Contrastive Instruction-Trajectory Learning for Vision-Language Navigation

Xiwen Liang[1], Fengda Zhu[2], Yi Zhu[3], Bingqian Lin[1], Bing Wang[4], Xiaodan Liang[1]*

[1]Shenzhen Campus of Sun Yat-sen University, Shenzhen
[2]Monash University
[3]Huawei Noah's Ark Lab
[4]Alibaba Group
liangxw29@mail2.sysu.edu.cn, fengda.zhu@monash.edu, zhu.yee@outlook.com, linbq6@mail2.sysu.edu.cn, fengquan.wb@alibaba-inc.com, liangxd9@mail.sysu.edu.cn

Abstract

The vision-language navigation (VLN) task requires an agent to reach a target with the guidance of natural language instruction. Previous works learn to navigate step-by-step following an instruction. However, these works may fail to discriminate the similarities and discrepancies across instruction-trajectory pairs and ignore the temporal continuity of sub-instructions. These problems hinder agents from learning distinctive vision-and-language representations, harming the robustness and generalizability of the navigation policy. In this paper, we propose a Contrastive Instruction-Trajectory Learning (CITL) framework that explores invariance across similar data samples and variance across different ones to learn distinctive representations for robust navigation. Specifically, we propose: (1) a coarse-grained contrastive learning objective to enhance vision-and-language representations by contrasting semantics of full trajectory observations and instructions, respectively; (2) a fine-grained contrastive learning objective to perceive instructions by leveraging the temporal information of the sub-instructions; (3) a pairwise sample-reweighting mechanism for contrastive learning to mine hard samples and hence mitigate the influence of data sampling bias in contrastive learning. Our CITL can be easily integrated with VLN backbones to form a new learning paradigm and achieve better generalizability in unseen environments. Extensive experiments show that the model with CITL surpasses the previous state-of-the-art methods on R2R, R4R, and RxR. Code is available at https://github.com/liangcici/CITL-VLN.

Introduction

Vision-Language Navigation (VLN) task (Anderson et al. 2018b) requires an agent to navigate following a natural language instruction. This task is closely connected to many real-world applications, such as household robots and rescue robots (Zhu et al. 2021). The VLN task is challenging since it requires an agent to acquire diverse skills, such as vision-language alignment, sequential vision perception and long-term decision making.

Early method (Anderson et al. 2018b) is developed upon an encoder-decoder framework (Sutskever, Vinyals, and Le 2014). Later methods (Fried et al. 2018; Wang et al. 2019b; Zhu et al. 2020a; Ke et al. 2019; Ma et al. 2019a) improve the agent with vision-language attention layers and auxiliary tasks. Coupling with BERT-like methods (Devlin et al. 2019; Lu et al. 2019; Li et al. 2020), the navigation agent obtains better generalization ability (Majumdar et al. 2020; Hong et al. 2021). However, these VLN methods only use the context within an instruction-trajectory pair while ignoring the knowledge across the pairs. For instance, they only recognize the correct actions that follow the instruction while ignoring the actions that do not follow the instruction. The differences between the correct actions and the wrong actions contain extra knowledge for navigation. On the other hand, previous methods do not explicitly exploit the temporal continuity inside an instruction, which may fail if the agent focuses on a wrong sub-instruction. Thus, learning a fine-grained sub-instruction representation by leveraging the temporal continuity of sub-instructions could improve the robustness of navigation.

Recently, self-supervised contrastive learning shows superior capacity in improving the instance discrimination and generalization of vision models (Chen et al. 2020; He et al. 2020; Xie et al. 2021; Li et al. 2021; Sun et al. 2019). Inspired by the success of contrastive learning, we propose our Contrastive Instruction-Trajectory Learning (CITL) framework to explore fine/coarse-grained knowledge of the instruction-trajectory pairs. Our CITL consists of two coarse-grained trajectory-instruction contrastive objectives and a fine-grained sub-instruction contrastive objective to learn from cross-instance trajectory-instruction pairs and sub-instructions. Firstly, we propose **coarse-grained contrastive learning** to learn distinctive long-horizon representations for trajectories and instructions respectively. The idea of coarse-grained contrastive learning is computing inter-intra cross-instance contrast: enforcing embedding to be similar for positive trajectory-instruction pairs and dissimilar for intra-negative and inter-negative ones. To obtain positive samples, we propose data augmentation methods for instructions and trajectories respectively. Intra-negative samples are generated through changing the temporal information of the instruction and selecting longer sub-optimal trajectories which deviate from the anchor one severely. In contrast, inter-negative samples are different trajectory-instruction pairs. In this way, the semantics of full trajectory observations and instructions can be captured for better

*Corresponding author.

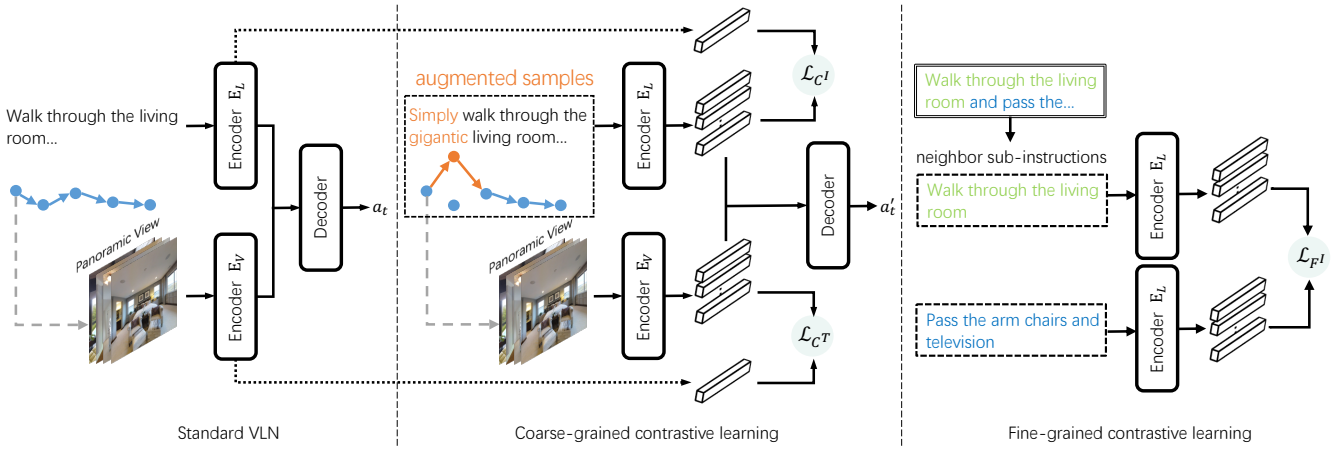

Figure 1: We propose contrastive instruction-trajectory learning for VLN (middle and right). Most previous works use a follower model to produce actions in VLN (left), neglecting different transformations of the instruction-trajectory pairs. As a result, representations may be variant with similar instruction-trajectory pairs. By contrast, our CITL learns representations similar to the input instruction-trajectory pair with different transformations (middle) and retains rich semantic information. Our CITL also leverages the temporal continuity of the sub-instructions (right).

shaping representations with less variance under diverse data transformations. Secondly, we propose **fine-grained contrastive learning** to learn fine-grained representations by focusing on the temporal information of sub-instructions. We generate sub-instructions as in (Hong et al. 2020a), and train the agent to learn embedding distances of these sub-instructions by contrastive learning. Specifically, neighbor sub-instructions are positive samples, while non-neighbor sub-instructions are intra-negative samples and different sub-instructions from other instructions are inter-negative samples. These learning objectives help the agent leverage richer knowledge to learn better embedding for instructions and trajectories, and therefore, obtain a more robust navigation policy and better generalizability. Fig. 1 shows an overview of our CITL framework.

We also overcome several challenges in adopting contrastive learning in VLN by introducing **pairwise sample-reweighting mechanism**. Firstly, a large scale of easy samples dominates the gradient, causing the performance to plateau quickly. Some false-negative samples may exist and introduce noise. To avoid these problems, we introduce pair mining to mine hard samples and remove false-negative ones online, making the model focuses on hard samples during training. Secondly, the generated positive trajectories may be close to or heavily deviate from the anchor one. Previous multi-pair contrastive learning methods (Xie et al. 2020; Cai et al. 2020) adopt InfoNCE loss (Oord, Li, and Vinyals 2018), which fails to explicitly penalize samples differently. Therefore we introduce the circle loss (Sun et al. 2020) to penalize different positive and negative samples.

Our experiments demonstrate that our CITL framework can be easily combined with different VLN models and significantly improves their navigation performance (2%-4% in terms of SPL in R2R and R4R). Our ablation studies show that CITL helps the model learn more distinct knowledge with different data transformations since coarse/fine-grained contrastive objectives introduce cross-instance long-horizon information and intra-instance fine-grained information.

Related Work

Vision-and-Language Navigation Learning navigation with vision-language clues has attracted a lot of attention of researchers. Room-to-Room (R2R) (Anderson et al. 2018b) and Touchdown (Chen et al. 2019) datasets introduce natural language and photo-realistic environment for navigation. Following this, dialog-based navigation, such as VNLA (Nguyen et al. 2019), HANNA (Nguyen and Daumé III 2019) and CVDN (Thomason et al. 2019), is proposed for further research. REVERIE (Qi et al. 2020b) introduces the task of localizing remote objects. A number of methods have been proposed to solve VLN. Speaker-Follower (Fried et al. 2018) introduces a speaker model and a panoramic representation to expand the limited data. Similarly, EnvDrop (Tan, Yu, and Bansal 2019) proposes a back-translation method to learn on augmented data. In (Ke et al. 2019), an asynchronous search combined with global and local information is adopted to decide whether the agent should backtrack. To align the visual observation and the partial instruction better, a visual-textual co-grounding module is proposed in (Ma et al. 2019a; Wang et al. 2019b). Progress monitor and other auxiliary losses are proposed in (Ma et al. 2019a,b; Zhu et al. 2020a; Qi et al. 2020a; Wang, Wu, and Shen 2020). RelGraph (Hong et al. 2020b) develops a language and visual relationship graph to model inter/intra-modality relationships. PRESS (Li et al. 2019) applies the pre-trained BERT (Devlin et al. 2019) to process instructions. RecBERT (Hong et al. 2021) further implements a recurrent function based on ViLBERT. However, current VLN methods only focus on individual instruction-trajectory pairs and ignore the invariance of different data transformations. As a result, representations may be variant with similar instruction-trajectory pairs.

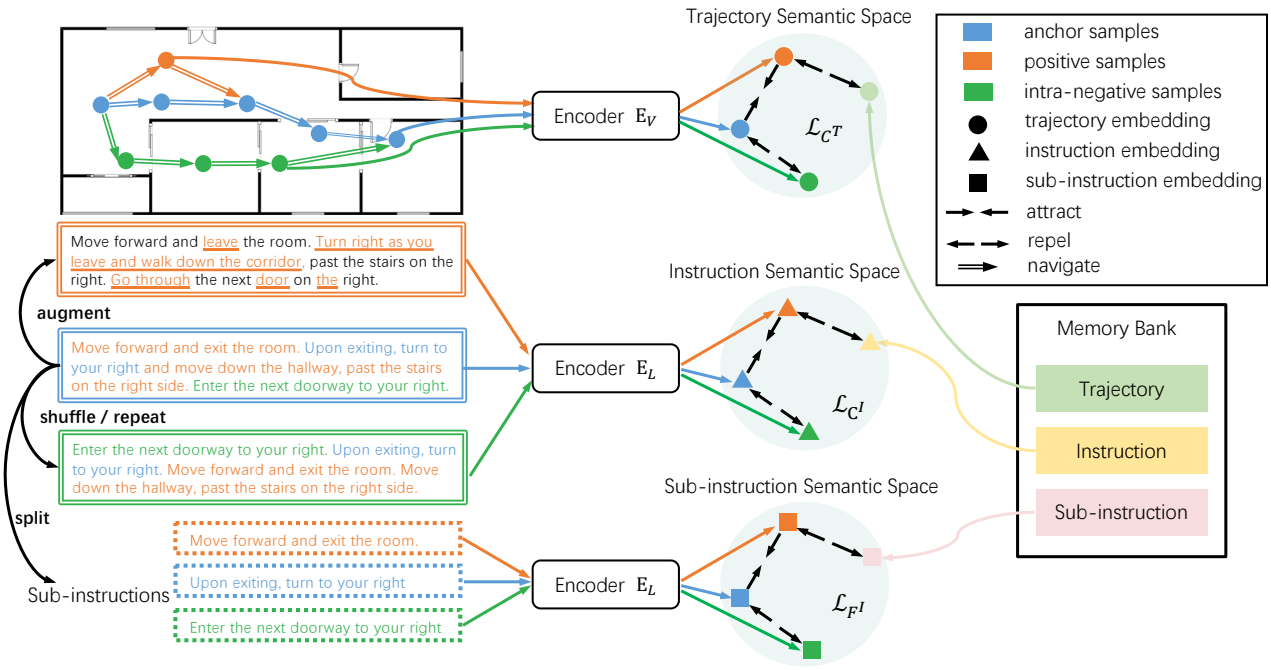

Figure 2: Overview of our coarse-fine contrastive learning-based vision-and-language navigation framework. Both trajectories and instructions are utilized for contrastive learning. Here trajectories are augmented by generating sub-optimal ones, and intra-negative trajectories deviate far away from the anchor one. Positive instructions are produced by back translation, inserting meaningful words and substituting with synonyms. Positive sub-instructions are neighboring to the anchor one. $\mathcal{L}_{C^T}$ is the coarse contrastive loss for trajectories, $\mathcal{L}_{C^I}$ represents the coarse contrastive loss for instructions, and $\mathcal{L}_{F^I}$ indicates the fine-grained contrastive loss for sub-instructions.

Contrastive Learning Contrastive loss (Hadsell, Chopra, and Lecun 2006) is adopted to encourage representations to be close for similar samples and distant for dissimilar samples. Recently, state-of-the-art methods on unsupervised representation learning (Wu et al. 2018; He et al. 2020; Grill et al. 2020; Caron et al. 2020; Chen et al. 2020; Chen and He 2021) are based on contrastive learning. Most methods adopt different transformations of an image as similar samples as in (Dosovitskiy et al. 2014). Similar to contrastive loss, mutual information (MI) is maximized in (Oord, Li, and Vinyals 2018; Henaff 2020; Hjelm et al. 2019; Bachman, Hjelm, and Buchwalter 2019) to learn representations. In (Hjelm et al. 2019; Bachman, Hjelm, and Buchwalter 2019; Henaff 2020), MI is maximized between global and local features from the encoder. (Chaitanya et al. 2020) integrates knowledge of medical imaging to define positive samples and focuses on distinguishing different areas in an image. Memory bank (Wu et al. 2018) and momentum contrast (He et al. 2020) are proposed to use more negative pairs per batch. No work has attempted to life VLN models with the merits of contrastive learning. The success of contrastive learning motivates us to rethink the training paradigm of VLN and design contrastive learning objectives for VLN.

Embedding Losses Contrastive loss (Hadsell, Chopra, and Lecun 2006) is a classic pair-based method in embedding learning. Triplet margin loss (Weinberger, Blitzer, and Saul 2006) is proposed to capture variance in inter-class dissimilarities. Following these works, the margin of angular loss (Wang et al. 2017) is based on angles of triplet vectors. Lifted structure loss (Song et al. 2016) applies LogSumExp, a smooth approximation of the maximum function, to all negative pairs. Softmax function is applied to each positive pair relative to all negative pairs in N-Pairs loss (Sohn 2016; Oord, Li, and Vinyals 2018; Chen et al. 2020). Similarities among each embedding and its neighbors are weighted explicitly or implicitly in (Wang et al. 2019a; Yu and Tao 2019; Sun et al. 2020). Unlike all previous work, our CITL is the first to adopt contrastive learning to learn distinct representations in VLN. Our proposed CITL differs from existing contrastive learning methods in several ways. Firstly, most previous single-modal contrastive learning approaches focus on image-level or pixel-level (Xie et al. 2021) comparison, and cross-modal contrastive learning methods mainly handle image-text pairs (Li et al. 2021) and video-text pairs (Sun et al. 2019), while we focus on trajectory-instruction pairs. Secondly, we introduce a pairwise sample-reweighting mechanism to learn trajectory-instruction representations effectively.

Preliminaries

Vision-Language Navigation

Given a natural language instruction I with a sequence of words, at each time step t, the agent observes a panoramic view, which is divided into 36 single-view images V_t =

$\{v_i\}_{i=1}^{36}$ for the agent to learn. The agent has N_t navigable viewpoint as candidates, whose views from the current point are denoted as $C_t = \{c_i\}_{i=1}^{N_t}$. The agent predicts an action a_t by selecting a viewpoint from C_t to navigate each timestep.

A language encoder E_L and a vision encoder E_V are adopted to encode instructions and viewpoints respectively. The language encoder encodes the instruction I as a global language feature $X \in \mathbb{R}^{N_I}$ and the vision encoder encodes the panoramic views and the candidate views as follows:

$$\begin{aligned} X &= \mathrm{E}_L(I), \\ f_t^v = \mathrm{E}_V(V_t), &\quad f_t^c = \mathrm{E}_V(C_t), \end{aligned} \tag{1}$$

where f_t^v and f_t^c are features of the current viewpoint and candidates. A cross-modal attention function $\mathrm{Attn}(\cdot)$ (Tan and Bansal 2019) is introduced to compute visual attention based on textual information. Then a policy network π is applied to predict action a_t:

$$\begin{aligned} s_t &= \mathrm{Attn}(X, f_t^v, s_{t-1}), \\ a_t &= \pi(s_t, f_t^c). \end{aligned} \tag{2}$$

Contrastive Learning

In contrastive learning, representations of positive and negative samples are extracted with an encoder $\mathrm{E}(\cdot)$ followed by a mapping function $\mathrm{U}(\cdot)$. For example, x_i is one of H positive examples for the anchor sample x. The representation of x_i is denoted by $p_i = \mathrm{U}(\mathrm{E}(x_i))$. For the anchor example x, representation is extracted with the encoder $\mathrm{E}(\cdot)$ followed by a projection $\mathrm{U}(\cdot)$ and a predictor $\mathrm{G}(\cdot)$. Thus, the representation of the anchor x is formulated as $q = \mathrm{G}(\mathrm{U}(\mathrm{E}(x)))$. x_j is one of J negative samples, whose representations are $n_j = \mathrm{U}(\mathrm{E}(x_j))$. Let $\mathcal{P}$ be the set of positive representations and $\mathcal{N}$ be the set of negative representations for each anchor representation q. Then for each anchor q, we have $\mathcal{P} = \{p_i\}_{i=1}^{H}$ and $\mathcal{N} = \{n_j\}_{j=1}^{J}$. Circle loss (Sun et al. 2020) is one of the embedding losses maximizing within-class similarity and minimizing between-class similarity, and meanwhile updating pair weights more accurately. It is formulated as:

$$\mathcal{L}_{\text{circle}} = \log \left[1 + \sum_{j=1}^{J} \exp\left(l_n^j\right) \sum_{i=1}^{H} \exp\left(l_p^i\right) \right], \tag{3}$$

where logits l_n^j and l_p^i are defined as follows:

$$\begin{aligned} l_n^j &= \gamma \left[\mathrm{sim}\left(q, n_j\right) - O_n \right]_+ \left(\mathrm{sim}\left(q, n_j\right) - \Delta_n \right), \\ l_p^i &= -\gamma \left[O_p - \mathrm{sim}\left(q, p_i\right) \right]_+ \left(\mathrm{sim}\left(q, p_i\right) - \Delta_p \right), \end{aligned} \tag{4}$$

where γ is a scale factor, $+$ is a cut-off at zero operation, and sim computes the cosine similarity. O_n, Δ_n, O_p and Δ_p are set as $-m$, m, $1+m$ and $1-m$ respectively, where m is the margin for similarity separation. If the similarity score deviates severely from its optimum (O_p for positive pairs and O_n for negative pairs), it will get a larger weighting factor.

CITL

In this section, we propose our Contrastive Instruction-trajectory Learning (CITL), consisting of coarse-fine contrastive objectives and a pairwise sample-reweighting mechanism. Fig. 2 shows the framework of our CITL.

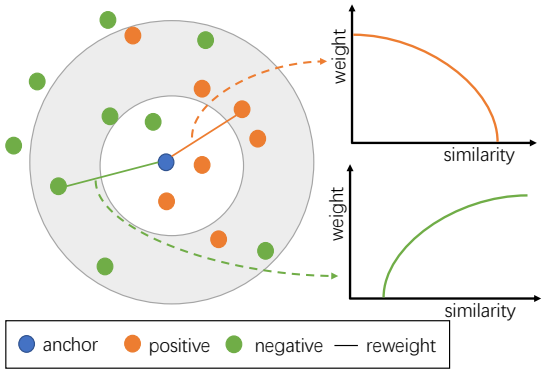

Figure 3: Illustration of the pairwise sample-reweighting mechanism. Samples distributed in white areas are easy samples or false-negative samples. Correlations of similarities and weights in the loss are shown at the right.

Coarse-grained Contrastive Learning

Our coarse-grained contrastive learning consists of two contrastive objectives: 1) coarse contrastive loss for trajectories $\mathcal{L}_{C^T}$ and 2) coarse contrastive loss for instructions $\mathcal{L}_{C^I}$.

Trajectory Loss The optimal trajectory is the shortest path from the starting position to the ending position. Learning from only the optimal trajectories may lead to over-fitting problems since the optimal trajectories only occupy a small proportion of the feasible navigation trajectories. To alleviate the over-fitting problems, we propose to learn not only from optimal trajectories but also from sub-optimal trajectories. As shown in Fig. 2, we define sub-optimal trajectories as the ones that have the same starting and ending points as the optimal trajectory and their lengths are shorter than a threshold. Positive sub-optimal trajectories should be close to the anchor, while intra-negative ones should deviate heavily from the anchor. The hop (step count) of a sub-optimal trajectory T is denoted as $h(T)$, and the hop of the optimal one is denoted as h_{gt}. We introduce two hyper-parameters α_p and α_n ($2>\alpha_n>\alpha_p>1$) to separate these trajectories into positive samples and intra-negative samples:

$$\begin{aligned} \mathcal{P}^T &= \{T_i | h(T_i) \le \alpha_p \cdot h_{gt}\}, \\ \mathcal{N}^T &= \{T_i | h(T_i) \ge \alpha_n \cdot h_{gt}\}, \end{aligned} \tag{5}$$

In this way, we get all initial positive trajectories $\mathcal{P}^T$ and intra-negative trajectories $\mathcal{N}^T$. To help the model distinguish different instances and improve efficiency, we introduce a memory bank $\mathcal{M}^T$ to make use of representations of inter-negative samples from previous batches. Then in the current batch, we get all negative representations by unifying intra-negative trajectories and inter-negative samples:

$$\mathcal{N}_{full}^T = \mathcal{N}^T \cup \mathcal{M}^T. \tag{6}$$

Therefore the coarse contrastive loss for trajectories is formulated as:

$$\mathcal{L}_{C^T} = \mathcal{L}_{\text{circle}}\left(q, \mathcal{P}^T, \mathcal{N}_{full}^T\right). \tag{7}$$

After computing $\mathcal{L}_{C^T}$, the memory bank $\mathcal{M}^T$ is updated by replacing oldest positive samples with $\mathcal{P}^T$.

Methods	R2R Val Seen				R2R Val Unseen				R2R Test Unseen			
	TL	NE↓	SR↑	SPL↑	TL	NE↓	SR↑	SPL↑	TL	NE↓	SR↑	SPL↑
Random	9.58	9.45	16	-	9.77	9.23	16	-	9.89	9.79	13	12
Human	-	-	-	-	-	-	-	-	11.85	1.61	86	76
Seq2Seq	11.33	6.01	39	-	8.39	7.81	22	-	8.13	7.85	20	18
Speaker-Follower	-	3.36	66	-	-	6.62	35	-	14.82	6.62	35	28
SMNA	-	3.22	67	58	-	5.52	45	32	18.04	5.67	48	35
RCM+SIL (train)	10.65	3.53	67	-	11.46	6.09	43	-	11.97	6.12	43	38
PRESS	10.57	4.39	58	55	10.36	5.28	49	45	10.77	5.49	49	45
FAST-Short	-	-	-	-	21.17	4.97	56	43	22.08	5.14	54	41
AuxRN	-	3.33	70	67	-	5.28	55	50	-	5.15	55	51
PREVALENT	10.32	3.67	69	65	10.19	4.71	58	53	10.51	5.30	54	51
RelGraph	10.13	3.47	67	65	9.99	4.73	57	53	10.29	4.75	55	52
EnvDrop	11.00	3.99	62	59	10.70	5.22	52	48	11.66	5.23	51	47
+ CITL	11.84	3.23	70	66	15.47	5.06	52	48	10.69	5.39	54	50
RecBERT (init OSCAR)	10.79	3.11	71	67	11.86	4.29	59	53	12.34	4.59	57	53
+ CITL	11.22	2.99	72	68	15.91	4.34	60	54	15.83	4.30	61	55
RecBERT (init PREVALENT)	11.13	2.90	72	68	12.01	3.93	**63**	57	12.35	4.09	63	57
+ CITL	11.20	**2.65**	**75**	**70**	11.88	**3.87**	**63**	**58**	12.30	**3.94**	**64**	**59**

Table 1: Comparison with state-of-the-art methods on R2R. Black indicates best results.

Methods	R4R Val Seen						R4R Val Unseen					
	NE↓	SR↑	SPL↑	CLS↑	nDTW↑	SDTW↑	NE↓	SR↑	SPL↑	CLS↑	nDTW↑	SDTW↑
Speaker-Follower	5.35	51.9	37.3	46.4	-	-	8.47	23.8	12.2	29.6	-	-
RCM (goal)	5.11	55.5	32.3	40.4	-	-	8.45	28.6	10.2	20.4	-	-
RCM (fidelity)	5.37	52.6	30.6	55.3	-	-	8.08	26.1	7.7	34.6	-	-
PTA high-level	4.54	58	39	**60**	**58**	41	8.25	24	10	37	32	10
EGP	-	-	-	-	-	-	8.00	30.2	-	44.4	37.4	17.5
BabyWalk	-	-	-	-	-	-	8.2	27.3	14.7	**49.4**	**39.6**	17.3
RecBERT*	4.27	60.5	51.9	53.3	51.6	37.7	6.73	41.2	31.7	39.6	36.8	21.6
+ CITL	**3.48**	**66.8**	**57.0**	56.4	55.2	**42.7**	**6.42**	**44.4**	**35.1**	39.6	37.4	**23.4**

Table 2: Comparison with agents on the R4R dataset. * indicates results we reproduce (init PREVALENT).

Instruction Loss Natural languages contain considerable noise due to their diversity, like multiple synonyms. To overcome this problem, we implement a contrastive objective for instruction-level comparison among the diversified language descriptions. First of all, we adopt three natural language processing augmentation methods to generate high-quality positive instructions given a query instruction: 1) using the WordNet to substitute words with their **synonyms** (Zhang, Zhao, and LeCun 2015); 2) using a pre-trained BERT to **insert** or **substitute** words according to context (Anaby-Tavor et al. 2020; Kumar, Choudhary, and Cho 2020); and 3) **back-translation** (Xie et al. 2020). We assume that the augmented instructions should preserve semantic information of the original ones. To obtain the intra-negative instruction of the query instruction, we first generate sub-instructions as in (Hong et al. 2020a). These sub-instructions are shuffled or repeated randomly and then reassembled to become an intra-negative instruction. All augmented samples are fed into the language encoder E_L to get positive and intra-negative language representations. After that, we get the positive representations $\mathcal{P}^I$ and the intra-negative representations $\mathcal{N}^I$. We also introduce a memory bank $\mathcal{M}^I$ for instruction to store inter-negative representations. Then we unify $\mathcal{N}^I$ and $\mathcal{M}^I$ to get full negative set $\mathcal{N}^I_{full}$ following Eq. 6. For each positive instruction representation q, the coarse contrastive loss for instruction is defined as:

$$\mathcal{L}_{C^I} = \mathcal{L}_{\text{circle}}\left(q, \mathcal{P}^I, \mathcal{N}^I_{full}\right). \tag{8}$$

Similar to $\mathcal{L}_{C^T}$, the memory bank $\mathcal{M}^I$ is updated with $\mathcal{P}^I$.

Model	SR↑	SPL↑	CLS↑	nDTW↑	SDTW↑
EnvDrop	38.5	34	54	51	32
Syntax	39.2	35	56	52	32
RecBERT*	44.9	39.3	56.2	52.5	36.3
+ CITL	**47.2**	**40.7**	**56.9**	**53.5**	**37.6**

Table 3: Comparison on the RxR monolingual unseen validation set. * indicates results we reproduce (init PREVALENT). Black indicates best results.

Fine-grained Contrastive Learning

The coarse-grained contrastive learning focuses on whole trajectories and instructions. In contrast, the fine-grained contrastive loss focuses on sub-instructions and introduces temporal information to help the agent analyze the coherence of sub-instructions. Here we propose a fine-grained contrastive loss for sub-instructions $\mathcal{L}_{F^I}$.

Sub-instruction Loss We propose a fine-grained contrastive strategy for sub-instructions to help the agent learn the tem-

Model	R2R Val Unseen (1%)				R2R Val Unseen (5%)				R2R Val Unseen (10%)			
	TL	NE↓	SR↑	SPL↑	TL	NE↓	SR↑	SPL↑	TL	NE↓	SR↑	SPL↑
RecBERT (init O.)*	8.69	9.08	17.79	16.49	9.77	**8.24**	24.39	22.48	10.74	7.46	31.89	29.14
+ CITL	10.03	**8.90**	**18.90**	**17.31**	10.31	8.35	**25.67**	**23.34**	10.21	**7.10**	**34.48**	**31.47**
RecBERT (init P.)*	13.77	7.49	**32.65**	27.96	11.56	6.07	42.32	37.67	12.86	5.37	48.11	42.69
+ CITL	11.13	**7.18**	32.52	**29.00**	11.23	**5.72**	**45.38**	**41.54**	12.10	**5.28**	**50.02**	**45.05**

Table 4: Results on semi-supervised evaluation (trained with 1%, 5% and 10% training data). * indicates results we reproduce (O. and P. represent OSCAR and PREVALENT).

poral information of sub-instructions and analyze instructions better. We assume that adjoining sub-instructions have a sense of coherence. Thus their representations should be similar to some degree. Those sub-instructions which are not neighbors should be pulled apart. To generate positive and intra-negative samples, we first generate sub-instructions given instruction as in (Hong et al. 2020a). Then we randomly select a sub-instruction as the query sub-instruction. The nearest neighbors of this query sub-instruction are positive samples, while others are intra-negative samples. Similar to the coarse contrastive losses, a memory bank $\mathcal{M}^{SI}$ is introduced to store inter-negative sub-instructions from other instructions. Similar to the instruction loss $\mathcal{L}_{C^I}$, positive and intra-negative language representations are extracted via the language encoder E_L. For the query sub-instruction q, the fine-grained contrastive loss is formulated as:

$$\mathcal{L}_{F^I} = \mathcal{L}_{\text{circle}}\left(q, \mathcal{P}^{SI}, \mathcal{N}^{SI}_{full}\right). \tag{9}$$

The memory bank $\mathcal{M}^{SI}$ is updated with $\mathcal{P}^{SI}$.

Pairwise Sample-reweighting Mechanism

There are large amounts of easy samples in augmented samples and memory banks, causing the training to plateau quickly and occupy extensive memory usage. To alleviate this, we propose the pairwise sample-reweighting mechanism equipped with a novel pair mining strategy to explore hard samples and reweight different pairs. The overview is shown in Fig. 3.

Pair Mining We introduce our pair mining strategy that aims to select informative samples and discard less informative ones. For the anchor q, positive and negative sets are denoted as P and N. Negative samples are selected as follows compared with the hardest positive sample:

$$S_n = \{n_j | 1 - m > \mathrm{sim}\left(q, n_j\right) > \min\{\mathrm{sim}\left(q, p_i\right)\} - m\}, \tag{10}$$

where $p_i \in \mathcal{P}$ and $n_j \in \mathcal{N}$. If the similarity score is greater than $1 - m$, this negative sample will be regarded as false negative and then discarded. After selecting negative samples, positive samples are compared with the remaining hardest negative sample:

$$S_p = \{p_i | \mathrm{sim}\left(q, p_i\right) < \max\{\mathrm{sim}\left(q, n_j\right)\} + m\}, \tag{11}$$

where $n_j \in S_n(\mathcal{N})$.

Sample reweighting The remaining samples will be reweighted by self-paced reweighting following Eq. 4. Unlike previous methods, our sample reweighting focuses on sequantial data. The reweighted loss can be formulated as $\mathcal{L}_{\text{circle}}(q, S_p(\mathcal{P}), S_n(\mathcal{N}))$. Hence the coarse-fine contrastive losses are rewrited as follows:

$$\begin{aligned}
\mathcal{L}^{'}_{C^T} &= \mathcal{L}_{C^T}(q, S_p(\mathcal{P}^T), S_n(\mathcal{N}^T_{full})), \\
\mathcal{L}^{'}_{C^I} &= \mathcal{L}_{C^I}(q, S_p(\mathcal{P}^I), S_n(\mathcal{N}^I_{full})), \\
\mathcal{L}^{'}_{F^I} &= \mathcal{L}_{F^I}(q, S_p(\mathcal{P}^{SI}), S_n(\mathcal{N}^{SI}_{full})).
\end{aligned} \tag{12}$$

Training

We train the model with a mixture of contrastive learning, reinforcement learning (RL) and imitation learning (IL). The agent learns by following teacher actions a^*_t:

$$\mathcal{L}_{IL} = \sum_t -a^*_t log\left(p_t\right). \tag{13}$$

RL is adopted to avoid overfitting in VLN. Here we adopt A2C (Mnih et al. 2016) algorithm:

$$\mathcal{L}_{RL} = -\sum_t a_t log\left(p_t\right) A_t. \tag{14}$$

p_t and A_t are predicted logits and the advantage function. The full loss in our proposed model is as:

$$\mathcal{L} = \mathcal{L}_{IL} + \mathcal{L}_{RL} + \lambda_1 \mathcal{L}^{'}_{C^P} + \lambda_2 \mathcal{L}^{'}_{C^I} + \lambda_3 \mathcal{L}^{'}_{F^I}. \tag{15}$$

where λ_1, λ_2 and λ_3 are weighting factors.

Experiments

Datasets We evaluate the CITL on several popular VLN datasets. The R2R (Anderson et al. 2018b) dataset consists of 90 housing environments. The training set comprises 61 scenes, and the validation unseen set and test unseen set contain 11 and 18 scenes respectively. R4R (Jain et al. 2019) concatenates the trajectories and instructions in R2R. RxR (Ku et al. 2020) is a larger dataset containing more extended instructions and trajectories.

Experimental Setup All experiments are conducted on an NVIDIA 3090 GPU. We also use the MindSpore Lite tool[1]. In all contrastive losses, the margin m is set to 0.25, and λ_1, λ_2 and λ_3 are fixed to 0.1, 0.01 and 0.01 respectively. The size of all memory banks is fixed to 240. α_p and α_n are set to 1.2 and 1.4 respectively. Training schedules are the same as baselines (Tan, Yu, and Bansal 2019; Hong et al. 2021). We use the same augmentation data as in (Hao et al. 2020) when adopting RecBERT (Hong et al. 2021) as the baseline.

Evaluation Metrics For R2R, the agent is evaluated using the following metrics (Anderson et al. 2018a,b): Trajectory

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

Length (TL), Navigation Error (NE), Success Rate (SR) and Success weighted by Path Length (SPL). Additional metrics are used for R4R and RxR, including Coverage weighted by Length Score (CLS) (Jain et al. 2019) and Normalized Dynamic Time Warping (nDTW) (Magalhaes et al. 2019) and Success rate weighted normalized Dynamic Time Warping (SDTW) (Magalhaes et al. 2019).

Comparison with SoTA

Results in Table 1 compare the single-run (greedy search, no pre-exploration (Wang et al. 2019b)) performance of different agents on the R2R benchmark. Previous methods include Seq2Seq (Anderson et al. 2018b), Speaker-Follower (Fried et al. 2018), SMNA (Ma et al. 2019a), RCM (Wang et al. 2019b), PRESS (Li et al. 2019), FAST-Short (Ke et al. 2019), AuxRN (Zhu et al. 2020a), PREVALENT (Hao et al. 2020), RelGraph (Hong et al. 2020b), EnvDrop (Tan, Yu, and Bansal 2019), RecBERT (Hong et al. 2021). Our base model initialised from PREVALENT, a pre-trained model for VLN, performs better than previous methods over all dataset splits, achieving 59% SPL (+2%) on the test set. Comparing to previous methods, we can see that the improvement of the test set is greater than the unseen validation split, which suggests the strong generalization of our agent by equipping with coarse/fine-grained semantic contrast. Table 2 shows results compared with previous methods (Speaker-Follower, RCM, PTA (Landi et al. 2019), EGP (Deng, Narasimhan, and Russakovsky 2020), Baby-Walk (Zhu et al. 2020b), RecBERT) on the R4R dataset. Our CITL performs consistently better than the RecBERT baseline, showing that our model can generalize well to long instruction and trajectory. Table 3 compares CITL with previous state-of-the-art methods (EnvDrop, Syntax (Li, Tan, and Bansal 2021) and RecBERT) on the RxR dataset. Our model gets significant improvement (+1.4% in SPL and +2.3% in SR) compared with its RecBERT backbone, and outperforms previous state-of-the-art models on all metrics.

Ablation Study

We further study the effectiveness of each component of CITL over the R2R dataset without augmentation data generated by the speaker.

Semi-Supervised Evaluation To validate the robustness of the proposed method and the ability to acquire exceptional knowledge with less training data, we conduct some experiments in the semi-supervised setting, in which we train the agent with only 1%, 5% and 10% of the training data. Table 4 presents results on the validation unseen split.

Pairwise Sample-reweighting Mechanism We present detailed comparisons on each module to validate our pairwise sample-reweighting mechanism as in Table 5. Our proposed pairwise sample-reweighting mechanism performs better than multi-pair InfoNCE loss (55.47% vs. 52.62% SPL). Simply using circle loss (Sun et al. 2020) as the contrastive loss does not help the agent fully leverage semantic information. Adding a memory bank can store more samples for contrastive learning, but many easy samples and some noisy data harm the training. Thus, pair mining to select hard samples improves the agent's performance (53.14% to 55.47% SPL). This evidence confirms that hard positive and negative samples are crucial in our contrastive losses.

Loss	Module				R2R Val Unseen		
	$\mathcal{L}_{\text{mul}}$	$\mathcal{L}_{\text{circle}}$	$\mathcal{M}$	PM	NE↓	SR↑	SPL↑
①	✓		✓		4.45	57.47	52.62
②	✓		✓	✓	4.30	59.17	54.10
③		✓			4.28	59.05	53.35
④		✓	✓		4.44	58.49	53.14
⑤		✓		✓	**4.11**	59.98	54.54
Full		✓	✓	✓	4.29	**60.90**	**55.47**

Table 5: Ablation study on NCE/circle loss and pairwise sample-reweighting mechanism in trajectory loss $\mathcal{L}_{C^T}$. $\mathcal{M}$ is the memory bank for trajectories, and PM is pair mining.

Models	Losses			R2R Val Unseen		
	$\mathcal{L}'_{C^T}$	$\mathcal{L}'_{C^I}$	$\mathcal{L}'_{F^I}$	NE↓	SR↑	SPL↑
Baseline				4.47	57.17	52.90
①	✓			4.29	60.90	55.47
②		✓		4.22	61.00	55.17
③			✓	4.37	58.24	53.58
Full	✓	✓	✓	**3.98**	**62.11**	**55.83**

Table 6: Results on different coarse-fine contrastive losses.

We also conduct experiments on InfoNCE loss with pair mining in Table 5. The number of positive samples is set to 16 to get better results in pair mining. We can see that it can improve the performance of InfoNCE loss. However, the final result is worse than our pairwise sample-reweighting mechanism since InfoNCE loss cannot reweight hard and easy samples differently.

Coarse/fine-grained Contrastive Losses Table 6 shows comprehensive ablation experiments on our coarse/fine-grained contrastive losses. As the results suggested, employing coarse contrastive losses leads to substantial performance gains, which suggests that exploiting the semantics of the cross-instance instruction-trajectory pairs in contrastive learning improves navigation. Meanwhile, employing fine-grained contrastive loss to learn temporal information of sub-instructions also enhances the performance, which indicates that the agent may benefit from analyzing relations of sub-instructions. Combining coarse/fine-grained contrastive loss further improves the agent's performance (52.90% to 55.83% SPL).

Conclusion

In this paper, we propose a novel framework named CITL, with coarse/fine-grained contrastive learning. Coarse-grained contrastive learning fully explores the semantics of cross-instance samples and enhances vision-and-language representations to improve the performance of the agent. The fine-grained contrastive learning learns to leverage the temporal information of sub-instructions. The pairwise sample-reweighting mechanism mines hard samples and eliminates the effects of false-negative samples, hence mitigating the influence of augmentation bias and improving the robustness of the agent. Our CITL is more robust.

Acknowledgements

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) under Grant No.U19A2073 and 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) and CAAI-Huawei MindSpore Open Fund. We thank MindSpore for the partial support of this work, which is a new deep learning computing framwork[2].

References

Anaby-Tavor, A.; Carmeli, B.; Goldbraich, E.; Kantor, A.; Kour, G.; Shlomov, S.; Tepper, N.; and Zwerdling, N. 2020. Do Not Have Enough Data? Deep Learning to the Rescue! In *AAAI*.

Anderson, P.; Chang, A. X.; Chaplot, D. S.; Dosovitskiy, A.; Gupta, S.; Koltun, V.; Kosecka, J.; Malik, J.; Mottaghi, R.; Savva, M.; and Zamir, A. R. 2018a. On Evaluation of Embodied Navigation Agents. *CoRR*.

Anderson, P.; Wu, Q.; Teney, D.; Bruce, J.; Johnson, M.; Sünderhauf, N.; Reid, I.; Gould, S.; and van den Hengel, A. 2018b. Vision-and-Language Navigation: Interpreting Visually-Grounded Navigation Instructions in Real Environments. In *CVPR*.

Bachman, P.; Hjelm, R. D.; and Buchwalter, W. 2019. Learning Representations by Maximizing Mutual Information Across Views. In *NeurIPS*.

Cai, Q.; Wang, Y.; Pan, Y.; Yao, T.; and Mei, T. 2020. Joint Contrastive Learning with Infinite Possibilities. In *NeurIPS*.

Caron, M.; Misra, I.; Mairal, J.; Goyal, P.; Bojanowski, P.; and Joulin, A. 2020. Unsupervised Learning of Visual Features by Contrasting Cluster Assignments. In *NeurIPS*.

Chaitanya, K.; Erdil, E.; Karani, N.; and Konukoglu, E. 2020. Contrastive learning of global and local features for medical image segmentation with limited annotations. In *NeurIPS*.

Chen, H.; Suhr, A.; Misra, D.; Snavely, N.; and Artzi, Y. 2019. TOUCHDOWN: Natural Language Navigation and Spatial Reasoning in Visual Street Environments. In *CVPR*.

Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020. A Simple Framework for Contrastive Learning of Visual Representations. In *ICML*.

Chen, X.; and He, K. 2021. Exploring Simple Siamese Representation Learning. *CVPR*.

Deng, Z.; Narasimhan, K.; and Russakovsky, O. 2020. Evolving Graphical Planner: Contextual Global Planning for Vision-and-Language Navigation. In *NeurIPS*.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. N. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL-HLT*.

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

Dosovitskiy, A.; Springenberg, J. T.; Riedmiller, M.; and Brox, T. 2014. Discriminative Unsupervised Feature Learning with Convolutional Neural Networks. In *NeurIPS*.

Fried, D.; Hu, R.; Cirik, V.; Rohrbach, A.; Andreas, J.; Morency, L.-P.; Berg-Kirkpatrick, T.; Saenko, K.; Klein, D.; and Darrell, T. 2018. Speaker-Follower Models for Vision-and-Language Navigation. In *NeurIPS*.

Grill, J.-B.; Strub, F.; Altché, F.; Tallec, C.; Richemond, P.; Buchatskaya, E.; Doersch, C.; Avila Pires, B.; Guo, Z.; Gheshlaghi Azar, M.; Piot, B.; kavukcuoglu, k.; Munos, R.; and Valko, M. 2020. Bootstrap Your Own Latent - A New Approach to Self-Supervised Learning. In *NeurIPS*.

Hadsell, R.; Chopra, S.; and Lecun, Y. 2006. Dimensionality Reduction by Learning an Invariant Mapping. In *CVPR*.

Hao, W.; Li, C.; Li, X.; Carin, L.; and Gao, J. 2020. Towards Learning a Generic Agent for Vision-and-Language Navigation via Pre-Training. In *CVPR*.

He, K.; Fan, H.; Wu, Y.; Xie, S.; and Girshick, R. 2020. Momentum Contrast for Unsupervised Visual Representation Learning. In *CVPR*.

Henaff, O. 2020. Data-Efficient Image Recognition with Contrastive Predictive Coding. In *ICML*.

Hjelm, R. D.; Fedorov, A.; Lavoie-Marchildon, S.; Grewal, K.; Bachman, P.; Trischler, A.; and Bengio, Y. 2019. Learning deep representations by mutual information estimation and maximization. In *ICLR*.

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

Hong, Y.; Rodríguez, C.; Qi, Y.; Wu, Q.; and Gould, S. 2020b. Language and Visual Entity Relationship Graph for Agent Navigation. In *NeurIPS*.

Hong, Y.; Wu, Q.; Qi, Y.; Rodriguez-Opazo, C.; and Gould, S. 2021. A Recurrent Vision-and-Language BERT for Navigation. *CVPR*.

Jain, V.; Magalhães, G.; Ku, A.; Vaswani, A.; Ie, E.; and Baldridge, J. 2019. Stay on the Path: Instruction Fidelity in Vision-and-Language Navigation. In *ACL*.

Ke, L.; Li, X.; Bisk, Y.; Holtzman, A.; Gan, Z.; Liu, J.; Gao, J.; Choi, Y.; and Srinivasa, S. 2019. Tactical Rewind: Self-Correction via Backtracking in Vision-And-Language Navigation. In *CVPR*.

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

Kumar, V.; Choudhary, A.; and Cho, E. 2020. Data Augmentation using Pre-trained Transformer Models. In *LifeLongNLP*.

Landi, F.; Baraldi, L.; Cornia, M.; Corsini, M.; and Cucchiara, R. 2019. Perceive, Transform, and Act: Multi-Modal Attention Networks for Vision-and-Language Navigation. *CoRR*.

Li, J.; Tan, H.; and Bansal, M. 2021. Improving Cross-Modal Alignment in Vision Language Navigation via Syntactic Information. In *NAACL*.

Li, W.; Gao, C.; Niu, G.; Xiao, X.; Liu, H.; Liu, J.; Wu, H.; and Wang, H. 2021. UNIMO: Towards Unified-Modal Understanding and Generation via Cross-Modal Contrastive Learning. *ACL/IJCNLP*.

Li, X.; Li, C.; Xia, Q.; Bisk, Y.; Çelikyilmaz, A.; Gao, J.; Smith, N. A.; and Choi, Y. 2019. Robust Navigation with Language Pretraining and Stochastic Sampling. In *EMNLP/IJCNLP*.

Li, X.; Yin, X.; Li, C.; Hu, X.; Zhang, P.; Zhang, L.; Wang, L.; Hu, H.; Dong, L.; Wei, F.; Choi, Y.; and Gao, J. 2020. Oscar: Object-Semantics Aligned Pre-training for Vision-Language Tasks. In *ECCV*.

Lu, J.; Batra, D.; Parikh, D.; and Lee, S. 2019. ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks. In *NeurIPS*.

Ma, C.-Y.; Lu, J.; Wu, Z.; AlRegib, G.; Kira, Z.; Socher, R.; and Xiong, C. 2019a. Self-Monitoring Navigation Agent via Auxiliary Progress Estimation. In *ICLR*.

Ma, C.-Y.; Wu, Z.; AlRegib, G.; Xiong, C.; and Kira, Z. 2019b. The Regretful Agent: Heuristic-Aided Navigation Through Progress Estimation. In *CVPR*.

Magalhaes, G. I.; Jain, V.; Ku, A.; Ie, E.; and Baldridge, J. 2019. General Evaluation for Instruction Conditioned Navigation using Dynamic Time Warping. In *NeurIPS ViGIL Workshop*.

Majumdar, A.; Shrivastava, A.; Lee, S.; Anderson, P.; Parikh, D.; and Batra, D. 2020. Improving Vision-and-Language Navigation with Image-Text Pairs from the Web. In *ECCV*.

Mnih, V.; Badia, A. P.; Mirza, M.; Graves, A.; Lillicrap, T.; Harley, T.; Silver, D.; and Kavukcuoglu, K. 2016. Asynchronous Methods for Deep Reinforcement Learning. In *ICML*.

Nguyen, K.; and Daumé III, H. 2019. Help, Anna! Visual Navigation with Natural Multimodal Assistance via Retrospective Curiosity-Encouraging Imitation Learning. In *EMNLP*.

Nguyen, K.; Dey, D.; Brockett, C.; and Dolan, B. 2019. Vision-Based Navigation With Language-Based Assistance via Imitation Learning With Indirect Intervention. In *CVPR*.

Oord, A. v. d.; Li, Y.; and Vinyals, O. 2018. Representation learning with contrastive predictive coding. In *NeurIPS*.

Qi, Y.; Pan, Z.; Zhang, S.; van Hengel, A.; and Wu, Q. 2020a. Object-and-Action Aware Model for Visual Language Navigation. In *ECCV*.

Qi, Y.; Wu, Q.; Anderson, P.; Wang, X.; Wang, W. Y.; Shen, C.; and van den Hengel, A. 2020b. REVERIE: Remote Embodied Visual Referring Expression in Real Indoor Environments. In *CVPR*.

Sohn, K. 2016. Improved Deep Metric Learning with Multi-class N-pair Loss Objective. In *NeurIPS*.

Song, H. O.; Xiang, Y.; Jegelka, S.; and Savarese, S. 2016. Deep Metric Learning via Lifted Structured Feature Embedding. In *CVPR*.

Sun, C.; Baradel, F.; Murphy, K.; and Schmid, C. 2019. Learning video representations using contrastive bidirectional transformer. *arXiv preprint arXiv:1906.05743*.

Sun, Y.; Cheng, C.; Zhang, Y.; Zhang, C.; Zheng, L.; Wang, Z.; and Wei, Y. 2020. Circle Loss: A Unified Perspective of Pair Similarity Optimization. In *CVPR*.

Sutskever, I.; Vinyals, O.; and Le, Q. V. 2014. Sequence to Sequence Learning with Neural Networks. In *NeurIPS*.

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

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

Thomason, J.; Murray, M.; Cakmak, M.; and Zettlemoyer, L. 2019. Vision-and-Dialog Navigation. In *CoRL*.

Wang, H.; Wu, Q.; and Shen, C. 2020. Soft Expert Reward Learning for Vision-and-Language Navigation. In *ECCV*.

Wang, J.; Zhou, F.; Wen, S.; Liu, X.; and Lin, Y. 2017. Deep Metric Learning With Angular Loss. In *ICCV*.

Wang, X.; Han, X.; Huang, W.; Dong, D.; and Scott, M. R. 2019a. Multi-Similarity Loss With General Pair Weighting for Deep Metric Learning. In *CVPR*.

Wang, X.; Huang, Q.; Celikyilmaz, A.; Gao, J.; Shen, D.; Wang, Y.-F.; Wang, W. Y.; and Zhang, L. 2019b. Reinforced Cross-Modal Matching and Self-Supervised Imitation Learning for Vision-Language Navigation. In *CVPR*.

Weinberger, K. Q.; Blitzer, J.; and Saul, L. 2006. Distance Metric Learning for Large Margin Nearest Neighbor Classification. In *NeurIPS*.

Wu, Z.; Xiong, Y.; Yu, S. X.; and Lin, D. 2018. Unsupervised Feature Learning via Non-Parametric Instance Discrimination. In *CVPR*.

Xie, Q.; Dai, Z.; Hovy, E.; Luong, T.; and Le, Q. 2020. Unsupervised Data Augmentation for Consistency Training. In *NeurIPS*.

Xie, S.; Gu, J.; Guo, D.; Qi, C. R.; Guibas, L. J.; and Litany, O. 2020. PointContrast: Unsupervised Pre-training for 3D Point Cloud Understanding. In *ECCV*.

Xie, Z.; Lin, Y.; Zhang, Z.; Cao, Y.; Lin, S.; and Hu, H. 2021. Propagate Yourself: Exploring Pixel-Level Consistency for Unsupervised Visual Representation Learning. *CVPR*.

Yu, B.; and Tao, D. 2019. Deep Metric Learning With Tuplet Margin Loss. In *ICCV*.

Zhang, X.; Zhao, J.; and LeCun, Y. 2015. Character-level Convolutional Networks for Text Classification. In *NeurIPS*.

Zhu, F.; Zhu, Y.; Chang, X.; and Liang, X. 2020a. Vision-Language Navigation With Self-Supervised Auxiliary Reasoning Tasks. In *CVPR*.

Zhu, F.; Zhu, Y.; Lee, V.; Liang, X.; and Chang, X. 2021. Deep Learning for Embodied Vision Navigation: A Survey. *arXiv preprint arXiv:2108.04097*.

Zhu, W.; Hu, H.; Chen, J.; Deng, Z.; Jain, V.; Ie, E.; and Sha, F. 2020b. BabyWalk: Going Farther in Vision-and-Language Navigation by Taking Baby Steps. In *ACL*.