CLEAR: Improving Vision-Language Navigation with Cross-Lingual, Environment-Agnostic Representations

Jialu Li **Hao Tan** **Mohit Bansal**
UNC Chapel Hill
{jialuli, airsplay, mbansal}@cs.unc.edu

Abstract

Vision-and-Language Navigation (VLN) tasks require an agent to navigate through the environment based on language instructions. In this paper, we aim to solve two key challenges in this task: utilizing multilingual instructions for improved instruction-path grounding and navigating through new environments that are unseen during training. To address these challenges, first, our agent learns a shared and visually-aligned cross-lingual language representation for the three languages (English, Hindi and Telugu) in the Room-Across-Room dataset. Our language representation learning is guided by text pairs that are aligned by visual information. Second, our agent learns an environment-agnostic visual representation by maximizing the similarity between semantically-aligned image pairs (with constraints on object-matching) from different environments. Our environment agnostic visual representation can mitigate the environment bias induced by low-level visual information. Empirically, on the Room-Across-Room dataset, we show that our multi-lingual agent gets large improvements in all metrics over the strong baseline model when generalizing to unseen environments with the cross-lingual language representation and the environment-agnostic visual representation. Furthermore, we show that our learned language and visual representations can be successfully transferred to the Room-to-Room and Cooperative Vision-and-Dialogue Navigation task, and present detailed qualitative and quantitative generalization and grounding analysis.[1]

1 Introduction

The Vision-and-Language Navigation task requires an agent to navigate through the environment based on language instructions. This task has two unsolved challenges. First, directly introducing pre-trained linguistic and visual representations into these agents suffers from domain shift (i.e., pre-trained linguistic and visual representation might not generalize to VLN task) (Huang et al., 2019b). Learning the instruction representation while also learning how to navigate based on the instruction is even more challenging for a multi-lingual agent, since more language variance is injected via multi-lingual instructions. At the same time, it also poses the important question that whether we can utilize multi-lingual instructions to learn a better cross-lingual representation and improve instruction-path grounding and referencing. Second, previous works (Fried et al., 2018; Wang et al., 2019a; Landi et al., 2021; Wang et al., 2020a; Huang et al., 2019a; Ma et al., 2019a; Majumdar et al., 2020; Qi et al., 2020a) on vision-language navigation have seen that agents tend to perform substantially worse in environments that are unseen during training, indicating the lack of generalizability of the navigation agent. In this paper, we propose to address these two challenges via cross-lingual and environment agnostic representations.

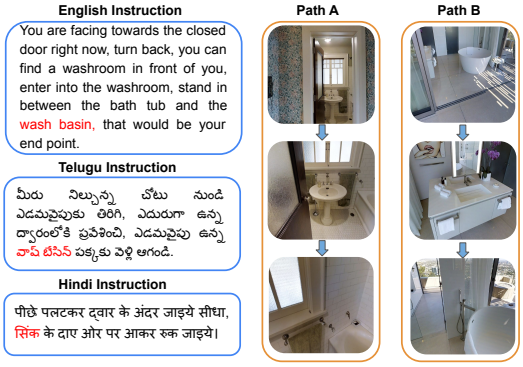


Figure 1: Motivation for cross-lingual and environment-agnostic visual representations: The English instruction, Telugu instruction, Hindi instruction on the left all correspond to the same path – Path A. The words in red correspond to the same visual object “wash basin". Path A and Path B are similar paths (i.e., the instruction for these two paths are semantically similar) in different environments.

[1]Code and model are available at https://github.com/jialuli-luka/CLEAR.

Findings of the Association for Computational Linguistics: NAACL 2022, pages 633 - 649
July 10-15, 2022

Although some initial progress (Huang et al., 2019b; Majumdar et al., 2020; Hong et al., 2021; Chen et al., 2021) has been made towards introducing pre-trained linguistic representations into vision-language navigation agents, how to understand and utilize paired multilingual instructions to transfer the pre-trained linguistic representation to multilingual navigation agent still remains unexplored. We argue that for a multilingual agent, the linguistic representation can capture more visual concepts from learning the similarity between paired multilingual instructions. As shown in Figure 1, though the three instructions shown here are in different languages and vary in length and level of detail[2], all of them correspond to the sample path – Path A. Hence, by learning the similarity between these paired instructions, the cross-lingual language representation of the same visual concept mentioned in these paired instructions (e.g., the red words correspond to the same visual object "wash basin") will be close to each other, making it easier for the agent to comprehend. Furthermore, the cross-lingual language representation will benefit from the complementary information from instructions in different languages since they elicit more references to visible entities. For example, in Figure 1, the target room environment "washroom" is only mentioned in English instructions. Hindi and Telugu instructions could benefit from learning the connection between "washroom" and "wash basin" through learning from the English instruction.

Moreover, many methods have been proposed to encourage agent generalization to unseen environments during training (Tan et al., 2019; Wang et al., 2020c; Fu et al., 2020; Zhang et al., 2020). Zhang et al. (2020) has shown that it is the low-level appearance information that causes the environment bias. To mitigate this bias, previous works only consider one single environment when learning the visual representation for a given path. We instead learn an environment-agnostic visual representation by exploring the connections between multiple environments. For the example shown in Figure 1, Path A and Path B are two semantically aligned paths in different environments. In both cases, the agent needs to head into the washroom and stop beside the wash basin. Learning the relationship between these paired paths helps the agent comprehend concepts like "bath tub", and not be distracted by the low-level appearance of the objects in unseen environments.

Overall, in this paper, we propose '**CLEAR**: **C**ross-**L**ingual and **E**nvironment-**A**gnostic **R**epresentations' to address the two challenges above. First, we define a visually-aligned instruction pair as two instructions that correspond to the same navigation path. Given the instruction pairs, we transfer the pre-trained multilingual BERT (Devlin et al., 2019) to the Vision-Language Navigation task by encouraging these paired instructions to be embedded close to each other. Second, we identify semantically-aligned path pairs based on the similarity between instructions. Intuitively, if the similarity between the two instructions is high, then their corresponding navigation path will be semantically similar (i.e., mentioning the same objects like "wash basin"). We further filter out image pairs (a pair of paths will contain multiple image pairs) that do not contain the same objects, for higher path pair similarity. Then, we train an environment agnostic visual representation that learns the connection between these semantically-aligned path pairs.

We conduct experiments on the Room-Across-Room (RxR) dataset (Ku et al., 2020), which contains instruction in three languages (English, Hindi, and Telugu). Empirical results show that our proposed representations significantly improves the performance over the mono-lingual model (Shen et al., 2022) by 2.59% in nDTW score on RxR test leaderboard. We further show that our CLEAR approach outperforms our baseline that utilizes ResNet (He et al., 2016) to extract image features by 5.3% in success rate and 4.3% in nDTW score (and it also outperforms a stronger baseline that utilizes the recent CLIP (Radford et al., 2021) method to extract image features). Moreover, our CLEAR approach shows better generalizability when transferred to Room-to-Room (R2R) dataset (Anderson et al., 2018b) and Cooperative Vision-and-Dialogue Navigation dataset (Thomason et al., 2019), and adapted to other SotA VLN Agent (Chen et al., 2021). We also demonstrate the advantage of optimizing similarity between all the three languages in RxR dataset for language representation learning and the effectiveness of the

[2]We translate Telugu instruction and Hindi instruction into English instruction with Google Translation for reference here (the translated instructions are not used in representation learning or navigation learning). Telugu: Return to the left from where you are standing, enter the door on the opposite side, and go to the side of the wash basin on the left and wait. Hindi: Turn back and go inside the door directly, come to the right side of the sink and stop.

way we generate positive path pairs for visual representation learning. Lastly, we demonstrate that our cross-lingual language representation captures visual semantics underlying the instructions, and our environment-agnostic visual representation generalizes better to the unseen environment with both qualitative and quantitative analysis.

2 Related Work

Vision-and-language navigation. Vision-and-Language Navigation (VLN) requires an agent to find the routes to the desired target based on instructions (Jain et al., 2019; Thomason et al., 2020; Nguyen and Daumé III, 2019; Qi et al., 2020b; Chen et al., 2019; Krantz et al., 2020). Specifically, there are two key challenges in VLN: grounding the natural language instruction to visual environments and generalizing to unseen environments. To address the first challenge, one line of research in VLN utilizes carefully designed cross-modal attention modules (Wang et al., 2018, 2019a; Tan et al., 2019; Landi et al., 2021; Xia et al., 2020; Wang et al., 2020b,a; Zhu et al., 2020; Li et al., 2021; Zhu et al., 2021; An et al., 2021; Kim et al., 2021), progress monitor modules (Ma et al., 2019b,a; Ke et al., 2019), and object-action aware modules (Qi et al., 2020a). Another line of research improves vision and language co-grounding by improving vision and language representations with pre-training techniques (Li et al., 2019; Huang et al., 2019b; Hao et al., 2020; Majumdar et al., 2020; Hong et al., 2021). Li et al. (2019) directly adopts pre-trained BERT for encoding instructions, Hao et al. (2020) and Hong et al. (2021) learn from a large amount of image-text-action triplets, Majumdar et al. (2020) learns from large amount of text-image pairs from the web, and Huang et al. (2019b) transfers language and visual representation to in-domain representation with auxiliary tasks. Different from them, we utilize the visually-aligned multilingual instructions to learn a cross-lingual language representation that inherently captures visual semantics underlying the instruction.

Multiple methods have been proposed to encourage generalization to unseen environments during training (Zhang et al., 2020; Tan et al., 2019; Wang et al., 2020c; Fu et al., 2020; Li et al., 2022). Zhang et al. (2020) demonstrates that it is the low-level appearance information that causes the large performance gap between seen and unseen environments. Tan et al. (2019) proposes to use environment dropout on visual features to create new environments and Fu et al. (2020) utilizes adversarial path sampling to encourage generalization. However, both of these methods rely on a speaker module to generate synthetic training data and can be considered as data augmentation methods, which are complementary to our proposed environment-agnostic visual representation. The closest work to ours is Wang et al. (2020c), where they proposes to pair an environment classifier with gradient reversal layer to learn an environment-agnostic representation. However, they only consider one single environment when learning the visual representation for a given path (i.e., given one path and predict its environment). In our environment-agnostic representation learning, we explore the connections between multiple environments (i.e., maximize the similarity between paths from different environments).

Vision-and-language with multilinguality. There has been growing interest in combining vision and language for tasks such as visual-guided machine translation (Sigurdsson et al., 2020; Surís et al., 2022; Huang et al., 2020), multi-lingual visual question answering (Gao et al., 2015; Gupta et al., 2020; Shimizu et al., 2018), multi-lingual image captioning (Gu et al., 2018; Lan et al., 2017), multi-lingual video captioning (Wang et al., 2019b), and multi-lingual image-sentence retrieval (Kim et al., 2020; Burns et al., 2020). In this paper, we work on multi-lingual vision-and-language navigation. We use vision (i.e., navigation path) as a bridge between multi-lingual instructions and learn a cross-lingual representation that captures visual concepts. Moreover, our method also use language as a bridge between different visual environments to learn an environment-agnostic visual representation.

3 Method

In this section, we present our CLEAR method that learns cross-lingual language representations and environment-agnostic visual representations. Given these learned language and visual representations, we then train the agent on the vision-and-language navigation task with imitation learning and reinforcement learning. The overall representation learning and navigation agent training processes are illustrated in Figure 2. We next describe our representation learning methods in Sec. 3.1 and Sec. 3.2. The navigation model (Tan et al., 2019) and training process are detailed in Appendix.

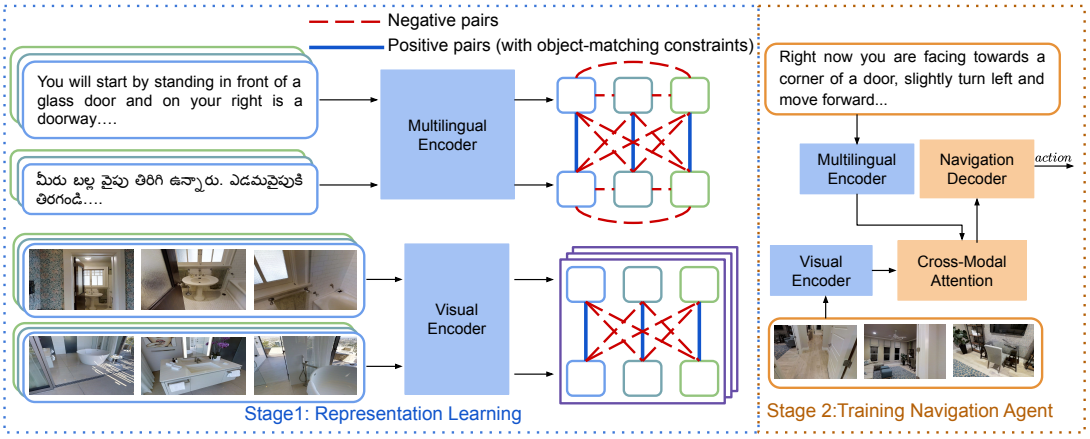


Figure 2: Left: the agent learns a cross-lingual language representation and an environment-agnostic visual representation via maximizing the similarity between positive pairs (connected with blue line) and minimizing the similarity between negative pairs (connected with red dashed line). For simplicity, we use 3 as batch size when illustrating the positive pairs and negative pairs. Right: then the agent is trained on the vision-and-language navigation task based on these learned representations.

3.1 Language Representation Learning

The goal of our language representation learning approach is to learn a cross-lingual language representation that can mitigate the natural ambiguity and variance in multilingual instructions and improve the path-instruction alignment by capturing the shared and salient visual concepts underlying the instructions. We define visually-aligned instruction pairs as instructions that correspond to the same navigation path. Since these instruction pairs refer to the same navigation path, the visual concepts underlying these instructions (e.g., visual objects mentioned in the instruction) are shared. Thus, we could train the language representation to emphasize these visual concepts by learning the connection between these visually-aligned instruction pairs.

For each navigation path, the Room-Across-Room (RxR) dataset (Ku et al., 2020) provides 9 corresponding language instructions in 3 languages (English, Hindi, and Telugu). During training, for each navigation path, we randomly sample two instructions out of the nine corresponding instructions as the visually-aligned instruction pairs. The two instructions can be in different languages, which helps the agent learn a cross-lingual language representation. Exclusively learning connections between instructions in the same language will lose crucial information across languages, and we quantitatively illustrate this result in Sec. 6.1.

Given the instruction $\{w_i\}_{i=0}^{m}$ with m words, we use feature of the [CLS] token (i.e., w_0) in the pre-trained multilingual BERT (Devlin et al., 2019) outputs as the sentence representation $\widetilde{w}$:

$$\{\widehat{w}_i\}_{i=0}^{m} = \text{m-BERT}(\{w_i\}_{i=0}^{m}) \quad (1)$$

$$\widetilde{w} = \widehat{w}_0 \quad (2)$$

In a batch of size N, we have N positive pairs of instructions with representations $(\widetilde{w}_j, \widetilde{u}_j)_{j=1}^{N}$ from Eqn. 2. Each positive pair is matched with $2(N-1)$ negatives in the batch (i.e., $\{\widetilde{w}_k\}_{k\neq i}$ and $\{\widetilde{u}_k\}_{k\neq j}$). Our goal is to learn a representation that maps instructions for the same path closer to each other in the representation space, regardless of the language and the natural variance in human-generated instructions. We learn the representation by optimizing a contrastive loss:

$$L_{lang} = -\sum_{i=1}^{N} \log \frac{\exp(\alpha_{i,i}/\tau)}{\sum_{k=1}^{2N} \mathbb{1}_{k\neq i} \exp(\alpha_{i,k}/\tau)} \quad (3)$$

$$\alpha_{i,j} = \frac{\widetilde{w}_i^T \widetilde{u}_j}{\|\widetilde{w}_i\| \|\widetilde{u}_j\|} \quad (4)$$

where $\alpha_{i,j}$ is the similarity between the instruction $\widetilde{w}_i$ and $\widetilde{u}_j$, and τ is the temperature hyperparameter.

3.2 Visual Representation Learning

Our goal in visual representation learning is to learn an environment-agnostic visual representation that can mitigate the environment bias caused by objects' low-level appearance, such that it could generalize better to unseen environments. Intuitively, the agent would learn the general concept of objects instead of the low-level appearance if the agent can identify the same objects in two images

in different environment. Thus, we train the agent to learn the connected visual semantics between the semantically-aligned navigation paths (i.e., paths that mention the same objects or mention similar actions in different environments).

Identifying semantically-aligned path pairs: Although the appearance of the path varies a lot in different environments, the instructions that describe the similar paths are more consistent across environments. Based on this intuition, we use language as the bridge between paths in multiple visual environments. Specifically, we propose to use instruction similarity as a direct measurement of how semantically similar two paths are. For each instruction-path pairs (I, P) given in the Room-Across-Room (RxR) dataset, we first represent each instruction I as in Eqn. 2. Then, we compute the cosine similarity between the representation of instruction I and all the other instructions in the training set. We pick the instruction $\widehat{I}$ that is most similar to I and also constraints that $\widehat{I}$'s corresponding path $\widehat{P}$ has the same path length as P. Thus, we group P and $\widehat{P}$ as the semantically-similar path pair.

Constraint on object-matching: In a batch of size N, we have N positive semantically-aligned path pairs $(P_k, Q_k)_{k=1}^{N}$. We represent the positive path pair (P_k, Q_k) as sequences of panoramic views $(\{p_{k,t}\}_{t=1}^{L_k}, \{q_{k,t}\}_{t=1}^{L_k})$ with length L_k. Since paths might not be fully aligned (i.e., correspondence between image pairs $\{p_{k,t}\}$ and $\{q_{k,t}\}$ might not hold), we use object-matching to filter out image pairs that don't contain the same objects. Specifically, we use Mask-RCNN (He et al., 2017) model trained on LVIS dataset (Gupta et al., 2019) in detectron2 (Wu et al., 2019) to detect objects in the 36 discretized views of the panoramic view. We filter out object classes that appear less than 1% of the time in all panoramic views. 27 object classes left, including objects like 'cabinet', 'chair', and 'sofa'. All object classes can be found in Appendix. During training, we randomly sample 10 out of 27 object classes in each iteration and filter out image pairs that don't contain same objects of the sampled 10 object classes. Our object-matching constraint ensures that the corresponding image pairs $\{p_{k,t}\}$ and $\{q_{k,t}\}$ also have a high semantic similarity.

Visual encoder: The panoramic view of time step t is discretized into 36 single views $\{o_{t,i}\}_{i=1}^{36}$. We encode the visual representation for each view as:

$$\widehat{o}_{t,i} = \text{pre-trained model}(o_{t,i}) \quad (5)$$

$$v_{t,i} = W_{v1}\text{ReLU}(W_{v2}\widehat{o}_{t,i}) \quad (6)$$

$$\widehat{v}_{t,i} = \text{LayerNorm}(v_{t,i} + \widehat{o}_{t,i}) \quad (7)$$

We first encode images with pre-trained vision models. Then the encoded view features are passed through two fully-connected layers with ReLU as activation function. Layer normalization and residual connection are applied on top of the fully-connected layer.

Learning visual representation: Given the N positive semantically-aligned path pairs $(P_k, Q_k)_{k=1}^{N}$, at each time step t, we have N_p panoramic views (computed as the average of 36 single views as in Eqn. 10) that have a positive pair (i.e., the paired view contain at least one same object). For each view $p_{k,t}$ that has a positive pair, the visual encoder is trained to predict which of the N possible panoramic views $\{q_{k,t}\}_{k=1}^{N}$ contain similar semantic information. Specifically, we train the visual encoder to maximize the cosine similarity of the N_p positive image pairs in the batch while minimizing the cosine similarity of the $N * N_p - N_p$ negative image pairs (i.e. each view has $N - 1$ negatives). We optimize the contrastive loss as:

$$L_{visual} = -\sum_{k=1}^{N_p}\sum_{t=1}^{L_k}\log(\text{Softmax}_k(\beta_{k,t}/\tau)) \quad (8)$$

$$\beta_{k,t} = \frac{p_{k,t}^{T} q_{k,t}}{\|p_{k,t}\|\|q_{k,t}\|} \quad (9)$$

where $\beta_{k,t}$ is the similarity between positive panoramic view pair $p_{k,t}$ and $q_{k,t}$, and τ is the temperature hyperparameter. We compute the panoramic view representation as the average of 36 single views:

$$p_{k,t} = \frac{1}{36}\sum_{i=1}^{36}\widehat{v}_{p,k,t,i} \quad (10)$$

where $\widehat{v}_{p,k,t,i}$ is the output representation from the visual encoder. $q_{k,t}$ is computed similarly.

3.3 Learning

Our CLEAR agent has two stages of learning: representation learning and navigation learning.

In the representation learning stage, we train the multilingual encoder and visual encoder by optimizing the contrastive loss L_{lang} in Eqn. 3 and

Models	SR↑	SPL↑	NDTW↑	SDTW↑
RxR	20.98	18.55	36.81	16.88
CLIP	38.34	35.17	51.10	32.42
Our	40.29	36.57	53.69	34.86

Table 1: Test leaderboard results under single run setup. RxR is the mono-lingual baseline in Ku et al. (2020), CLIP is the mono-lingual agent in Shen et al. (2022)

L_{visual} in Eqn. 8 respectively. The representation learning process transfers the language representation to domain-specific language representation and adapts the visual representation to learn the correlation underlying the navigation environments.

In the navigation learning stage, we use a mixture of imitation learning and reinforcement learning to train the agent on the navigation task as in Tan et al. (2019). Details can be found in Appendix.

4 Experimental Setup

4.1 Dataset

We evaluate our agent on the Room-Across-Room (RxR) dataset (Ku et al., 2020). The dataset is split into training set, seen and unseen validation set, and test set. In the unseen validation set and test set, the environments are not appeared in training set. Thus the performance on these two sets show the model's generalizability to new environments. More details can be found in Appendix.

4.2 Evaluation Metrics

To evaluate the performance of our model, we follow the metrics used in the Room-Across-Room paper (Ku et al., 2020) (details in Appendix): Success Rate (SR), Success rate weighted by Path Length (SPL) (Anderson et al., 2018a), normalized Dynamic Time Warping (nDTW) (Magalhaes et al., 2019), and success rate weighted by Dynamic Time Warping (sDTW) (Magalhaes et al., 2019). nDTW and sDTW are the main metrics for RxR and SR and SPL are the main metrics for R2R.

4.3 Implementation Details

In our experiments, we learn the shared cross-lingual representation based on cased multilingual $\text{BERT}_{\text{BASE}}$. For the pre-trained vision model, we compare performance between image features extracted from ImageNet-pre-trained (Russakovsky et al., 2015) ResNet-152 (He et al., 2016) and CLIP-pre-trained (Radford et al., 2021) vision transformer (ViT-B/32) (Dosovitskiy et al., 2021) (abbreviated as 'CLIP feature' later). More details about representation learning and navigation training can be found in Appendix.

5 Results

5.1 Test Set Results

We compare our final agent model with results on the Room-Across-Room (RxR) leaderboard. Our agent is a multilingual model that learn three languages in the same model. Compared with mono-lingual agents that learn instructions in three languages separately, a multilingual agent performs worse due to high-resource languages degradation (Ku et al., 2020; Aharoni et al., 2019; Pratap et al., 2020). Our agent is tested under the single-run setup. In the single-run setting, the agent only navigates once and does not pre-explore the test environment. As shown in Table 1, our CLEAR model with CLIP features is 16.88% higher in nDTW score than the baseline mono-lingual model (Ku et al., 2020) ('RxR') that utilizes ResNet features and other base navigation model. Furthermore, our model is 2.59% higher in nDTW score than the mono-lingual model (Shen et al., 2022) ('CLIP') that utilizes CLIP features and the same base navigation model as ours.

5.2 Ablation Results

We demonstrate the effectiveness of our learned visual and language representations with ablation studies. The baseline model (annotated as 'ResNet' in Table 2) uses multilingual BERT and pre-trained ResNet to encode instructions and images without the representation learning stage. Our CLEAR-ResNet ('ResNet+both' in Table 2) outperforms its baseline models in all evaluation metrics on average. Specifically, it improves the baseline model by 5.3% in success rate (SR) and 4.3% in nDTW score on average over three languages. These results demonstrate that our CLEAR agent is not only more capable of reaching the target, but also follows the ground-truth path better.

We then show that both the cross-lingual language representation and environment-agnostic visual representation contribute to the overall improvement. When the cross-lingual language representation is added ('+text'), we see consistent improvement on the averaged metrics and observe that Hindi benefits most from the cross-lingual language representation. When adding the environment-agnostic visual representation ('+visual'), the nDTW score improves by 2.6%. These

Models	SR↑				SPL↑				NDTW↑				SDTW↑			
	avg	en	hi	te	avg	en	hi	te	avg	en	hi	te	avg	en	hi	te
RxR	22.8	22.2	23.0	23.1	20.4	19.8	20.7	20.7	38.9	38.6	39.2	38.8	18.2	17.8	18.3	18.4
ResNet	35.1	35.4	36.4	33.4	31.6	31.6	33.0	30.4	51.1	50.7	52.3	50.3	30.1	30.1	31.4	28.7
+text	36.0	36.1	37.6	34.3	31.7	31.7	33.2	30.3	52.0	52.3	53.4	50.2	30.5	30.5	32.0	29.1
+visual	35.6	35.8	36.9	33.9	32.5	32.6	33.9	31.0	53.7	53.6	55.1	52.5	30.5	30.5	31.7	29.1
+both	40.4	41.5	42.2	37.6	36.5	36.7	38.5	34.3	55.4	54.4	57.8	54.1	34.6	35.1	36.4	32.2
CLIP	41.7	42.5	44.0	38.6	37.1	37.2	39.2	34.8	55.8	55.6	57.3	54.5	35.6	36.3	37.6	33.3
+both	**44.4**	**46.0**	**46.0**	**41.1**	**39.3**	**40.1**	**41.0**	**36.9**	**57.0**	**57.2**	**58.1**	**55.7**	**37.8**	**38.7**	**39.3**	**35.3**

Table 2: Ablation study of our model with ResNet features and CLIP features on validation unseen sets. 'avg' is the agent's average performance on English, Hindi, and Telugu instructions.

Methods	SR↑	SPL↑	NDTW↑	SDTW↑
m-BERT	35.1	31.6	51.1	30.1
Mono	32.9	30.4	51.4	28.0
Multi	36.0	31.7	52.0	30.5

Table 3: Comparison between language representation trained with mono-lingual instruction pairs ('Mono') and multi-lingual instruction pairs ('Multi') on validation unseen sets. 'm-BERT' is the method that uses original multilingual BERT as language representation.

improvements validate the effectiveness of our learned language and visual representations.

Moreover, we show that our CLEAR approach could generalize to other pre-trained visual features. We implement another model (annotated as 'CLIP' in Table 2) that uses CLIP to encode images, which is a stronger baseline compared with the ResNet baseline ('ResNet' in Table 2). Our CLEAR-CLIP model ('CLIP+both' in Table 2) also shows 2.7% improvement in success rate (SR) and 1.2% improvement in nDTW score on average over three languages. This demonstrates the effectiveness of our CLEAR approach over different pre-trained visual features.

6 Analysis

6.1 Effectiveness of Cross-Lingual Representations

In this section, we show the effectiveness of our language representation learning method described in Sec. 3.1. We first show the effectiveness of using paired multilingual instructions instead of monolingual instructions in the language representation learning stage. Then, we show that our learned cross-lingual language representation captures the visual concepts behind the instruction better than the original multilingual BERT representation.

Multilingual vs. monolingual. To show that the multilingual instruction pairs are crucial for our cross-lingual language representation learning, we experiment with fine-tuning multilingual BERT with instruction pairs in same language only ('Mono' in Table 3). We observe that compared with the agent with cross-lingual representation ('Multi'), the success rate decreases by 3.1% and sDTW score decreases by 2.5%. Furthermore, compared with the baseline model that uses the original multilingual-BERT ('m-BERT'), the success rate drops 2.2% and the sDTW score drops 2.1%. This result indicates that instruction representations in one language cannot benefit from learning representation in other languages if the multi-lingual representation is only supervised by contrastive loss between mono-lingual instruction pairs.

Capturing visual concepts. Our cross-lingual language representation can ground to the visual environment more easily by capturing the visual concepts in the instruction. We demonstrate that shared visual concepts in different paths are captured by our language representation. We first encode the instruction as in Eqn. 2 with cross-lingual representation and original multilingual BERT separately. For every instruction, we retrieve another instruction with the highest cosine similarity under the constraints that two instructions don't correspond to the same path and equal path length. As shown in Figure 3, the second row is the query instruction and the first row is its corresponding path. The following four rows correspond to the instruction-path picked with cross-lingual representation and multilingual-BERT representation. First, we observe in Figure 3 that our cross-lingual representation retrieves a Hindi instruction while the multilingual-BERT picks an English instruction. This indicates that our cross-lingual representation learns to encode instructions with similar semantics in different languages closer to each other. Besides, we observe that in all three paths, the agent passes tables and chairs, but only in the query path and the cross-lingual paired path, the agent stops at places similar to "bar stools". This demonstrates that the visual objects in the cross-lingual picked path are

Figure 3: Comparison of the most similar instruction picked with cross-lingual representation and multilingual-BERT. Our cross-lingual picked instruction mentions more visual object as in the query instruction. Besides, the path corresponding to the cross-lingual picked instruction contains more accurate visual objects as in the query path.

Similarity	SR↑	SPL↑	NDTW↑	SDTW↑
mono	32.9	30.4	51.4	28.0
en+hi	32.0	28.3	48.4	26.9
en+te	30.0	27.2	48.8	25.4
hi+te	27.8	25.0	46.1	23.2
multi	36.0	31.7	52.0	30.5

Table 4: Comparison between language representation trained with different instruction pairs. 'mono' indicates representation trained with mono-lingual instruction pairs, 'multi' indicates representation trained with multi-lingual instruction pairs in all three languages, and *'en+hi'* indicates representation trained with multilingual instruction pairs in English and Hindi only.

more similar to the objects in the query path.

6.2 Effectiveness of Optimizing Similarity between Three Languages

In this section, we further show that only optimizing the similarity between a subset of languages (i.e., two out of three languages) will hurt the performance. Specifically, we train the language representation that optimizes similarity between only English and Hindi (*'en+hi'*), only English and Telugu (*'en+te'*), only Hindi and Telugu (*'hi+te'*), and only single language ('mono'). Given paired language instructions in English and Hindi in unseen set, the average distance is 0.61 for our language representation (i.e., optimizes similarity between all three languages), 0.43 for *en+te*, 1.67 for *hi+te*, and 1.55 for same language only, indicating that explicitly optimizing the similarity between *en+hi* helps reduce the distance between *en+hi* most. Adding *te* in optimization will make *en+hi* farther from each other, but still much better than only optimizing *hi+te*, and could also make the distance between all three languages to be closer to each other. We further show the performance of training the navigation agent with these language representations in Table 4. We observe that both the success rate and the nDTW score drop significantly when only training on a subset of languages. This result shows that it's crucial to train the language representation with instruction pairs in all three languages.

6.3 Decreasing Gap between Seen and Unseen Environments

Most previous navigation models (Wang et al., 2019a; Ma et al., 2019a; Majumdar et al., 2020) suffer from a large performance drop when moving from seen validation to unseen validation because the visual encoder overfits the low-level appearance features (Zhang et al., 2020). Our environment agnostic visual representation can decrease the performance gap between validation seen and unseen environments. As shown in Table 5, the nDTW gap is decreased from 1.6 to 1.0 compared with baseline model. It is also lower than the gap of multi-lingual agent in Ku et al. (2020).

6.4 Comparison with Other Contrastive Learning Approaches

In this section, we compare with SimCSE (Gao et al., 2021), an effective contrastive learning approach for text representation learning. We use SimCSE on our visual representation learning, where we use dropout as positives in contrastive learning. Using SimCSE to train the visual representation gets 34.8/53.0 (SR/nDTW),

Models	seen				unseen				|Δ|			
	SR	SPL	NDTW	SDTW	SR	SPL	NDTW	SDTW	SR	SPL	NDTW	SDTW
Ku et al. (2020)	25.2	-	42.2	20.7	22.8	-	38.9	18.2	2.4	-	3.3	2.5
ResNet	38.4	34.1	52.7	32.6	35.1	31.6	51.1	30.1	3.3	2.5	1.6	2.5
+visual	34.1	31.1	52.7	28.8	35.6	32.5	53.7	30.5	1.5	1.4	1.0	1.7

Table 5: The results of adding our learned visual representation on validation seen environments and validation unseen environments. |Δ| indicates absolute performance difference between seen and unseen environments.

Models	SR↑	SPL↑	NDTW↑	SDTW↑
ResNet	49.1	44.7	58.8	42.0
+text	49.0	45.2	59.5	42.3
+visual	50.4	46.3	60.3	43.4
CLEAR	**50.5**	**46.4**	**60.6**	**43.3**
ResNet-zero	30.9	27.9	49.0	26.3
CLEAR-zero	**35.4**	**30.1**	**49.0**	**28.2**

Table 6: Results on R2R validation unseen environments. "CLEAR" (based on ResNet) transfers the language and visual representation from RxR dataset, and "ResNet" is the baseline model that uses multilingual BERT and pre-trained ResNet. "ResNet-zero" and "CLEAR-zero" are zero-shot performance of baseline and our approach on R2R dataset.

which is lower than our visual representation (35.6/53.7). Furthermore, we experiment with using both dropout as positives and our identified path pairs as positives. The performance decreases in nDTW score (52.4) compared with only using our identified path pairs as positives (53.7).

6.5 Generalization to Other VLN Tasks

We further evaluate our CLEAR approach's generalizability on Room-to-Room (R2R) dataset (Anderson et al., 2018b) and Cooperative Vision-and-Dialog Navigation (CVDN) dataset (Thomason et al., 2019), in which we directly transfer our CLEAR approach and train on the navigation task on R2R and CVDN. R2R and CVDN follows the same training, validation seen, and validation unseen split of environments as Room-Across-Room dataset. The main difference is that the language instructions in R2R and CVDN is monolingual (i.e., English). Besides, instructions in CVDN are multi-round dialogues between navigator and the oracle. Our baseline model uses multilingual BERT to encode instructions and the ResNet pretrained on ImageNet to extract image features. The cross-lingual language representation and environment-agnostic visual representation is trained on RxR dataset (as in Sec. 3.1 and Sec. 3.2). We then train the navigation agent on R2R dataset and CVDN dataset with the language and visual encoder initialized from our CLEAR representation.

As shown in Table 6, on R2R dataset, our learned representation outperforms the baseline by 1.4% in success rate and 1.8% in nDTW. Furthermore, we show that the zero-shot performance of our approach improves the baseline by 4.5% in success rate and 2.2% in SPL on R2R dataset. On CVDN dataset, our learned representation outperforms the baseline by 0.74 in Goal Progress (4.05 vs. 3.31) after training on CVDN dataset, and outperforms the baseline by 0.42 in Goal Progress (0.92 vs. 0.50) in the zero-shot setting. Goal Progress measures the progress made towards the target location and is the main evaluation metric in CVDN. This result demonstrates that our learned cross-lingual and environment agnostic representation could generalize to other tasks.

6.6 Generalization to Other VLN Agents

We further evaluate our CLEAR approach's generalizability to other VLN agent. Specifically, we adapt CLEAR to SotA VLN agent HAMT (Chen et al., 2021). With the pre-trained weights released in HAMT, we further learn the text representation and visual representation with our approach. Adapting CLEAR to HAMT achieves 57.2% in success rate and 65.6% in nDTW score, which is 0.7% higher than HAMT in success rate and 2.5% higher than HAMT in nDTW score on RxR validation unseen set, demonstrating the effectiveness of our proposed approach over SotA VLN models.

7 Conclusion

In this paper, we presented the CLEAR method that learns a cross-lingual and environment-agnostic representation. We demonstrated that our cross-lingual language representation captures more visual semantics and our environment-agnostic representation generalizes better to unseen environments. Our experiments on Room-Across-Room dataset suggest that our CLEAR method improved the performance in all evaluation metrics over a strong baseline. Furthermore, we qualitatively and quantitatively analyze the effectiveness of every component of our CLEAR approach and its generalizability to other tasks and base VLN agents.

Ethics Statement

In this paper, we presented a method to learn cross-lingual and environment-agnostic representations for Vision-and-Language Navigation. Vision-and-Language Navigation task can be used in many real-world applications, for example, a home service robot can bring things to the owner based on natural language instructions, making people's life easier. Our learned representations enable the agent to understand multi-lingual instructions and improve agents' generalizability to unseen environments. However, currently we learn our cross-lingual representation from three languages (i.e., English, Hindi, and Telugu) only due to dataset availability, which might limit its generalization to other languages. Besides, similar to other instruction-following agent, our agent might fail to reach the target given some instructions, which requires further human assistance.

Acknowledgement

We thank the reviewers for their helpful comments. This work was supported by ARO W911NF2110220, ONR N000141812871, DARPA KAIROS FA8750-19-2-1004, Google Focused Award. The views contained in this article are those of the authors and not of the funding agency.

References

Roee Aharoni, Melvin Johnson, and Orhan Firat. 2019. Massively multilingual neural machine translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3874–3884, Minneapolis, Minnesota. Association for Computational Linguistics.

Dong An, Yuankai Qi, Yan Huang, Qi Wu, Liang Wang, and Tieniu Tan. 2021. Neighbor-view enhanced model for vision and language navigation. In *MM '21: ACM Multimedia Conference, Virtual Event, China, October 20 - 24, 2021*, pages 5101–5109. ACM.

Peter Anderson, Angel Chang, Devendra Singh Chaplot, Alexey Dosovitskiy, Saurabh Gupta, Vladlen Koltun, Jana Kosecka, Jitendra Malik, Roozbeh Mottaghi, Manolis Savva, et al. 2018a. On evaluation of embodied navigation agents. *arXiv preprint arXiv:1807.06757*.

Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton van den Hengel. 2018b. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3674–3683.

Andrea Burns, Donghyun Kim, Derry Wijaya, Kate Saenko, and Bryan A Plummer. 2020. Learning to scale multilingual representations for vision-language tasks. In *European Conference on Computer Vision*, pages 197–213. Springer.

Howard Chen, Alane Suhr, Dipendra Misra, Noah Snavely, and Yoav Artzi. 2019. Touchdown: Natural language navigation and spatial reasoning in visual street environments. In *CVPR*, pages 12538–12547.

Shizhe Chen, Pierre-Louis Guhur, Cordelia Schmid, and Ivan Laptev. 2021. History aware multimodal transformer for vision-and-language navigation. *Advances in Neural Information Processing Systems*, 34.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, pages 4171–4186.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*.

Daniel Fried, Ronghang Hu, Volkan Cirik, Anna Rohrbach, Jacob Andreas, Louis-Philippe Morency, Taylor Berg-Kirkpatrick, Kate Saenko, Dan Klein, and Trevor Darrell. 2018. Speaker-follower models for vision-and-language navigation. In *Advances in Neural Information Processing Systems*, pages 3314–3325.

Tsu-Jui Fu, Xin Eric Wang, Matthew F Peterson, Scott T Grafton, Miguel P Eckstein, and William Yang Wang. 2020. Counterfactual vision-and-language navigation via adversarial path sampler. In *European Conference on Computer Vision*, pages 71–86. Springer.

Haoyuan Gao, Junhua Mao, Jie Zhou, Zhiheng Huang, Lei Wang, and Wei Xu. 2015. Are you talking to a machine? dataset and methods for multilingual image question answering. *Neurips*.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. In *EMNLP*.

Jiuxiang Gu, Shafiq Joty, Jianfei Cai, and Gang Wang. 2018. Unpaired image captioning by language pivoting. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 503–519.

Agrim Gupta, Piotr Dollar, and Ross Girshick. 2019. Lvis: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5356–5364.

Deepak Gupta, Pabitra Lenka, Asif Ekbal, and Pushpak Bhattacharyya. 2020. A unified framework for multilingual and code-mixed visual question answering. In *ACL*, pages 900–913.

Weituo Hao, Chunyuan Li, Xiujun Li, Lawrence Carin, and Jianfeng Gao. 2020. Towards learning a generic agent for vision-and-language navigation via pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13137–13146.

Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. 2017. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Geoffrey Hinton, Nitish Srivastava, and Kevin Swersky. 2012. Neural networks for machine learning lecture 6a overview of mini-batch gradient descent. *Cited on*, 14(8).

Yicong Hong, Qi Wu, Yuankai Qi, Cristian Rodriguez-Opazo, and Stephen Gould. 2021. A recurrent vision-and-language bert for navigation. In *CVPR*.

Haoshuo Huang, Vihan Jain, Harsh Mehta, Jason Baldridge, and E. Ie. 2019a. Multi-modal discriminative model for vision-and-language navigation. *SpLU-RoboNLP Workshop at NAACL*, abs/1905.13358.

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

Po-Yao Huang, Junjie Hu, Xiaojun Chang, and Alexander Hauptmann. 2020. Unsupervised multimodal neural machine translation with pseudo visual pivoting. *ACL*.

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

Liyiming Ke, Xiujun Li, Yonatan Bisk, Ari Holtzman, Zhe Gan, Jingjing Liu, Jianfeng Gao, Yejin Choi, and Siddhartha Srinivasa. 2019. Tactical rewind: Self-correction via backtracking in vision-and-language navigation. In *CVPR*, pages 6741–6749.

Donghyun Kim, Kuniaki Saito, Kate Saenko, Stan Sclaroff, and Bryan Plummer. 2020. Mule: Multimodal universal language embedding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11254–11261.

Hyounghun Kim, Jialu Li, and Mohit Bansal. 2021. Ndh-full: Learning and evaluating navigational agents on full-length dialogue. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6432–6442.

Jacob Krantz, Erik Wijmans, Arjun Majumdar, Dhruv Batra, and Stefan Lee. 2020. Beyond the nav-graph: Vision-and-language navigation in continuous environments. In *ECCV*, pages 104–120.

Alexander Ku, Peter Anderson, Roma Patel, Eugene Ie, and Jason Baldridge. 2020. Room-across-room: Multilingual vision-and-language navigation with dense spatiotemporal grounding. In *EMNLP*, pages 4392–4412.

Weiyu Lan, Xirong Li, and Jianfeng Dong. 2017. Fluency-guided cross-lingual image captioning. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1549–1557.

Federico Landi, Lorenzo Baraldi, Marcella Cornia, Massimiliano Corsini, and Rita Cucchiara. 2021. Multimodal attention networks for low-level vision-and-language navigation. *Computer Vision and Image Understanding*.

Jialu Li, Hao Tan, and Mohit Bansal. 2021. Improving cross-modal alignment in vision language navigation via syntactic information. In *ACL*, pages 1041–1050.

Jialu Li, Hao Tan, and Mohit Bansal. 2022. Envedit: Environment editing for vision-and-language navigation. In *CVPR*.

Xiujun Li, C. Li, Qiaolin Xia, Yonatan Bisk, A. Çelikyilmaz, Jianfeng Gao, Noah A. Smith, and Yejin Choi. 2019. Robust navigation with language pretraining and stochastic sampling. In *EMNLP/IJCNLP*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. *ICLR*.

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

Chih-Yao Ma, Zuxuan Wu, Ghassan AlRegib, Caiming Xiong, and Zsolt Kira. 2019b. The regretful agent: Heuristic-aided navigation through progress estimation. In *CVPR*, pages 6732–6740.

Gabriel Ilharco Magalhaes, Vihan Jain, Alexander Ku, Eugene Ie, and Jason Baldridge. 2019. General evaluation for instruction conditioned navigation using dynamic time warping. In *NeurIPS Visually Grounded Interaction and Language (ViGIL) Workshop*.

Arjun Majumdar, Ayush Shrivastava, Stefan Lee, Peter Anderson, Devi Parikh, and Dhruv Batra. 2020. Improving vision-and-language navigation with image-text pairs from the web. In *ECCV*.

Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. 2016. Asynchronous methods for deep reinforcement learning. In *International conference on machine learning*, pages 1928–1937.

Khanh Nguyen and Hal Daumé III. 2019. Help, anna! visual navigation with natural multimodal assistance via retrospective curiosity-encouraging imitation learning. In *EMNLP-IJCNLP*, pages 684–695.

Vineel Pratap, Anuroop Sriram, Paden Tomasello, Awni Hannun, Vitaliy Liptchinsky, Gabriel Synnaeve, and Ronan Collobert. 2020. Massively multilingual asr: 50 languages, 1 model, 1 billion parameters. *arXiv preprint arXiv:2007.03001*.

Yuankai Qi, Zizheng Pan, S. Zhang, A. V. D. Hengel, and Qi Wu. 2020a. Object-and-action aware model for visual language navigation. In *ECCV*.

Yuankai Qi, Qi Wu, Peter Anderson, Xin Wang, William Yang Wang, Chunhua Shen, and Anton van den Hengel. 2020b. Reverie: Remote embodied visual referring expression in real indoor environments. In *CVPR*, pages 9982–9991.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. *Image*, 2:T2.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. 2015. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252.

Sheng Shen, Liunian Harold Li, Hao Tan, Mohit Bansal, Anna Rohrbach, Kai-Wei Chang, Zhewei Yao, and Kurt Keutzer. 2022. How much can clip benefit vision-and-language tasks? *ICLR*.

Nobuyuki Shimizu, Na Rong, and Takashi Miyazaki. 2018. Visual question answering dataset for bilingual image understanding: A study of cross-lingual transfer using attention maps. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1918–1928.

Gunnar A Sigurdsson, Jean-Baptiste Alayrac, Aida Nematzadeh, Lucas Smaira, Mateusz Malinowski, João Carreira, Phil Blunsom, and Andrew Zisserman. 2020. Visual grounding in video for unsupervised word translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10850–10859.

Dídac Surís, Dave Epstein, and Carl Vondrick. 2022. Globetrotter: Unsupervised multilingual translation from visual alignment. *CVPR*.

Hao Tan, Licheng Yu, and Mohit Bansal. 2019. Learning to navigate unseen environments: Back translation with environmental dropout. In *NAACL*, pages 2610–2621.

Jesse Thomason, Michael Murray, Maya Cakmak, and Luke Zettlemoyer. 2019. Vision-and-dialog navigation. In *arXiv*.

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

Hanqing Wang, Wenguan Wang, Tianmin Shu, Wei Liang, and Jianbing Shen. 2020a. Active visual information gathering for vision-language navigation. In *ECCV*.

Hu Wang, Qi Wu, and Chunhua Shen. 2020b. Soft expert reward learning for vision-and-language navigation. *ECCV*.

Xin Wang, Qiuyuan Huang, Asli Celikyilmaz, Jianfeng Gao, Dinghan Shen, Yuan-Fang Wang, William Yang Wang, and Lei Zhang. 2019a. 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*, pages 6629–6638.

Xin Wang, Vihan Jain, Eugene Ie, William Yang Wang, Zornitsa Kozareva, and Sujith Ravi. 2020c. Environment-agnostic multitask learning for natural language grounded navigation. *ECCV*.

Xin Wang, Jiawei Wu, Junkun Chen, Lei Li, Yuan-Fang Wang, and William Yang Wang. 2019b. Vatex: A large-scale, high-quality multilingual dataset for video-and-language research. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4581–4591.

Xin Wang, Wenhan Xiong, Hongmin Wang, and William Wang. 2018. Look before you leap: Bridging model-free and model-based reinforcement learning for planned-ahead vision-and-language navigation. In *ECCV*.

Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. 2019. Detectron2. https://github.com/facebookresearch/detectron2.

Qiaolin Xia, Xiujun Li, C. Li, Yonatan Bisk, Zhifang Sui, Yejin Choi, and N. A. Smith. 2020. Multi-view learning for vision-and-language navigation. *ArXiv*, abs/2003.00857.

Yubo Zhang, Hao Tan, and Mohit Bansal. 2020. Diagnosing the environment bias in vision-and-language navigation. In *IJCAI 2020*, pages 890–897. ijcai.org.

Fengda Zhu, Xiwen Liang, Yi Zhu, Qizhi Yu, Xiaojun Chang, and Xiaodan Liang. 2021. Soon: Scenario oriented object navigation with graph-based exploration. In *CVPR*, pages 12689–12699.

Wang Zhu, Hexiang Hu, Jiacheng Chen, Zhiwei Deng, Vihan Jain, Eugene Ie, and Fei Sha. 2020. Babywalk: Going farther in vision-and-language navigation by taking baby steps. In *ACL*, pages 2539–2556.

Appendix

A Overview

In this supplementary, we provide a detailed description of our navigation model structure (Sec. B), representation learning and navigation learning objective (Sec. C), dataset (Sec. D), evaluation metrics (Sec. E), implementation details (Sec. F), and additional analysis in the last four sections. In this analysis, we first show that using object-matching as constraints during visual representation learning improves the nDTW score (Sec. G). Then we show that our CLEAR approach decreases the performance variance among different environments (Sec. H) and learn better alignment between the instruction and the environment (Sec. J). We further analyze whether the word representation from our learned cross-lingual representation also learn the visual/spatial information (Sec. I). Moreover, we investigate the effect of filtering out low-quality paths (Sec. K). Lastly, we show the high correspondence between instruction similarity and path pair alignment in Sec. L.

B Navigation Model

Our navigation agent follows the decoder structure as Tan et al. (2019).

At each time step t, the agent perceives a panoramic view of the current location. The panoramic view is discretized into 36 single views $\{o_{t,m}\}_{m=1}^{36}$ (12 angles and 3 camera poses per angle). Given the visual representation for each view $\widehat{v}_{t,m}$, we concatenate it with the orientation feature to get the view features $\{f_{t,m}\}_{m=1}^{36}$:

$$f_{t,m} = [\widehat{v}_{t,m}; (\cos\theta_{t,m}, \sin\theta_{t,m}, \cos\phi_{t,m}, \sin\phi_{t,m})] \quad (11)$$

where $\theta_{t,m}$ and $\phi_{t,m}$ the heading and elevation of view $o_{t,m}$.

As a reaction to the input, the agent needs to select one of the K navigable locations as an action. The action is represented as the orientation features (heading and elevation) between the current viewpoint and the chosen navigable viewpoint. The navigation decoder takes the attended visual feature $\widehat{f}_t$ of the current viewpoint and the previous action embedding a_{t-1} as input, and updates its environment-aware context vector h_t:

$$\gamma_{t,m} = \text{Softmax}_m(f_{t,m}^T W_f \widehat{h}_{t-1}) \quad (12)$$

$$\widehat{f}_t = \sum_m \gamma_{t,m} f_{t,m} \quad (13)$$

$$h_t = \text{LSTM}([\widehat{f}_t; a_{t-1}], \widehat{h}_{t-1}) \quad (14)$$

where a_{t-1} is represented as the orientation features $(\cos\theta_{t-1,k^\star}, \sin\theta_{t-1,k^\star}, \cos\phi_{t-1,k^\star}, \sin\phi_{t-1,k^\star})$ of the chosen navigable viewpoint $k^\star$ at time step $t-1$, and $\widehat{h}_{t-1}$ is the instruction-aware context vector that incorporates the attended instruction information. The navigator calculates the probability of moving to the k-th navigable location based on the alignment between the visual feature $g_{t,k}$ of that navigable location and the instruction-aware context vector $\widehat{h}_t$:

$$\rho_{t,j} = \text{Softmax}_j(\widehat{w}_j^T W_l h_t) \quad (15)$$

$$u_t = \sum_j \rho_{t,j} \widehat{w}_j \quad (16)$$

$$\widehat{h}_t = \tanh(W_m[u_t; h_t]) \quad (17)$$

$$p(a_t = k) = \text{Softmax}_k(g_{t,k}^T W_a \widehat{h}_t) \quad (18)$$

where $g_{t,k}$ is constructed similarly as $f_{t,i}$ in Eqn. 11, and $\widehat{w}_j$ is the language representation.

C Learning

Our CLEAR agent has two stages of learning: representation learning and navigation learning.

In the representation learning stage, given a pair of instructions that correspond to the same navigation path, we train the shared multilingual encoder to generate representations of paired instructions close to each other by optimizing a contrastive loss L_{lang}. Furthermore, we train the visual encoder to learn the connections between paths with similar instructions by optimizing the contrastive loss L_{visual}. The representation learning process transfers the language representation to domain-specific

language representation and adapts the visual representation to learn the correlation underlying the navigation environments.

In the navigation learning stage, we use a mixture of imitation learning and reinforcement learning to train the agent on the navigation task as in Tan et al. (2019).

In imitation learning, we use teacher-forcing to determine the next navigable viewpoint. Different from previous methods (Hong et al., 2021; Tan et al., 2019; Huang et al., 2019b) that takes the shortest path as the teacher action, our teacher action $a_t^\star$ at each time step t is picked based on the given ground truth path between the start point and target point. The agent tries to imitate the teacher action $a_t^\star$ by minimizing the negative log probability:

$$L_{IL} = \sum_t -a_t^\star \log p_t \tag{19}$$

We combine reinforcement learning with imitation learning to learn a more generalizable agent. At each time step t, the agent samples an action a_t from the predicted distribution $p_t(a_t)$. We follow (Hong et al., 2021) to do the reward shaping. The immediate reward at each time step t consists of three parts. First, if the agent moves closer to the target viewpoint, a positive reward +1 is given, otherwise the agent receives a negative reward -1. Second, to encourage instruction following, we include normalized Dynamic Time Warping (nDTW) score in the reward. The agent gets a positive reward if the nDTW score for the navigated path increases. Lastly, the agent receives a negative reward if it misses the target. When the agent predicts the "STOP" action, the agent will receive a +3/-3 reward based on whether the agent is within 3 meters from the target viewpoint. We use Advantage Actor-Critic (Mnih et al., 2016) to train the agent.

The navigation loss L_{nav} is a weighted combination of imitation learning loss and reinforcement learning loss.

$$L_{nav} = L_{RL} + \lambda L_{IL} \tag{20}$$

D Dataset

We evaluate our agent on the Room-Across-Room (RxR) dataset (Ku et al., 2020). The dataset is built on the Matterport3D simulator (Anderson et al., 2018b). It contains 126,069 human-annotated instructions with an average instruction length of 78.

Methods	SR↑	SPL↑	NDTW↑	SDTW↑
+visual	35.6	32.5	53.7	30.5
-sample	37.8	33.7	53.0	32.1
-object	36.6	33.0	52.4	30.9

Table 7: Comparison between visual representation trained with objects constraints ('+visual'), without sampling strategy ('-sample') and without object constraints ('-object') on validation unseen sets. nDTW is the main metric for Room-Across-Room (RxR) dataset.

The dataset is split into training set, seen validation set, unseen validation set, and test set. In the unseen validation set and test set, the environments do not appear in the training set. Thus the performance on these two sets show the model's generalizability to new environments. There are 16,522 paths in total, and each path is annotated in 3 languages (and 3 instructions per language on average). The training set contains 11,089 paths, the seen validation set contains 1,232 paths, the unseen validation contains 1,517 paths, and the test set contains 2,684 paths.

E Evaluation Metrics

To evaluate the performance of our model, we follow the metrics used in the Room-Across-Room paper (Ku et al., 2020). The metrics include: (1) Success Rate (SR): We consider a success for navigation if the agent stops less than 3m from the target location. (2) Success rate weighted by Path Length (SPL) (Anderson et al., 2018a): This metric penalizes the navigation with long paths (i.e., when both navigations reach the target, the navigation with shorter path length has a higher SPL score). (3) normalized Dynamic Time Warping (nDTW) (Magalhaes et al., 2019): This metric measures the path fidelity by penalizing deviations from the reference path. The agent navigates to the target through the shortest path instead of instruction following will be penalized. (4) success rate weighted by Dynamic Time Warping (sDTW) (Magalhaes et al., 2019): This metric only considers nDTW of successful navigation and ignores failed navigation. Normalized Dynamic Time Warping (nDTW) is the main metrics for RxR and Success Rate (SR) and Success rate weighted by Path Length (SPL) are the main metrics for R2R.

F Implementation Details

In our experiments, we learn the shared multilingual representation based on cased multilingual $\text{BERT}_{\text{BASE}}$. The instruction is truncated from the end with a maximum sequence length of 160. For the pre-trained vision model, we compare performance between image features extracted from ImageNet-pre-trained (Russakovsky et al., 2015) ResNet-152 (He et al., 2016) and CLIP-pre-trained (Radford et al., 2021) vision transformer (ViT-B/32) (Dosovitskiy et al., 2021) (abbreviated as 'CLIP feature' later). The 27 object classes are: 'drawer', 'faucet', 'cabinet', 'hinge', 'cushion', 'sofa', 'chair', 'pillow', 'armchair', 'lamp', 'vase', 'knob', 'curtain', 'statue(sculpture)', 'doorknob', 'vent', 'lightbulb', 'flowerpot', 'book', 'pipe', 'painting', 'wall socket', 'bed', 'mirror', 'television set', 'flower arrangement', 'chandelier'. The navigation decoder's hidden size is 768 and the action embedding size is 128. The language encoder is optimized with AdamW (Loshchilov and Hutter, 2019) with linear-decayed learning rate. The peak learning rate is 4e-5 for both the representation learning and the navigation agent learning stage. The visual encoder, the navigation decoder, and the discriminator are optimized with RMSProp (Hinton et al., 2012) with learning rate 1e-4. The weight λ we use to combine loss is set to be 0.4 for the ResNet-based full model and 0.2 for the CLIP-based full model. The batch size for training ResNet features and CLIP features are 12 and 16, respectively. During training, CLIP model is around 1.5 times faster than ResNet model in this setting since CLIP features are 512 dimensions while ResNet features are 2048 dimensions. To keep roughly the same amount of training time, we train the agent with ResNet features for 100K iterations, while we train model CLIP-ViT features for 150K iterations.

G Analysis: Effectiveness of Object-Matching Constraints

Our visual representation learning optimizes the similarity between panoramic views at each step of the semantically-aligned path pairs. Since paths are not fully-aligned, we use object-matching as a constraints to filter out panoramic view pairs that don't contain same objects. As shown in Table 7, the visual representation trained with fixed object classes as constraints ('-sample') improve the nDTW score (the main metric for RxR dataset) by 0.6% compared with the visual representation trained without object-matching constraints ('-object'), suggesting that using object-matching as constraints help learn a better visual representation. Besides, the sampling strategy (i.e., randomly sample 10 object classes from 27 object classes during each iteration) also helps the visual representation learning ('+visual'), further improving the nDTW score by 0.7% compared with the visual representation learned with fixed 27 object classes ('-sample'). In total, our object-matching constraints and sampling strategy ('+visual') improves the performance by 1.3% in nDTW score compared with learning without object constraints ('-object').

H Analysis: Performance Variance Reduction among Different Environments

We demonstrate that our CLEAR approach could decrease the performance variance (i.e., performance's standard deviation) among different environments. Intuitively, we hope the agent to perform equally well in different environments instead of getting high performance by only learning to navigate through several easy environments. We show the results for 11 environments in validation unseen set in Table 8. Our CLEAR approach ('+both' as in Table 2 in the main paper) outperforms the baseline model ('ResNet' as in Table 2 in the main paper) in most of the environments. Moreover, the weighted standard deviation (weighted by # Data in Table 8) of our CLEAR approach is lower than the baseline model. Specifically, the standard deviation of nDTW score for our CLEAR approach is 9.24 while the standard deviation of nDTW score for the baseline model is 10.01, suggesting that our CLEAR approach decreases the performance variance between different environments.

I Analysis: Word Representation from Cross-Lingual Representation

The visual semantics are injected during learning the cross-lingual language representation by maximizing the similarity between full instruction sentences (representation of 'CLS' token). However, it's unclear that whether the word-level representation also learned such visual information. In this section, we investigate whether the learning encodes spatially close words/objects closer to each other. As shown in Table 9, we check the top-5 close words to 'kitchen', and 'fire' from a vocabu-

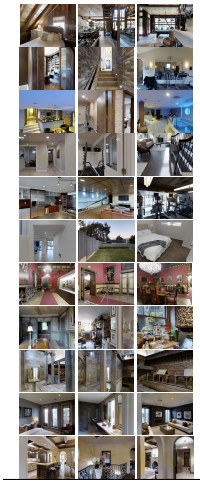
Environment	# Data	ResNet				CLEAR			
		SR	SPL	NDTW	SDTW	SR	SPL	NDTW	SDTW
	1206	32.4	26.7	49.8	26.5	35.9	30.2	50.0	28.0
	2177	27.0	23.8	41.3	22.3	28.6	26.0	47.5	24.4
	567	38.1	34.3	56.9	31.8	48.0	44.8	64.4	40.4
	1692	38.3	34.8	56.1	33.2	39.5	36.0	57.6	33.4
	153	57.5	54.7	72.1	53.1	64.1	60.0	74.5	57.3
	1404	42.7	38.7	58.3	37.8	41.3	39.1	61.1	36.6
	900	52.0	49.5	67.9	46.3	45.7	44.2	65.9	41.0
	2223	40.8	36.9	57.7	35.5	44.0	39.2	60.3	38.2
	18	38.9	32.7	59.4	34.0	50.0	45.1	70.8	47.1
	1152	42.4	38.6	54.6	36.1	38.1	35.2	55.1	33.3
	2160	18.1	16.4	34.6	14.7	15.9	13.1	37.1	13.1

Table 8: The results of our CLEAR method and ResNet baseline on different environments in validation unseen set. # Data means the number of instruction-path pairs for each environment.

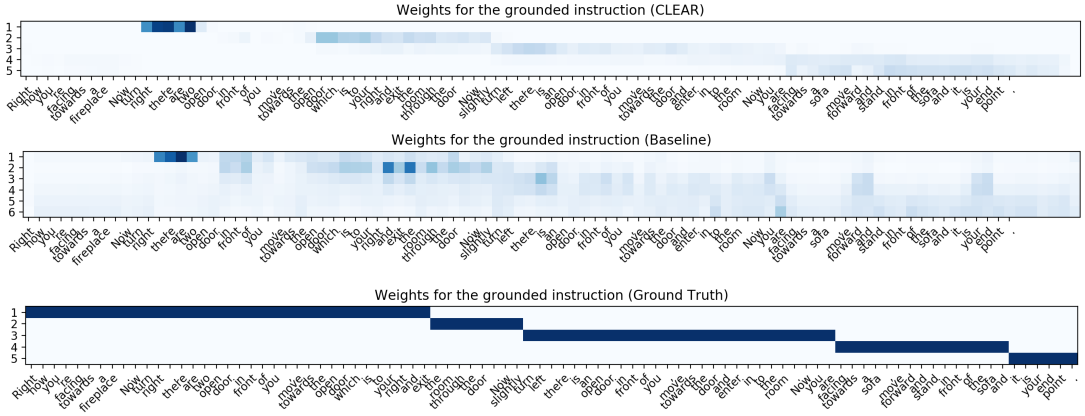


Figure 4: The attention weights for the grounded instruction for our CLEAR model, ResNet based baseline model, and ground truth from RxR dataset.

lary of 2754 English tokens. We see that our cross-lingual representation puts words that appear spatially near each other close (e.g. 'kitchen' and 'island'/'dinning', 'fire' and 'chair'/'fireplace') while m-BERT representation fails (e.g. 'kitchen' and 'room'/'house', 'fire' and 'family'/'study').

J Analysis: Alignment between Instructions and Environments

The Room-Across-Room dataset provides ground-truth alignment between instructions and navigation paths. To demonstrate that our CLEAR approach learns a good alignment between instructions and paths, we not only compare our CLEAR approach with the baseline approach, but also compare it with the ground truth alignment provided in the RxR dataset. The attention weights for grounded instruction for CLEAR, Baseline, and Ground Truth are shown in Figure 4. We observe that our CLEAR model successfully attends to sub-instructions "turn right", "move towards the open door to your right and exit the room through the door", "slightly turn left", "move towards and stand in front of the sofa" sequentially. Although the baseline model also successfully executes the first two sub-instructions "turn right" and "move towards the open door", yet the baseline agent gets lost in the later navigation. Furthermore, the alignment learned by our CLEAR approach matches better with the ground truth alignment provided in the RxR dataset.

K Analysis: Filtering out Low Quality Path Pairs

We investigate whether filtering out low-quality path pairs during visual representation learning

Word	Top-5
kitchen	'island', 'counter', 'maker', 'din', '##iding'
	'living', 'counter', 'room', 'table', 'house'
fire	'##place', 'over', 'place', 'chair', '##fas'
	'display', 'study', 'family', 'living', 'coffee'

Table 9: Top-5 closest tokens for 'Kitchen' and 'fire'. Top-row: tokens picked by our cross-lingual representation. Bottom-row: tokens picked by multi-lingual BERT baseline.

Similarity	SR↑	SPL↑	NDTW↑	SDTW↑
0.00	35.6	32.5	53.7	30.5
0.90	36.2	32.2	51.9	30.7
0.95	38.6	34.3	53.3	33.0
0.98	38.6	34.3	52.9	32.9
0.99	37.8	33.5	52.6	32.0
1.00	30.9	28.0	49.7	26.1

Table 10: Performance in validation unseen environment when filtering out different percentages of data in training our visual representation. 0.90 means filter out data with similarity score less than 0.90.

could further improve the performance. Since our identified path pairs are retrieved based on the similarity between instructions, we hypothesize that the path pair is aligned better if having a higher instruction similarity score. Thus, we experiment with filtering out instruction pairs that have a cosine similarity score less than 0.90, 0.95, 0.98, and 0.99, and then train the visual representation with filtered data and object-matching constraints. The proportion of filtered-out data is 1%, 6%, 28% and 58% respectively. We also experiment with filtering out 0% and 100%. Filtering out 0% of the data is the same to our proposed environment-agnostic visual representation ('+visual' in Table 2) and filtering out 100% of the data is analogous to randomly initialize the visual encoder[3]. We then train our environment-agnostic representation (in Sec. 3.2) based on the remaining data and show its performance on the validation unseen environments. As shown in Table 10, though the success rate improves when filtering out some path pairs with lower quality, not filtering out any path pairs achieve the highest nDTW score. This demonstrates that using object-matching constraints without filtering out path pairs with low instruction similarity is enough for learning a good visual representation. Furthermore, we see a significant performance drop when not fine-tuning the visual representation on any data, which indicates that training the visual encoder with semantically-aligned path pairs is important for agent performance.

[3]Note that filtering out 100% of the data is not the same as the baseline model ('ResNet' in Table 2). The baseline model does not have the visual encoder we introduced in Sec. 3.2

L Analysis: Correspondence between Instruction Similarity and Path Pair Alignment

In this section, we show that instruction pairs that have high similarity have similar BLEU score and ROUGE score to the instruction pairs that corresponding to the same path. Specifically, the BLEU-1 and ROUGE-L score for instruction pairs that have high similarity are 0.42 and 0.320, and the BLEU-1 and ROUGE-L score for the instruction pairs that corresponding to the same path are 0.41 and 0.323. Randomly picking gets 0.37 BLEU-1 score and 0.295 ROUGE-L score. These results indicate that high similarity instruction pairs may be of competitive quality as the instruction pairs that corresponding to the same path, and can be used to pick the semantically-aligned path pairs.