# ADAPT: Vision-Language Navigation with Modality-Aligned Action Prompts

Bingqian Lin[1,*], Yi Zhu[2], Zicong Chen[1], Xiwen Liang[1], Jianzhuang Liu[2], Xiaodan Liang[1†]
[1]Shenzhen Campus of Sun Yat-sen University [2]Huawei Noah's Ark Lab
{linbq6@mail2,chenzc7@mail2,liangxw29@mail2, liangxd9@mail}.sysu.edu.cn,
{zhuyi36, liu.jianzhuang}@huawei.com

## Abstract

*Vision-Language Navigation (VLN) is a challenging task that requires an embodied agent to perform action-level modality alignment, i.e., make instruction-asked actions sequentially in complex visual environments. Most existing VLN agents learn the instruction-path data directly and cannot sufficiently explore action-level alignment knowledge inside the multi-modal inputs. In this paper, we propose modAlity-aligneD Action PrompTs (ADAPT), which provides the VLN agent with action prompts to enable the explicit learning of action-level modality alignment to pursue successful navigation. Specifically, an action prompt is defined as a modality-aligned pair of an image sub-prompt and a text sub-prompt, where the former is a single-view observation and the latter is a phrase like "walk past the chair". When starting navigation, the instruction-related action prompt set is retrieved from a pre-built action prompt base and passed through a prompt encoder to obtain the prompt feature. Then the prompt feature is concatenated with the original instruction feature and fed to a multi-layer transformer for action prediction. To collect high-quality action prompts into the prompt base, we use the Contrastive Language-Image Pretraining (CLIP) model which has powerful cross-modality alignment ability. A modality alignment loss and a sequential consistency loss are further introduced to enhance the alignment of the action prompt and enforce the agent to focus on the related prompt sequentially. Experimental results on both R2R and RxR show the superiority of ADAPT over state-of-the-art methods.*

## 1. Introduction

In the Vision-Language Navigation (VLN) task [1,4], an embodied agent is required to navigate through complex scenes following a given language instruction. To accomplish successful navigation, the agent needs to implement both object-level and action-level modality alignment accurately given the instruction and visual observations. For example, given an instruction of "exit the bedroom", the agent should not only locate the "bedroom" in its observation but also find the door of the bedroom to make the action of "exit". With great potential in the applications such as in-home robots and personal assistants, VLN has received wide spread attention in the robotic visual applications.

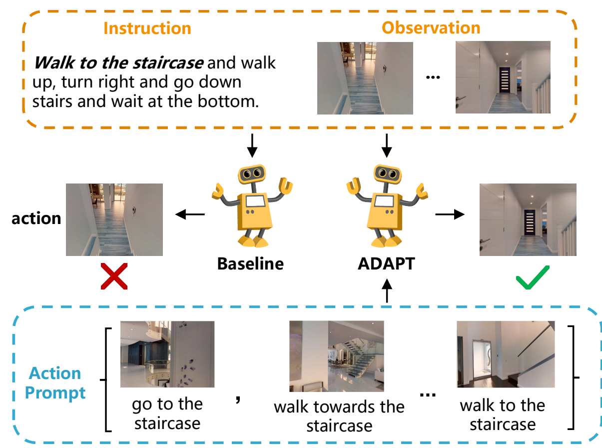


Figure 1. The action decision comparison between a baseline [14] and our ADAPT. With the help of action prompts related to "walk to the staircase" in the instruction, our ADAPT successfully makes correct action from the current observation.

Early VLN approaches explore diverse data augmentation strategies [8,9,27,38], efficient learning paradigms [15, 24, 40, 46, 47] and useful model architecture [7, 13, 29, 40] to improve the agent performance. Motivated by the significant progress made by large-scale cross-modal pre-trained models in vision-language tasks [6,21,23,25,37], more and more works attempt to introduce the pretraining paradigms and models into the VLN task. PREVALENT [11] pretrains the model on a large amount of image-text-action triplets in a self-supervised learning manner. VLN⟳BERT [14] introduces a recurrent function into the pretrained models to make the VLN agent time-aware. Although the object-level alignment ability may be significantly enhanced through the

*Part of this work was done during an internship in Huawei Noah's Ark Lab.

†Corresponding Author

pretraining process, these VLN agents still learn the action-level modality alignment in an implicit way, which largely limits the robust action decision under different scenes.

Recently, the prompt engineering paradigm has shown great potential in endowing pretrained models with diverse capabilities through simply providing prompts designed by experts or optimized with task-specific objectives [20,28,39,43,45]. Inspired by this, we propose to introduce the prompt into the VLN task to improve the action-level modality alignment ability of the pretrained VLN agents. To this end, we propose mod**A**lity-aligne**D** **A**ction **P**romp**T**s (ADAPT), where the agent is provided with explicit *action prompts* to make action decision. An action prompt contains a pair of multi-modal sub-prompts, where the image sub-prompt is a single-view observation indicating a salient visual object or location, and the paired text sub-prompt is an object-related action phrase like "go to the staircase".

Before navigating, the instruction-related action prompts are retrieved from a pre-constructed action prompt base. Then the action prompts are passed through a prompt encoder and the output feature is concatenated with the original instruction feature. The prompt-based instruction feature, together with the visual feature are fed to a multi-layer transformer for making action decision. Note that different from the common prompt engineering methods which change the output prediction form of a downstream task by introducing the prompt [28], in this work, we keep the same form of the action prediction as the baseline model and focus on the design of the prompts. Through these provided action prompts, the agent can learn the action-level modality alignment explicitly and make robust actions in different scenes. To enhance the discriminative power of the action prompts and enforce the agent to attend to related action prompts at each timestep, a modality alignment loss and a sequential consistency loss are further introduced into the training. Fig. 1 presents an action decision comparison between the baseline agent [14] and our ADAPT. As shown in Fig. 1, with the help of the action prompts related to "walk to the staircase", our ADAPT can choose the correct action in the given observations to navigate successfully.

To collect high-quality action prompts into the action prompt base, we resort to the recently developed Contrastive Language-Image Pretraining (CLIP) [32] model which has powerful cross-modal object/location-level alignment ability. Concretely, the image sub-prompt is obtained by retrieving object/location-related images using CLIP from the action image sequence where each image contains the action information itself. The text sub-prompt is derived through a simple nearest-verb-search scheme.

Experimental results on both Room-to-Room (R2R) [1] and Room-across-Room (RxR) [19] benchmarks show the superiority of our proposed ADAPT over the state-of-the-art methods, demonstrating that introducing explicit action prompts is promising for improving the agent navigation performance. Our ablation study indicates the effectiveness of each method component and the good generalization ability of ADAPT. The visualization analysis also shows its good interpretability.

To summarize, the main contributions of this paper are: 1) We propose modality-aligned action prompts (ADAPT) to enforce the VLN agent to learn cross-modal action knowledge explicitly for improving action decision during navigation. To the best of our knowledge, this is the first attempt to develop prompt-based agents in the VLN task. 2) We develop a modality alignment loss and a sequential consistency loss for enabling efficient learning of action prompts. The Contrastive Language-Image Pretraining (CLIP) model is employed to ensure the quality of the action prompts. 3) ADAPT establishes new state-of-the-art results on both R2R and RxR. It also shows good interpretability and generalization ability.

## 2. Related Work

**Vision-Language Navigation.** Given the language instruction, a VLN agent is required to follow it to reach a predefined goal position. Early methods usually employ a sequence-to-sequence model architecture [8, 38, 46]. Speaker-follower [8] introduces synthetic instructions to alleviate the annotation burden of instructions. EnvDrop [38] develops an environmental dropout strategy to generate augmented data by mimicking unseen environments.

Recently, large-scale vision-language pretraining models [6,21,23,25,37] have shown significant superiority on multiple vision-language understanding tasks like Visual Commonsense Reasoning [42] and Visual Question Answering [2]. Inspired by this, more and more works have introduced vision-language pretrained models into the VLN area [11, 14, 31]. PREVALENT [11] collects plenty of image-text-action triplets to pretrain the agent with self-supervised tasks such as attended masked language modeling and action prediction. VLN↻BERT [14] adds a recurrent function to help the agent recognize time-dependent input. However, in these pretrained VLN methods, the agent learns the relationship between the action decision and multi-modal information implicitly, leading to inefficient training and limited generalization abilities. In this paper, we take the first step to develop a prompt-based VLN agent, which receives explicit action prompts indicating cross-modal action knowledge for assisting the action decision during navigation.

**Prompt Engineering.** Recent studies have shown that prompts play a vital role in improving pretrained language models in many downstream NLP tasks [3,20,26,28,33,43]. Jiang et al. [18] apply the text mining and paraphrasing techniques to generate the candidate prompts and choose the one with the highest accuracy. For facilitating prompt learning, Shin et al. [36] propose to generate prompts auto-

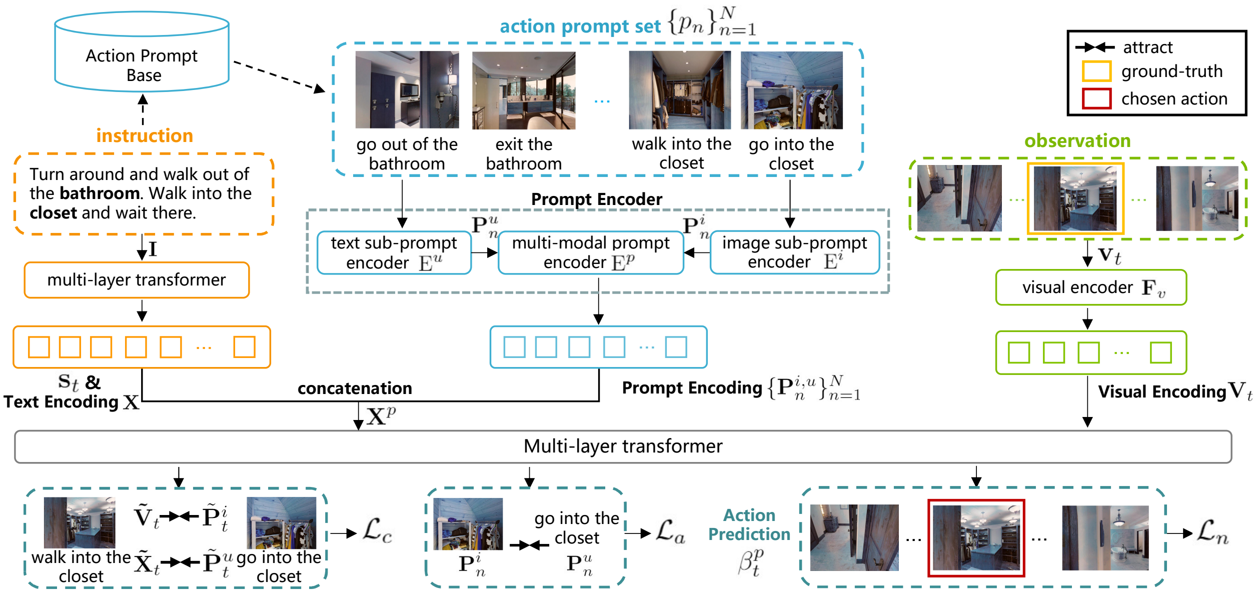


Figure 2. Overview of our ADAPT. At timestep $t$, the agent receives the instruction, visual observation, and retrieved action prompts. The action prompts are passed through the prompt encoder and the output feature is concatenated with the instruction encoding $\mathbf{X}$ to obtain prompt-based instruction feature $\mathbf{X}^p$. The action decision is made based on $\mathbf{X}^p$ and the visual encoding $\mathbf{V}_t$. The navigation loss $\mathcal{L}_n$, the sequential consistency loss $\mathcal{L}_c$ and the modality alignment loss $\mathcal{L}_a$ are applied to optimize ADAPT. (Best viewed in color.)

matically through the gradient-based search. Lately, some works [20, 26, 43] propose to generate continuous prompts instead of hand-crafted text prompts.

Inspired by the progress that prompt learning has made in NLP, some works attempt to introduce it into the pre-trained vision-language models recently [39, 41, 45]. CoOp [45] models the context in prompts using continuous representations and keeps the pretrained model parameters fixed to conduct end-to-end learning. CPT [41] reformulates the visual grounding task into a fill-in-the-blank problem with color-based cross-modal prompts. Frozen [39] encodes the image as a sequence of continuous embeddings to serve as the prefix to implement multi-modal few-shot learning. In the light of the prompt engineering paradigm, we introduce the modality-aligned action prompts during navigation for enabling VLN agents to learn cross-modal action knowledge explicitly. Through these action prompts, the agent can effectively learn action-level modality alignment for implementing successful navigation.

**Contrastive Language-Image Pretraining (CLIP).** CLIP [32] is a large-scale pre-trained model that relies on natural language supervision to learn visual representations. For an image-text pair, a visual encoder and a text encoder are used to encode the input representations independently. And the dot product between the two encoder's output serves as the alignment score of the image-text pair. Through training on 400M noisy image-text pairs, CLIP has shown strong zero-shot capabilities on benchmarks such as ImageNet classification. Recently, some works propose to resort to the knowledge learned in CLIP to improve the generalization ability of downstream models, including object detection [10], image manipulation [30], and vision-language tasks [35]. In this paper, we employ CLIP to retrieve the image containing instruction-referred visual object/location in a specific action image sequence for building action prompts. With the powerful cross-modal alignment ability of CLIP, the instruction-referred visual object/location images can be effectively retrieved for ensuring the quality of the action prompts.

## 3. Method

The overview of our ADAPT is given in Fig. 2. Before navigation, the agent retrieves the instruction-related action prompts from the action prompt base. Then the agent makes the action decision at each timestep based on the given instruction, the visual observation, and retrieved action prompts. The navigation is optimized by the navigation loss $\mathcal{L}_n$, the sequential consistency loss $\mathcal{L}_c$, and the modality alignment loss $\mathcal{L}_a$.

### 3.1. VLN Problem Setup

Given a language instruction $\mathbf{I} = \{w_0, ..., w_L\}$ with $L$ words, a VLN agent is required to find a route from a start viewpoint $c_0$ to the target viewpoint $c_T$. At each timestep $t$, the agent observes a panoramic view, which contains 36 image views $\{o_{t,i}\}_{i=1}^{36}$. Each image view $o_{t,i}$ includes an RGB image $b_{t,i}$ accompanied with its orientation $(\theta_{t,i}^1, \theta_{t,i}^2)$, where $\theta_{t,i}^1$ and $\theta_{t,i}^2$ are the angles of heading and elevation, respectively. With the instructions and current visual observations, the agent infers the action for each step $t$ from the candidate actions list, which consists of $J$ neighbors of the current node in the navigation connectivity graph $\mathcal{G} = (V, E)$ and a stop action. $V$ and $E$ represent the nodes and edges in the navigation connectivity graph, respectively.

## 3.2. VLN Agent with Action Prompts

### 3.2.1 Baseline Agent

Our baseline agent follows the architecture of VLN↻BERT [14], which is a multi-layer transformer model consisting of the self-attention module and cross-modal attention module. At each timestep, the model receives the cross-modal inputs for the action prediction.

**Visual Input.** For each image view $o_{t,i}$ in the candidate views at timestep $t$, a pretrained Convolutional Neural Network (CNN) [14] or a transformer [35] is applied in advance to extract image feature $\mathbf{v}_{t,i}$. Then $\mathbf{v}_{t,i}$ is projected by a visual encoder $\mathbf{F}_v$ [14] to get the visual encoding $\mathbf{V}_{t,i}$:

$$\mathbf{V}_{t,i} = \mathbf{F}_v(\mathbf{v}_{t,i}; \theta_v), \tag{1}$$

where $\theta_v$ denotes the parameters of $\mathbf{F}_v$. The set $\mathbf{V}_t = \{\mathbf{V}_{t,i}\}_{i=1}^{36}$ denotes the candidate visual encodings at timestep $t$.

**Language Input.** When initialization, the instruction encoding $\mathbf{X}$ and the initialized state feature $\mathbf{s}_0$ are obtained by feeding the instruction sequence $\mathbf{I}$ together with [CLS] and [SEP] tokens to the self-attention module in the transformer:

$$\mathbf{s}_0, \mathbf{X} = \mathrm{SelfAttn}(\mathrm{Concat}([\mathrm{CLS}], \mathbf{I}, [\mathrm{SEP}]); \theta_s^1), \tag{2}$$

where $\mathrm{Concat}(\cdot)$ represents the concatenation operation, and $\theta_s^1$ denotes the parameters of the self-attention module. $\mathbf{s}_0$ will be updated to obtain $\mathbf{s}_t$ at each timestep $t$.

**Action Decision.** During the action decision at timestep $t$, the state feature $\mathbf{s}_t$ is concatenated with the visual feature $\mathbf{V}_t$ to obtain the state-visual feature $\mathbf{K}_t$. Then the cross-modal attention $\alpha_t$ between $\mathbf{K}_t$ and the instruction feature $\mathbf{X}$ is calculated to update $\mathbf{K}_t$:

$$\tilde{\mathbf{K}}_t, \alpha_t = \mathrm{CrossAttn}(\mathbf{K}_t, \mathbf{X}; \theta_c), \tag{3}$$

where $\theta_c$ represents the parameters of the cross-modal attention module. The attended instruction feature $\tilde{\mathbf{X}}_t$ is derived by weighting the instruction feature $\mathbf{X}$ by $\alpha_t$. The updated state-visual feature $\tilde{\mathbf{K}}_t$ is further fed to another self-attention module $\mathrm{SelfAttn}(\cdot)$ to obtain the attention scores $\beta_t$ of the state feature $\mathbf{s}_t$ over the visual feature $\mathbf{V}_t$, which is also treated as the action prediction probability:

$$\beta_t = \mathrm{SelfAttn}(\tilde{\mathbf{K}}_t; \theta_s^2), \tag{4}$$

where $\theta_s^2$ represents the module parameters. The attended visual feature $\tilde{\mathbf{V}}_t$ is obtained through weighting the visual feature $\mathbf{V}_t$ by $\beta_t$. Then $\tilde{\mathbf{X}}_t$ and $\tilde{\mathbf{V}}_t$ are used for updating the state feature $\mathbf{s}_t$ which is used for the next timestep action prediction. For more model details, refer to [14].


Figure 3. Examples of action prompts.

### 3.2.2 Action Prompts

Before describing our prompt-based VLN agent, we first define the action prompts. An action prompt is a modality-aligned pair of an image sub-prompt and a text sub-prompt, where the former is a single-view observation and the latter is an action phrase. The observation indicates a salient visual object or a location. The action phrase contains two main elements, i.e., a word/phrase representing the *action* such as "exit" or "walk into", and a *object/location* word such as "chair" or "bedroom". Fig. 3 shows some examples of the action prompts. From Fig. 3 we can find that an action prompt not only contains an aligned visual object or location in both modalities but also indicates the modality-aligned action knowledge. For example, the paired image sub-prompt of the text sub-prompt "walk out of bedroom" contains the appearance of the bedroom and its door, through which the agent can complete the action of "walk out of" the bedroom. Therefore, by explicitly providing the action prompts into the training, the agent is able to better explore the cross-modal action knowledge which is important for guiding correct action decision. The construction of the action prompt base is described in Sec. 3.3.

### 3.2.3 Action Decision with Action Prompts

At the beginning of the navigation, the agent retrieves instruction-correlated action prompts from the action prompt base. Specifically, the object/location-related action phrases in the given instruction are derived following the strategy for obtaining text sub-prompts (see Sec. 3.3). Then the sentence similarity between each object/location-related action phrase and the text sub-prompts in the prompt base is calculated to retrieve the instruction-related action prompt set $\{p_n\}_{n=1}^N$, where $N$ is the size of the set.

With $\{p_n\}_{n=1}^N$, we obtain the prompt encoding $\{\mathbf{P}_n^{i,u}\}_{n=1}^N$ through the prompt encoder (see Fig. 2). The prompt encoder consists of two single-modal sub-prompt encoders and a multi-modal prompt encoder. Denote the image and text sub-prompts in the action prompt $p_n$ as $p_n^i$ and $p_n^u$, respectively, i.e., $p_n = \{p_n^i, p_n^u\}$. $p_n^i$ and $p_n^u$ are firstly passed through the single-modal sub-prompt en-

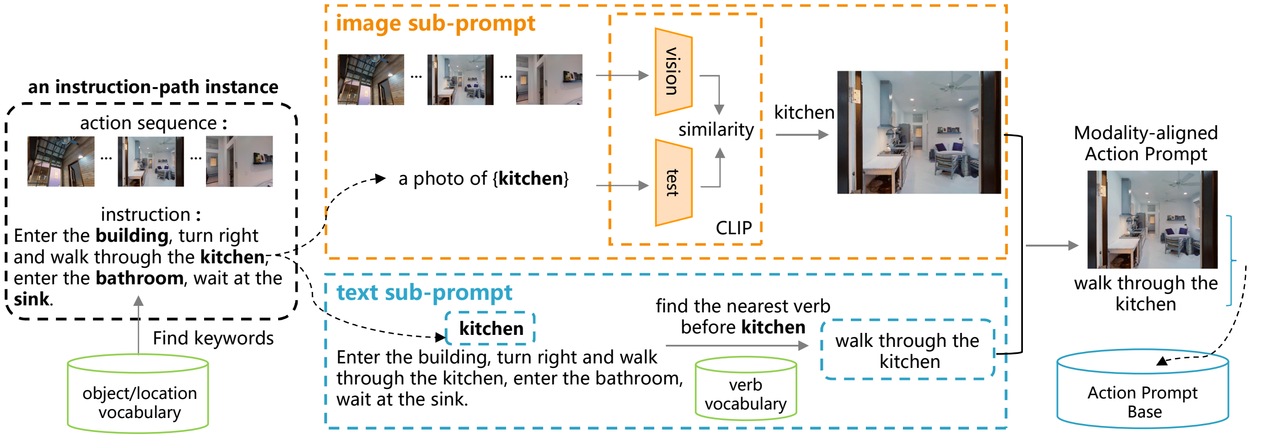


Figure 4. Illustration of action prompt collection for building the action prompt base. Given a training instruction-path instance, the image and text sub-prompts are firstly obtained via CLIP and nearest verb search, respectively. Then the multi-modal sub-prompts related to the same visual object/location and action are aligned to form an action prompt. Here the word "kitchen" is taken as an example.

coders to obtain the sub-prompt features $\mathbf{P}_n^i$ and $\mathbf{P}_n^u$:

$$\mathbf{P}_n^i = \mathrm{E}^i(p_n^i; \theta^i), \tag{5}$$
$$\mathbf{P}_n^u = \mathrm{E}^u(p_n^u; \theta^u), \tag{6}$$

where $\mathrm{E}^i(\cdot)$ with parameters $\theta^i$ and $\mathrm{E}^u(\cdot)$ with parameters $\theta^u$ represent the image sub-prompt encoder and text sub-prompt encoder, respectively. Then $\mathbf{P}_n^i$ and $\mathbf{P}_n^u$ are fed to the multi-modal prompt encoder $\mathrm{E}^p(\cdot)$ to obtain the prompt encoding $\mathbf{P}_n^{i,u}$:

$$\mathbf{P}_n^{i,u} = \mathrm{E}^p(\mathrm{Concat}(\mathbf{P}_n^i, \mathbf{P}_n^u); \theta^p), \tag{7}$$

where $\theta^p$ denotes the parameters of $\mathrm{E}^p(\cdot)$, and $\mathrm{Concat}(\cdot)$ is the concatenation operation. In our ADAPT, the encoders $\mathrm{E}^i(\cdot)$, $\mathrm{E}^u(\cdot)$ and $\mathrm{E}^p(\cdot)$ consists of one linear layer followed by the dropout operation to reduce the over-fitting.

With the prompt encoding $\{\mathbf{P}_n^{i,u}\}$ and the instruction encoding $\mathbf{X}$, we obtain the prompt-based instruction feature $\mathbf{X}^p$ by simply concatenating $\mathbf{X}$ and $\{\mathbf{P}_n^{i,u}\}$. Then the state-visual feature $\mathbf{K}_t$ is updated based on the cross-modal attention $\alpha_t^p$ between $\mathbf{K}_t$ and $\mathbf{X}^p$:

$$\tilde{\mathbf{K}}_t^p, \alpha_t^p = \mathrm{CrossAttn}(\mathbf{K}_t, \mathbf{X}^p; \theta_c). \tag{8}$$

$\alpha_t^p$ is then split to $\alpha_t^{p_1}$ and $\alpha_t^{p_2}$ for obtaining different attended features. Concretely, the attended instruction feature $\tilde{\mathbf{X}}_t$ is derived via weighting $\mathbf{X}$ by $\alpha_t^{p_1}$. The attended image sub-prompt feature $\tilde{\mathbf{P}}_t^i$ and the attended text sub-prompt feature $\tilde{\mathbf{P}}_t^u$ are obtained through weighting $\mathbf{P}_n^i$ and $\mathbf{P}_n^u$ by $\alpha_t^{p_2}$. $\tilde{\mathbf{P}}_t^i$ and $\tilde{\mathbf{P}}_t^u$ are used for calculating the sequential consistency loss $\mathcal{L}_c$. $\tilde{\mathbf{X}}_t$ is used for updating the state feature like the baseline agent. Finally, the prompt-based action prediction probability $\beta_t^p$ is obtained by feeding $\tilde{\mathbf{K}}_t^p$ into the self-attention module like that in Eq. 4.

### 3.3. Construction of the Action Prompt Base

Although it is easy to assign an object category label to an image through object recognition, associating an image with an action phrase is not straightforward. To better align the image and the action phrase to form the action prompt, we design a two-branch scheme to collect the image and text sub-prompts, as shown in Fig. 4. At first, for an instruction-path instance in the training dataset, we use a pre-constructed visual object/location vocabulary to find the referred visual objects/locations in the instruction. Then for each visual object/location, we obtain the related image and text sub-prompts separately as described below.

Note that the ground-truth path sequence contains a set of single-view images, each of which indicates an action needed to make at the specific timestep. Therefore, for deriving the image sub-prompt in an action prompt, we only need to retrieve the object/location-related image from the ground-truth path sequence, which itself contains the action information. Instead of resorting to existing object classifiers or detectors trained on a fixed set of class categories [12, 34], we use CLIP [32] which shows excellent zero-shot cross-modal alignment ability to locate the object/location-related image. To adapt to the inference process of CLIP, we replace the {CLASS} token in the phrase "a photo of {CLASS}" with the visual object/location whose category label is $c$. The probability that an image $\boldsymbol{B}$ in the action sequence belongs to the class $c$ is calculated by:

$$p(y = c|\boldsymbol{B}) = \frac{\exp(\mathrm{sim}(\boldsymbol{b}, \boldsymbol{w}_c)/\tau_1)}{\sum_{i=1}^{M}(\exp(\mathrm{sim}(\boldsymbol{b}, \boldsymbol{w}_i))/\tau_1)}, \tag{9}$$

where $\tau_1$ is the temperature parameter, sim represents the cosine similarity, $\boldsymbol{b}$ and $\boldsymbol{w}_c$ are the image and phrase features generated by CLIP, respectively, and $M$ is the size of the vocabulary. Then the image having the maximum similarity with the phrase is selected as the image sub-prompt.

For obtaining the text sub-prompt, we use a simple nearest-verb-search scheme, that is, finding the nearest verb (which is in a pre-built verb vocabulary) just before a specific object/location word. As shown in Fig. 4, for the word “kitchen”, the verb “walk” is located and then the phrase “walk through the kitchen” is extracted as the text sub-prompt. Finally, the image and text sub-prompts with the same visual object/location and action are formed as an aligned action prompt.

### 3.4. Training and Inference

**Modality Alignment Loss.** While an action prompt has the matched image and text sub-prompts, they may not be aligned in the feature space. To address this problem, following the contrastive learning paradigm used in CLIP [32] that enforces paired image and text features to be similar and non-paired ones to be distant, we use the infoNCE loss [5] to encourage the feature alignment of the image and text sub-prompts in each action prompt:

$$\mathcal{L}_a = -\log\left(\frac{e^{\mathrm{sim}(\mathbf{P}_n^i,\mathbf{P}_n^u)/\tau_2}}{e^{\mathrm{sim}(\mathbf{P}_n^i,\mathbf{P}_n^u)/\tau_2} + \sum_{\overline{\mathbf{P}}_n^u} e^{\mathrm{sim}(\mathbf{P}_n^i,\overline{\mathbf{P}}_n^u)/\tau_2}}\right), \quad (10)$$

where $\tau_2$ is the temperature parameter, $\mathbf{P}_n^i$ and $\mathbf{P}_n^u$ represent the features of the paired image and text sub-prompts of action prompt $p_n$, and $\mathbf{P}_n^i$ and $\overline{\mathbf{P}}_n^u$ denote the non-paired sub-prompts. Through the modality alignment loss, the action prompts can become more discrminative for guiding the learning of action-level modality alignment.

**Sequential Consistency Loss.** Since an instruction usually refers to different visual landmarks sequentially, the action prompts in the retrieved action prompt set $\{p_n\}$ are also related to different objects/locations. To encourage the agent to focus on related action prompts in the retrieved prompt set sequentially according to its visual observations, we develop a sequential consistency loss which is the sum of two single-modal consistency losses. Take the text modality as an example, at each timestep $t$, the attended text sub-prompt feature $\tilde{\mathbf{P}}_t^u$ and the attended instruction feature $\tilde{\mathbf{X}}_t$ are enforced to be close:

$$\mathcal{L}_c^u = ||\tilde{\mathbf{P}}_t^u - \tilde{\mathbf{X}}_t||^2. \quad (11)$$

Similarly, define $\mathcal{L}_c^i = ||\tilde{\mathbf{P}}_t^i - \tilde{\mathbf{V}}_t||^2$, which is used to promote the similarity between the attended image sub-prompt feature $\tilde{\mathbf{P}}_t^i$ and the attended visual feature $\tilde{\mathbf{V}}_t$. Then the sequential consistency loss $\mathcal{L}_c$ is obtained by:

$$\mathcal{L}_c = \mathcal{L}_c^i + \mathcal{L}_c^u. \quad (12)$$

**Total Objective.** Following most of existing works [13, 14, 38], we also use the navigation loss $\mathcal{L}_n$, which is the sum of an imitation loss $\mathcal{L}_{IL}$ and a reinforcement learning loss $\mathcal{L}_{RL}$. Thus, the total training objective of our ADAPT is:

$$\mathcal{L} = \mathcal{L}_{RL} + \lambda_1\mathcal{L}_{IL} + \lambda_2\mathcal{L}_c + \lambda_3\mathcal{L}_a, \quad (13)$$

where $\lambda_1$, $\lambda_2$, and $\lambda_3$ are the loss weights to balance the loss items.

**Inference.** During inference, the agent retrieves instruction-related action prompts from the action prompt base built in the training stage.

## 4. Experiments

### 4.1. Experimental Setup

**Datasets.** We evaluate ADAPT on two public benchmarks, i.e., R2R [1] and RxR [19]. R2R [1] includes 10,800 panoramic views and 7,189 trajectories. Since the baseline [14] is pretrained on English language data, we test our ADAPT on the English subset of RxR (both en-IN and en-US), which includes 26,464 path-instruction pairs for training and 4,551 pairs in the val-unseen split.

**Evaluation Metrics.** We use four popular metrics [1] for the performance evaluation on R2R: 1) Trajectory Length (TL) calculates the average length of the trajectory, 2) Navigation Error (NE) is the distance between target viewpoint and agent stopping position, 3) Success Rate (SR) calculates the success rate of reaching the goal, and 4) Success rate weighted by Path Length (SPL) makes the tradeoff between SR and TL. Three more metrics are used for RxR following other works [19, 22]: Coverage weighted by Length Score (CLS) [17], Normalized Dynamic Time Warping (nDTW) [16], and Success rate weighted normalized Dynamic Time Warping (SDTW) [16].

**Implementation Details.** All experiments are conducted on an NVIDIA V100 GPU. Two kinds of image features are used, i.e., the features extracted from a ResNet-152 [12] pretrained on Places365 [44] and the features extracted through the visual encoder of CLIP [35]. The model is trained for 300K and 100K iterations for R2R and RxR, respectively. The max sizes of the action prompt set are 60 and 100 for R2R and RxR, respectively. The instance whose number of retrieved action prompts less than the max size is padded. The values of $\lambda_1$, $\lambda_2$, and $\lambda_3$ are 0.2, 0.01, and 0.0001, respectively. The same augmented data in [14] is used for R2R for fair comparison.

### 4.2. Quantitative Results

**Comparison with the State-of-the-Arts (SOTAs).** Table 1 and Table 2 give the comparison between existing methods and our ADAPT. Table 1 shows that ADAPT with ResNet-152 feature outperforms previous SOTA methods on RxR. Moreover, ADAPT significantly improves the performance of the baseline [14] with different visual features in both Val Seen and Val Unseen settings on RxR, showing that introducing explicit action prompts can effectively promote the agent navigation capability. From Table 2 we can

Table 1. Comparison with the SOTA methods on RxR. * indicates that the results are obtained by our re-implementation of the model.

| Method | Model | RxR Val Seen | | | | | RxR Val Unseen | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | SR↑ | SPL↑ | CLS↑ | nDTW↑ | SDTW↑ | SR↑ | SPL↑ | CLS↑ | nDTW↑ | SDTW↑ |
| EnvDrop [38] | ResNet-152 | 48.1 | 44 | 61 | 57 | 40 | 38.5 | 34 | 54 | 51 | 32 |
| Syntax [22] | | 48.1 | 44 | 61 | 58 | 40 | 39.2 | 35 | 56 | 52 | 32 |
| VLN⟳BERT* [14] | | 50.9 | 45.4 | 60.3 | 56.9 | 41.3 | 45.5 | 39.3 | 56.6 | 52.9 | 36.3 |
| ADAPT (ours) | | **52.7** | **47.0** | **61.3** | **58.5** | **42.9** | **46.7** | **40.3** | **56.6** | **53.6** | **37.3** |
| VLN⟳BERT* [14] | CLIP | 48.6 | 43.4 | 58.8 | 55.7 | 39.8 | 45.7 | 39.5 | 56.0 | 52.8 | 36.7 |
| ADAPT (ours) | | **50.3** | **44.6** | **59.6** | **56.3** | **40.6** | **46.9** | **40.2** | **57.2** | **54.1** | **37.7** |

Table 2. Comparison with the SOTA methods on R2R. * indicates that the results are obtained by our re-implementation of the model.

| Method | Val Seen | | | | Val Unseen | | | | Test Unseen | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | TL | NE ↓ | SR ↑ | SPL ↑ | TL | NE ↓ | SR ↑ | SPL ↑ | TL | NE ↓ | SR ↑ | SPL ↑ |
| Seq2Seq [1] | 11.33 | 6.01 | 39 | - | 8.39 | 7.81 | 22 | - | 8.13 | 7.85 | 20 | 18 |
| Speaker-Follower [8] | - | 3.36 | 66 | - | - | 6.62 | 35 | - | 14.82 | 6.62 | 35 | 28 |
| EnvDropout [38] | 11.00 | 3.99 | 62 | 59 | 10.70 | 5.22 | 52 | 48 | 11.66 | 5.23 | 51 | 47 |
| PREVALENT [11] | 10.32 | 3.67 | 69 | 65 | 10.19 | 4.71 | 58 | 53 | 10.51 | 5.30 | 54 | 51 |
| VLN⟳BERT [14] | 11.13 | 2.90 | 72 | 68 | 12.01 | 3.93 | 63 | 57 | 12.35 | 4.09 | 63 | 57 |
| ADAPT (ResNet-152) | 10.97 | **2.54** | **76** | **72** | 12.21 | **3.77** | **64** | **58** | 12.99 | **3.79** | **65** | **59** |
| VLN⟳BERT* (CLIP) | 11.37 | 3.17 | 70 | 66 | 12.03 | 3.81 | 65 | 58 | 12.73 | 4.26 | 61 | 55 |
| ADAPT (CLIP) | 11.39 | **2.70** | **74** | **69** | 12.33 | **3.66** | **66** | **59** | 13.16 | **4.11** | **63** | **57** |

Table 3. Ablation study of ADAPT on R2R Val Unseen. ResNet-152 and CLIP represent using different visual features. ADAPT-1: using action prompts only; ADAPT-2: using action prompts with the modality alignment loss; ADAPT-3: using action prompts with the sequential consistency loss; ADAPT-Full: our full model. All models are trained for 100K iterations.

| Method | ResNet-152 | | | CLIP | | |
|---|---|---|---|---|---|---|
| | NE ↓ | SR ↑ | SPL | NE ↓ | SR ↑ | SPL ↑ |
| Baseline | 4.17 | 60.4 | 54.7 | 4.11 | 61.5 | 55.3 |
| ADAPT-1 | 4.19 | 60.5 | 55.2 | 3.90 | 61.6 | 56.0 |
| ADAPT-2 | 4.16 | 61.7 | 55.4 | **3.78** | 62.8 | 56.3 |
| ADAPT-3 | 4.07 | 60.7 | 56.1 | 4.05 | 61.9 | 56.6 |
| ADAPT-Full | **4.07** | **62.5** | **56.1** | 4.10 | **63.1** | **57.2** |

see that ADAPT (ResNet-152) establishes new SOTA results on R2R. Moreover, from the results of VLN⟳BERT* (CLIP) and ADAPT (CLIP) we can find that by introducing the CLIP visual feature, both models show a performance enhancement in Val Unseen while a performance drop in both Val Seen and Test Unseen. However, ADAPT (CLIP) outperforms VLN⟳BERT* (CLIP) on all the metrics, showing the effectiveness of the proposed method.

**Ablation Study.** Table 3 presents the ablation study results of ADAPT. As shown in Table 3, explicitly introducing the action prompts can effectively improve the performance of the strong baseline model [14]. By comparing the results between "ADAPT-1" and "ADAPT-2" we can find that introducing the modality alignment loss can effectively enhance the navigation performance, demonstrating that the action prompts with good discriminative power are useful for learning better action-level modality alignment. Comparing the results between "ADAPT-2" and "ADAPT-Full", we can see that the introduction of the sequential consistency loss further improves the navigation performance, which shows that attending to related action prompts sequentially is helpful for making correct action decision.

To verify the generalization ability of ADAPT when a small amount of training data is available, we set up two training settings: "Scan" and "Instance". "Scan" means that extracting part of the training scans with all the instances for training. "Instance" means that extracting all the training scans but with part of the instances for training. From the evaluation results given in Table 4, we can find that in both "Scan" and "Instance" settings, our ADAPT is superior over the strong baseline method, showing that by learning explicit action knowledge, the agent can have better generalization ability in different scenes.

## 4.3. Visualization

We present some visualization results in this subsection to further analyze how introducing the explicit action prompts can contribute to correct navigation action decision. From Fig. 5 we can see that by introducing the action prompts related to "walk around the bed" and "walk into the hallway" in the instruction, our ADAPT can successful enforce the agent to choose the correct actions of walking around the bed and walking into the hallway in different visual observations. The baseline agent, however, leaves the original room and makes the wrong navigation trajectory.

We further validate the action-level modality alignment ability of ADAPT by comparing the action prompt alignment between the CLIP features and the sub-prompt features of ADAPT. For the action phrase feature, the top 5 similar image features are retrieved from the object-related image set. From Fig. 6 we can find that compared with CLIP, ADAPT can perform better action-level modality alignment. Given the action phrase of "walk up the stairs", the top 5 results retrieved by CLIP from a set of stairs images all indicate the action of "walk down" the stairs. Our ADAPT, however, can obtain 3 images indicating the action of "walk up" the stairs in the top 5 results.

Table 4. Results of the baseline [14] and our ADAPT on R2R Val Unseen with fewer training data. * indicates that the results are obtained by our re-implementation of the model.

| Model | Scan | | | | | | | | Instance | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 20% | | 40% | | 60% | | 80% | | 20% | | 40% | | 60% | | 80% | |
| | SR↑ | SPL↑ | SR↑ | SPL↑ | SR↑ | SPL↑ | SR↑ | SPL↑ | SR↑ | SPL↑ | SR↑ | SPL↑ | SR↑ | SPL↑ | SR↑ | SPL↑ |
| VLN↻BERT* [14] | 50.8 | 44.0 | 53.7 | 48.1 | **57.7** | 51.7 | 57.4 | 53.1 | 51.3 | 47.0 | 55.8 | 49.7 | 57.1 | 52.1 | 57.9 | 52.7 |
| ADAPT (ours) | **52.5** | **46.4** | **55.1** | **48.8** | 57.2 | **51.8** | **59.1** | **53.3** | **52.5** | **47.3** | **56.6** | **49.8** | **58.8** | **53.5** | **59.4** | **54.6** |


Figure 5. Visualization of panoramic views and action comparison in a trajectory example between the baseline [14] and our ADAPT.


Figure 6. Action prompt alignment comparison between the CLIP features and the sub-prompt features of our ADAPT.

## 5. Conclusion and Limitation

In this work, we propose modality-aligned action prompts (ADAPT), which prompts the VLN agent with explicit cross-modal action knowledge for enhancing the navigation performance. During navigation, the agent retrieves the action prompts from a pre-built action prompt base. Then the prompt-based instruction features are obtained for improving action decision. The CLIP model is used to collect high-quality action prompts into the prompt base. We also propose a modality alignment loss and a sequential consistency loss for training. Experiments on the public VLN benchmarks show the effectiveness of our ADAPT, which establishes new SOTA results. We hope this work can offer new directions for prompt-based navigation research.

With regards to the limitation of our work, our constructed action prompt base in ADAPT contains more or less noise due to the ability of CLIP, the scene complexity and instruction diversity in the VLN task. The future work includes finding action prompts of better quality.

## Acknowledgement

This work was supported in part by National Key R&D Program of China under Grant No. 2020AAA0109700, National Natural Science Foundation of China (NSFC) No.61976233, Guangdong Province Basic and Applied Basic Research (Regional Joint Fund-Key) Grant No.2019B1515120039, Guangdong Outstanding Youth Fund (Grant No. 2021B1515020061), Shenzhen Fundamental Research Program (Project No. RCYX20200714114642083, No. JCYJ20190807154211365), National Natural Science Foundation of China under Grant No.62006255. We thank MindSpore for the partial support of this work, which is a new deep learning computing framework[1].

[1] https://www.mindspore.cn/

## References


[1] Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sunderhauf, 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 (CVPR)*, pages 3674–3683, 2018. 1, 2, 6, 7

[2] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 2425–2433, 2015. 2

[3] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Proceedings of the Conference on Neural Information Processing Systems (NeurIPS)*, volume 33, pages 1877–1901, 2020. 2

[4] Howard Chen, Alane Suhr, Dipendra Misra, Noah Snavely, and Yoav Artzi. Touchdown: Natural language navigation and spatial reasoning in visual street environments. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12538–12547, 2019. 1

[5] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2020. 6

[6] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Universal image-text representation learning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 104–120, 2020. 1, 2

[7] Zhiwei Deng, Karthik Narasimhan, and Olga Russakovsky. Evolving graphical planner: Contextual global planning for vision-and-language navigation. In *Proceedings of the Conference on Neural Information Processing Systems (NeurIPS)*, volume 33, pages 20660–20672, 2020. 1

[8] Daniel Fried, Ronghang Hu, Volkan Cirik, Anna Rohrbach, Jacob Andreas, Louis-Philippe Morency, Taylor Berg-Kirkpatrick, Kate Saenko, Dan Klein, and Trevor Darrell. Speaker-follower models for vision-and-language navigation. In *Proceedings of the Conference on Neural Information Processing Systems (NeurIPS)*, pages 3314–3325, 2018. 1, 2, 7

[9] 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 *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 71–86, 2020. 1

[10] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. *arXiv preprint arXiv:2104.13921*, 2021. 3

[11] 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 Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13137–13146, 2020. 1, 2, 7

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. 5, 6

[13] Yicong Hong, Cristian Rodriguez, Yuankai Qi, Qi Wu, and Stephen Gould. Language and visual entity relationship graph for agent navigation. In *Proceedings of the Conference on Neural Information Processing Systems (NeurIPS)*, volume 33, 2020. 1, 6

[14] 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 Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1643–1653, 2021. 1, 2, 4, 6, 7, 8

[15] Haoshuo Huang, Vihan Jain, Harsh Mehta, Alexander Ku, Gabriel Magalhaes, Jason Baldridge, and Eugene Ie. Transferable representation learning in vision-and-language navigation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 7404–7413, 2019. 1

[16] Gabriel Ilharco, Vihan Jain, Alexander Ku, Eugene Ie, and Jason Baldridge. General evaluation for instruction conditioned navigation using dynamic time warping. *ViGIL@NeurIPS*, 2019. 6

[17] 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 Association for Computational Linguistics (ACL)*, pages 1862–1872, 2019. 6

[18] Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. How can we know what language models know. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020. 2

[19] 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 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4392–4412, 2020. 2, 6

[20] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*, 2021. 2, 3

[21] Gen Li, Nan Duan, Yuejian Fang, Ming Gong, and Daxin Jiang. Unicoder-vl: A universal encoder for vision and language by cross-modal pre-training. In *Proceedings of the Association for the Advance of Artificial Intelligence (AAAI)*, volume 34, pages 11336–11344, 2020. 1, 2

[22] Jialu Li, Hao Tan, and Mohit Bansal. Improving cross-modal alignment in vision language navigation via syntactic infor-

mation. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 1041–1050, 2021. 6, 7

[23] Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*, 2019. 1, 2

[24] Xiujun Li, Chunyuan Li, Qiaolin Xia, Yonatan Bisk, Asli Çelikyilmaz, Jianfeng Gao, Noah A. Smith, and Yejin Choi. Robust navigation with language pretraining and stochastic sampling. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1494–1499, 2019. 1

[25] Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, Yejin Choi, and Jianfeng Gao. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 121–137, 2020. 1, 2

[26] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the Association for Computational Linguistics (ACL)*, pages 4582–4597, 2021. 2, 3

[27] 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 International Conference on Computer Vision (ICCV)*, pages 1644–1654, 2021. 1

[28] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *arXiv preprint arXiv:2107.13586*, 2021. 2

[29] 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. 1

[30] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. Styleclip: Text-driven manipulation of stylegan imagery. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 2085–2094, 2021. 3

[31] 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 International Conference on Computer Vision (ICCV)*, pages 1655–1664, 2021. 2

[32] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 8748–8763, 2021. 2, 3, 5, 6

[33] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020. 2

[34] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1137–1149, 2017. 5

[35] Sheng Shen, Liunian Harold Li, Hao Tan, Mohit Bansal, Anna Rohrbach, Kai-Wei Chang, Zhewei Yao, and Kurt Keutzer. How much can clip benefit vision-and-language tasks? *arXiv preprint arXiv:2107.06383*, 2021. 3, 4, 6

[36] Taylor Shin, Yasaman Razeghi, Robert L. Logan, Eric Wallace, and Sameer Singh. Autoprompt: Eliciting knowledge from language models with automatically generated prompts. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4222–4235, 2020. 2

[37] Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. Vl-bert: Pre-training of generic visual-linguistic representations. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020. 1, 2

[38] Hao Tan, Licheng Yu, and Mohit Bansal. Learning to navigate unseen environments: Back translation with environmental dropout. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 2610–2621, 2019. 1, 2, 6, 7

[39] Maria Tsimpoukelli, Jacob Menick, Serkan Cabi, S. M. Ali Eslami, Oriol Vinyals, and Felix Hill. Multimodal few-shot learning with frozen language models. *arXiv preprint arXiv:2106.13884*, 2021. 2, 3

[40] 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 Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6629–6638, 2019. 1

[41] Yuan Yao, Ao Zhang, Zhengyan Zhang, Zhiyuan Liu, Tat-Seng Chua, and Maosong Sun. Cpt: Colorful prompt tuning for pre-trained vision-language models. *arXiv preprint arXiv:2109.11797*, 2021. 3

[42] Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. From recognition to cognition: Visual commonsense reasoning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6720–6731, 2019. 2

[43] Zexuan Zhong, Dan Friedman, and Danqi Chen. Factual probing is [mask]: Learning vs. learning to recall. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 5017–5033, 2021. 2, 3

[44] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database

for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(6):1452–1464, 2018. 6

[45] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *arXiv preprint arXiv:2109.01134*, 2021. 2, 3

[46] Fengda Zhu, Yi Zhu, Xiaojun Chang, and Xiaodan Liang. Vision-language navigation with self-supervised auxiliary reasoning tasks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10012–10022, 2020. 1, 2

[47] 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 *ACL*, 2020. 1