Diagnosing Vision-and-Language Navigation: What Really Matters

Anonymous ACL submission

Abstract

Vision-and-language navigation (VLN) is a multimodal task where an agent follows natural language instructions and navigates in visual environments. Multiple setups have been proposed, and researchers apply new model architectures or training techniques to boost navigation performance. However, there still exist non-negligible gaps between machines' performance and human benchmarks. Moreover, the agents' inner mechanisms for navigation decisions remain unclear. To the best of our knowledge, how the agents perceive the multimodal input is under-studied and needs investigation. In this work, we conduct a series of diagnostic experiments to unveil agents' focus during navigation. Results show that indoor navigation agents refer to both object and direction tokens when making decisions. In contrast, outdoor navigation agents heavily rely on direction tokens and poorly understand the object tokens. The differences in dataset designs and the visual features lead to distinct behaviors on visual environment understanding. Many models claim that they can align object tokens with specific visual targets when it comes to vision-and-language alignments. We find unbalanced attention on the vision and text input and doubt the reliability of such cross-modal alignments.

1 Introduction

A key challenge for Artificial Intelligence (AI) research is to move beyond Independent and Identically Distributed (i.i.d.) data analysis: We need to teach AI agents to understand multimodal input data, and jointly learn to reason and perform incremental and dynamic decision-making with the help from humans. Vision-and-Language Navigation (VLN) has received much attention due to its active perception and multimodal grounding setting, dynamic decision-making nature, rich applications, and accurate evaluation of agents' performances in language-guided visual grounding. As the AI research community gradually shifts the attention from the static empirical analysis of datasets to more challenging settings that require incremental decision-making processes, the interactive task of VLN deserves a more in-depth analysis of why it works and how it works.

	R2R	RxR	Touchdown
Human Performance	86	94	92
SoTA Model Performance	78	53	17

Table 1: There exists salient gaps between machines' vision-and-language navigation (VLN) performance and human benchmarks. Navigation success rates are reported on the R2R (Anderson et al., 2018) and the RxR dataset (Ku et al., 2020b) for indoor VLN and the Touchdown dataset (Chen et al., 2019) for outdoor VLN.

Various setups have been proposed to address to the VLN task. Researchers generate visual trajectories and collect human-annotated instructions for indoor (Anderson et al., 2018; Jain et al., 2019a; Ku et al., 2020a; Chen et al., 2021) and outdoor environment (Chen et al., 2019; Mehta et al., 2020; Mirowski et al., 2018). There are also interactive VLN settings based on dialogues (Nguyen et al., 2019; Nguyen and III, 2019; Zhu et al., 2020c), and task that navigates agents to localize a remote object (Qi et al., 2020c). However, few studies ask the *Why* and *How* questions: Why do these agents work (or do not work)? How do agents make decisions in different setups?

Through the years, agents with different model architectures and training mechanisms have been proposed for indoor VLN (Anderson et al., 2018; Fried et al., 2018; Hao et al., 2020; Hong et al., 2020a,b; Huang et al., 2019; Ke et al., 2019; Li et al., 2019; Ma et al., 2019a; Qi et al., 2020b; Tan et al., 2019; Wang et al., 2020a, 2019, 2018, 2020b; Zhu et al., 2020a) and outdoor VLN (Chen et al., 2019; Ma et al., 2019b; Mirowski et al., 2018; Xia et al., 2020; Xiang et al., 2020; Zhu et al., 2020b). Back-translation eases the urgent problem of data scarcity (Fried et al., 2018). Imitation learning and

reinforcement learning enhance agents' generalization ability (Wang et al., 2019, 2018). With the rise of BERT-based models, researchers also apply Transformer and pre-training to further improve navigation performance (Hao et al., 2020; Hong et al., 2020b; Zhu et al., 2020b). While applying new techniques to the navigation agents might boost their performance, we still know little about how agents make each turning decision. Treatment of the agents' processing of instructions and perception of the visual environment as a black box might hinder the design of a generic model that fully understands visual and textual input regardless of VLN setups. Table 1 shows that there are still non-negligible performance gaps between neural agents and humans on both indoor and outdoor VLN tasks[1].

Therefore, we focus on analyzing how the navigation agents understand the multimodal input data in this work. We conduct our investigation from the perspectives of natural language instruction, visual environment, and the interpretation of vision-language alignment. We create counterfactual interventions to alter the instructions and the visual environment in the validation dataset, focusing on variables related to object and direction. More specifically, we modify the instruction by removing or replacing the object/direction tokens, and we adjust the environment by masking out visual instances or horizontally flipping the viewpoint images. Subsequently, we examine the interventions' treatment effects on agents' evaluation performance while keeping other variables unchanged. We set up experiments on the R2R (Anderson et al., 2018) and the RxR dataset (Ku et al., 2020b) for indoor VLN and the Touchdown dataset (Chen et al., 2019) for outdoor VLN. We examine nine VLN agents on the three datasets with quantitative ablation diagnostics on the text and visual inputs.

In summary, our key findings include:

1. Indoor navigation agents refer to both objects and directions in the instruction when making decisions. In contrast, outdoor navigation agents heavily rely on directions and poorly understand visual objects. (Section 4)
2. The differences of dataset designs and the visual features lead to distinct behaviors on visual environment understanding. R2R agents rely more on background information to navigate over RxR agents. Compared to ImageNet ResNet-152 features, CLIP-ViT features are less affected by the loss of visual object information. (Section 5)
3. Indoor agents can align object tokens to certain targets in the visual environment to a certain extent, but display in-balanced attention on text and visual input. (Section 6)

We hope these findings reveal opportunities and obstacles of current VLN models and lead to new research directions.

2 Related Work

Instruction Following is a long-standing topic in AI studies that ask an agent to follow natural language instructions and accomplish target tasks, which can be dated back to the SHRLDU (Winograd, 1971). Efforts made to tackle this classic problem spans broadly from defining templates (Klingspor et al., 1997; Antoniol et al., 2011), designing hard-encoded concepts to ground visual attributes and spatial relations (Steels and Vogt, 1997; Roy, 2002; Guadarrama et al., 2013; Kollar et al., 2013; Matuszek et al., 2014), to constructing varies datasets and learning environments (Anderson et al., 1991; Bisk et al., 2016; Misra et al., 2018).

Vision-and-Language Navigation is a task where an agent comprehends the natural language instructions and reasons through the visual environment. To enrich training data, a line of work (Fried et al., 2018; Zhu et al., 2020b) use back-translation to generate augmented instructions. To enforce cross-modal grounding, RPA and RCM (Wang et al., 2018, 2019) use reinforcement learning, SMNA (Ma et al., 2019a) uses a visual-textual co-grounding module to improve cross-modal alignment, RelGraph (Hong et al., 2020a) uses graphs for task formulation. To address the generalizability problem to unseen environment, PRESS (Li et al., 2019) introduces a stochastic sampling scheme, EnvDrop (Tan et al., 2019) proposes environment dropout. To utilize visual information from the environment, AuxRN (Zhu et al., 2020a) uses auxiliary tasks to assist semantic information extraction, VLN-HAMT (Chen et al., 2021) incorporates panorama history with a hierarchical vision transformer.

With the success of BERT-related models in NLP, researchers also started to build Transformer-based navigation agents and add a pre-training pro-

[1]We record the published state-of-the-art performance on R2R, RxR and Touchdown leaderboards on Dec.15th, 2021.

cess before fine-tuning on the downstream VLN task (Hao et al., 2020; Hong et al., 2020b; Zhu et al., 2020b; Chen et al., 2021). The increased model size and additional training phase help improve navigation performance to a certain extent. Above mentioned studies aim at improving agents' performance in one way or another.

Model Behavior Analysis As multimodal studies gain more and more attention, there are lines of works that focus on explaining models' behaviors to understand better and handle the tasks. Some generate textual explanations by training another model to mimic human explanations (Hendricks et al., 2016; Park et al., 2018; Wu and Mooney, 2019). Others generate visual explanations with the help of attention mechanism (Lu et al., 2016) or gradient analysis (Selvaraju et al., 2017). There are also attempts on providing multimodal explanations, e.g., (Li et al., 2018) breaks up the end-to-end VQA process and examines the intermediate results by extracting attributes from the visual instances. Another line of works examines model performance by conducting ablation studies on input data. Two recent analyses on language modelling (O'Connor and Andreas, 2021) and machine translation (Fernandes et al., 2021) ablate both training and validation data. A study on multimodal models (Frank et al., 2021) only applies ablation during evaluation, which is the same with our settings.

3 Background and Research Questions

3.1 Vision-and-Language Navigation

In the vision-and-language navigation task, the navigation agent is asked to find the path to reach the target location following the instructions $\mathcal{X}$. The navigation procedure can be viewed as a sequential decision-making process. At each time step t, the visual environment presents an image view $\boldsymbol{v}_t$. With reference to the instruction $\mathcal{X}$ and the visual view $\boldsymbol{v}_t$, the agent is expected to choose an action a_t such as *turn left* or *stop*.

Datasets We conduct indoor navigation experiments on the Room-to-Room (R2R) dataset (Anderson et al., 2018) and the Room-across-Room (RxR) dataset (Ku et al., 2020b), and test outdoor VLN on Touchdown (Chen et al., 2019). R2R and RxR are built upon real estate layouts and contain separate graphs for each apartment/house. Unlike R2R that shoots for the shortest path, RxR contains longer and more variable paths. R2R only contains English instructions, while RxR also includes instructions in Hindi and Telugu.[2] Navigation in Touchdown occurs in the urban environment, where the viewpoints form a huge connected graph. Compared to indoor environments, Touchdown has more complicated visual environments and a more extensive search space. The evaluation results in this study are reported on the validation unseen sets for R2R and RxR, and the test set for Touchdown.

Dataset	Model	Trans?	Visual Feature
R2R	EnvDrop (Tan et al., 2019)	×	ResNet-152
	VLN ⟳ BERT (Hong et al., 2020b)	×	
	FAST (Ke et al., 2019)	✓	
	PREVALENT (Hao et al., 2020)	✓	
R2R	CLIP-ViL (Shen et al., 2021)	×	CLIP-ViT
	VLN-HAMT (Chen et al., 2021)	✓	
Touchdown	RCONCAT (Chen et al., 2019)	×	ResNet-18
	ARC (Xiang et al., 2020)	×	
	VLN-Transfomer (Zhu et al., 2020b)	✓	

Table 2: The VLN datasets and models covered in this study. We record whether the model structure is Transformer-based, and the pre-trained feature extractor used to encode visual environment.

Models Table 2 lists out the models in our study. We use the code and trained checkpoints shared by the authors in the following experiments.

For indoor navigation on R2R, we study a widely adopted base model Envdrop (Tan et al., 2019), a backtracking framework for self-correction FAST (Ke et al., 2019), and two SoTA models VLN ⟳ BERT (Hong et al., 2020b) and PREVALENT (Hao et al., 2020). The Envdrop introduces environment dropout on top of the Speaker-Follower (Fried et al., 2018) model, FAST conducts an asynchronous search for backtracking, PREVALENT, and VLN ⟳ BERT are Transformer-based agents with pre-trained models.

For navigation on RxR, we examine CLIP-ViL (Shen et al., 2021) and VLN-HAMT (Chen et al., 2021). CLIP-ViL shares the same model structure with EnvDrop. The only difference is that CLIP-ViL use CLIP (Radford et al., 2021) to extract visual features, while EnvDrop use ImageNet ResNet (Szegedy et al., 2017) features. VLN-HAMT incorporates a long-horizon history into decision-making by encoding all the past panoramic observations via a hierarchical vision transformer.

For outdoor navigation on Touchdown, we use the common baseline RCONCAT (Chen et al.,

[2] In this study, we only cover the English subset for RxR.

2019), and two SoTA models ARC (Xiang et al., 2020) and VLN-Transfomer (Zhu et al., 2020b). RCONCAT encodes the trajectory and the instruction in an LSTM-based manner. ARC improves RCONCAT by paying special attention to stop signals. VLN-Transfomer is a Transformer-based agent that applies pre-training on a multimodal dataset.

Metrics In the following experiments, we evaluate navigation performance with Success Rate (SR) for indoor agents and Task Completion (TC) rate for outdoor agents. Both SR and TC measure the accuracy of completing the navigation task, which reflects the agents' overall ability to finish navigation correctly. An indoor navigation task is considered complete if the agent's final position locates within 3 meters of the target location. For outdoor navigation, the task is considered completed if the agent stops at the target location or one of its adjacent nodes in the environment graph.

3.2 Research Questions

Current VLN studies have reached their bottleneck as only minor performance improvements have been achieved recently, while a significant gap still exists between machine and human performance. This motivates us to find the reasons.

To better understand how VLN agents make decisions during navigation, we conduct a series of experiments on indoor and outdoor VLN tasks, aiming to answer the following questions that might help us locate the deficiencies of current model designs and explore future research directions:

1. *What can the agents learn from the instructions? Do they pay more attention to object tokens or directions tokens? Do they have the ability to count? (Section 4)*
2. *What do agents see in the visual environment? Are they staring at the closely surrounded objects or also browsing further layout? Do they focus on individual visual instances or perceive the overall outline? (Section 5)*
3. *Can agents match textual tokens to visual entities? How reliable are such connections? (Section 6)*

4 Analysis on Instruction Understanding

This section examines whether and to what extent the agent understands navigation instructions. We focus on how the agent perceives object-related tokens, direction-related tokens, and numeric tokens, and their effects on final navigation performance. Table 3 shows exemplar instructions of these three datasets. As shown in Table 4, the ratio of object and direction tokens in R2R, RxR and Touchdown are comparable. RxR and Touchdown have longer instructions. Instructions in all three datasets involve about two times more object tokens than direction tokens.

Dataset	Instruction
R2R	Walk through the **door** by the **sink** into the **middle** of the next **room**. Turn ***right*** and walk down the **hallway** and enter the **third** **door** on your ***right***.
RxR	You are standing inside a **living room**, turn ***right*** and exit, move towards the **stairs** on your ***right***, and then turn ***left*** take few **steps** towards two **entrances** in ***front*** and then turn ***left***, their should be a **wine room** on your ***right*** and on your ***right*** a **glass window** on your ***left*** and an open **door** in ***front*** of you, go towards the open **door** which has a black **bench**, on your ***right***, once you are their **turn** ***left*** and then proceed straight ahead towards the **garage door**, more towards the **lockers** on your ***left***, once you are their you are done .
Touchdown	Orient yourself so that you are moving in the same **direction** as **traffic**. Go straight through **3** **intersections**. Keep moving ***forward***, after the **3rd** **intersection**, you should see a **signs** for a **store** with a white **background** and red **dots** as well as a red and white **bullseye target**. Continue going straight past this **store** and at the next **intersection**, turn ***left***. Go through **one** **intersection** and ***stop*** just after the **wall** on your ***left*** with the purple **zig zag patterns**.

Table 3: Instructions from R2R, RxR and Touchdown with **object-tokens**, ***direction-tokens*** and **numeric-tokens** highlighted.

Dataset	#Data	Avg_Len	#Object	Avg_#Obj	Avg_#Direc
R2R	2.3k	29.4	1.3k	19.6%	7.6%
RxR	4.6k	114.0	2.9k	16.0%	6.5%
Touchdown	1.4k	91.1	1.7k	17.2%	6.8%

Table 4: Statistics of R2R, RxR and Touchdown datasets. #Data is the number of data samples used for evalution in the following sections. Avg_Len is the average #tokens in instructions, and #Object is the number of unique objects in visual environments. Avg_#Obj and Avg_#Direc denotes the percentage of object/direction tokens per instruction.

4.1 The Effect of Object-related Tokens

We create counterfactual interventions on instructions by masking out the object tokens. We use Stanza (Qi et al., 2020a) part-of-speech (POS) tagger to locate object-related tokens. A token will be regarded as an object token if its POS tag is *NOUN* or *PROPN*. During masking, we replace the object token with a specified mask token *[MASK]*. Then we examine the average treatment effects of the intervention on agents' performance, while keeping other variables unchanged.

Table 5 gives an example of removing object tokens by masking. Noticeably, when we mask out the object tokens, the tokens visible to the agent also decrease, which is a coherent factor with #object tokens and might interfere with our analysis. To eliminate the effect of reducing visible tokens, we add a controlled trial in which we randomly

Setting	Instruction
Original	Go ***left*** down the **hallway** toward the **exit sign**. Turn ***right*** and go down the **hallway**. Go into the **door** on the ***left*** and ***stop*** by the **table**.
Mask Object Tokens	Go left down the **[MASK]** toward the **[MASK] [MASK]**. Turn right and go down the **[MASK]**. To into the **[MASK]** on the left and stop by the **[MASK]**.
Replace Object Tokens	Go left down the **portrait** toward the **sofa fountains**. turn right and go down the **door**. Go into the **football** on the left and stop by the **boats**.
Controlled Trial	Go left down the hallway **[MASK]** the exit sign. **[MASK]** right and go down the **[MASK]**. To into the door on **[MASK]** left and **[MASK]** by **[MASK]** table.
Mask Direction Tokens	Go ***[MASK]*** down the hallway toward the exit sign. Turn ***[MASK]*** and go down the hallway. Go into the door on the ***[MASK]*** and ***[MASK]*** by the table.
Replace Direction Tokens	Go ***right*** down the hallway toward the exit sign. Turn ***left*** and go down the hallway. Go into the door on the ***right*** and ***forward*** by the table.
Controlled Trial	Go left down the ***[MASK] [MASK]*** the exit sign. Turn right and go down the ***[MASK]***. Go into the door on the left and ***[MASK]*** by the table.

Table 5: Example of instruction modification. In the original instruction, there are six **object-related tokens**, and four ***direction-related tokens***. In the object token ablations, we mask out the object tokens, or replace them with randomly sampled object tokens. The controlled trial randomly masked out six tokens from the instruction for a fair comparison. Likewise the direction tokens.

Ablation	Setting	SR ↑ on R2R				SR ↑ on RxR		TC ↑ on Touchdown		
		EnvDrop	FAST	VLN⟳BERT	PREVALENT	CLIP-ViL	VLN-HAMT	RCONCAT	ARC	VLN-Trans
–	Vanilla	49.77	63.39	53.30	57.13	40.21	52.52	11.78	15.19	16.11
Object	Mask	-36%	-38%	-21%	-20%	-48%	-32%	-35%	-35%	-7%
	Controlled Trial	-30%	-26%	-9%	-8%	-35%	-24%	-50%	-55%	-19%
Direction	Mask	-23%	-23%	-15%	-11%	-39%	-28%	-74%	-89%	-45%
	Controlled Trial	-12%	-11%	-5%	-3%	-18%	-9%	-19%	-26%	-11%

Table 6: The navigation performance for indoor and outdoor agents on object-token and direction-token ablations. The "vanilla" setting reports the validation score on R2R and RxR validation unseen set, and Touchdown test set. For object-token ablations, the "mask" setting masks out all the object-tokens, while the controlled trial masks out the same amount of random tokens. The same applies for direction-token ablations.

mask out the same amount of tokens.

We follow each agent's original experiment setting for all the experiments in this study and train it on the original train set. Then we apply masking to object tokens in the validation set, and report agents' performance under each setting. We conduct five repetitive experiments and report the average scores for settings that involve random masking or replacing.

Table 6 presents how the agents' navigation performance change when object tokens are masked out. Intuitively, not knowing what objects are mentioned in the instruction lowers all models' performance. Comparing the masking ablations with the controlled trial for indoor VLN, we notice that masking out the object tokens result in a more drastic decrease in success rate than masking out random tokens. This holds for all indoor agents, which verifies that indoor agents depend on object tokens more than other tokens. However, when we compare results on the Touchdown for outdoor VLN, we notice in surprise that masking out the object tokens has a weaker impact on task completion rate than masking out random tokens. This suggests that current outdoor navigation agents do not fully take object tokens into consideration in Touchdown instruction when making decisions. This may be caused by the weak visual recognition module in current outdoor agents. As addressed in Table 4, all three outdoor agents rely on visual features extracted by ResNet-18, which may not be powerful enough to incorporate the complicated urban environments fully.

4.2 The Effect of Direction-related Tokens

We explicitly select the following tokens as direction-related tokens: *left, right, back, front, forward, stop*. Similar to how we ablate the object tokens, we also mask out direction tokens from the instruction and examine their impact on agents' navigation performance. Table 5 provides examples of masking out direction tokens, and the corresponding controlled trial where the same amount of random tokens are masked out.

Table 6 shows indoor and outdoor agents' performance when masking out direction tokens. For indoor agents, masking out the direction tokens cause the success rate to drop more than masking out random tokens, which means the indoor navigation agents do consider the direction tokens during navigation. We also notice that agents are more sensitive to the loss of direction guidance on RxR than on R2R. Such difference may be caused by the way these two datasets are designed. R2R's ground-truth trajectories are the shortest path from start to goal. Previous studies have noted that R2R has the danger of exposing structural bias and leaking hidden shortcuts (Thomason et al., 2019), and

that such design encourages goal-seeking over path adherence (Jain et al., 2019b). RxR is crafted to include longer and more variable paths to avoid such biases. Naturally, agents on RxR pay more attention to direction tokens since they may approach their goal indirectly.

For outdoor navigation agents, masking out direction tokens leads to a drastic decline in task completion rate, compared to random masking. This indicates that current outdoor navigation agents heavily rely on the direction tokens when making decisions. Given the complicated visual environments and instructions in the outdoor navigation task, current agents fail to fully use the instructions, especially ignoring the rich object-related information. We notice that the ARC model shows the most salient performance decline of 89% to the instructions ablated by direction token masking. Aside from the classifier that predicts the next direction to take, ARC also uses a stop indicator to decide whether to stop at each step or not. Its unique mechanism for detecting stop signals might explain why it is more sensitive to the instruction's masking.

4.3 The Effect of Numeric Tokens

	#Data	Avg_Len	Avg_#Num
RxR	2.4K	122.9	1.2%
Touchdown	0.9k	98.2	1.8%

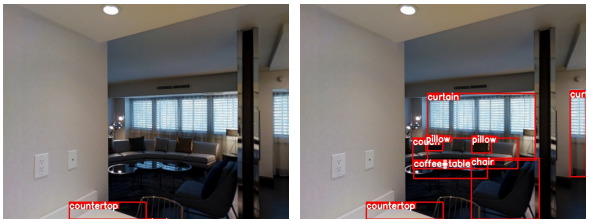
Table 7: The number of data samples with numeric tokens in the RxR and Touchdown dataset. For instructions containing numeric tokens, Avg_Len is average length, and Avg_#Num denotes the percentage of numeric tokens per instruction.

Setting	SR ↑ on RxR		TC ↑ on Touchdown		
	CLIPViL	HAMT	RCONCAT	ARC	VLN-Trans
Vanilla	38.48	50.63	10.23	15.15	15.55
Mask Number	-4%	-5%	-4%	-6%	-2%
Replace Number	-11%	-15%	-5%	-4%	-4%
Controlled Trial	-4	-4%	-8%	-6%	-5%

Table 8: Navigation performance on different numeric-token ablations settings.

We conduct ablation studies on agents' understanding of numeric tokens on RxR for indoor agents and Touchdown for outdoor agents. We select a subset of examples whose instructions contain numeric tokens, and construct ablated instructions on top. Table 7 provides the statistics of the instructions for numeric ablations. Table 8 lists out the results. The indoor agents on RxR have comparable performance when masking numeric tokens over random tokens, and have worse performance when replacing numeric tokens. This suggests that agents on RxR have a certain ability to count. In contrast, the outdoor agents on Touchdown have similar performance drops in all three ablation settings, which implies their insufficient counting ability.

5 Analysis on Visual Environment



(a) Foreground Objects (b) All Visible Objects

Figure 1: Accessible objects within different ranges.

Setting	#Object
All Visible Objects (except wall/floor/ceiling)	33.1
Foreground Objects	2.8
Objects Mentioned in the Instruction	6.5

Table 9: The average number of visual objects at each viewpoint under different settings.

Setting	SR ↑ on R2R				SR ↑ on RxR	
	EnvDrop	FAST	Recur	PVLT	CLIPViL	HAMT
Vanilla	49.77	63.39	53.30	57.13	40.21	52.52
Mask All Range	-41%	-55%	-37%	-47%	-30%	-43%
Mask Foreground	-3%	-5%	-1%	-8%	-1%	-2%
Controlled Trial	-26%	-12%	-5%	-10%	-2%	-2%

Table 10: The indoor navigation performance on different visual object ablation settings: mask all visible objects, mask foreground objects, and the controlled trial. Recur: VLN ⟳ BERT. PVLT: PREVALENT.

This section investigates what the agent perceives in the visual environment. We set an eye on inspecting the agent's understanding of the surrounding objects. We also verify the indoor agents' understandings of direction-related information in the appendix.

Built upon the Matterport dataset (Chang et al., 2017), R2R obtains detailed object instances annotations and serves as an excellent source for our visual object studies. Touchdown is based on Google Street View and does not acquire object-related annotations. Thus, we conduct experiments on the indoor VLN environment.

We designed several ablation settings for visual objects. The "mask all range" setting applies

masking to all the visible visual objects in the environment (except for walls/ceiling/floor). The "mask foreground" setting ablates the visual objects within 3 meters of the camera viewpoint, which we refer to as the foreground area. The region beyond 3 meters from the camera viewpoint is regarded as the background area. Figure 1 shows an example for comparison. Table 9 compares the number of foreground and background visual objects. We choose 3 meters as the boundary because the bounding box annotations for objects within 3 meters are provided in REVERIE (Qi et al., 2020c). We denote the number of visual objects within 3 meters as k, and add a controlled trial that masks out k random visual objects from all the visible objects at the current viewpoint, regardless of its depth.

We mask out the objects in each view by filling the corresponding bounding boxes with the mean color of the surrounding. Then we follow original experiment settings and use ImageNet ResNet-152 (He et al., 2016) CNN to extract image features for R2R agents (Anderson et al., 2018; Hao et al., 2020; Ke et al., 2019; Hong et al., 2020b; Tan et al., 2019), and use CLIP ViT (Radford et al., 2021) to extract visual features for RxR agents (Shen et al., 2021; Chen et al., 2021).

Results for visual object ablations are shown in Table 10. We examine the influence of masking out different quantities of visual objects by comparing the "mask all range" setting with the controlled trial. It comes naturally that masking out all the visible objects has a more salient impact on the success rate for all the listed indoor agents. We study the influence of masking visual objects at different distances from the viewpoint by comparing the "mask foreground" setting with the controlled trial. If we only mask the foreground objects, all the agents' performance rarely changes. This is because there are only a few foreground objects. On the R2R dataset, EnvDrop has much worse performance on the controlled trial, while FAST and the two Transformer-based agents have mild drops in success rates. On the RxR dataset, both agents have comparable performance on the controlled trial and the "mask foreground" setting.

Noted here that the dataset domains and the visual feature extractors are two coherent factors that may result in the performance difference between R2R agents and RxR agents. We further justify this by adding another set of ablation studies, where we apply ImageNet ResNet-152 to extract visual features for RxR. Results are shown in Table 11. Comparing line 1 and line 2, we notice that the agent is affected more heavily in the R2R controlled trial, suggesting that agents rely more on background information to navigate in R2R. This may be caused by structural bias and hidden shortcuts leakage in R2R (Thomason et al., 2019; Jain et al., 2019b). Comparing line 2 and line 3, we found out that with CLIP-ViT features, the agent has a higher success rate and is less affected when all the visual objects are masked out. Such difference originates from the different architectures and training data for ImageNet ResNet-152 and CLIP-ViT.

Dataset	Vi-Feat	Vanilla	AR	FG	CT
R2R	ResNet-152	49.77	-41%	-3%	-26%
RxR	ResNet-152	35.27	-42%	-1%	-3%
RxR	CLIP-ViT	40.21	-30%	-1%	-2%

Table 11: EnvDrop's navigation performance on R2R and RxR with different visual object ablation settings when using different visual features. AR: mask all range. FG: mask foreground. CT: controlled trial.

6 Analysis on Vision-Language Alignment

This section examines the agents' ability to learn vision-language alignment in the VLN task. We focus on whether the agents can understand the objects mentioned in the instruction and align them to the correct visual instance in the environment, which is crucial to completing this multimodal task. To verify the existence of vision-language alignment, we add perturbations to the visual and textual input.

6.1 Instruction Side Perturbation

We add noise to the textual input by randomly replacing object tokens with other object tokens in the instruction. Table 5 shows an example. This experiment aims to verify whether the agent can line the object tokens up to certain visual targets. The assumption is that if the agent can correctly align objects mentioned in the instruction to some targets in the visual environment, then replacing the object token will confuse and misguide the agent.

Examining Figure 2, we notice that for all three datasets, the Transformer-based models have worse performance when replacing the object tokens, compared to simple masking. This indicates that Transformer-based models have a better cross-modal understanding of objects, and can align

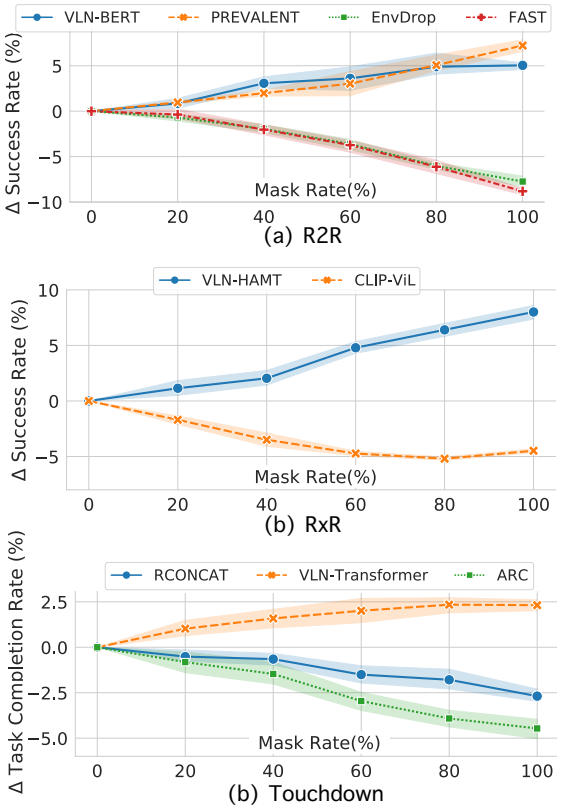


Figure 2: Performance gap between masking and replacing object tokens from instructions. If $\Delta > 0$, then replacing object tokens leads to worse navigation performance, which suggests a better understanding of object tokens.

object tokens to the visual targets. Such superior performance may result from the fact that the Transformer-based models are often pre-trained on multimodal resources, thus displaying a slightly more vital ability to form alignment.

6.2 Environment Side Perturbation

Setting	SR ↑ on R2R				SR ↑ on RxR	
	EnvDrop	FAST	Recur	PVLT	CLIPViL	HAMT
Vanilla	49.77	63.39	53.30	57.13	40.21	52.52
Dynamic Mask	-13%	-34%	-11%	-19%	-20%	-28%
Controlled Trial	-8%	-28%	-8%	-11%	-13%	-12%
Mask Tokens	-36%	-38%	-21%	-20%	-48%	-32%

Table 12: The indoor navigation performance when dynamically masking out the visual objects mentioned in the instructions. Recur: VLN ⟳ BERT. PVLT: PREVALENT.

We add noise to the visual input by conducting the following ablations. In the "dynamic mask" setting, we dynamically mask out the visual object regions mentioned in the instruction. In its controlled trial, we randomly mask out the same amount of visual objects at each viewpoint. We also compare with the "mask tokens" setting, where we mask out all the object tokens in the instruction, while leaving the visual environment untouched. This experiment aims to determine if the agent aligns the textual object tokens to the correct visual target. The assumption is that if the agent builds proper vision-language alignment and we mask out visual objects mentioned in the instruction, then the agent may get confused since it can not find the counterpart in the visual environment.

Results are shown in Table 12. The success rate witnesses a decline when dynamically masking out the visual objects. However, comparing the "dynamic mask" setting with its controlled trial, we notice in surprise that specifically masking out the target visual objects only displays a slightly more significant impact over random masking. Noticeably, when all visual objects mentioned in the instruction are masked out, the agents can still reach a success rate higher than 40% on R2R and higher than 32% on RxR. This contradicts the previous assumption and casts doubt on the reliability of the navigation agents' vision-language alignment.

Comparing "dynamic mask" with the "mask tokens" setting, we notice that visual object ablation has less impact on navigation performance than text object ablations, which suggests that current models have unbalanced attention on vision and text for the VLN task. Recent studies on pre-trained vision-and-language models (Frank et al., 2021) reveals that such asymmetry is also witnessed in other multimodal tasks. Future study may follow the line of constructing a more balanced VLN agent.

7 Conclusion

In this paper, we inspect how the navigation agents understand the multimodal information by conducting ablation diagnostics input data. We find out that indoor navigation agents refer to both object tokens and direction tokens in the instruction when making decisions. In contrast, outdoor navigation agents heavily rely on direction tokens and poorly understand the object tokens. When it comes to vision-and-language alignments, we witness unbalanced attention on text and vision, and doubt the reliability of cross-modal alignments. We hope this work encourages more investigation and research into understanding neural VLN agents' black-box and improves the task setups and navigation agents' capacity for future studies.

References

Anne H. Anderson, Miles Bader, Ellen Gurman Bard, Elizabeth Boyle, Gwyneth Doherty, Simon Garrod, Stephen D. Isard, Jacqueline C. Kowtko, Jan McAllister, Jim Miller, Catherine Sotillo, Henry S. Thompson, and Regina Weinert. 1991. The hcrc map task corpus. *Language and Speech*, 34:351 – 366.

Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian D. Reid, Stephen Gould, and Anton van den Hengel. 2018. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 3674–3683. IEEE Computer Society.

Giuliano Antoniol, Roldano Cattoni, and Mauro Cettolo. 2011. Robust speech understanding for robot telecontrol.

Yonatan Bisk, Deniz Yuret, and Daniel Marcu. 2016. Natural language communication with robots. In *NAACL*.

Angel X. Chang, Angela Dai, Thomas A. Funkhouser, Maciej Halber, Matthias Nießner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. 2017. Matterport3d: Learning from RGB-D data in indoor environments. In *2017 International Conference on 3D Vision, 3DV 2017, Qingdao, China, October 10-12, 2017*, pages 667–676. IEEE Computer Society.

Howard Chen, Alane Suhr, Dipendra Misra, Noah Snavely, and Yoav Artzi. 2019. TOUCHDOWN: natural language navigation and spatial reasoning in visual street environments. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 12538–12547. Computer Vision Foundation / IEEE.

Shizhe Chen, Pierre-Louis Guhur, Cordelia Schmid, and Ivan Laptev. 2021. History aware multimodal transformer for vision-and-language navigation. In *NeurIPS*.

Patrick Fernandes, Kayo Yin, Graham Neubig, and André F. T. Martins. 2021. Measuring and increasing context usage in context-aware machine translation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6467–6478, Online. Association for Computational Linguistics.

Stella Frank, Emanuele Bugliarello, and Desmond Elliott. 2021. Vision-and-language or vision-for-language? on cross-modal influence in multimodal transformers. In *EMNLP*.

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 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 3318–3329.

Sergio Guadarrama, Lorenzo Riano, David Hamilton Golland, Daniel Goehring, Yangqing Jia, Dan Klein, P. Abbeel, and Trevor Darrell. 2013. Grounding spatial relations for human-robot interaction. *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1640–1647.

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 *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 13134–13143. IEEE.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society.

Lisa Anne Hendricks, Zeynep Akata, Marcus Rohrbach, Jeff Donahue, Bernt Schiele, and Trevor Darrell. 2016. Generating visual explanations. In *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part IV*, volume 9908 of *Lecture Notes in Computer Science*, pages 3–19. Springer.

Yicong Hong, Cristian Rodriguez Opazo, Yuankai Qi, Qi Wu, and Stephen Gould. 2020a. Language and visual entity relationship graph for agent navigation. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Yicong Hong, Qi Wu, Yuankai Qi, Cristian Rodriguez Opazo, and Stephen Gould. 2020b. A recurrent vision-and-language BERT for navigation. *CoRR*, abs/2011.13922.

Haoshuo Huang, Vihan Jain, Harsh Mehta, Alexander Ku, Gabriel Magalhães, Jason Baldridge, and Eugene Ie. 2019. Transferable representation learning in vision-and-language navigation. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 7403–7412. IEEE.

Vihan Jain, Gabriel Magalhães, Alexander Ku, Ashish Vaswani, Eugene Ie, and Jason Baldridge. 2019a. Stay on the path: Instruction fidelity in vision-and-language navigation. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 1862–1872. Association for Computational Linguistics.

Vihan Jain, Gabriel Magalhães, Alexander Ku, Ashish Vaswani, Eugene Ie, and Jason Baldridge. 2019b. Stay on the path: Instruction fidelity in vision-and-language navigation. *ArXiv*, abs/1905.12255.

Liyiming Ke, Xiujun Li, Yonatan Bisk, Ari Holtzman, Zhe Gan, Jingjing Liu, Jianfeng Gao, Yejin Choi, and Siddhartha S. Srinivasa. 2019. Tactical rewind: Self-correction via backtracking in vision-and-language navigation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 6741–6749. Computer Vision Foundation / IEEE.

Volker Klingspor, John Demiris, and Michael Kaiser. 1997. Human-robot-communication and machine learning. *APPLIED ARTIFICIAL INTELLIGENCE JOURNAL*, 11(11):719–746.

Thomas Kollar, Jayant Krishnamurthy, and Grant P. Strimel. 2013. Toward interactive grounded language acqusition. In *Robotics: Science and Systems*.

Alexander Ku, Peter Anderson, Roma Patel, Eugene Ie, and Jason Baldridge. 2020a. Room-across-room: Multilingual vision-and-language navigation with dense spatiotemporal grounding. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 4392–4412. Association for Computational Linguistics.

Alexander Ku, Peter Anderson, Roma Patel, Eugene Ie, and Jason Baldridge. 2020b. Room-Across-Room: Multilingual vision-and-language navigation with dense spatiotemporal grounding. In *Conference on Empirical Methods for Natural Language Processing (EMNLP)*.

Qing Li, Jianlong Fu, Dongfei Yu, Tao Mei, and Jiebo Luo. 2018. Tell-and-answer: Towards explainable visual question answering using attributes and captions. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 1338–1346. Association for Computational Linguistics.

Xiujun Li, Chunyuan Li, Qiaolin Xia, Yonatan Bisk, Asli Celikyilmaz, Jianfeng Gao, Noah A. Smith, and Yejin Choi. 2019. Robust navigation with language pretraining and stochastic sampling. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 1494–1499. Association for Computational Linguistics.

Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. 2016. Hierarchical question-image co-attention for visual question answering. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 289–297.

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 *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Chih-Yao Ma, Zuxuan Wu, Ghassan AlRegib, Caiming Xiong, and Zsolt Kira. 2019b. The regretful agent: Heuristic-aided navigation through progress estimation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 6732–6740. Computer Vision Foundation / IEEE.

Cynthia Matuszek, Liefeng Bo, Luke Zettlemoyer, and Dieter Fox. 2014. Learning from unscripted deictic gesture and language for human-robot interactions. In *AAAI*.

Harsh Mehta, Yoav Artzi, Jason Baldridge, Eugene Ie, and Piotr Mirowski. 2020. Retouchdown: Adding touchdown to streetlearn as a shareable resource for language grounding tasks in street view. *CoRR*, abs/2001.03671.

Piotr Mirowski, Matthew Koichi Grimes, Mateusz Malinowski, Karl Moritz Hermann, Keith Anderson, Denis Teplyashin, Karen Simonyan, Koray Kavukcuoglu, Andrew Zisserman, and Raia Hadsell. 2018. Learning to navigate in cities without a map. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 2424–2435.

Dipendra Misra, Andrew Bennett, Valts Blukis, Eyvind Niklasson, Max Shatkhin, and Yoav Artzi. 2018. Mapping instructions to actions in 3d environments with visual goal prediction. In *EMNLP*.

Khanh Nguyen, Debadeepta Dey, Chris Brockett, and Bill Dolan. 2019. Vision-based navigation with language-based assistance via imitation learning with indirect intervention. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 12527–12537. Computer Vision Foundation / IEEE.

Khanh Nguyen and Hal Daumé III. 2019. Help, anna! visual navigation with natural multimodal assistance via retrospective curiosity-encouraging imitation learning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 684–695. Association for Computational Linguistics.

Joe O'Connor and Jacob Andreas. 2021. What context features can transformer language models use? In *ACL/IJCNLP*.

Dong Huk Park, Lisa Anne Hendricks, Zeynep Akata, Anna Rohrbach, Bernt Schiele, Trevor Darrell, and

Marcus Rohrbach. 2018. Multimodal explanations: Justifying decisions and pointing to the evidence. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 8779–8788. IEEE Computer Society.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020a. Stanza: A python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, ACL 2020, Online, July 5-10, 2020*, pages 101–108. Association for Computational Linguistics.

Yuankai Qi, Zizheng Pan, Shengping Zhang, Anton van den Hengel, and Qi Wu. 2020b. 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*, volume 12355 of *Lecture Notes in Computer Science*, pages 303–317. Springer.

Yuankai Qi, Qi Wu, Peter Anderson, Xin Wang, William Yang Wang, Chunhua Shen, and Anton van den Hengel. 2020c. REVERIE: remote embodied visual referring expression in real indoor environments. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 9979–9988. IEEE.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. In *ICML*.

Deb K. Roy. 2002. Learning visually grounded words and syntax for a scene description task. *Comput. Speech Lang.*, 16:353–385.

Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. 2017. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 618–626. IEEE Computer Society.

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*, abs/2107.06383.

Luc L. Steels and Paul Vogt. 1997. Grounding adaptive language games in robotic agents.

Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander Amir Alemi. 2017. Inception-v4, inception-resnet and the impact of residual connections on learning. In *AAAI*.

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, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 2610–2621. Association for Computational Linguistics.

Jesse Thomason, Daniel Gordon, and Yonatan Bisk. 2019. Shifting the baseline: Single modality performance on visual navigation & qa. In *NAACL*.

Hu Wang, Qi Wu, and Chunhua Shen. 2020a. Soft expert reward learning for vision-and-language navigation. In *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part IX*, volume 12354 of *Lecture Notes in Computer Science*, pages 126–141. Springer.

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 *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 6629–6638. Computer Vision Foundation / IEEE.

Xin Wang, Wenhan Xiong, Hongmin Wang, and William Yang Wang. 2018. Look before you leap: Bridging model-free and model-based reinforcement learning for planned-ahead vision-and-language navigation. In *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XVI*, volume 11220 of *Lecture Notes in Computer Science*, pages 38–55. Springer.

Xin Eric Wang, Vihan Jain, Eugene Ie, William Yang Wang, Zornitsa Kozareva, and Sujith Ravi. 2020b. Environment-agnostic multitask learning for natural language grounded navigation. In *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XXIV*, volume 12369 of *Lecture Notes in Computer Science*, pages 413–430. Springer.

Terry Winograd. 1971. Procedures as a representation for data in a computer program for understanding natural language.

Jialin Wu and Raymond Mooney. 2019. Faithful multimodal explanation for visual question answering. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 103–112, Florence, Italy. Association for Computational Linguistics.

Qiaolin Xia, Xiujun Li, Chunyuan Li, Yonatan Bisk, Zhifang Sui, Jianfeng Gao, Yejin Choi, and Noah A. Smith. 2020. Multi-view learning for vision-and-language navigation. *CoRR*, abs/2003.00857.

Jiannan Xiang, Xin Wang, and William Yang Wang. 2020. Learning to stop: A simple yet effective approach to urban vision-language navigation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings, EMNLP 2020, Online Event, 16-20 November 2020*, pages 699–707. Association for Computational Linguistics.

Fengda Zhu, Yi Zhu, Xiaojun Chang, and Xiaodan Liang. 2020a. Vision-language navigation with self-supervised auxiliary reasoning tasks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 10009–10019. IEEE.

Wanrong Zhu, Xin Wang, Tsu-Jui Fu, An Yan, Pradyumna Narayana, Kazoo Sone, Sugato Basu, and William Yang Wang. 2020b. Multimodal text style transfer for outdoor vision-and-language navigation. *CoRR*, abs/2007.00229.

Yi Zhu, Fengda Zhu, Zhaohuan Zhan, Bingqian Lin, Jianbin Jiao, Xiaojun Chang, and Xiaodan Liang. 2020c. Vision-dialog navigation by exploring cross-modal memory. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 10727–10736. IEEE.

A Appendix

A.1 Heuristics Model

In this work, we use Stanza (Qi et al., 2020a) part-of-speech (POS) tagger to locate object-related tokens. A token is regarded as object token if its POS tag is *NOUN* or *PROPN*. In this section, we verify the accuracy of heuristics models. We first randomly sample 50 instructions from R2R validation unseen set. Then the authors of this work manually mark out the object tokens in the instructions, and compare the annotations with the results yield by Stanza POS tagger. We report the recall and precision score in Table 13. A few misaligned tokens are "turn", "stand", and the tokens mis-spelled in original instructions. However, the misalignment occurs infrequently. The heuristic POS tagging model is sufficient to detect object tokens in our study.

Precision	Recall
98.7%	99.1%

Table 13: The precision and recall when comparing object token annotation results provided by human and Stanza POS tagger.

A.2 Effect of Directions in the Environment

We randomly flip some of the viewpoints horizontally. The objects' relative positions at the flipped viewpoints will be reversed. Presumably, suppose the agent can follow the instruction and find the corresponding direction to approach. In that case, the flipped viewpoints will misguide the agent in the opposite direction and lower the navigation success rate. As shown in Table 14, with more and more viewpoints being swapped from left to right, the SR drastically declines for all three agents. This verifies our previous finding that indoor agents can understand directions in the instruction.

Setting	SR ↑ on R2R				SR ↑ on RxR	
	EnvDrop	FAST	Recur	PVLT	CLIPViL	HAMT
Horizontal Flip	-51%	-29%	-48%	-59%	-36%	-47%

Table 14: The indoor navigation performance for visual direction ablations. Recur: VLN ⟳ BERT. PVLT: PREVALENT.