporate a preference for the optimality of the future paths into the policy. We achieve this by introducing a fourth condition into the definition of the path set $\mathbb{T}(\mathbf{T}, \mathcal{A}, \mathcal{N}, t)$ in Equation (7): in each trajectory $\tilde{\mathbf{T}}$, the segment from $\mathbf{T}[-1]$ to $\tilde{\mathbf{T}}[-1]$ must be a shortest path. With this additional requirement, it can be proven that for any partial trajectory $\mathbf{T}$ and any node $\mathcal{N}$, there exists at most one pair of (a, t) that satisfies $\mathrm{P}_{\pi}(\mathcal{N}, t|\mathbf{T}, a) > 0$. This implies that the feature of each possible future node is accumulated into the Q-feature of a single action through a unique path. Figure 3 illustrates the sets of future nodes accumulated to different action candidates, along with the corresponding rollout steps t for each node. This policy design enables the learned Q-features to comprehensively aggregate diverse future observations while reflecting differences in navigation efficiency across actions, achieving a balance between coverage and optimality.

To sum up, the data generation pipeline for training our Q-model is: (i) sample a trajectory $\mathbf{T}$ of arbitrary length in the scene; (ii) sample an action a at the last node of the trajectory; (iii) use Eq (8) to compute the ground-truth Q-feature as supervision.

3.3.2. Model and Training

The Q-model is designed as a Transformer. As shown in Figure 3, the input consists of interleaved node features and action features of the partial trajectory, followed by the features of candidate actions at current location. Multiple candidates can be processed in a single forward pass as they share the trajectory prefix. We use the set of view features, $\{r_i^t\}$, as the descriptor of each node, while the actions are encoded by sin and cos values of the orientations. The outputs corresponding to the candidate tokens are used as predicted Q-features. MSE loss between the predictions and the ground-truth is employed to train the network.

The key consideration in pre-training the Q-model is to achieve generalizability. The model is expected to learn the common patterns regarding room layouts and object placements, rather than simply memorizing the details of the training scenes. Using large-scale random trajectories for training can solve this problem to some extent. Yet, due to the limited number of training scenes, the model is still at risk of overfitting. We further alleviate this issue through the following designs.

(1) Text-based Prediction. The visual features of RGB views inevitably carry some stylistic and texture information, which is usually unrelated to the navigation task. The Q-model trained on these features may establish some spurious correlations, making it difficult to generalize to new scenes. We propose to learn the Q-model in the latent text space, *i.e.,* the feature extractor $\boldsymbol{R}$ is designed to be the feature of the natural language description of a node. These descriptions can be obtained by pre-processing the scenes with an off-the-shelf image captioning model [60, 92]. By predicting the abstracted text-based features of future observations, the Q-model can better focus on high-level semantic relationships, thereby providing more reliable guidance for the navigation task.

(2) Warm-up Pre-training. Self-supervised pre-training is proven beneficial in many vision and language tasks. Before performing regression on the Q-features, we first carry out an MAE pre-training [35]. The input format is the same as described above, with some randomly selected tokens set to zero. An additional MLP is appended after the Transformer to reconstruct the masked tokens. This training process provides a good initialization for the Q-training and guides the model to fully analyze the information in the trajectory history.

3.4. Future Encoder

With the Q-model at hand, we can generate Q-features for the candidate actions at each navigation step, representing the scenarios the agent may encounter after it takes the action. The future encoder (FE) is responsible for transforming the task-agnostic feature into goal-oriented information. Formally,

$$\{\hat{q}_i^t\}_{i=0}^{|\tilde{G}^t|} = \mathrm{FE}(\tilde{G}^t, w). \tag{9}$$

$\tilde{G}^t$ is a graph with the same topology as G^t (Eq (1)), while it is updated with the Q-features of the candidate nodes in-

Table 1. The results on REVERIE. The best and second-best results are marked as **bold** and <u>underline</u>, respectively.

	Val Unseen					Test Unseen				
	OSR↑	SR↑	SPL↑	RGS↑	RGSPL↑	OSR↑	SR↑	SPL↑	RGS↑	RGSPL↑
HAMT [13] [NeurIPS21]	36.84	32.95	30.20	18.92	17.28	33.41	30.40	26.67	14.88	13.08
HOP [98] [CVPR22]	36.24	31.78	26.11	18.85	15.73	33.06	30.17	24.34	17.69	14.34
LANA [121] [CVPR23]	52.97	48.31	33.86	32.86	22.77	57.20	51.72	36.45	32.95	22.85
AZHP [28] [CVPR23]	53.65	48.31	36.63	34.00	25.79	55.31	51.57	35.85	32.25	22.44
KERM [64] [CVPR23]	55.21	50.44	35.38	34.51	24.45	57.58	52.43	39.21	32.39	23.64
BEV-Bert [2] [ICCV23]	-	51.78	36.37	34.71	24.44	-	52.81	36.41	32.06	22.09
BSG [78] [ICCV23]	58.05	52.12	35.59	35.36	24.24	**62.83**	56.45	38.70	33.15	22.34
GridMM [124] [ICCV23]	58.48	51.37	36.47	34.57	24.56	59.55	53.13	36.60	<u>34.87</u>	23.45
FDA [36] [NeurIPS23]	51.41	47.57	35.90	32.06	24.31	53.54	49.62	36.45	30.34	22.08
GOAT [117] [CVPR24]	-	<u>53.37</u>	36.70	**38.43**	<u>26.09</u>	-	**57.72**	**40.53**	**38.32**	**26.70**
VER [80] [CVPR24]	**61.09**	**55.98**	**39.66**	33.71	23.70	<u>62.22</u>	<u>56.82</u>	38.76	33.88	23.19
ENP [79] [NeurIPS24]	54.70	48.90	33.78	34.74	23.39	59.38	53.19	36.26	33.10	22.14
baseline [14] [CVPR22]	51.07	46.98	33.73	32.15	23.03	56.91	52.51	36.06	31.88	22.06
NavQ	<u>60.47</u>	53.22	<u>38.89</u>	<u>36.84</u>	**27.12**	60.39	53.29	<u>39.50</u>	34.82	<u>25.16</u>
	(+9.40)	(+6.24)	(+5.16)	(+4.69)	(+4.09)	(+3.48)	(+0.78)	(+3.44)	(+2.94)	(+3.10)
Methods with additional scenes										
AutoVLN [15] [ECCV22]	**62.14**	<u>55.89</u>	<u>40.85</u>	<u>36.58</u>	<u>26.76</u>	**62.30**	<u>55.17</u>	38.88	32.23	22.68
Lily [74] [ICCV23]	53.71	48.11	34.43	32.15	23.43	60.51	54.32	37.34	32.02	21.94
ScaleVLN [123] [ICCV23]	-	**56.97**	**41.84**	35.76	26.05	-	**56.13**	<u>39.52</u>	<u>32.53</u>	<u>22.78</u>
PanoGen [57] [NeurIPS23]	-	51.18	34.99	33.26	22.99	-	-	-	-	-
NavQ (w.o. speaker annotation)	<u>62.00</u>	54.10	39.22	**37.57**	**27.29**	<u>61.25</u>	54.91	**40.08**	**35.87**	**25.14**

stead of the view features. FE is designed as a Graph Transformer that shares the same architecture as GE. The output $\{\hat{q}_i^t\}_{i=0}^{|\tilde{G}^t|}$ is integrated into the fusion process described in Eq (4).

Ideally, the GE branch is tasked with analyzing historical information, while the FE branch handles future information. To ensure this decomposition and improve the performance of each branch, we introduce some additional supervisory signals. In previous works, progress monitor [82] is a widely-used auxiliary task, which requires the model to predict at each timestep the progress it has made towards the destination. Here we adopt this idea and designs two progress-related subtasks. For each node, on one hand, we send GE's node feature $\hat{v}_i^t$ to a lightweighted MLP to predict the traversed distance up to now. On the other hand, we send FE's output $\hat{q}_i^t$ to another MLP to predict the remaining distance to go. The ground-truth for them are designed as:$s_1(\mathcal{A}) = (\text{dist}(\mathcal{S}, \mathcal{C}) + \text{dist}(\mathcal{C}, \mathcal{A}))/D_1, s_2(\mathcal{A}) = \text{dist}(\mathcal{A}, \mathcal{G})/D_2$, where $\mathcal{S}$, $\mathcal{C}$, and $\mathcal{G}$ are the starting, current, and goal nodes, $\text{dist}(\cdot)$ is the shortest distance between two nodes, D_1 and D_2 are normalizing constants. The combination of these two sub-tasks also reflects the idea of integrating the cost function with a goal-directed heuristic function in the A* algorithm [33], allowing the future information embedded in the Q-feature to be effectively utilized by the navigation agent.

3.5. Training Scheme

The training process of NavQ is divided into three stages.

Stage 1: Q-model pre-training. As detailed in Section 3.3, we first pre-train the Q-model on randomly sampled trajectories in the training scenes. The Q-model will be kept frozen and used as a feature extractor in the following stages.

Stage 2: Agent pre-training. Pre-training on offline instruction-trajectory pairs is proven effective by many recent works [14, 31, 32, 41, 98]. We adopt the four pre-training tasks implemented by DUET. Besides, to give direct guidance to FE and GE, the two progress-related tasks mentioned in Section 3.4 are also included. Details of these tasks can be found in the supplementary material.

Stage 3: Agent finetuning. We still follow DUET to finetune the agent on online data using DAgger [105] wth a pseudo expert policy.

4. Experiments

4.1. Datasets and Metrics

Experiments are performed on two popular VLN benchmarks, REVERIE [96] and SOON [143]. Both are goal-oriented VLN datasets based on the MP3D simulator [3], requiring the agent to navigate to a target object instance. REVERIE includes a set of high-level instructions that guide the agent toward the target object located 4 to 7 steps away. SOON is a more challenging dataset with longer target descriptions and an average trajectory length of 9.5. We evaluate the model on the official validation set and test set, both consisting of previously unseen scenes during training. The metrics include success rate (SR), oracle SR (OSR), SR penalized by path length (SPL), remote grounding success (RGS), RGS penalized by path length (RGSPL). Detailed descriptions of the datasets and metrics can be found in the supplementary material.

Table 2. The results on SOON. The best and second-best results are marked as **bold** and underline, respectively.

		OSR	SR	SPL	RGSPL
Val Unseen	GBE [23]	28.54	19.52	13.34	1.16
	GridMM [124]	53.39	37.46	24.81	3.91
	KERM [64]	51.62	38.05	23.16	4.04
	AZHP [28]	56.19	**40.71**	26.58	5.53
	GOAT [117]	54.69	40.35	**28.05**	**6.10**
	baseline [14]	50.91	36.28	22.58	3.75
	NavQ (Ours)	**58.79**	39.09	26.65	5.51
		(+7.88)	(+2.81)	(+4.07)	(+1.76)
Test Unseen	GBE [23]	21.45	12.90	9.23	0.45
	GridMM [124]	48.02	36.27	21.25	4.15
	GOAT [117]	**50.63**	**40.50**	**25.18**	**6.10**
	baseline [14]	43.00	33.44	21.42	4.17
	NavQ (Ours)	48.92	38.59	24.50	4.48
		(+5.92)	(+5.15)	(+3.08)	(+0.31)

4.2. Implementation Details

The Q-model is implemented as a 4-layer Transformer, and the FE is a 4-layer Graph Transformer. The remaining parts of the model follow the same architecture as DUET [14]. The batch size, learning rate, and iterations for the three training stages are set to 128/32/4, 1e-5/5e-5/1e-5, 30k/100k/20k, respectively. CLIP-ViT/B is used as the visual and textual feature extractors for its cross-modal performance. The training can be conducted on a single NVIDIA RTX 3090 GPU. More details are presented in the supplementary material.

4.3. Main Results

Table 1 shows the performance comparison on REVERIE. Our NavQ agent consistently outperforms the DUET [14] baseline across all evaluation metrics, showing the effectiveness of incorporating the future branch. Compared to state-of-the-art models based on techniques such as causal learning [117] and volumetric representation [80], our model also demonstrates competitive performance, *e.g.*, +3.4%/+2.0% RGSPL than VER on the validation/test set, +2.2% SPL than GOAT on the validation set. One advantage of our method is that the Q-model could benefit from training on large-scale unlabeled scenes. To prove this, we borrow scenes from the HM3D [102] and Gibson [128] simulator following [123], and obtain a total of 1,351 scenes for Q-training. Note that we do not employ any speaker model [25] to label the trajectories in the additional scenes, and these scenes are only used in training stage 1. As illustrated in the lower part of Table 1, using additional scenes further boosts NavQ's navigation capability, reaching a performance comparable or higher than the methods that utilize additional annotated trajectories [15, 57, 74, 123].

Similarly, as in Table 2, our model also performs better than the baseline for all metrics on SOON. Note that the pre-trained Q-model is shared across REVERIE and SOON. Thus, the results highlight the task-agnostic nature of the learned Q-model, and demonstrate the generalizability of our approach.

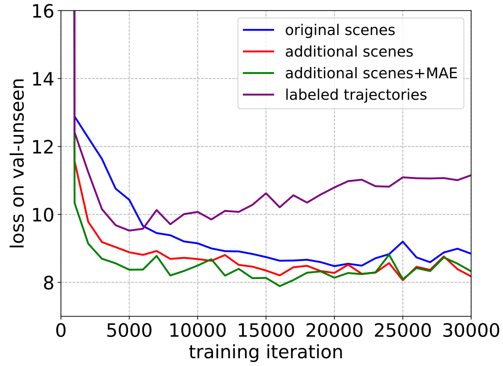

Figure 4. A comparison among different training techniques for the Q-model. We plot the MSE loss on the val-unseen scenes during the training process.

Table 3. Ablation study on the future branch. The results are obtained on REVERIE's unseen validation set.

	QM	FE	OSR	SR	SPL	RGS	RGSPL
(1)	✗	✗	54.42	48.14	33.38	30.19	21.05
(2)	✗	✓	54.84	48.20	33.92	32.52	23.14
(3)	✓	✗	53.25	48.48	32.22	33.03	21.86
(4)	✓	w.o. loss	55.98	51.55	35.79	34.51	23.81
(5)	✓	w. loss	**60.47**	**53.22**	**38.89**	**36.84**	**27.12**
(6)	GT	✗	60.18	54.36	41.71	37.03	28.59
(7)	GT	✓	**65.38**	**59.27**	**47.04**	**39.68**	**31.62**

4.4. Ablation Studies

We conduct an ablational experiment on the role of the Q-model (QM) and the future encoder (FE) in the future branch. The compared architectures include: (1) A variant without the future branch, *i.e.*, a reproduced version of the baseline model. (2) A variant that utilizes FE but not QM, where FE receives the same input as GE. (3) A variant that utilizes QM but not FE, where the output of QM is concatenated with the view features $\{r_i^t\}_{i=1}^N$ and fed into GE (Eq(1-2)). (4) A model that utilizes both QM and FE, but without supervision from the progress-related subtasks during the second training stage. (5) The full NavQ model.

The results are shown in Table 3. We first notice that our reproduced baseline has higher OSR/SR but lower RGS/RGSPL than the reported performance of DUET, which may be attributed to the use of different visual backbones. (We use CLIP-ViT/B to enhance the cross-modal capability of the Q-model, while DUET employs an ViT-B/16 pre-trained on ImageNet which is the same as the object feature extractor.) Upon that, merely introducing FE provides only limited improvement, suggesting that leveraging historical information alone may not be sufficient. On the other hand, the improvement achieved by solely using the Q model is minor too, indicating the significance of employ-

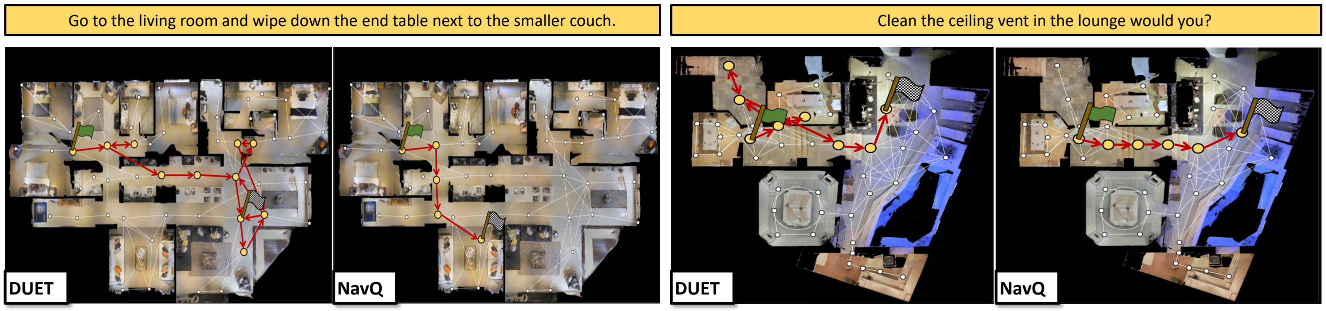

Figure 5. A qualitative comparison of our method and the baseline agent. In both examples, the NavQ agent performs the instruction correctly while the baseline agent fails.

ing FE to extract task-relevant information from the rich future context. The progress-related losses also contribute to the overall performance, validating the benefits of applying direct supervisions to decouple the history branch and the future branch.

In addition, we experiment with replacing the outputs of QM with the ground-truth (GT) Q-features, which serves as an upper bound for our method. The GT Q-features are sent to FE ((7) in Table 3) or concatenated with the view features and sent to GE ((6) in Table 3). It can be observed that using the GT Q-features significantly enhances performance, especially on the metrics related to navigation efficiency (*e.g.*, +14% SPL and +11% RGSPL over the baseline). These results validate the design of the Q-feature (Eq (8)) and the choice of the rollout policy π that incorporates a preference for shortest paths. Also, the superiority of using FE remains valid when high-quality Q-features are available.

To take a closer look at the training process of the Q-model, we plot the curve of validation loss on the scenes in the val-unseen set. As shown in Figure 4, training the Q-model with randomly sampled paths achieves better generalization than training solely with annotated paths, due to the vast difference in the number of training samples. This observation is a key factor motivating us to design a Q-learning paradigm without instruction annotations. Meanwhile, introducing additional training scenes and adding the MAE pre-training for Q-model also show positive influence on the quality of Q-features, which in turn leads to better navigation performance as in Table 1.

Besides, we also conduct an analysis on the decay ratio γ, which is a key hyper-parameter in the design of Q-model. When $\gamma = 0$, the Q-model is reduced to only predicting the observation of the immediate next step, like a world model or a novel view synthesis model. As γ grows larger, the ground-truth Q-features will encompass richer future information. At the same time, the training of the Q-model will become more challenging, and the discrepancy between the predicted Q-feature and the ground-truth will increase. We choose $\gamma = 0.5$ as a default setting, which makes a balance between the feature quality and the training difficulty.

Table 4. Analysis on the effect of the decay ratio for Q-features. The results are obtained on REVERIE's unseen validation set.

γ	OSR	SR	SPL	RGS	RGSPL
0	56.66	51.12	37.42	34.42	25.16
0.3	59.73	51.95	**38.90**	35.13	26.61
0.5	**60.47**	**53.22**	38.89	**36.84**	**27.12**
0.7	57.06	50.89	36.15	33.48	23.85

As shown in Table 4, it achieves higher overall navigation performance than using other values for γ. In particular, it clearly outperforms the $\gamma = 0$ variant which only reconstructs the feature of neighboring nodes, demonstrating the essential role of long-term future information.

4.5. Qualitative Results

In Figure 5, we visualize the trajectories predicted by the model on top-down floor maps. Thanks to the informative Q-features, our method can find the correct direction to explore when the items mentioned in the instruction are not yet observed. Therefore, compared to the baseline, NavQ demonstrates a higher likelihood of reaching the correct destination and exhibits greater navigation efficiency.

5. Conclusion

In this work, we propose a foresighted agent for goal-oriented VLN that efficiently integrates future-relevant information into a baseline model. A novel Q-model is developed to represent the future outcomes of a given action in the form of aggregated features. Based on scenes without instruction annotation, we design a self-supervised training paradigm using random trajectories and put forward a series of techniques for collecting training data and enhancing model generalization. Furthermore, we propose a future encoder that leverages instructions to transform the Q-features into assessments of candidate actions' anticipated future prospects, complementing the decision-making process that relies solely on historical information. In future work, we plan to further optimize the design of the Q-model, and explore extending the proposed approach to continuous environments.

Acknowledgments

This work is supported by an internal grant of Peking University (2024JK28).

References

[1] Dong An, Yuankai Qi, Yan Huang, Qi Wu, Liang Wang, and Tieniu Tan. Neighbor-view enhanced model for vision and language navigation. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 5101–5109, 2021. 2

[2] Dong An, Yuankai Qi, Yangguang Li, Yan Huang, Liang Wang, Tieniu Tan, and Jing Shao. Bevbert: Multimodal map pre-training for language-guided navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2737–2748, 2023. 2, 6, 4

[3] Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton Van Den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3674–3683, 2018. 1, 2, 6, 4

[4] Peter Anderson, Ayush Shrivastava, Devi Parikh, Dhruv Batra, and Stefan Lee. Chasing ghosts: Instruction following as bayesian state tracking. *Advances in neural information processing systems*, 32, 2019. 2

[5] Peter Anderson, Ayush Shrivastava, Joanne Truong, Arjun Majumdar, Devi Parikh, Dhruv Batra, and Stefan Lee. Sim-to-real transfer for vision-and-language navigation. In *Conference on Robot Learning*, pages 671–681. PMLR, 2021. 2

[6] Shurjo Banerjee, Jesse Thomason, and Jason Corso. The robotslang benchmark: Dialog-guided robot localization and navigation. In *Conference on Robot Learning*, pages 1384–1393. PMLR, 2021. 2

[7] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. *International Conference on 3D Vision (3DV)*, 2017. 2

[8] Matthew Chang, Arjun Gupta, and Saurabh Gupta. Semantic visual navigation by watching youtube videos. *Advances in Neural Information Processing Systems*, 33:4283–4294, 2020. 3

[9] Yevgen Chebotar, Quan Vuong, Karol Hausman, Fei Xia, Yao Lu, Alex Irpan, Aviral Kumar, Tianhe Yu, Alexander Herzog, Karl Pertsch, et al. Q-transformer: Scalable offline reinforcement learning via autoregressive q-functions. In *Conference on Robot Learning*, pages 3909–3928. PMLR, 2023. 3

[10] Jinyu Chen, Chen Gao, Erli Meng, Qiong Zhang, and Si Liu. Reinforced structured state-evolution for vision-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15450–15459, 2022. 2

[11] Jiaqi Chen, Bingqian Lin, Ran Xu, Zhenhua Chai, Xiaodan Liang, and Kwan-Yee Wong. Mapgpt: Map-guided prompting with adaptive path planning for vision-and-language navigation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9796–9810, 2024. 2

[12] Kevin Chen, Junshen K Chen, Jo Chuang, Marynel Vázquez, and Silvio Savarese. Topological planning with transformers for vision-and-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11276–11286, 2021. 2

[13] Shizhe Chen, Pierre-Louis Guhur, Cordelia Schmid, and Ivan Laptev. History aware multimodal transformer for vision-and-language navigation. *Advances in neural information processing systems*, 34:5834–5847, 2021. 1, 2, 6

[14] Shizhe Chen, Pierre-Louis Guhur, Makarand Tapaswi, Cordelia Schmid, and Ivan Laptev. Think global, act local: Dual-scale graph transformer for vision-and-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16537–16547, 2022. 1, 2, 3, 6, 7, 4

[15] Shizhe Chen, Pierre-Louis Guhur, Makarand Tapaswi, Cordelia Schmid, and Ivan Laptev. Learning from unlabeled 3d environments for vision-and-language navigation. In *European Conference on Computer Vision*, pages 638–655. Springer, 2022. 2, 6, 7, 1

[16] Wenhao Cheng, Xingping Dong, Salman Khan, and Jianbing Shen. Learning disentanglement with decoupled labels for vision-language navigation. In *European Conference on Computer Vision*, pages 309–329. Springer, 2022. 2

[17] Ta-Chung Chi, Minmin Shen, Mihail Eric, Seokhwan Kim, and Dilek Hakkani-Tur. Just ask: An interactive learning framework for vision and language navigation. In *Proceedings of the AAAI conference on artificial intelligence*, pages 2459–2466, 2020. 2

[18] Kyunghyun Cho and Shuhei Kurita. Generative language-grounded policy in vision-and-language navigation with bayes'rule. In *International Conference on Learning Representations*, 2020. 2

[19] Xinru Cui, Qiming Liu, Zhe Liu, and Hesheng Wang. Frontier-enhanced topological memory with improved exploration awareness for embodied visual navigation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024. 1, 2

[20] Yibo Cui, Liang Xie, Yakun Zhang, Meishan Zhang, Ye Yan, and Erwei Yin. Grounded entity-landmark adaptive pre-training for vision-and-language navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12043–12053, 2023. 2

[21] Zhiwei Deng, Karthik Narasimhan, and Olga Russakovsky. Evolving graphical planner: Contextual global planning for vision-and-language navigation. *Advances in Neural Information Processing Systems*, 33:20660–20672, 2020. 2

[22] Yilun Du, Chuang Gan, and Phillip Isola. Curious representation learning for embodied intelligence. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10408–10417, 2021. 2

[23] Sheng Fan, Rui Liu, Wenguan Wang, and Yi Yang. Navigation instruction generation with bev perception and large language models. In *European Conference on Computer Vision*, pages 368–387. Springer, 2025. 2, 4, 7

[24] Jesse Farebrother, Jordi Orbay, Quan Vuong, Adrien Ali Taïga, Yevgen Chebotar, Ted Xiao, Alex Irpan, Sergey Levine, Pablo Samuel Castro, Aleksandra Faust, Aviral Kumar, and Rishabh Agarwal. Stop regressing: training value functions via classification for scalable deep rl. In *Proceedings of the 41st International Conference on Machine Learning*. JMLR.org, 2024. 3

[25] Daniel Fried, Ronghang Hu, Volkan Cirik, Anna Rohrbach, Jacob Andreas, Louis-Philippe Morency, Taylor Berg-Kirkpatrick, Kate Saenko, Dan Klein, and Trevor Darrell. Speaker-follower models for vision-and-language navigation. *Advances in neural information processing systems*, 31, 2018. 2, 4, 7

[26] Tsu-Jui Fu, Xin Eric Wang, Matthew F Peterson, Scott T Grafton, Miguel P Eckstein, and William Yang Wang. Counterfactual vision-and-language navigation via adversarial path sampler. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VI 16*, pages 71–86. Springer, 2020. 2

[27] Chen Gao, Jinyu Chen, Si Liu, Luting Wang, Qiong Zhang, and Qi Wu. Room-and-object aware knowledge reasoning for remote embodied referring expression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3064–3073, 2021. 2

[28] Chen Gao, Xingyu Peng, Mi Yan, He Wang, Lirong Yang, Haibing Ren, Hongsheng Li, and Si Liu. Adaptive zone-aware hierarchical planner for vision-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14911–14920, 2023. 2, 6, 7

[29] Junyu Gao, Xuan Yao, and Changsheng Xu. Fast-slow test-time adaptation for online vision-and-language navigation. In *Forty-first International Conference on Machine Learning*. 2

[30] Georgios Georgakis, Bernadette Bucher, Karl Schmeckpeper, Siddharth Singh, and Kostas Daniilidis. Learning to map for active semantic goal navigation. In *The Tenth International Conference on Learning Representations (ICLR 2022)*, 2022. 2

[31] Pierre-Louis Guhur, Makarand Tapaswi, Shizhe Chen, Ivan Laptev, and Cordelia Schmid. Airbert: In-domain pretraining for vision-and-language navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1634–1643, 2021. 2, 6

[32] Weituo Hao, Chunyuan Li, Xiujun Li, Lawrence Carin, and Jianfeng Gao. Towards learning a generic agent for vision-and-language navigation via pre-training. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13137–13146, 2020. 2, 6

[33] Peter E. Hart, Nils J. Nilsson, and Bertram Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968. 1, 3, 6

[34] Keji He, Yan Huang, Qi Wu, Jianhua Yang, Dong An, Shuanglin Sima, and Liang Wang. Landmark-rxr: Solving vision-and-language navigation with fine-grained alignment supervision. *Advances in Neural Information Processing Systems*, 34:652–663, 2021. 2

[35] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022. 5

[36] Keji He, Chenyang Si, Zhihe Lu, Yan Huang, Liang Wang, and Xinchao Wang. Frequency-enhanced data augmentation for vision-and-language navigation. *Advances in Neural Information Processing Systems*, 36, 2024. 2, 6

[37] Matteo Hessel, Joseph Modayil, Hado Van Hasselt, Tom Schaul, Georg Ostrovski, Will Dabney, Dan Horgan, Bilal Piot, Mohammad Azar, and David Silver. Rainbow: Combining improvements in deep reinforcement learning. In *Proceedings of the AAAI conference on artificial intelligence*, 2018. 3

[38] Haodong Hong, Yanyuan Qiao, Sen Wang, Jiajun Liu, and Qi Wu. General scene adaptation for vision-and-language navigation. In *The Thirteenth International Conference on Learning Representations*, 2025. 2

[39] Yicong Hong, Cristian Rodriguez, Yuankai Qi, Qi Wu, and Stephen Gould. Language and visual entity relationship graph for agent navigation. *Advances in Neural Information Processing Systems*, 33:7685–7696, 2020. 2

[40] Yicong Hong, Cristian Rodriguez, Qi Wu, and Stephen Gould. Sub-instruction aware vision-and-language navigation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3360–3376, 2020. 2

[41] Yicong Hong, Qi Wu, Yuankai Qi, Cristian Rodriguez-Opazo, and Stephen Gould. Vln bert: A recurrent vision-and-language bert for navigation. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pages 1643–1653, 2021. 2, 6

[42] Yicong Hong, Yang Zhou, Ruiyi Zhang, Franck Dernoncourt, Trung Bui, Stephen Gould, and Hao Tan. Learning navigational visual representations with semantic map supervision. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3055–3067, 2023. 2

[43] Ronghang Hu, Daniel Fried, Anna Rohrbach, Dan Klein, Trevor Darrell, and Kate Saenko. Are you looking? grounding to multiple modalities in vision-and-language navigation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6551–6557, 2019. 2

[44] Haoshuo Huang, Vihan Jain, Harsh Mehta, Jason Baldridge, and Eugene Ie. Multi-modal discriminative model for vision-and-language navigation. In *Proceedings of the Combined Workshop on Spatial Language Understanding (SpLU) and Grounded Communication for Robotics (RoboNLP)*, pages 40–49, 2019. 2, 4

[45] Haoshuo Huang, Vihan Jain, Harsh Mehta, Alexander Ku, Gabriel Magalhaes, Jason Baldridge, and Eugene Ie. Trans-

ferable representation learning in vision-and-language navigation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7404–7413, 2019. 2

[46] Jingyang Huo, Qiang Sun, Boyan Jiang, Haitao Lin, and Yanwei Fu. Geovln: Learning geometry-enhanced visual representation with slot attention for vision-and-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23212–23221, 2023. 2

[47] Minyoung Hwang, Jaeyeon Jeong, Minsoo Kim, Yoonseon Oh, and Songhwai Oh. Meta-explore: Exploratory hierarchical vision-and-language navigation using scene object spectrum grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6683–6693, 2023. 2

[48] Vihan Jain, Gabriel Magalhaes, Alexander Ku, Ashish Vaswani, Eugene Ie, and Jason Baldridge. Stay on the path: Instruction fidelity in vision-and-language navigation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1862–1872, 2019. 1, 2

[49] Aishwarya Kamath, Peter Anderson, Su Wang, Jing Yu Koh, Alexander Ku, Austin Waters, Yinfei Yang, Jason Baldridge, and Zarana Parekh. A new path: Scaling vision-and-language navigation with synthetic instructions and imitation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10813–10823, 2023. 2

[50] Liyiming Ke, Xiujun Li, Yonatan Bisk, Ari Holtzman, Zhe Gan, Jingjing Liu, Jianfeng Gao, Yejin Choi, and Siddhartha Srinivasa. Tactical rewind: Self-correction via backtracking in vision-and-language navigation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6741–6749, 2019. 2

[51] Jing Yu Koh, Honglak Lee, Yinfei Yang, Jason Baldridge, and Peter Anderson. Pathdreamer: A world model for indoor navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14738–14748, 2021. 1, 2

[52] Jing Yu Koh, Harsh Agrawal, Dhruv Batra, Richard Tucker, Austin Waters, Honglak Lee, Yinfei Yang, Jason Baldridge, and Peter Anderson. Simple and effective synthesis of indoor 3d scenes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1169–1178, 2023. 2

[53] Xianghao Kong, Jinyu Chen, Wenguan Wang, Hang Su, Xiaolin Hu, Yi Yang, and Si Liu. Controllable navigation instruction generation with chain of thought prompting. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024. 2, 4

[54] Jacob Krantz, Erik Wijmans, Arjun Majumdar, Dhruv Batra, and Stefan Lee. Beyond the nav-graph: Vision-and-language navigation in continuous environments. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVIII 16*, pages 104–120. Springer, 2020. 1, 2

[55] Jacob Krantz, Shurjo Banerjee, Wang Zhu, Jason Corso, Peter Anderson, Stefan Lee, and Jesse Thomason. Iterative vision-and-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14921–14930, 2023. 2

[56] Alexander Ku, Peter Anderson, Roma Patel, Eugene Ie, and Jason Baldridge. Room-across-room: Multilingual vision-and-language navigation with dense spatiotemporal grounding. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4392–4412, 2020. 2, 4

[57] Jialu Li and Mohit Bansal. Panogen: Text-conditioned panoramic environment generation for vision-and-language navigation. *Advances in Neural Information Processing Systems*, 36:21878–21894, 2023. 2, 6, 7

[58] Jialu Li and Mohit Bansal. Improving vision-and-language navigation by generating future-view image semantics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10803–10812, 2023. 1, 2

[59] Jialu Li, Hao Tan, and Mohit Bansal. Improving cross-modal alignment in vision language navigation via syntactic information. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1041–1050, 2021. 2

[60] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pages 12888–12900. PMLR, 2022. 5, 1

[61] Jialu Li, Hao Tan, and Mohit Bansal. Envedit: Environment editing for vision-and-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15407–15417, 2022. 2

[62] Mingxiao Li, Zehao Wang, Tinne Tuytelaars, and Marie-Francine Moens. Layout-aware dreamer for embodied visual referring expression grounding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1386–1395, 2023. 2

[63] Xiujun Li, Chunyuan Li, Qiaolin Xia, Yonatan Bisk, Asli Celikyilmaz, Jianfeng Gao, Noah A Smith, and Yejin Choi. Robust navigation with language pretraining and stochastic sampling. 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)*, pages 1494–1499, 2019. 2

[64] Xiangyang Li, Zihan Wang, Jiahao Yang, Yaowei Wang, and Shuqiang Jiang. Kerm: Knowledge enhanced reasoning for vision-and-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2583–2592, 2023. 2, 6, 7, 1

[65] Xiwen Liang, Fengda Zhu, Li Lingling, Hang Xu, and Xiaodan Liang. Visual-language navigation pretraining via prompt-based environmental self-exploration. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4837–4851, 2022. 2, 4

[66] Xiwen Liang, Fengda Zhu, Yi Zhu, Bingqian Lin, Bing Wang, and Xiaodan Liang. Contrastive instruction-

trajectory learning for vision-language navigation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1592–1600, 2022. 2

[67] Yiqing Liang, Boyuan Chen, and Shuran Song. Sscnav: Confidence-aware semantic scene completion for visual semantic navigation. In *2021 IEEE international conference on robotics and automation (ICRA)*, pages 13194–13200. IEEE, 2021. 2

[68] Bingqian Lin, Yi Zhu, Yanxin Long, Xiaodan Liang, Qixiang Ye, and Liang Lin. Adversarial reinforced instruction attacker for robust vision-language navigation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44 (10):7175–7189, 2021. 2

[69] Bingqian Lin, Yi Zhu, Zicong Chen, Xiwen Liang, Jianzhuang Liu, and Xiaodan Liang. Adapt: Vision-language navigation with modality-aligned action prompts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15396–15406, 2022. 2

[70] Bingqian Lin, Yanxin Long, Yi Zhu, Fengda Zhu, Xiaodan Liang, Qixiang Ye, and Liang Lin. Towards deviation-robust agent navigation via perturbation-aware contrastive learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(10):12535–12549, 2023. 2

[71] Bingqian Lin, Yi Zhu, Xiaodan Liang, Liang Lin, and Jianzhuang Liu. Actional atomic-concept learning for demystifying vision-language navigation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1568–1576, 2023. 2, 1

[72] Bingqian Lin, Yunshuang Nie, Ziming Wei, Yi Zhu, Hang Xu, Shikui Ma, Jianzhuang Liu, and Xiaodan Liang. Correctable landmark discovery via large models for vision-language navigation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 2

[73] Chuang Lin, Yi Jiang, Jianfei Cai, Lizhen Qu, Gholamreza Haffari, and Zehuan Yuan. Multimodal transformer with variable-length memory for vision-and-language navigation. In *European Conference on Computer Vision*, pages 380–397. Springer, 2022. 2

[74] Kunyang Lin, Peihao Chen, Diwei Huang, Thomas H Li, Mingkui Tan, and Chuang Gan. Learning vision-and-language navigation from youtube videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8317–8326, 2023. 2, 6, 7

[75] Xiangru Lin, Guanbin Li, and Yizhou Yu. Scene-intuitive agent for remote embodied visual grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7036–7045, 2021. 2

[76] Zongyu Lin, Yao Tang, Da Yin, Stuart X Yao, Ziniu Hu, Yizhou Sun, and Kai-Wei Chang. Q* agent: Optimizing language agents with q-guided exploration. 3

[77] Chong Liu, Fengda Zhu, Xiaojun Chang, Xiaodan Liang, Zongyuan Ge, and Yi-Dong Shen. Vision-language navigation with random environmental mixup. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1644–1654, 2021. 2

[78] Rui Liu, Xiaohan Wang, Wenguan Wang, and Yi Yang. Bird's-eye-view scene graph for vision-language navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10968–10980, 2023. 1, 2, 6

[79] Rui Liu, Wenguan Wang, and Yi Yang. Vision-language navigation with energy-based policy. *arXiv preprint arXiv:2410.14250*, 2024. 2, 6, 1

[80] Rui Liu, Wenguan Wang, and Yi Yang. Volumetric environment representation for vision-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16317–16328, 2024. 1, 2, 6, 7

[81] Yuxing Long, Xiaoqi Li, Wenzhe Cai, and Hao Dong. Discuss before moving: Visual language navigation via multi-expert discussions. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 17380–17387. IEEE, 2024. 2

[82] Chih-Yao Ma, Jiasen Lu, Zuxuan Wu, Ghassan AlRegib, Zsolt Kira, Richard Socher, and Caiming Xiong. Self-monitoring navigation agent via auxiliary progress estimation. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019. 2, 6

[83] Chih-Yao Ma, Zuxuan Wu, Ghassan AlRegib, Caiming Xiong, and Zsolt Kira. The regretful agent: Heuristic-aided navigation through progress estimation. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pages 6732–6740, 2019. 2

[84] Arjun Majumdar, Ayush Shrivastava, Stefan Lee, Peter Anderson, Devi Parikh, and Dhruv Batra. Improving vision-and-language navigation with image-text pairs from the web. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VI 16*, pages 259–274. Springer, 2020. 2

[85] Volodymyr Mnih. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013. 3

[86] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015. 4

[87] Bahram Mohammadi, Yicong Hong, Yuankai Qi, Qi Wu, Shirui Pan, and Javen Qinfeng Shi. Augmented commonsense knowledge for remote object grounding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 4269–4277, 2024. 2, 1

[88] Abhinav Moudgil, Arjun Majumdar, Harsh Agrawal, Stefan Lee, and Dhruv Batra. Soat: A scene-and object-aware transformer for vision-and-language navigation. *Advances in Neural Information Processing Systems*, 34:7357–7367, 2021. 2

[89] Khanh Nguyen and Hal Daumé III. Help, anna! visual navigation with natural multimodal assistance via retrospective curiosity-encouraging imitation learning. 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)*, pages 684–695, 2019. 2

[90] Khanh Nguyen, Debadeepta Dey, Chris Brockett, and Bill Dolan. Vision-based navigation with language-based assis-

tance via imitation learning with indirect intervention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12527–12537, 2019. 2

[91] Joaquín Ossandón, Benjamín Earle, and Álvaro Soto. Bridging the visual semantic gap in vln via semantically richer instructions. In *European Conference on Computer Vision*, pages 54–69. Springer, 2022. 2, 4

[92] Bowen Pan, Rameswar Panda, SouYoung Jin, Rogerio Feris, Aude Oliva, Phillip Isola, and Yoon Kim. Langnav: Language as a perceptual representation for navigation. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 950–974, 2024. 2, 5, 1

[93] Amin Parvaneh, Ehsan Abbasnejad, Damien Teney, Javen Qinfeng Shi, and Anton Van den Hengel. Counterfactual vision-and-language navigation: Unravelling the unseen. *Advances in neural information processing systems*, 33:5296–5307, 2020. 2

[94] Pranav Putta, Edmund Mills, Naman Garg, Sumeet Motwani, Chelsea Finn, Divyansh Garg, and Rafael Rafailov. Agent q: Advanced reasoning and learning for autonomous ai agents. *arXiv preprint arXiv:2408.07199*, 2024. 3

[95] Yuankai Qi, Zizheng Pan, Shengping Zhang, Anton van den Hengel, and Qi Wu. Object-and-action aware model for visual language navigation. In *European Conference on Computer Vision*, pages 303–317. Springer, 2020. 2

[96] Yuankai Qi, Qi Wu, Peter Anderson, Xin Wang, William Yang Wang, Chunhua Shen, and Anton van den Hengel. Reverie: Remote embodied visual referring expression in real indoor environments. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1, 2, 6, 3, 4

[97] Yuankai Qi, Zizheng Pan, Yicong Hong, Ming-Hsuan Yang, Anton Van Den Hengel, and Qi Wu. The road to know-where: An object-and-room informed sequential bert for indoor vision-language navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1655–1664, 2021. 2

[98] Yanyuan Qiao, Yuankai Qi, Yicong Hong, Zheng Yu, Peng Wang, and Qi Wu. Hop: History-and-order aware pretraining for vision-and-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15418–15427, 2022. 2, 6

[99] Yanyuan Qiao, Yuankai Qi, Zheng Yu, Jing Liu, and Qi Wu. March in chat: Interactive prompting for remote embodied referring expression. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15758–15767, 2023. 2

[100] Yanyuan Qiao, Zheng Yu, and Qi Wu. Vln-petl: Parameter-efficient transfer learning for vision-and-language navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15443–15452, 2023. 2

[101] Yanyuan Qiao, Qianyi Liu, Jiajun Liu, Jing Liu, and Qi Wu. Llm as copilot for coarse-grained vision-and-language navigation. 2024. 2

[102] Santhosh Kumar Ramakrishnan, Aaron Gokaslan, Erik Wijmans, Oleksandr Maksymets, Alexander Clegg, John M Turner, Eric Undersander, Wojciech Galuba, Andrew Westbury, Angel X Chang, et al. Habitat-matterport 3d dataset (hm3d): 1000 large-scale 3d environments for embodied ai. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*. 7, 2

[103] Santhosh Kumar Ramakrishnan, Devendra Singh Chaplot, Ziad Al-Halah, Jitendra Malik, and Kristen Grauman. Poni: Potential functions for objectgoal navigation with interaction-free learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18890–18900, 2022. 2

[104] Homero Roman Roman, Yonatan Bisk, Jesse Thomason, Asli Celikyilmaz, and Jianfeng Gao. Rmm: A recursive mental model for dialogue navigation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1732–1745, 2020. 2

[105] Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. 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*, pages 627–635. JMLR Workshop and Conference Proceedings, 2011. 6

[106] Hao Tan, Licheng Yu, and Mohit Bansal. Learning to navigate unseen environments: Back translation with environmental dropout. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2610–2621, 2019. 2, 4

[107] Jesse Thomason, Michael Murray, Maya Cakmak, and Luke Zettlemoyer. Vision-and-dialog navigation. In *Conference on Robot Learning*, pages 394–406. PMLR, 2020. 1, 2

[108] Hado Van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double q-learning. In *Proceedings of the AAAI conference on artificial intelligence*, 2016. 3

[109] Chaojie Wang, Yanchen Deng, Zhiyi Lv, Shuicheng Yan, and An Bo. Q*: Improving multi-step reasoning for llms with deliberative planning. *arXiv preprint arXiv:2406.14283*, 2024. 3

[110] Hanqing Wang, Wenguan Wang, Tianmin Shu, Wei Liang, and Jianbing Shen. Active visual information gathering for vision-language navigation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXII 16*, pages 307–322. Springer, 2020. 2

[111] Hu Wang, Qi Wu, and Chunhua Shen. Soft expert reward learning for vision-and-language navigation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IX 16*, pages 126–141. Springer, 2020. 2, 4

[112] Hanqing Wang, Wenguan Wang, Wei Liang, Caiming Xiong, and Jianbing Shen. Structured scene memory for vision-language navigation. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pages 8455–8464, 2021. 2

[113] Hanqing Wang, Wei Liang, Luc V Gool, and Wenguan Wang. Towards versatile embodied navigation. *Advances in neural information processing systems*, 35:36858–36874, 2022. 2

[114] Hanqing Wang, Wei Liang, Jianbing Shen, Luc Van Gool, and Wenguan Wang. 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*, pages 15471–15481, 2022. 2

[115] Hanqing Wang, Wei Liang, Luc Van Gool, and Wenguan Wang. Dreamwalker: Mental planning for continuous vision-language navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10873–10883, 2023. 1, 2

[116] Liuyi Wang, Zongtao He, Jiagui Tang, Ronghao Dang, Naijia Wang, Chengju Liu, and Qijun Chen. A dual semantic-aware recurrent global-adaptive network for vision-and-language navigation. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, pages 1479–1487, 2023. 2

[117] Liuyi Wang, Zongtao He, Ronghao Dang, Mengjiao Shen, Chengju Liu, and Qijun Chen. Vision-and-language navigation via causal learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13139–13150, 2024. 1, 2, 6, 7

[118] Su Wang, Ceslee Montgomery, Jordi Orbay, Vighnesh Birodkar, Aleksandra Faust, Izzeddin Gur, Natasha Jaques, Austin Waters, Jason Baldridge, and Peter Anderson. Less is more: Generating grounded navigation instructions from landmarks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15428–15438, 2022. 2, 4

[119] Xin Wang, Wenhan Xiong, Hongmin Wang, and William Yang Wang. 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)*, pages 37–53, 2018. 1, 2, 4

[120] Xin Wang, Qiuyuan Huang, Asli Celikyilmaz, Jianfeng Gao, Dinghan Shen, Yuan-Fang Wang, William Yang Wang, and Lei Zhang. Reinforced cross-modal matching and self-supervised imitation learning for vision-language navigation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6629–6638, 2019. 2, 4

[121] Xiaohan Wang, Wenguan Wang, Jiayi Shao, and Yi Yang. Lana: A language-capable navigator for instruction following and generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19048–19058, 2023. 2, 6

[122] Xin Eric Wang, Vihan Jain, Eugene Ie, William Yang Wang, Zornitsa Kozareva, and Sujith Ravi. Environment-agnostic multitask learning for natural language grounded navigation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIV 16*, pages 413–430. Springer, 2020. 2

[123] Zun Wang, Jialu Li, Yicong Hong, Yi Wang, Qi Wu, Mohit Bansal, Stephen Gould, Hao Tan, and Yu Qiao. Scaling data generation in vision-and-language navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12009–12020, 2023. 2, 6, 7, 1

[124] Zihan Wang, Xiangyang Li, Jiahao Yang, Yeqi Liu, and Shuqiang Jiang. Gridmm: Grid memory map for vision-and-language navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15625–15636, 2023. 2, 6, 7

[125] Zihan Wang, Xiangyang Li, Jiahao Yang, Yeqi Liu, Junjie Hu, Ming Jiang, and Shuqiang Jiang. Lookahead exploration with neural radiance representation for continuous vision-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13753–13762, 2024. 1, 2

[126] Zun Wang, Jialu Li, Yicong Hong, Songze Li, Kunchang Li, Shoubin Yu, Yi Wang, Yu Qiao, Yali Wang, Mohit Bansal, and Limin Wang. Bootstrapping language-guided navigation learning with self-refining data flywheel. In *The Thirteenth International Conference on Learning Representations*, 2025. 2

[127] Christopher JCH Watkins and Peter Dayan. Q-learning. *Machine learning*, 8:279–292, 1992. 3

[128] Fei Xia, Amir R Zamir, Zhiyang He, Alexander Sax, Jitendra Malik, and Silvio Savarese. Gibson env: Real-world perception for embodied agents. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9068–9079, 2018. 7, 2

[129] Taku Yamagata, Ahmed Khalil, and Raul Santos-Rodriguez. Q-learning decision transformer: Leveraging dynamic programming for conditional sequence modelling in offline rl. In *International Conference on Machine Learning*, pages 38989–39007. PMLR, 2023. 3

[130] Albert J Zhai and Shenlong Wang. Peanut: Predicting and navigating to unseen targets. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10926–10935, 2023. 2

[131] Jiwen Zhang, Jianqing Fan, Jiajie Peng, et al. Curriculum learning for vision-and-language navigation. *Advances in Neural Information Processing Systems*, 34:13328–13339, 2021. 2

[132] Jiazhao Zhang, Kunyu Wang, Rongtao Xu, Gengze Zhou, Yicong Hong, Xiaomeng Fang, Qi Wu, Zhizheng Zhang, and He Wang. Navid: Video-based vlm plans the next step for vision-and-language navigation. *arXiv preprint arXiv:2402.15852*, 2024. 2

[133] Sixian Zhang, Xinyao Yu, Xinhang Song, Xiaohan Wang, and Shuqiang Jiang. Imagine before go: Self-supervised generative map for object goal navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16414–16425, 2024. 2

[134] Weixia Zhang, Chao Ma, Qi Wu, and Xiaokang Yang. Language-guided navigation via cross-modal grounding and alternate adversarial learning. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(9):3469–3481, 2020. 2

[135] Yue Zhang and Parisa Kordjamshidi. Vln-trans: Translator for the vision and language navigation agent. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13219–13233, 2023. 2

[136] Yubo Zhang, Hao Tan, and Mohit Bansal. Diagnosing the environment bias in vision-and-language navigation. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 890–897, 2021. 2

[137] Xinxin Zhao, Wenzhe Cai, Likun Tang, and Teng Wang. Imaginenav: Prompting vision-language models as embodied navigator through scene imagination. In *The Thirteenth International Conference on Learning Representations*, 2025. 1, 2

[138] Yusheng Zhao, Jinyu Chen, Chen Gao, Wenguan Wang, Lirong Yang, Haibing Ren, Huaxia Xia, and Si Liu. Target-driven structured transformer planner for vision-language navigation. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 4194–4203, 2022. 2

[139] Duo Zheng, Shijia Huang, Lin Zhao, Yiwu Zhong, and Liwei Wang. Towards learning a generalist model for embodied navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13624–13634, 2024. 2

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

[141] Gengze Zhou, Yicong Hong, Zun Wang, Xin Eric Wang, and Qi Wu. Navgpt-2: Unleashing navigational reasoning capability for large vision-language models. In *European Conference on Computer Vision*, pages 260–278. Springer, 2025. 2, 1

[142] Fengda Zhu, Yi Zhu, Xiaojun Chang, and Xiaodan Liang. Vision-language navigation with self-supervised auxiliary reasoning tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10012–10022, 2020. 2

[143] Fengda Zhu, Xiwen Liang, Yi Zhu, Qizhi Yu, Xiaojun Chang, and Xiaodan Liang. Soon: Scenario oriented object navigation with graph-based exploration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12689–12699, 2021. 1, 2, 6, 3, 4

[144] Wang Zhu, Hexiang Hu, Jiacheng Chen, Zhiwei Deng, Vihan Jain, Eugene Ie, and Fei Sha. Babywalk: Going farther in vision-and-language navigation by taking baby steps. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2539–2556, 2020. 2

[145] Yi Zhu, Yue Weng, Fengda Zhu, Xiaodan Liang, Qixiang Ye, Yutong Lu, and Jianbin Jiao. Self-motivated communication agent for real-world vision-dialog navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1594–1603, 2021. 2