Explicit Object Relation Alignment for Vision and Language Navigation

Yue Zhang
Michigan State University
zhan1624@msu.edu

Parisa Kordjamshidi
Michigan State University
kordjams@msu.edu

Abstract

In this paper, we investigate the problem of vision and language navigation. To solve this problem, grounding the landmarks and spatial relations in the textual instructions into visual modality is important. We propose a neural agent named Explicit Object Relation Alignment Agent (EXOR), to explicitly align the spatial information in both instruction and the visual environment, including landmarks and spatial relationships between the agent and landmarks. Empirically, our proposed method surpasses the baseline by a large margin on the R2R dataset. We provide a comprehensive analysis to show our model's spatial reasoning ability and explainability.

1 Introduction

Vision and Language Navigation (VLN) problem (Anderson et al., 2018) requires the agent to carry out a sequence of actions in an indoor photo-realistic simulated environment in response to corresponding natural language instructions. The first VLN benchmark to appear was Room-to-Room navigation (R2R) (Anderson et al., 2018) , as shown in Figure 1, the agent needs to generate a navigation trajectory in a visual environment rendered from real images following an instruction.

This task is challenging because, apart from understanding the language and vision modalities, the agent needs to learn the connection between them without explicit intermediate supervision.

To address this challenge, several recent work started to consider the semantic structure from both language and vision sides. Hong et al. (2020a) train an implicit entity-relationship graph allowing an agent to learn the latent concepts and relationships between different components (scene, object and direction). They use the object features extracted from Faster-RCNN (Ren et al., 2015) instead of only using ResNet visual features which can easily overfit on the training environment (Hu et al., 2019).

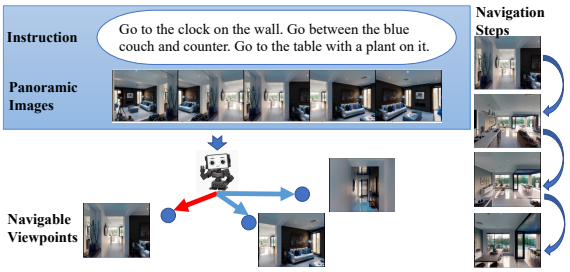


Figure 1: **VLN Task Demonstration.** The agent generates a navigation trajectory composed of navigable viewpoints selected based on the given instruction and the panoramic images at each step. The **red** arrow shows the ground-truth navigable viewpoint.

Although the grounding ability of their agent improves, their experimental results show that the object features do not help the navigation independently unless their relationships to the scene and direction are modeled. This issue indicates their loosely modeled latent space and motivates us to explore the ways the object features can be further exploited.

The recent research finds that indoor navigation agents rely on both landmark and direction tokens in the instruction when taking actions (Zhu et al., 2021). However, it is difficult to identify which landmarks the agent should pay attention to at each navigation step. Previous works (Tan et al., 2019; Ma et al., 2018; Wang et al., 2019; Zhu et al., 2020) mainly use the surrounding visual information as a clue to indicate the landmark tokens that the agent should focus on. However, the semantics of instruction should also play an important role. For example, with the understanding of the instruction "go to the table with chair, and then walk towards the door", the agent needs to give the same attention to "table" and "chair", and less attention to "door" at the first navigation step. In terms of direction tokens, the prior works concentrate most on the direction tokens related to motions, such as "turn left", and ignore the spatial description of

Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics
Student Research Workshop, pages 322 - 331

landmarks, such as "table on the left". We believe distinguishing those different direction tokens can benefit the navigation performance. The last but not least, modeling the landmarks and their spatial relations can improve the explainability of the agent's actions.

In this paper, we propose a neural agent, called *Explicit Object Relation Alignment Agent* (EXOR), to explicitly align the spatial semantics between linguistic instructions and the visual environment. Specifically, we first split the long instruction into spatial configurations (Dan et al., 2020; Zhang et al., 2021), and then we select the important landmarks based on such configurations. After that, in the visual environment, we retrieve the most relevant objects according to their similarity with the selected landmarks in the instructions. Moreover, we obtain **textual spatial relation encoding** to model the spatial relations between the agent and landmarks in the textual instructions, and use **visual spatial relation encoding** to represent the relation between agent and the image in the visual environment. We then establish a mapping between the two encodings to achieve a better alignment. Finally, we use the representations of the aligned objects and spatial relations to enrich the image representations. To the best of our knowledge, none of the previous work modeled the explicit spatial relations considering the agent's perspective for this task.

Our contribution is summarized as follows:

1. Our model achieves the explicit alignments between textual and visual spatial information, and such alignments guide the agent to pay more attention to the objects in the visual environment given landmark mentions in the instructions.

2. We explicitly model the spatial relations between the agent and landmarks from both instruction and visual environments, which enhance their alignments and improve the overall navigation performance.

3. Our method surpasses the baseline performance by a large margin. Also, we provide a comprehensive analysis to show the spatial reasoning ability and explainability of our model.

2 Related Work

Vision and Language Navigation The vision-language navigation problem nowadays has gained an increasing popularity, and various navigation datasets and platforms(Savva et al., 2019; Kolve et al., 2017) are proposed to assist the development of this topic in the community, for example, R2R (Anderson et al., 2018) and Touchdown (Chen et al., 2019) datasets, which have extended navigation to the photo-realistic simulation environments. More broadly, there are work also related to instruction-guided household task benchmarks such as ALFRED (Shridhar et al., 2020). RXR (Ku et al., 2020) is a multilingual navigation dataset with spatial-temporal grounding, CVDN and HANNA are a dialog-based interactive navigation dataset (Thomason et al., 2020; Nguyen and Daumé III, 2019), and REVERI (Qi et al., 2020b) navigates to localize a remote object.

Accompanied with these benchmark works, numerous deep learning methods (Tan et al., 2019; Hong et al., 2021, 2020a) have been proposed. For R2R task, Anderson et al. (2018) propose a Sequence-to-Sequence baseline model to encode the instructions and decode the embeddings to the low-level action sequence with the observed images. Speaker-Follower agent proposed by Fried et al. (2018) trains a speaker model to generate the augmented samples to improve the generalizability. They also start modeling a panoramic action space for navigation, which further promotes fast iteration of different VLN approaches.

Grounding in VLN It has been observed that the connection between linguistic instruction and visual environment can yield a great improvement in VLN task, hence many research efforts for modeling such visual-linguistic relation have recently been developed. In general, we categorize these research works into three directions.

The first main thread (Anderson et al., 2018; Ma et al., 2018; Tan et al., 2019; Wang et al., 2019; Ma et al., 2019) tends to adopt attention mechanisms for establishing language and vision connections in neural navigation agents. For instance, Ma et al. (2019) apply a visual-textual co-grounding module and a progress monitor to guide the execution progress. The second branch of prior works (Hu et al., 2019; Hao et al., 2020; Majumdar et al., 2020; Hong et al., 2021; Shen et al., 2021) explores the pre-trained Vision and Language (VL) representation from the transformer-based models. Hong et al. (2021) design a recurrent unit on the VL transformer models, and fine-tune them on the down stream VLN task. Notably, the increased model size and additional training process help improve navigation performance and surpass the previous

performance by a large margin.

The third branch works model the semantic structure, based on both language and vision perspectives, to improve the grounding ability, such as (Qi et al., 2020a; Zhang et al., 2021; Hong et al., 2020a,b; Li et al., 2021), and our work also follows such paradigm. From the language side, Hong et al. (2020b) segment the long instruction into sub-instructions and annotate their corresponding trajectories to supervise the agent to learn the alignments. From the image side, instead of using only the ResNet visual features that easily over-fits on the training environment, some recent work (Hu et al., 2019; Qi et al., 2020a; Zhang et al., 2020) use object representations to improve the generalizability. Most importantly, one should bridge both linguistic and visual semantics, and Ent-Rel (Hong et al., 2020a) obtains the best results in the third branch of work by building an implicit language-visual entity relation graph to learn the connections between the two modalities. Our work serves as a new method in the third method category. We explicitly model the alignments between landmarks and visual objects and model the spatial relations to improve the spatial reasoning ability of the agent.

3 Method

3.1 Problem Description

In our study, the agent is given an instruction with length l, denoted as $w =< w_1, w_2, \cdots, w_l >$. At each time step t, the agent observes its surrounding and receives 360-degree panoramic views of images, which are denoted as $v^p =< v_1^p, v_2^p, \cdots, v_{36}^p >$.[1] In those panoramic views, there are q candidate navigable viewpoints which the agent can navigate. We denote the viewpoints as $v^c =< v_1^c, v_2^c, \cdots, v_q^c >$. The goal of the task is to select the next viewpoint among the navigable viewpoints for generating a trajectory that takes the agent close to a goal destination. The agent terminates when the current viewpoint is selected, or a predefined maximum number of navigation steps have been reached.

3.2 Base Model

We follow the modeling approach of (Tan et al., 2019) which uses an Long short-term Memory (LSTM) based sequence-to-sequence architecture. The encoder is a bidirectional LSTM-RNN with an embedding layer to obtain language representation, denoted as, $[s_1, s_2, \cdots, s_l] = BiLSTM(F(< w_1, w_2, \cdots, w_l >)$, where F represents the embedding function. The decoder is also an attentive LSTM-RNN. At each decoding step t of navigation, the agent first attends to the panoramic image representation f^p with the previous hidden context feature $\tilde{h}_{t-1}$. The visual representation of ith panoramic image is denoted as $f_i^p = [ResNet(v_i^p); d_i]$, which is the concatenation of the ResNet visual features $ResNet(v_i^p)$ and the corresponding 128 dimensional direction encoding d_i. The direction encoding for panoramic images d_i is the replication of $[cos\theta_i, sin\theta_i, cos\phi_i, sin\phi_i]$ by 32 times, where θ_i and ϕ_i are the angles of heading and elevation of ith panoramic image. The attentive panoramic visual feature $\tilde{f}_t^p$ is computed by $\tilde{f}_t^p = SoftAttn(Q = \tilde{h}_{t-1}, K = f_t^p, V = f_t^p)$, and then is used as input to the LSTM of the decoder to represent the agent's current state as,

$$h_t = LSTM([a_{t-1}; \tilde{f}_t^p], \tilde{h}_{t-1}), \quad (1)$$

where a_{t-1} is the selected action direction of the previous navigation step, and $\tilde{h}_{t-1}$ is the hidden context after considering the grounded objects. The details will be discussed in the following sections.

3.3 Landmark-object alignment and spatial relations modeling

The proposed model has been shown in Figure 2, and we describe its four components as follows.

Spatial Configuration Representation

We split the long instructions into smaller sub-instructions, called spatial configurations. A spatial configuration contains fine-grained spatial roles, such as motion indicator, landmark, spatial indicator, and trajector (Dan et al., 2020). For example, the instruction "go to the bathroom and stop" can be split into two spatial configurations, which are "go to the bathroom" and "stop". In the first configuration, "go" is the motion indicator; "bathroom" is the landmark. In the second configuration, "stop" is the motion indicator.

We follow Zhang et al. (2021) to split a navigation instruction w into m spatial configurations based on the verbs or verb phrases. Each spatial configuration contains the flexible number of tokens and a [SEP] token as the last token. Formally, we re-organize the contextual embeddings of tokens into the array of spatial configurations representation $C = [C_1, C_2 \ldots C_m]$, where m is the number of configurations. Each configuration

[1] 12 headings and 3 elevations with 30 degree intervals.

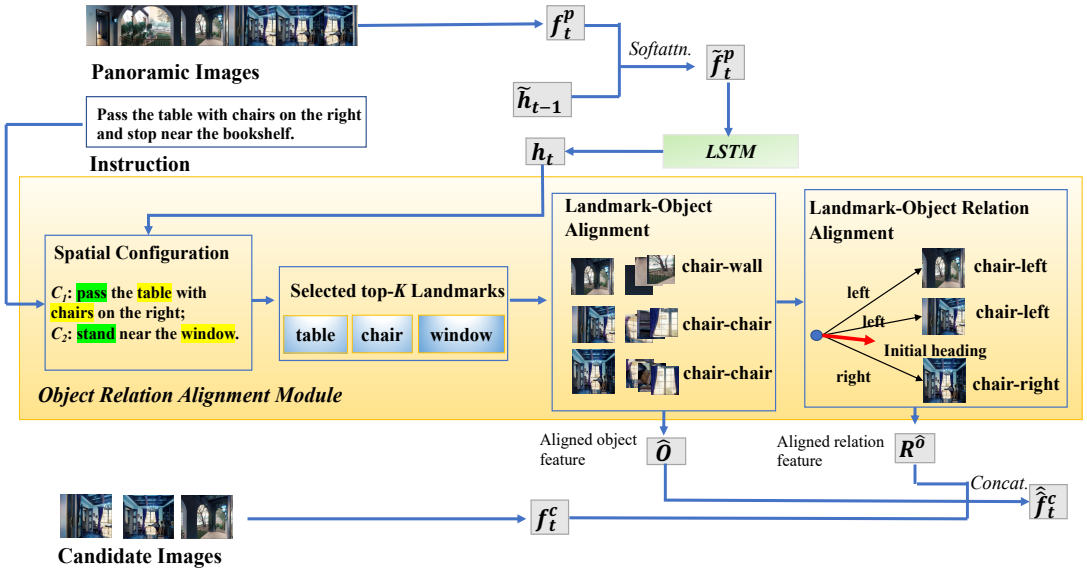


Figure 2: **Model Architecture.** The model has four sub-modules, (1) Spatial Configuration (2) Select top-k landmark selection (3) Landmark-Object alignment (4) Landmark-Object Spatial Relation relation alignment. The text highlighted in **green** and yellow in (1) shows motion indicators and landmarks, respectively. The **red** arrow in (4) is the initial agent heading (i.e. orientation).

is composed of tokens generated by the encoder, denoted as $[s_1, s_2, \cdots s_p]$, where s_p is the embedding of [SEP] token that contains the most comprehensive contextual information about the preceding words. This is because LSTM encoder is used for propagating the information throughout the sequence. Also, to enrich the spatial configuration representations, we consider the spatial semantic elements. We extract the verbs or verb phrases as motion indicators, s_m, and nouns or noun phrases as landmarks, s_l. Then we apply soft attention to each configuration representation with the representations of the [SEP] token s_p, the motion indicator s_m, and landmark s_l separately. The enriched spatial configuration is represented as $\bar{C} = [\bar{C}_1, \bar{C}_2 \ldots \bar{C}_m]$. In the base model, we attend the current hidden context h_t of the LSTM to the spatial configuration features C to form the weighted spatial configurations output $\tilde{C}$. This process is defined as follows,

$$\beta_{t,j} = softmax(\bar{C}_j^T W_c h_t), \quad (2)$$

$$\tilde{C}_t = \sum_j \beta_{t,j} C_j, \quad (3)$$

where β is the attended configuration weights, j is the index of spatial configuration and W_c is the learned weights.

Landmark Selection

Landmark phrases in instructions are split into groups according to the spatial configuration. We assign the attention weights of each spatial configuration to all its included landmarks. The attention weights of landmarks are the same once they appear in the same configuration. Then we sort all weighted landmarks and select the top-k important ones for the agent to focus on at each navigation step. Formally, each configuration contains n landmarks, denoted as $L =< L_1, L_2, \cdots, L_n >$. The total number of landmarks is $m * n$ in m spatial configurations. After sorting all landmarks based on the spatial configuration weights β, we can obtain top-k selected landmark representations, as $\tilde{L} =< \tilde{L}_1, \tilde{L}_2, \cdots, \tilde{L}_k >$. We obtain the best result when k is 3 (see 5.1 for the experiment).

Landmark-Object Alignment

After selecting top-k landmarks, the next step is to align them with the corresponding objects in the image. We use Faster-RCNN to detect 36 objects in each image, and the object representation of the i-th image is $O_i = [o_{i,1}, o_{i,2}, \cdots, o_{i,36}]$. We compute the cosine similarity scores between the j-th landmark in top-k landmarks and all objects in the i-th image, and select the object with the highest similarity score as the most relevant object to the j-th landmark, as $\hat{O}_{i,L_j} = max(cos\_sim(\tilde{L}_j, O_i))$. The aligned objects in the i-th image are denoted as

$\hat{O}_i = [\hat{O}_{i,L_1}, \hat{O}_{i,L_2}, \cdots, \hat{O}_{i,L_k}]$. We get k aligned objects since we have top-k landmarks. Finally, we concatenate the aligned object representations with the candidate image features f^c. The ith candidate image is represented as $f_i^p = [ResNet(v_i^c); d_i]$. After aligned with the corresponding objects, its representation is updated as $\hat{f}_i^c = [f_i^c; \hat{O}_i^c]$.

Landmark-Object Spatial Relation Alignment

We model both textual spatial relations and visual spatial relations. On the text side, there are mainly three different cases of spatial relations described in the navigation instructions.

- Case 1. Motions verbs, such as "turn left to the table";
- Case 2. Relative spatial relationships between agent and landmarks, such as "table on your left";
- Case 3. Spatial relationships between landmarks, such as "vase on the table".

This work mainly investigates the spatial relations from the agent's perspective, and we only model the first two cases. We extract "landmark-relation" pairs for each landmark in the instructions (based on syntactic rules). For Case 1, we pair the spatial relation with all landmarks in the configuration. For example, "turn left to the table with chair", the extracted pairs are {table-left} and {chair-left}. For Case 2, we pair the relation with the related landmark. For example, "go to the sofa on the right.", the extracted pair is {sofa-right}.

We encode the spatial relations for the landmarks in six bits $[left, right, front, back, up, down]$ as the **textual spatial relation encoding**. Each bit is set to 1 for the landmark if its paired relation has the corresponding relation. On the image side, we encode the same six spatial relations as the **visual spatial relation encoding**. We obtain the spatial relations of objects in the visual environment based on the relative angle, the differences between the agent's initial direction and the navigable direction. The spatial relations are the same for all objects if they are in the same image.

Formally, for the obtained top-k landmarks, we denote their spatial encoding as $R^{\hat{L}} = [R_1^{\hat{L}}, R_2^{\hat{L}}, \cdots, R_k^{\hat{L}}]$. For the top-$k$ objects aligned with those landmarks, the spatial relations in i-th navigable image are represented as $R_i^{\hat{O}} = [R_{i,1}^{\hat{O}}, R_{i,2}^{\hat{O}}, \cdots, R_{i,k}^{\hat{O}}]$. We compute the inner product of the spatial encoding between top-k landmarks and the top-k aligned objects to obtain the spatial similarity score between the instruction and the i-th image, that is, $sim_i^R = R^{\hat{L}} \cdot R_i^{\hat{O}}$. Then we concatenate each aligned object spatial encoding with the corresponding similarity score, denoted as $\hat{O}_{i,R} = [[R_{i,1}^{\hat{O}}; sim_{i,1}^R], [R_{i,2}^{\hat{O}}; sim_{i,2}^R], \cdots, [R_{i,k}^{\hat{O}}; sim_{i,k}^R]]$. Finally, we further concatenate $\hat{O}_{i,R}$ with the candidate image features $\hat{f}_i^c$ which is concatenated with the aligned object features , and i-th candidate images features is updated as $\hat{\tilde{f}}_i^c = [\hat{f}_i^c; \hat{O}_{i,g}]$. The updated image representations are then used to make action decisions for the agent.

3.4 Action Prediction

After modeling alignment between landmark tokens in the instruction and visual objects, the panoramic image feature is enriched with the aligned visual objects, and candidate image feature is enriched with both visual objects and their spatial relations. Then based on the backbone sequence to sequence agent, the probability of moving to the k-th navigable viewpoint $p_t(a_{t,k})$ is calculated as softmax of the alignment between the navigable viewpoint features and a context-aware hidden output $\tilde{h}_t$, which can be calculate as

$$\tilde{h}_t = tanh(W_{\tilde{c}h}[\tilde{C}; h_t]) \tag{4}$$

$$p_t(a_{t,k}) = softmax(\hat{\tilde{f}}_i^c W_{\hat{c}} \tilde{h}_t) \tag{5}$$

where $W_{\tilde{c}h}$ and $W_{\hat{c}}$ are learnt weights.

3.5 Training and Inference

We follow the work of (Tan et al., 2019) for training the model with a mixture of Imitation Learning (IL) and Reinforcement Learning (RL). Imitation Learning minimizes the cross-entropy loss of the prediction and always samples the ground-truth navigable viewpoint at each time step, and Reinforcement Learning samples an action from the action probability p_t and learns from the rewards. During inference, we use a greedy search with the highest probability of the next viewpoints to generate the trajectory.

4 Experimental Setups

Dataset

We use Room-Room(R2R) dataset (Anderson et al., 2018) that is built upon the Matterport3D dataset.

	Method	Val Seen			Val Unseen			Test(Unseen)	
		SR ↑	**SPL** ↑	**SDTW** ↑	**SR** ↑	**SPL**↑	**SDTW**↑	**SR** ↑	**SPL** ↑
1	Speaker-Follower (Fried et al., 2018)	0.54	-	-	0.27	-	-	-	-
2	Env-Drop (Tan et al., 2019)	0.55	0.53	-	0.47	0.43	-	-	-
3	Env-Drop* (Tan et al., 2019)	0.63	0.60	0.53	0.50	0.48	0.37	0.50	0.47
4	SpC-NAV* (Zhang et al., 2021)	0.65	0.61	-	0.45	0.42	-	0.46	0.44
5	OAAM* (Qi et al., 2020a)	0.65	0.62	0.53	0.54	0.50	0.39	0.53	0.50
6	Entity-Relation (Hong et al., 2020a)	0.62	0.60	0.54	0.52	0.50	0.46	0.51	0.48
7	EXOR (ours)	0.60	0.58	0.53	0.52	0.49	0.46	0.49	0.46

Table 1: **Experimental Results Comparing with Baseline Models** (* means data augmentation).

		Ent-Rel		EXOR(ours)	
		SR↑	**SPL**↑	**SR**↑	**SPL**↑
1	Mask Scene	0.47	0.44	0.48	0.46
2	No Mask	0.52	0.50	0.50	0.48

Table 2: **Results on Scene & Object Alignment.**

		Val Seen			Val Unseen		
	Method	**SR**↑	**SPL**↑	**SDTW**↑	**SR**↑	**SPL**↑	**SDTW**↑
1	Baseline	0.55	0.53	0.49	0.47	0.43	0.37
2	Lan-Obj	0.59	0.55	0.52	0.50	0.48	0.43
3	Lan-Obj+Rel	**0.60**	**0.58**	**0.53**	**0.52**	**0.49**	**0.46**
4	Lan-Obj+Rel_v	0.59	0.56	0.52	**0.52**	0.47	0.44

Table 3: **Ablation Study.**

R2R dataset contains 7198 paths and 21567 instructions with an average length of 29 words. The whole dataset is partitioned into training, seen validation, unseen validation, and unseen test set. The seen set shares the same visual environments with the training set, while unseen sets contain different environments.

Evaluation Metrics

We mainly report three evaluation metrics. (1) Success Rate (SR): the percentage of the cases where the predicted final position lays within 3 meters from the goal location. (2) Success rate weighted by normalized inverse Path Length (SPL) (Anderson et al., 2018): normalizes Success Rate by trajectory length. It considers both the effectiveness and efficiency of navigation performance. (3) the Success weighted by normalized Dynamic Time Warping (SDTW) (Ilharco et al., 2019): penalizes deviations from the referenced path and also considers the success rate.

Baseline Models

Env_Drop (Tan et al., 2019) proposes a neural agent trained with the method of the mixture of Imitation Learning and Reinforcement Learning. Our model is built based on Env_Drop.

SpC-NAV (Zhang et al., 2021) models instructions using spatial configurations and designs a state attention to guarantee the sequential execution. Besides, it uses a similarity score between landmarks in the instruction and objects in the image to control this attention.

OAAM (Qi et al., 2020a) proposes an object-and-action aware model to learn the object and action attention separately, and also learns the object-vision and action-orientation matching.

Ent-Rel (Hong et al., 2020a) proposes a language and visual entity relation graph to exploit the connection among the scene, objects, and direction clues during navigation.

Implementation Details

We use PyTorch to implement our model[2]. We use 768 dimensional BERT-base (Devlin et al., 2018) (frozen) as the embedding of the raw instruction, and get its 512 dimensional contextual embedding by LSTM. We encode the representations of the motion indicator and the landmark in each configuration with 300 dimensional GloVe embedding respectively, and concatenate them with the 512 dimensional configuration representation to obtain the enriched configuration representation (1112 dimensional). We use 300 dimensional GloVe (Pennington et al., 2014) embedding to represent motion indicator, landmark, and object label. The optimizer is ADAM, and the learning rate is $1e-4$ with a batch size of 32.

5 Results and Analysis

Table 1 shows the performance of our model compared with baselines and the competitive models of the third branch of work as aforementioned in the related work (section 2) on unseen validation and test set. Our result is better than the baseline (Env-Drop) even with their augmented data (Tan et al., 2019) (Row#1 and Row#2), showing our improved generalizability. We obtain significantly improved

[2]Our code is available at https://github.com/HLR/Object-Grounding-for-VLN

results compared to SpC-NAV which models the semantic structure in language and image modalities. Compared with OAAM, which learns the object-vision matching with the augmented data, we get better SDTW, indicating that our agent can genuinely follow the instruction to the destination. However, Ent-Rel achieves better results that ours, for which we provide further analysis in the next section.

5.1 The Number of Selected Landmarks

We experimentally validated the best number of important landmarks the agent should select. Figure 3 shows the SPL results with different k values on validation seen and unseen dataset. We find that the best result is obtained when k is 3. It also shows that letting the agent focus on only one landmark or all landmarks in the instruction will hinder their navigation performance. Table4 shows the statistics of the extracted spatial configurations in train and validation seen/unseen dataset. On average, each instruction can be split into about four spatial configurations, and about 76% of spatial configurations contain landmarks. In fact, selecting top 3 landmarks means that the agent mainly focuses on the landmark-object alignment in 3 spatial configurations at most at each navigation step.

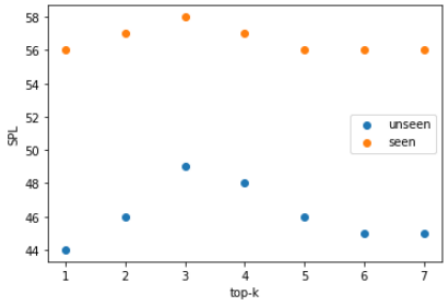


Figure 3: **SPL Results with Different K Values.**

5.2 Scene & Object Alignment

Ent-Rel(Hong et al., 2020a) distinguishes the landmarks which are *scene*s from *object*s. Scene tokens describe the location at a coarse level, such as “bathroom", while object tokens describe the exact landmarks, such as “table". To evaluate the agent’s performance given the instructions with only object tokens, we mask all scene tokens in the instructions and evaluate on Ent-Rel and our model. Table 2 shows the experimental results in the unseen validation set. Compared with Ent-Rel, our model performs slightly better given the instruction with only object tokens but worse with scene and object tokens. One of the reasons for such a phenomenon is that Faster-RCNN often fails to detect the scenes correctly. For example, the aligned object labels in the image for the landmark "bedroom" are “floor", “roof", “wall", which are parts of the bedroom. The explicitly modeling makes our model more sensitive to the wrong alignments, which further impacts the navigation performance.

5.3 Ablation Study

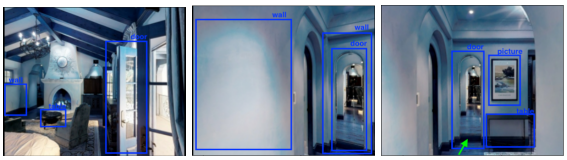
Table 3 shows the ablation study results. Row#1 is the baseline model. Row#2 (*Lan-Obj*) shows that explicitly modeling important landmarks and aligned objects improves the performance compared to the baseline. *Rel* (row#3) is the result after modeling the spatial relation tokens describing the relative relation between agent and landmark. *Rel_v* (row#4) is the result after modeling the spatial relations in motions. The improved SDTW shows the modeling of spatial relations can help the agent to follow the instructions. However, the spatial terms directly describing the landmark are more helpful than the spatial terms in motions.

5.4 Qualitative Analysis

Figure 4 shows qualitative analysis examples. The selected k-important landmarks are “door”, “table”, “painting” in Figure 4a. The agent makes a correct decision by selecting the viewpoint that contains the objects aligned with all three landmarks. Figure 4b shows an example after modeling spatial relations. Although three navigable viewpoints have the object "door", the agent selects the aligned object with the “left” direction. Also, in Figure 5, we provide an example to visualize the navigation process using the selected landmark based on the spatial configurations.

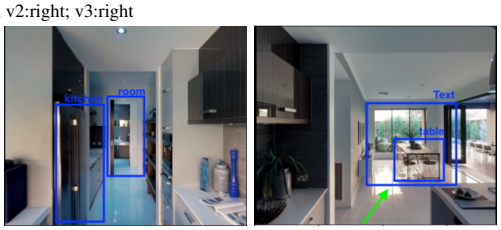
However, we find that relation alignments will be helpful when the object alignments are done correctly. Figure 4c shows another example of landmark and object alignments. It contains two spatial configurations: “walk past the kitchen towards the dining room” and “stop before you reach the table”. In the first configuration, the landmarks are “kitchen’ and “dining room”; in the second configuration, the landmark is “table”. By merely using the visual environment as a clue for viewpoint selection, the agent will select the second navigable viewpoint because of its detected “kitchen” view.

Nevertheless, based on the instruction semantics, the “kitchen” is an object the agent passes by, and

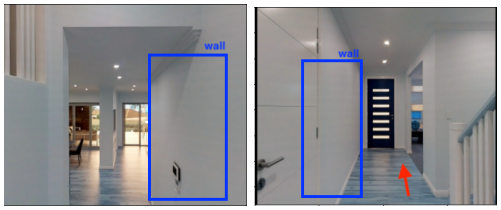


(a) Enter the "door" to the small "table" with a "painting" above.
v1: [door-door; table-table; painting-wall]
v2: [door-door; table-wall; painting-wall]
v3: [door-door; table-table; painting-picture]

(b) Head towards the "doors" on the left towards "kitchen".
v1:left; v2:right; v3:right



(c) Walk past the "kitchen" towards the "dining room". Stop before you reach the "table".
v1: [kitchen-room; dining room-room; table-table]
v2: [kitchen-kitchen; dining room-room; table-kitchen]



(d) Turn right toward "bathroom". Stop at the top of the steps.
v1:left; v2:right;

Figure 4: **Qualitative Examples.** Blue bounding boxes are the aligned objects. Green arrow is the selected correct viewpoint. v is the viewpoint, the alignment between landmarks and objects is [landmark-object].

		Train	Val Seen	Val Unseen
1	Instructions	14025	1021	2349
2	Configs	58277	4301	9625
3	Configs with Landmark	44053	3225	7303
4	Configs with Relation	13543	1142	2566

Table 4: **Statistics of Spatial Configuration**

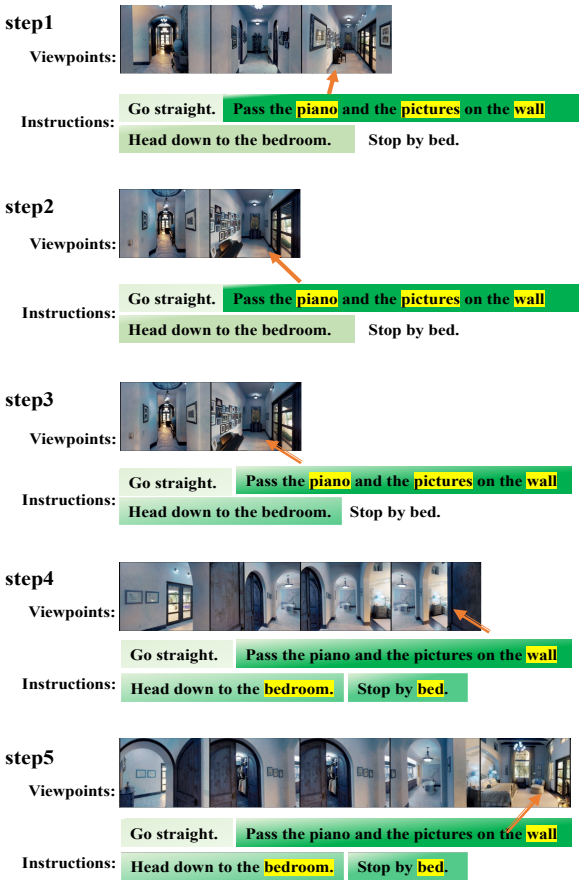


Figure 5: The **green** boxes are spatial configurations; **darker green** means higher weights; **yellow** boxes are the selected landmarks; the **orange** arrows are the path.

the "table" is the final goal. In some cases, our method can handle such situations by using the selected landmarks. In this example, the model allows the agent to focus on the aligned object such as "table", which appear later in the spatial configuration. It increases the probability of selecting the first viewpoint. Also, we find that relation alignments modeling will be helpful only when the object alignments are done correctly. If the object alignments fail, for example, when the agent makes mistakes during navigation or the aligned objects can not be detected, modeling relations can worsen the situation. For instance, in Figure 4d, for both navigable viewpoints, the object "bathroom" can not be detected, and in this case, further modeling relations leads to making wrong decisions.

6 Conclusion

In this work, we propose a neural architecture to solve the vision and language navigation problem. Our method achieves the alignments between textual landmarks and visual objects. In particular, we first select important landmarks based on spatial configurations, and then encourage the agent to concentrate on the relevant objects in the visual environment given the selected landmarks. Besides, We are the first to explicitly model the spatial relations between the agent and the landmarks from the agent's perspective on both instruction and image sides. Our experiments show that explicit object-

landmark alignments and the perspective information are important factors and lead to competitive results compared with strong baselines. We have conducted comprehensive analysis to support our conclusion that explicitly modeling the objects and spatial relation alignments improving the spatial reasoning ability, generalizability and explainability of the model. Though we do not achieve the SOTA compared to transformer-based models that rely on pre-training, we plan to apply the same ideas on top of such recent models in the future.

7 Acknowledgement

This project is supported by National Science Foundation (NSF) CAREER award 2028626 and partially supported by the Office of Naval Research (ONR) grant N00014-20-1-2005. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation nor the Office of Naval Research. We thank all reviewers for their thoughtful comments and suggestions.

References

Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton Van Den Hengel. 2018. 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.

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

Soham Dan, Parisa Kordjamshidi, Julia Bonn, Archna Bhatia, Zheng Cai, Martha Palmer, and Dan Roth. 2020. From spatial relations to spatial configurations. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 5855–5864.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

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. *arXiv preprint arXiv:1806.02724*.

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.

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

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

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

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

Gabriel Ilharco, Vihan Jain, Alexander Ku, Eugene Ie, and Jason Baldridge. 2019. General evaluation for instruction conditioned navigation using dynamic time warping. *arXiv preprint arXiv:1907.05446*.

Eric Kolve, Roozbeh Mottaghi, Winson Han, Eli VanderBilt, Luca Weihs, Alvaro Herrasti, Daniel Gordon, Yuke Zhu, Abhinav Gupta, and Ali Farhadi. 2017. Ai2-thor: An interactive 3d environment for visual ai. *arXiv preprint arXiv:1712.05474*.

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 *Conference on Empirical Methods for Natural Language Processing (EMNLP)*.

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

Chih-Yao Ma, Jiasen Lu, Zuxuan Wu, Ghassan AlRegib, Zsolt Kira, Richard Socher, and Caiming Xiong. 2018. Self-monitoring navigation agent via auxiliary progress estimation.

Chih-Yao Ma, Jiasen Lu, Zuxuan Wu, Ghassan AlRegib, Zsolt Kira, Richard Socher, and Caiming Xiong. 2019. Self-monitoring navigation agent via auxiliary progress estimation. *arXiv preprint arXiv:1901.03035*.

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 *European Conference on Computer Vision*, pages 259–274. Springer.

Khanh Nguyen and Hal Daumé III. 2019. Help, anna! visual navigation with natural multimodal assistance via retrospective curiosity-encouraging imitation learning. *arXiv preprint arXiv:1909.01871*.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Yuankai Qi, Zizheng Pan, Shengping Zhang, Anton van den Hengel, and Qi Wu. 2020a. Object-and-action aware model for visual language navigation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16*, pages 303–317. Springer.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9982–9991.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28:91–99.

Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. 2019. Habitat: A platform for embodied ai research. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9339–9347.

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

Mohit Shridhar, Jesse Thomason, Daniel Gordon, Yonatan Bisk, Winson Han, Roozbeh Mottaghi, Luke Zettlemoyer, and Dieter Fox. 2020. Alfred: A benchmark for interpreting grounded instructions for everyday tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10740–10749.

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

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

Xin Wang, Qiuyuan Huang, Asli Celikyilmaz, Jianfeng Gao, Dinghan Shen, Yuan-Fang Wang, William Yang Wang, and Lei Zhang. 2019. Reinforced cross-modal matching and self-supervised imitation learning for vision-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6629–6638.

Yubo Zhang, Hao Tan, and Mohit Bansal. 2020. Diagnosing the environment bias in vision-and-language navigation. *arXiv preprint arXiv:2005.03086*.

Yue Zhang, Quan Guo, and Parisa Kordjamshidi. 2021. Towards navigation by reasoning over spatial configurations. *arXiv preprint arXiv:2105.06839*.

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 *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2539–2556.

Wanrong Zhu, Yuankai Qi, Pradyumna Narayana, Kazoo Sone, Sugato Basu, Xin Eric Wang, Qi Wu, Miguel Eckstein, and William Yang Wang. 2021. Diagnosing vision-and-language navigation: What really matters. *arXiv preprint arXiv:2103.16561*.