Environment-agnostic Multitask Learning for Natural Language Grounded Navigation

Xin Eric Wang[1,2⋆ ⋆⋆], Vihan Jain[3⋆], Eugene Ie[3], William Yang Wang[2], Zornitsa Kozareva[3], and Sujith Ravi[3,4⋆⋆]

[1] University of California, Santa Cruz
[2] University of California, Santa Barbara
[3] Google Research
[4] Amazon

Abstract. Recent research efforts enable study for natural language grounded navigation in photo-realistic environments, e.g., following natural language instructions or dialog. However, existing methods tend to overfit training data in seen environments and fail to generalize well in previously unseen environments. To close the gap between seen and unseen environments, we aim at learning a generalized navigation model from two novel perspectives: (1) we introduce a multitask navigation model that can be seamlessly trained on both Vision-Language Navigation (VLN) and Navigation from Dialog History (NDH) tasks, which benefits from richer natural language guidance and effectively transfers knowledge across tasks; (2) we propose to learn environment-agnostic representations for the navigation policy that are invariant among the environments seen during training, thus generalizing better on unseen environments. Extensive experiments show that environment-agnostic multitask learning significantly reduces the performance gap between seen and unseen environments, and the navigation agent trained so outperforms baselines on unseen environments by 16% (relative measure on success rate) on VLN and 120% (goal progress) on NDH. Our submission to the CVDN leaderboard establishes a new state-of-the-art for the NDH task on the holdout test set. Code is available at https://github.com/google-research/valan.

Keywords: Vision-and-Language Navigation, Natural Language Grounding, Multitask Learning, Agnostic Learning

1 Introduction

Navigation in visual environments by following natural language guidance [18] is a fundamental capability of intelligent robots that simulate human behaviors, because humans can easily reason about the language guidance and navigate efficiently by interacting with the visual environments. Recent efforts [3, 9, 42,

⋆ Equal contribution.
⋆⋆ Work done at Google.

36] empower large-scale learning of natural language grounded navigation that is situated in photo-realistic simulation environments.

Nevertheless, the generalization problem commonly exists for these tasks, especially indoor navigation: the agent usually performs poorly on unknown environments that have never been seen during training. One of the leading causes of such behavior is data scarcity, as it is expensive and time-consuming to extend either visual environments or natural language guidance. The number of scanned houses for indoor navigation is limited due to high expense and privacy concerns. Besides, unlike vision-only navigation tasks [31, 32, 50, 30, 23, 25] where episodes can be exhaustively sampled in simulation, natural language grounded navigation is supported by human demonstrated interaction in natural language. It is impractical to fully collect all the samples for individual tasks.

Therefore, it is essential though challenging to efficiently learn a more generalized policy for natural language grounded navigation tasks from existing data [48, 49]. In this paper, we study how to resolve the generalization and data scarcity issues from two different angles. First, previous methods are trained for one task at the time, so each new task requires training a new agent instance from scratch that can only solve the one task on which it was trained. In this work, we propose a generalized multitask model for natural language grounded navigation tasks such as Vision-Language Navigation (VLN) and Navigation from Dialog History (NDH), aiming to efficiently transfer knowledge across tasks and effectively solve all the tasks simultaneously with one agent.

Furthermore, even though there are thousands of trajectories paired with language guidance, the underlying house scans are restricted. For instance, the popular Matterport3D environment [6] contains only 61 unique house scans in the training set. The current models perform much better in seen environments by taking advantage of the knowledge of specific houses they have acquired over multiple task completions during training, but fail to generalize to houses not seen during training. To overcome this shortcoming, we propose an environment-agnostic learning method to learn a visual representation that is invariant to specific environments but can still support navigation. Endowed with the learned environment-agnostic representations, the agent is further prevented from the overfitting issue and generalizes better on unseen environments.

To the best of our knowledge, we are the first to introduce natural language grounded multitask and environment-agnostic training regimes and validate their effectiveness on VLN and NDH tasks. Extensive experiments demonstrate that our environment-agnostic multitask navigation model can not only efficiently execute different language guidance in indoor environments but also outperform the single-task baseline models by a large margin on both tasks. Besides, the performance gap between seen and unseen environments is significantly reduced. Furthermore, our leaderboard submission for the NDH task establishes a new state-of-the-art outperforming the existing best agent by more than 66% on the primary metric of goal progress on the holdout test set.

2 Background

Vision-and-Language Navigation. As depicted in Figure 1, Vision-and-Language Navigation [3, 7] task requires an embodied agent to navigate in photo-realistic environments to carry out natural language instructions. For a given path, the associated natural language instructions describe the step-by-step guidance from the starting position to the target position. The agent is spawned at an initial pose $p_0 = (v_0, \phi_0, \theta_0)$, which includes the spatial location, heading and elevation angles. Given a natural language instruction $X = \{x_1, x_2, ..., x_n\}$, the agent is expected to perform a sequence of actions $\{a_1, a_2, ..., a_T\}$ and arrive at the target position v_{tar} specified by the language instruction X. In this work, we consider the VLN task defined for Room-to-Room (R2R) [3] dataset, which contains instruction-trajectory pairs across 90 different indoor environments (houses). The instructions for a given trajectory in the dataset on an average contain 29 words. Previous VLN methods have studied various aspects to improve the navigation performance, such as planning [46], data augmentation [14, 40, 15], cross-modal alignment [45, 20], progress estimation [28], error correction [29, 22], interactive language assistance [34, 33] etc. This work tackles VLN via multitask learning and environment-agnostic learning, which is orthogonal to all these prior arts.

Navigation from Dialog History. Different from Visual Dialog [10] that involves dialog grounded in a single image, the recently introduced Cooperative Vision-and-Dialog Navigation (CVDN) dataset [42] includes interactive language assistance for indoor navigation, which consists of over 2,000 embodied, human-human dialogs situated in photo-realistic home environments. The task of Navigation from Dialog History (NDH) demonstrated in Figure 1, is defined as: given a target object t_0 and a dialog history between humans cooperating to perform the task, the embodied agent must infer navigation actions towards the goal room that contains the target object. The dialog history is denoted as $< t_0, Q_1, A_1, Q_2, A_2, ..., Q_i, A_i >$, including the target object t_0, the questions Q and answers A till the turn i ($0 \leq i \leq k$, where k is the total number of Q-A turns from the beginning to the goal room). The agent, located in p_0, is trying to move closer to the goal room by inferring from the dialog history that happened before. The dialog for a given trajectory lasts 6 utterances (3 question-answer exchanges) and is 82 words long on an average.

Multitask Learning. The basis of multitask learning is the notion that tasks can serve as mutual sources of inductive bias for each other [5]. When multiple tasks are trained jointly, multitask learning causes the learner to prefer the hypothesis that explains all the tasks simultaneously, leading to more generalized solutions. Multitask learning has been successful in natural language processing [8], speech recognition [11], computer vision [17], drug discovery [37], and Atari games [41]. The deep reinforcement learning methods that have become very popular for training models on natural language grounded navigation tasks [45, 19, 20, 40] are known to be data inefficient. In this work, we introduce multitask reinforcement learning for such tasks to improve data efficiency by positive transfer across related tasks.

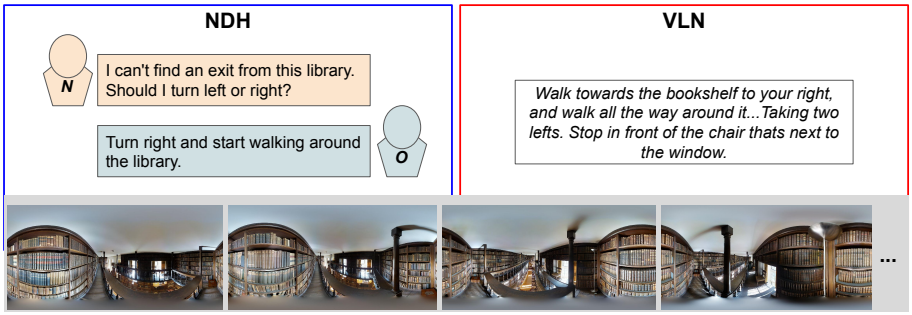

Fig. 1: While the NDH task (left) requires an agent to navigate using dialog history between two human players - a navigator (N) who is trying to find the goal room with the help of an oracle (O), the VLN task (right) requires navigating using instructions written by human annotators.

Agnostic Learning. A few studies on agnostic learning have been proposed recently. For example, Model-Agnostic Meta-Learning (MAML) [13] aims to train a model on a variety of learning tasks and solve a new task using only a few training examples. Liu *et al.* [27] proposes a unified feature disentangler that learns domain-invariant representation across multiple domains for image translation. Other domain-agnostic techniques are also proposed for supervised [26] and unsupervised domain adaption [38, 35]. In this work, we pair the environment classifier with a gradient reversal layer [16] to learn an environment-agnostic representation that can be better generalized on unseen environments in a zero-shot fashion where no adaptation is involved.

Distributed Actor-Learner Navigation Learning Framework. To train models for the various language grounded navigation tasks like VLN and NDH, we use the VALAN framework [24], a distributed actor-learner learning infrastructure. The framework is inspired by IMPALA [12] and uses its off-policy correction method called V-trace to scale reinforcement learning methods to thousands of machines efficiently. The framework additionally supports a variety of supervision strategies essential for navigation tasks such as teacher-forcing [3], student-forcing [3] and mixed supervision [42]. The framework is built using TensorFlow [1] and supports ML accelerators (GPU, TPU).

3 Environment-agnostic Multitask Learning

3.1 Overview

Our environment-agnostic multitask navigation model is illustrated in Figure 2. First, we adapt the reinforced cross-modal matching (RCM) model [45] and make it seamlessly transfer across tasks by sharing all the learnable parameters for both NDH and VLN, including joint word embedding layer, language encoder, trajectory encoder, cross-modal attention module (CM-ATT), and action

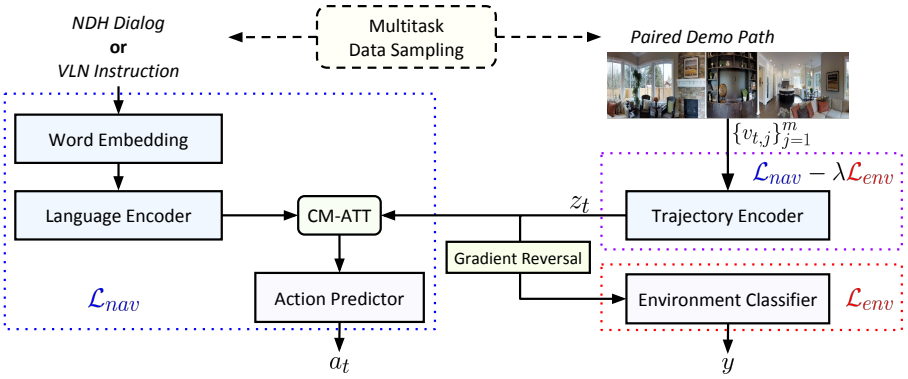

Fig. 2: Overview of environment-agnostic multitask learning.

predictor. Furthermore, to learn the environment-agnostic representation $\boldsymbol{z}_t$, we equip the navigation model with an environment classifier whose objective is to predict which house the agent is. However, note that between trajectory encoder and environment classifier, a gradient reversal layer [16] is introduced to reverse the gradients back-propagated to the trajectory encoder, making it learn representations that are environment-agnostic and thus more generalizable in unseen environments. During training, the environment classifier is minimizing the environment classification loss $\mathcal{L}_{env}$, while the trajectory encoder is maximizing $\mathcal{L}_{env}$ and minimizing the navigation loss $\mathcal{L}_{nav}$. The other modules are optimized with the navigation loss $\mathcal{L}_{nav}$ simultaneously. Below we introduce multitask reinforcement learning and environment-agnostic representation learning. A more detailed model architecture is presented in Sec. 4.

3.2 Multitask Reinforcement Learning

In this section, we describe how we adapted the RCM agent model to learn the two tasks of VLN and NDH simultaneously. It is worth noting that even though both the VLN and NDH tasks use the same Matterport3D indoor environments [6], there are significant differences in the motivations and the overall objectives of the two tasks. While the natural language descriptions associated with the paths in the VLN task are step-by-step instructions to follow the ground-truth paths, the descriptions of the paths in the NDH task are series of question-answer interactions (dialog) between two human players which need not necessarily align sequentially with the ground-truth paths. This difference in the style of the two tasks also manifests in their respective datasets — the average path description length and average path length in the NDH task's dataset are roughly three times that of the VLN task's dataset. Furthermore, while the objective in VLN is to find the exact goal node in the environment (i.e., point navigation), the objective in NDH is to find the goal room that contains the specified object (i.e., room navigation).

Interleaved Multitask Data Sampling. To avoid overfitting individual tasks, we adopt an interleaved multitask data sampling strategy to train the model. Particularly, each data sample within a mini-batch can be from either task, so that the VLN instruction-trajectory pairs and NDH dialog-trajectory pairs are interleaved in a mini-batch though they may have different learning objectives.

Reward Shaping. Following prior art [46, 45], we first implement a discounted cumulative reward function R for the VLN and NDH tasks:

$$R(s_t, a_t) = \sum_{t'=t}^{T} \gamma^{t'-t} r(s_{t'}, a_{t'}) \tag{1}$$

where γ is the discounted factor. For the VLN task, we choose the immediate reward function such that the agent is rewarded at each step for getting closer to (or penalized for getting further from) the target location. At the end of the episode, the agent receives a reward only if it terminated successfully. Formally,

$$r(s_{t'}, a_{t'}) = \begin{cases} d(s_{t'}, v_{tar}) - d(s_{t'+1}, v_{tar}) & \text{if } t' < T \\ \mathbb{1}[d(s_T, v_{tar}) \leq d_{th}] & \text{if } t' = T \end{cases} \tag{2}$$

where $d(s_t, v_{tar})$ is the distance between state s_t and the target location v_{tar}, $\mathbb{1}[.]$ is the indicator function and d_{th} is the maximum distance from v_{tar} that the agent is allowed to terminate for success.

Different from VLN, the NDH task is essentially room navigation instead of point navigation because the agent is expected to reach a room that contains the target object. Suppose the goal room is occupied by a set of nodes $\{v_i\}_1^N$, we replace the distance function $d(s_t, v_{tar})$ in Equation 2 with the minimum distance to the goal room $d_{room}(s_t, \{v_i\}_1^N)$ for NDH:

$$d_{room}(s_t, \{v_i\}_1^N) = \min_{1 \leq i \leq N} d(s_t, v_i) \tag{3}$$

Navigation Loss. Since human demonstrations are available for both VLN and NDH tasks, we use behavior cloning to constrain the learning algorithm to model state-action spaces that are most relevant to each task. Following previous works [45], we also use reinforcement learning to aid the agent's ability to recover from erroneous actions in unseen environments. During navigation model training, we adopt a mixed training strategy of reinforcement learning and behavior cloning, so the navigation loss function is:

$$\mathcal{L}_{nav} = -\mathbb{E}_{a_t \sim \pi}[R(s_t, a_t) - b] - \mathbb{E}[\log \pi(a_t^* | s_t)] \tag{4}$$

where we use REINFORCE policy gradients [47] and supervised learning gradients to update the policy π. b is the estimated baseline to reduce the variance and a_t^* is the human demonstrated action.

3.3 Environment-agnostic Representation Learning

To further improve the navigation policy's generalizability, we propose to learn a latent environment-agnostic representation that is invariant among seen environments. The objective is to not learn the intricate environment-specific features that are irrelevant to general navigation (e.g. unique house appearances), preventing the model from overfitting to specific seen environments. We can reformulate the navigation policy as

$$\pi(a_t|s_t) = p(a_t|\boldsymbol{z}_t, s_t)p(\boldsymbol{z}_t|s_t) \tag{5}$$

where $\boldsymbol{z}_t$ is a latent representation.

As shown in Figure 2, $p(a_t|\boldsymbol{z}_t, s_t)$ is modeled by the policy module (including CM-ATT and action predictor) and $p(\boldsymbol{z}_t|s_t)$ is modeled by the trajectory encoder. In order to learn the environment-agnostic representation, we employ an environment classifier and a gradient reversal layer [16]. The environment classifier is parameterized to predict the house identity, so its loss function $\mathcal{L}_{env}$ is defined as

$$\mathcal{L}_{env} = -\mathbb{E}[\log p(y = y^*|\boldsymbol{z}_t)] \tag{6}$$

where y^* is the ground-truth house label. The gradient reversal layer has no parameters. It acts as an identity transform during forward-propagation, but multiplies the gradient by $-\lambda$ and passes it to the trajectory encoder during back-propagation. Therefore, in addition to minimizing the navigation loss $\mathcal{L}_{nav}$, the trajectory encoder is also maximizing the environment classification loss $\mathcal{L}_{env}$. While the environment classifier is minimizing the classification loss conditioned on the latent representation $\boldsymbol{z}_t$, the trajectory encoder is trying to increase the classifier's entropy, resulting in an adversarial learning objective.

4 Model Architecture

Language Encoder. The natural language guidance (instruction or dialog) is tokenized and embedded into n-dimensional space $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_n\}$ where the word vectors $\boldsymbol{x}_i$ are initialized randomly. The vocabulary is restricted to tokens that occur at least five times in the training instructions (the vocabulary used when jointly training VLN and NDH tasks is the union of the two tasks' vocabularies.). All out-of-vocabulary tokens are mapped to a single out-of-vocabulary identifier. The token sequence is encoded using a bi-directional LSTM [39] to create $\boldsymbol{H}^{\boldsymbol{X}}$ following:

$$\boldsymbol{H}^{\boldsymbol{X}} = [\boldsymbol{h}_1^{\boldsymbol{X}}; \boldsymbol{h}_2^{\boldsymbol{X}}; ...; \boldsymbol{h}_n^{\boldsymbol{X}}], \quad \boldsymbol{h}_t^{\boldsymbol{X}} = \sigma(\overrightarrow{\boldsymbol{h}}_t^{\boldsymbol{X}}, \overleftarrow{\boldsymbol{h}}_t^{\boldsymbol{X}}) \tag{7}$$

$$\overrightarrow{\boldsymbol{h}}_t^{\boldsymbol{X}} = LSTM(\boldsymbol{x}_t, \overrightarrow{\boldsymbol{h}}_{t-1}^{\boldsymbol{X}}), \quad \overleftarrow{\boldsymbol{h}}_t^{\boldsymbol{X}} = LSTM(\boldsymbol{x}_t, \overleftarrow{\boldsymbol{h}}_{t+1}^{\boldsymbol{X}}) \tag{8}$$

where $\overrightarrow{\boldsymbol{h}}_t^{\boldsymbol{X}}$ and $\overleftarrow{\boldsymbol{h}}_t^{\boldsymbol{X}}$ are the hidden states of the forward and backward LSTM layers at time step t respectively, and the σ function is used to combine $\overrightarrow{\boldsymbol{h}}_t^{\boldsymbol{X}}$ and $\overleftarrow{\boldsymbol{h}}_t^{\boldsymbol{X}}$ into $\boldsymbol{h}_t^{\boldsymbol{X}}$.

Trajectory Encoder. Similar to benchmark models [14, 45, 20], at each time step t, the agent perceives a 360-degree panoramic view at its current location. The view is discretized into k view angles ($k = 36$ in our implementation, 3 elevations by 12 headings at 30-degree intervals). The image at view angle i, heading angle ϕ and elevation angle θ is represented by a concatenation of the pre-trained CNN image features with the 4-dimensional orientation feature [sin ϕ; cos ϕ; sin θ; cos θ] to form $\boldsymbol{v}_{t,i}$. The visual input sequence $\boldsymbol{V} = \{\boldsymbol{v}_1, \boldsymbol{v}_2, ..., \boldsymbol{v}_m\}$ is encoded using a LSTM to create $\boldsymbol{H^V}$ following:

$$\boldsymbol{H^V} = [\boldsymbol{h_1^V}; \boldsymbol{h_2^V}; ...; \boldsymbol{h_m^V}], \quad \text{where } \boldsymbol{h_t^V} = LSTM(\boldsymbol{v}_t, \boldsymbol{h_{t-1}^V}) \tag{9}$$

$\boldsymbol{v}_t = \text{Attention}(\boldsymbol{h_{t-1}^V}, \boldsymbol{v}_{t,1..k})$ is the attention-pooled representation of all view angles using previous agent state $\boldsymbol{h}_{t-1}$ as the query. We use the dot-product attention [43] hereafter.

Policy Module. The policy module comprises of cross-modal attention (CM-ATT) unit as well as an action predictor. The agent learns a policy π_θ over parameters θ that maps the natural language instruction $\boldsymbol{X}$ and the initial visual scene $\boldsymbol{v}_1$ to a sequence of actions $[a_1, a_2, ..., a_n]$. The action space which is common to VLN and NDH tasks consists of navigable directions from the current location. The available actions at time t are denoted as $\boldsymbol{u}_{t,1..l}$, where $\boldsymbol{u}_{t,j}$ is the representation of the navigable direction j from the current location obtained similarly to $\boldsymbol{v}_{t,i}$. The number of available actions, l, varies per location, since graph node connectivity varies. Following Wang *et al.* [45], the model predicts the probability p_d of each navigable direction d using a bilinear dot product:

$$p_d = \text{softmax}([\boldsymbol{h_t^V}; \boldsymbol{c_t^{\text{text}}}; \boldsymbol{c_t^{\text{visual}}}]\boldsymbol{W}_c(\boldsymbol{u}_{t,d}\boldsymbol{W}_u)^T) \tag{10}$$

where $\boldsymbol{c_t^{\text{text}}} = \text{Attention}(\boldsymbol{h_t^V}, \boldsymbol{h_{1..n}^X})$ and $\boldsymbol{c_t^{\text{visual}}} = \text{Attention}(\boldsymbol{c_t^{\text{text}}}, \boldsymbol{v}_{t,1..k})$. $\boldsymbol{W}_c$ and $\boldsymbol{W}_u$ are learnable parameters.

Environment Classifier. The environment classifier is a two-layer perceptron with a SoftMax layer as the last layer. Given the latent representation $\boldsymbol{z}_t$ (which is $\boldsymbol{h_t^V}$ in our setting), the classifier generates a probability distribution over the house labels.

5 Experiments

5.1 Experimental Setup

Implementation Details. We use a 2-layer bi-directional LSTM for the instruction encoder, where the size of LSTM cells is 256 in each direction. The inputs to the encoder are 300-dimensional embeddings initialized randomly. For the visual encoder, we use a 2-layer LSTM with a cell size of 512. The encoder inputs are image features derived as mentioned in Sec. 4. The cross-modal attention layer size is 128 units. The environment classifier has one hidden layer of size 128 units, followed by an output layer of size equal to the number of classes. The negative gradient multiplier λ in the gradient reversal layer is empirically tuned

and fixed at a value of 1.3 for all experiments. During training, some episodes in the batch are identical to available human demonstrations in the training dataset, where the objective is to increase the agent's likelihood of choosing human actions (behavioral cloning [4]). The rest of the episodes are constructed by sampling from the agent's own policy. For the NDH task, we deploy mixed supervision similar to Thomason *et al.* [42], where the navigator's or oracle's path is selected as ground-truth depending on if the navigator was successful in reaching the correct end node following the question-answer exchange with the oracle or not. In the experiments, unless otherwise stated, we use the entire dialog history from the NDH task for model training. *All the reported results in subsequent studies are averages of at least three independent runs.*

Evaluation Metrics. The agents are evaluated on two datasets, namely *Validation Seen* that contains new paths from the training environments and *Validation Unseen* that contains paths from previously unseen environments. The evaluation metrics for VLN task are as follows: *Path Length (PL)* measures the total length of the predicted path; *Navigation Error (NE)* measures the distance between the last nodes in the predicted and the reference paths; *Success Rate (SR)* measures how often the last node in the predicted path is within some threshold distance of the last node in the reference path; *Success weighted by Path Length (SPL)* [2] measures Success Rate weighted by the normalized Path Length; and *Coverage weighted by Length Score (CLS)* [21] measures predicted path's conformity to the reference path weighted by length score. For the NDH task, the agent's progress is defined as a reduction (in meters) from the distance to the goal region at the agent's first position versus at its last position [42].

5.2 Environment-agnostic Multitask Learning

Table 1 shows the results of training the navigation model using environment-agnostic learning (*EnvAg*) as well as multitask learning (*MT-RCM*). First, both learning methods independently help the agent learn more generalized navigation policy, as is evidenced by a significant reduction in agent's performance gap between seen and unseen environments (better visualized with Figure 3). For instance, the performance gap in goal progress on the NDH task drops from 3.85m to 0.92m using multitask learning, and the performance gap in success rate on the VLN task drops from 9.26% to 8.39% using environment-agnostic learning. Second, the two techniques are complementary—the agent's performance when trained with both the techniques simultaneously improves on unseen environments compared to when trained separately. Finally, we note here that *MT-RCM + EnvAg* outperforms the baseline goal progress of 2.10m [42] on NDH validation unseen dataset by more than 120%. At the same time, it outperforms the equivalent RCM baseline [45] of 40.6% success rate by more than 16% (relative measure) on VLN validation unseen dataset.

To further validate our results on NDH task, we evaluated the *MT-RCM + EnvAg* agent on the test set of NDH dataset which is held out as the CVDN

[5] The equivalent RCM model without intrinsic reward is used as the benchmark.

Table 1: The agent's performance under different training strategies. The single-task RCM (ST-RCM) model is independently trained and tested on VLN or NDH tasks. The standard deviation across 3 independent runs is reported.

Fold	Model	NDH	VLN				
		Progress ↑	PL	NE ↓	SR ↑	SPL ↑	CLS ↑
Val Seen	seq2seq [42]	5.92					
	RCM [45][5]		12.08	**3.25**	**67.60**	-	-
	Ours						
	ST-RCM	**6.49** ±0.95	10.75 ±0.26	5.09 ±0.49	52.39 ±3.58	48.86 ±3.66	63.91 ±2.41
	ST-RCM + EnvAg	6.07 ±0.56	11.31 ±0.26	4.93 ±0.49	52.79 ±3.72	48.85 ±3.71	63.26 ±2.31
	MT-RCM	5.28 ±0.56	10.63 ±0.10	5.09 ±0.05	**56.42** ±1.21	**49.67** ±1.07	**68.28** ±0.16
	MT-RCM + EnvAg	5.07 ±0.45	11.60 ±0.30	**4.83** ±0.12	53.30 ±0.71	49.39 ±0.74	64.10 ±0.16
Val Unseen	seq2seq [42]	2.10					
	RCM [45]		15.00	6.02	40.60	-	-
	Ours						
	ST-RCM	2.64 ±0.06	10.60 ±0.27	6.10 ±0.06	42.93 ±0.21	38.88 ±0.20	54.86 ±0.92
	ST-RCM + EnvAg	3.15 ±0.29	11.36 ±0.27	5.79 ±0.06	44.40 ±2.14	40.30 ±2.12	55.77 ±1.31
	MT-RCM	4.36 ±0.17	10.23 ±0.14	**5.31** ±0.18	46.20 ±0.55	**44.19** ±0.64	54.99 ±0.87
	MT-RCM + EnvAg	**4.65** ±0.20	12.05 ±0.23	5.41 ±0.20	**47.22** ±1.00	41.80 ±1.11	**56.22** ±0.87

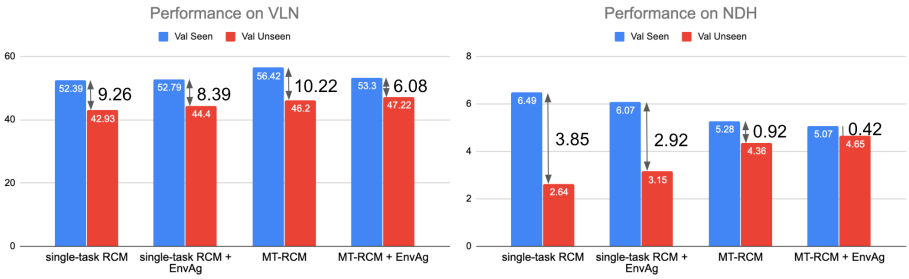

Fig. 3: Visualizing performance gap between seen and unseen environments for VLN (success rate) and NDH (progress) tasks.

challenge[6]. Table 2 shows that our submission to the leaderboard with *MT-RCM + EnvAg* establishes a new state-of-the-art on this task outperforming the existing best agent by more than 66%.

5.3 Multitask Learning

We then conduct studies to examine cross-task transfer using multitask learning alone. First, we experiment multitasking learning with access to different parts of the dialog—the target object t_o, the last oracle answer A_i, the prefacing navigator question Q_i, and the full dialog history. Table 3 shows the results of

[6] https://evalai.cloudcv.org/web/challenges/challenge-page/463/leaderboard/1292

Table 2: Comparison on CVDN Leaderboard Test Set. Note that the metric *Progress* is the same as *dist_to_end_reduction*.

	Agent	Progress ↑
Baselines	Random	0.83
	Shortest Path Agent (*upper bound*)	9.76
Leaderboard Submissions	Seq2Seq [42]	2.35
	MT-RCM + EnvAg	**3.91**

jointly training *MT-RCM* model on VLN and NDH tasks. (1) *Does VLN complement NDH?* Yes, consistently. On NDH Val Unseen, *MT-RCM* consistently benefits from following shorter paths with step-by-step instructions in VLN for all kinds of dialog inputs. It shows that VLN can serve as an essential task to boost learning of primitive action-and-instruction following and therefore support more complicated navigation tasks like NDH. (2) *Does NDH complement VLN?* Yes, under certain conditions. From the results on VLN Val Unseen, we can observe that MT-RCM with only target objects as the guidance performs equivalently or slightly worse than *VLN-RCM*, showing that extending visual paths alone (even with final targets) is not helpful in VLN. But we can see a consistent and gradual increase in the success rate of *MT-RCM* on the VLN task as it is trained on paths with richer dialog history from the NDH task. This shows that the agent benefits from more fine-grained information about the path implying the importance given by the agent to the language instructions in the task. (3) Multitask learning improves the generalizability of navigation models: the seen-unseen performance gap is narrowed. (4) As a side effect, results of different dialog inputs on NDH Val Seen *versus* Unseen verify the essence of language guidance in generalizing navigation to unseen environments.

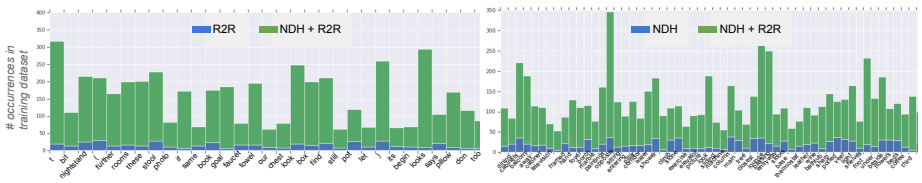

Fig. 4: Selected tokens from the vocabulary for VLN (left) and NDH (right) tasks which gained more than 40 additional occurrences in the training dataset due to joint-training.

Besides, we show multitask learning results in better language grounding through more appearance of individual words in Figure 4 and shared semantic encoding of the whole sentences in Table 4. Figure 4 illustrates that under-represented tokens in each of the individual tasks get a significant boost in the

Table 3: Comparison of agent performance when trained separately *vs.* jointly on VLN and NDH tasks.

Fold	Model	NDH Evaluation: Inputs for NDH t_o	A_i	Q_i	$A_{1:i-1}; Q_{1:i-1}$	Progress ↑	VLN Evaluation: PL	NE ↓	SR ↑	SPL ↑	CLS ↑
Val Seen	NDH-RCM	✓				**6.97**					
		✓	✓			**6.92**					
		✓	✓	✓		**6.47**					
		✓	✓	✓	✓	**6.49**					
	VLN-RCM						10.75	5.09	52.39	48.86	63.91
	MT-RCM	✓				3.00	11.73	4.87	54.56	52.00	65.64
		✓	✓			5.92	11.12	4.62	54.89	**52.62**	66.05
		✓	✓	✓		5.43	10.94	**4.59**	54.23	52.06	66.93
		✓	✓	✓	✓	5.28	10.63	5.09	**56.42**	49.67	**68.28**
Val Unseen	NDH-RCM	✓				1.25					
		✓	✓			2.69					
		✓	✓	✓		2.69					
		✓	✓	✓	✓	2.64					
	VLN-RCM						10.60	6.10	42.93	38.88	54.86
	MT-RCM	✓				**1.69**	13.12	5.84	42.75	38.71	53.09
		✓	✓			**4.01**	11.06	5.88	42.98	40.62	54.30
		✓	✓	✓		**3.75**	11.08	5.70	44.50	39.67	54.95
		✓	✓	✓	✓	**4.36**	10.23	**5.31**	**46.20**	**44.19**	**54.99**

Table 4: Comparison of agent performance when language instructions are encoded by separate *vs.* shared encoder for VLN and NDH tasks.

Language Encoder	Val Seen: NDH	VLN					Val Unseen: NDH	VLN				
	Progress ↑	PL	NE ↓	SR ↑	SPL ↑	CLS ↑	Progress ↑	PL	NE ↓	SR ↑	SPL ↑	CLS ↑
Shared	**5.28**	10.63	5.09	**56.42**	**49.67**	**68.28**	**4.36**	10.23	**5.31**	**46.20**	**44.19**	**54.99**
Separate	5.17	11.26	**5.02**	52.38	48.80	64.19	4.07	11.72	6.04	43.64	39.49	54.57

number of training samples. Table 4 shows that the model with shared language encoder for NDH and VLN tasks outperforms the model that has separate language encoders for the two tasks, hence demonstrating the importance of parameter sharing during multitask learning.

Furthermore, we observed that the agent's performance improves significantly when trained on a mixture of VLN and NDH paths even when the size of the training dataset is fixed, advancing the argument that multitask learning on NDH and VLN tasks complements the agent's learning. More details of the ablation studies can be found in the Appendix.

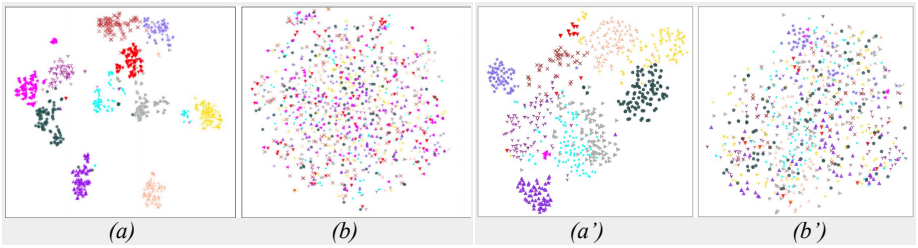

Fig. 5: t-SNE visualization of trajectory encoder's output for VLN task across 11 different color-coded seen (*a*,*b*) and unseen (*a'*,*b'*) environments. The depicted representations in *(a)* and *(a')* are learned with environment-aware objective while those in *(b)* and *(b')* are learned with environment-agnostic objective.

Table 5: Environment-agnostic *versus* environment-aware learning.

(a) Comparison on NDH.

Model	Val Seen	Val Unseen
	Progress ↑	Progress ↑
RCM	6.49	2.64
EnvAware	**8.38**	1.81
EnvAg	6.07	**3.15**

(b) Comparison on VLN.

Model	Val Seen					Val Unseen				
	PL	NE ↓	SR ↑	SPL ↑	CLS ↑	PL	NE ↓	SR ↑	SPL ↑	CLS ↑
RCM	10.75	5.09	52.39	48.86	63.91	10.60	6.10	42.93	38.88	54.86
EnvAware	10.30	**4.36**	**57.59**	**54.05**	**68.49**	10.13	6.30	38.83	35.65	54.79
EnvAg	11.31	4.93	52.79	48.85	63.26	11.36	**5.79**	**44.40**	**40.30**	**55.77**

5.4 Environment-agnostic Learning

From Table 1, it can be seen that both VLN and NDH tasks benefit from environment-agnostic learning independently. To further examine the generalization property of environment-agnostic learning, we train a model with the opposite objective—learn to correctly predict the navigation environments by removing the gradient reversal layer (*environment-aware learning*). The results in Table 5 demonstrate that environment-aware learning leads to overfitting on the training dataset as the performance on environments seen during training consistently increases for both the tasks. In contrast, environment-agnostic learning leads to a more generalized navigation policy that performs better on unseen environments. Figure 5 further shows that due to environment-aware learning, the model learns to represent visual inputs from the same environment closer to each other while the representations of different environments are farther from each other resulting in a clustering learning effect. On the other hand, environment-agnostic learning leads to more general representation across different environments, which results in better performance on unseen environments.

5.5 Reward Shaping for NDH task

As discussed in Sec. 3.2, we conducted studies to shape the reward for the NDH task. Table 6 presents the results of training the agent with access to different

Table 6: Average agent progress towards goal room when trained using different rewards and mixed supervision strategy.

	Model	Inputs				Goal Progress (m)	
		t_0	A_i	Q_i	$A_{1:i-1}; Q_{1:i-1}$	Val Seen	Val Unseen
Baselines	Shortest-Path Agent					9.52	9.58
	Random Agent					0.42	1.09
	Seq2Seq [42]	✓				5.71	**1.29**
		✓	✓			6.04	2.05
		✓	✓	✓		6.16	1.83
		✓	✓	✓	✓	5.92	2.10
Ours	NDH-RCM (distance to goal location)	✓				4.18	0.42
		✓	✓			4.96	2.34
		✓	✓	✓		4.60	2.25
		✓	✓	✓	✓	5.02	2.58
	NDH-RCM (distance to goal room)	✓				**6.97**	1.25
		✓	✓			**6.92**	**2.69**
		✓	✓	✓		**6.47**	**2.69**
		✓	✓	✓	✓	**6.49**	**2.64**

parts of the dialog history. The results demonstrate that the agents rewarded for getting closer to the goal room consistently outperform the agents rewarded for getting closer to the exact goal location. This proves that using a reward function better aligned with the NDH task's objective yields better performance than other reward functions.

6 Conclusion

In this work, we presented an environment-agnostic multitask learning framework to learn generalized policies for agents tasked with natural language grounded navigation. We applied the framework to train agents that can simultaneously solve two popular and challenging tasks in the space: Vision-and-Language Navigation and Navigation from Dialog History. We showed that our approach effectively transfers knowledge across tasks and learns more generalized environment representations. As a result, the trained agents not only close down the performance gap between seen and unseen environments but also outperform the single-task baselines on both tasks by a significant margin. Furthermore, the studies show the two approaches of multitask learning and environment-agnostic learning independently benefit the agent learning and complement each other. There are possible future extensions to our work—*MT-RCM* can further be adapted to other language-grounded navigation datasets (e.g., Touchdown [7], TalkTheWalk [44], StreetLearn [31]); and complementary techniques like environmental dropout [40] can be combined with environment-agnostic learning to learn more general representations.

References

1. Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M., Levenberg, J., Monga, R., Moore, S., Murray, D.G., Steiner, B., Tucker, P., Vasudevan, V., Warden, P., Wicke, M., Yu, Y., Zheng, X.: Tensorflow: A system for large-scale machine learning. In: 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16). pp. 265–283 (2016), https://www.usenix.org/system/files/conference/osdi16/osdi16-abadi.pdf
2. Anderson, P., Chang, A., Chaplot, D.S., Dosovitskiy, A., Gupta, S., Koltun, V., Kosecka, J., Malik, J., Mottaghi, R., Savva, M., Zamir, A.R.: On evaluation of embodied navigation agents. arXiv (2018), `arXiv:1807.06757 [cs.AI]`
3. Anderson, P., Wu, Q., Teney, D., Bruce, J., Johnson, M., Sünderhauf, N., Reid, I., Gould, S., van den Hengel, A.: Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3674–3683 (2018)
4. Bain, M., Sammut, C.: A framework for behavioural cloning. In: Machine Intelligence 15, Intelligent Agents [St. Catherine's College, Oxford, July 1995]. pp. 103–129. Oxford University, Oxford, UK, UK (1999), http://dl.acm.org/citation.cfm?id=647636.733043
5. Caruana, R.: Multitask learning: A knowledge-based source of inductive bias. In: Proceedings of the Tenth International Conference on Machine Learning. pp. 41–48. Morgan Kaufmann (1993)
6. Chang, A., Dai, A., Funkhouser, T., Halber, M., Niessner, M., Savva, M., Song, S., Zeng, A., Zhang, Y.: Matterport3d: Learning from rgb-d data in indoor environments. International Conference on 3D Vision (3DV) (2017)
7. Chen, H., Suhr, A., Misra, D., Snavely, N., Artzi, Y.: Touchdown: Natural language navigation and spatial reasoning in visual street environments. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 12538–12547 (2019)
8. Collobert, R., Weston, J.: A unified architecture for natural language processing: Deep neural networks with multitask learning. In: Proceedings of the 25th International Conference on Machine Learning. pp. 160–167. ICML '08, ACM, New York, NY, USA (2008). https://doi.org/10.1145/1390156.1390177, http://doi.acm.org/10.1145/1390156.1390177
9. Das, A., Datta, S., Gkioxari, G., Lee, S., Parikh, D., Batra, D.: Embodied question answering. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 2054–2063 (2018)
10. Das, A., Kottur, S., Gupta, K., Singh, A., Yadav, D., Moura, J.M., Parikh, D., Batra, D.: Visual Dialog. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
11. Deng, L., Hinton, G., Kingsbury, B.: New types of deep neural network learning for speech recognition and related applications: an overview. In: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing. pp. 8599–8603 (May 2013). https://doi.org/10.1109/ICASSP.2013.6639344
12. Espeholt, L., Soyer, H., Munos, R., Simonyan, K., Mnih, V., Ward, T., Doron, Y., Firoiu, V., Harley, T., Dunning, I., Legg, S., Kavukcuoglu, K.: IMPALA: Scalable distributed deep-RL with importance weighted actor-learner architectures. In: Dy, J., Krause, A. (eds.) Proceedings of the 35th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 80,

pp. 1407–1416. PMLR, Stockholmsmässan, Stockholm Sweden (10–15 Jul 2018), http://proceedings.mlr.press/v80/espeholt18a.html
13. Finn, C., Abbeel, P., Levine, S.: Model-agnostic meta-learning for fast adaptation of deep networks. In: Proceedings of the 34th International Conference on Machine Learning-Volume 70. pp. 1126–1135. JMLR. org (2017)
14. Fried, D., Hu, R., Cirik, V., Rohrbach, A., Andreas, J., Morency, L.P., Berg-Kirkpatrick, T., Saenko, K., Klein, D., Darrell, T.: Speaker-follower models for vision-and-language navigation. In: Neural Information Processing Systems (NeurIPS) (2018)
15. Fu, T.J., Wang, X., Peterson, M., Grafton, S., Eckstein, M., Wang, W.Y.: Counterfactual vision-and-language navigation via adversarial path sampling. arXiv preprint arXiv:1911.07308 (2019)
16. Ganin, Y., Lempitsky, V.: Unsupervised domain adaptation by backpropagation. In: Proceedings of the 32Nd International Conference on International Conference on Machine Learning - Volume 37. pp. 1180–1189. ICML'15, JMLR.org (2015), http://dl.acm.org/citation.cfm?id=3045118.3045244
17. Girshick, R.: Fast r-cnn. In: 2015 IEEE International Conference on Computer Vision (ICCV). pp. 1440–1448 (Dec 2015). https://doi.org/10.1109/ICCV.2015.169
18. Hemachandra, S., Duvallet, F., Howard, T.M., Roy, N., Stentz, A., Walter, M.R.: Learning models for following natural language directions in unknown environments. In: 2015 IEEE International Conference on Robotics and Automation (ICRA). pp. 5608–5615. IEEE (2015)
19. Huang, H., Jain, V., Mehta, H., Baldridge, J., Ie, E.: Multi-modal discriminative model for vision-and-language navigation. In: Proceedings of the Combined Workshop on Spatial Language Understanding (SpLU) and Grounded Communication for Robotics (RoboNLP). pp. 40–49. Association for Computational Linguistics, Minneapolis, Minnesota (2019). https://doi.org/10.18653/v1/W19-1605, https://www.aclweb.org/anthology/W19-1605
20. Huang, H., Jain, V., Mehta, H., Ku, A., Magalhães, G., Baldridge, J., Ie, E.: Transferable representation learning in vision-and-language navigation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision178(ICCV) (2019)
21. Jain, V., Magalhães, G., Ku, A., Vaswani, A., Ie, E., Baldridge, J.: Stay on the path: Instruction fidelity in vision-and-language navigation. In: ACL (2019)
22. Ke, L., Li, X., Bisk, Y., Holtzman, A., Gan, Z., Liu, J., Gao, J., Choi, Y., Srinivasa, S.: Tactical rewind: Self-correction via backtracking in vision-and-language navigation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6741–6749 (2019)
23. Kolve, E., Mottaghi, R., Han, W., VanderBilt, E., Weihs, L., Herrasti, A., Gordon, D., Zhu, Y., Gupta, A., Farhadi, A.: AI2-THOR: An Interactive 3D Environment for Visual AI. arXiv (2017)
24. Lansing, L., Jain, V., Mehta, H., Huang, H., Ie, E.: Valan: Vision and language agent navigation. ArXiv **abs/1912.03241** (2019)
25. Li, J., Wang, X., Tang, S., Shi, H., Wu, F., Zhuang, Y., Wang, W.Y.: Unsupervised reinforcement learning of transferable meta-skills for embodied navigation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12123–12132 (2020)
26. Li, Y., Baldwin, T., Cohn, T.: What's in a Domain? Learning Domain-Robust Text Representations using Adversarial Training. In: NAACL-HLT (2018)
27. Liu, A.H., Liu, Y.C., Yeh, Y.Y., Wang, Y.C.F.: A unified feature disentangler for multi-domain image translation and manipulation. In: Advances in Neural Information Processing Systems. pp. 2590–2599 (2018)

28. Ma, C.Y., Lu, J., Wu, Z., AlRegib, G., Kira, Z., Socher, R., Xiong, C.: Self-monitoring navigation agent via auxiliary progress estimation. arXiv preprint arXiv:1901.03035 (2019)
29. Ma, C.Y., Wu, Z., AlRegib, G., Xiong, C., Kira, Z.: The regretful agent: Heuristic-aided navigation through progress estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6732–6740 (2019)
30. Manolis Savva*, Abhishek Kadian*, Oleksandr Maksymets*, Zhao, Y., Wijmans, E., Jain, B., Straub, J., Liu, J., Koltun, V., Malik, J., Parikh, D., Batra, D.: Habitat: A Platform for Embodied AI Research. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2019)
31. Mirowski, P., Grimes, M., Malinowski, M., Hermann, K.M., Anderson, K., Teplyashin, D., Simonyan, K., Kavukcuoglu, K., Zisserman, A., Hadsell, R.: Learning to navigate in cities without a map. In: Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R. (eds.) Advances in Neural Information Processing Systems 31, pp. 2419–2430. Curran Associates, Inc. (2018), http://papers.nips.cc/paper/7509-learning-to-navigate-in-cities-without-a-map.pdf
32. Mirowski, P.W., Pascanu, R., Viola, F., Soyer, H., Ballard, A., Banino, A., Denil, M., Goroshin, R., Sifre, L., Kavukcuoglu, K., Kumaran, D., Hadsell, R.: Learning to navigate in complex environments. ArXiv **abs/1611.03673** (2016)
33. Nguyen, K., Daumé III, H.: Help, anna! visual navigation with natural multimodal assistance via retrospective curiosity-encouraging imitation learning. arXiv preprint arXiv:1909.01871 (2019)
34. Nguyen, K., Dey, D., Brockett, C., Dolan, B.: Vision-based navigation with language-based assistance via imitation learning with indirect intervention. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 12527–12537 (2019)
35. Peng, X., Huang, Z., Sun, X., Saenko, K.: Domain agnostic learning with disentangled representations. In: Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA. pp. 5102–5112 (2019)
36. Qi, Y., Wu, Q., Anderson, P., Wang, X., Wang, W.Y., Shen, C., Hengel, A.v.d.: Reverie: Remote embodied visual referring expression in real indoor environments. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9982–9991 (2020)
37. Ramsundar, B., Kearnes, S.M., Riley, P., Webster, D., Konerding, D.E., Pande, V.S.: Massively multitask networks for drug discovery. ArXiv **abs/1502.02072** (2015)
38. Romijnders, R., Meletis, P., Dubbelman, G.: A domain agnostic normalization layer for unsupervised adversarial domain adaptation. In: 2019 IEEE Winter Conference on Applications of Computer Vision (WACV). pp. 1866–1875. IEEE (2019)
39. Schuster, M., Paliwal, K.K.: Bidirectional recurrent neural networks. IEEE Trans. Signal Processing **45**, 2673–2681 (1997)
40. Tan, H., Yu, L., Bansal, M.: 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). pp. 2610–2621. Association for Computational Linguistics, Minneapolis, Minnesota (Jun 2019). https://doi.org/10.18653/v1/N19-1268

41. Teh, Y., Bapst, V., Czarnecki, W.M., Quan, J., Kirkpatrick, J., Hadsell, R., Heess, N., Pascanu, R.: Distral: Robust multitask reinforcement learning. In: Advances in Neural Information Processing Systems. pp. 4496–4506 (2017)
42. Thomason, J., Murray, M., Cakmak, M., Zettlemoyer, L.: Vision-and-dialog navigation. In: Conference on Robot Learning (CoRL) (2019)
43. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: Advances in neural information processing systems. pp. 5998–6008 (2017)
44. de Vries, H., Shuster, K., Batra, D., Parikh, D., Weston, J., Kiela, D.: Talk the walk: Navigating new york city through grounded dialogue. arXiv preprint arXiv:1807.03367 (2018)
45. Wang, X., Huang, Q., Celikyilmaz, A., Gao, J., Shen, D., Wang, Y.F., Wang, W.Y., Zhang, L.: Reinforced cross-modal matching and self-supervised imitation learning for vision-language navigation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6629–6638 (2019)
46. Wang, X., Xiong, W., Wang, H., Yang Wang, W.: Look before you leap: Bridging model-free and model-based reinforcement learning for planned-ahead vision-and-language navigation. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 37–53 (2018)
47. Williams, R.J.: Simple statistical gradient-following algorithms for connectionist reinforcement learning. Machine Learning **8**(3), 229–256 (May 1992). https://doi.org/10.1007/BF00992696, https://doi.org/10.1007/BF00992696
48. Wu, Y., Wu, Y., Gkioxari, G., Tian, Y.: Building generalizable agents with a realistic and rich 3d environment. arXiv preprint arXiv:1801.02209 (2018)
49. Wu, Y., Wu, Y., Tamar, A., Russell, S., Gkioxari, G., Tian, Y.: Learning and planning with a semantic model. arXiv preprint arXiv:1809.10842 (2018)
50. Xia, F., R. Zamir, A., He, Z.Y., Sax, A., Malik, J., Savarese, S.: Gibson env: real-world perception for embodied agents. In: Computer Vision and Pattern Recognition (CVPR), 2018 IEEE Conference on. IEEE (2018)