

VISITRON: Visual Semantics-Aligned Interactively Trained Object-Navigator

Ayush Shrivastava^{1*}, Karthik Gopalakrishnan², Yang Liu², Robinson Piramuthu²,
Gokhan Tür², Devi Parikh¹, Dilek Hakkani-Tür²

¹Georgia Tech, ²Amazon Alexa AI

{ayshrv, parikh}@gatech.edu

{karthgop, yangliud, robinpir, gokhatur, hakkanit}@amazon.com

Abstract

Interactive robots navigating photo-realistic environments need to be trained to effectively leverage and handle the dynamic nature of dialogue in addition to the challenges underlying vision-and-language navigation (VLN). In this paper, we present VISITRON, a multi-modal Transformer-based navigator better suited to the interactive regime inherent to Cooperative Vision-and-Dialog Navigation (CVDN). VISITRON is trained to: i) identify and associate object-level concepts and semantics between the environment and dialogue history, ii) identify when to interact vs. navigate via imitation learning of a binary classification head. We perform extensive pre-training and fine-tuning ablations with VISITRON to gain empirical insights and improve performance on CVDN. VISITRON’s ability to identify when to interact leads to a natural generalization of the gameplay mode introduced by Roman et al. (2020) for enabling the use of such models in different environments. VISITRON is competitive with models on the static CVDN leaderboard and attains state-of-the-art performance on the Success weighted by Path Length (SPL) metric.

1 Introduction

Large pre-trained Transformer-based language models (Vaswani et al., 2017) are ubiquitous in natural language processing (NLP) and have performed very well in interactive settings such as open-domain (Gopalakrishnan et al., 2019; Huang et al., 2020) and task-oriented dialogue (Kim et al., 2020). The success of Transformers and the pre-train/fine-tune paradigm in NLP has also inspired their adoption in vision-and-language research, with cross-modal representations being learned (Li et al., 2020) and utilized towards tasks like image and object captioning, visual question answering, visual commonsense reasoning and visual dialogue.

* Work done as an intern at Amazon Alexa AI. Code available at: www.github.com/alexa/visitron

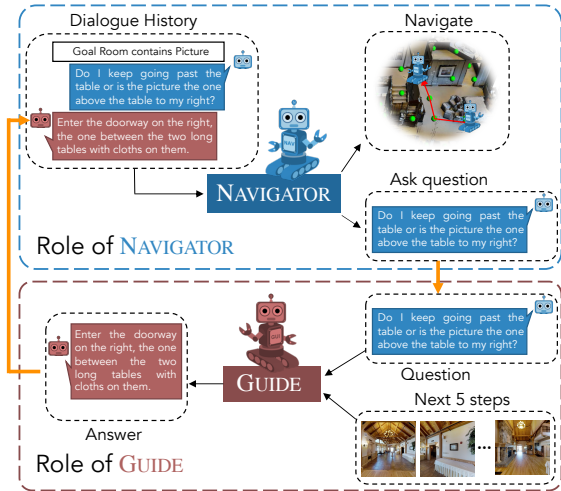


Figure 1: Cooperative Vision-and-Dialog Navigation (CVDN) with Dynamic Question-Asking

Vision-and-language navigation (VLN) is a challenging cross-modal research task in which agents need to learn to navigate in response to natural language instructions in simulated photo-realistic environments. VLN has been studied extensively with the advent of the Room-to-Room (R2R) dataset (Anderson et al., 2018b) and there has been growing interest recently in pushing the pre-train/fine-tune paradigm towards VLN, with work on leveraging disembodied corpora (Majumdar et al., 2020) to learn cross-modal pre-trained representations that can improve embodied VLN performance. As depicted in Figure 1, the Cooperative Vision-and-Dialog Navigation (CVDN) dataset (Thomason et al., 2020) allows for dialogue with a guide during navigation: a navigator can ask natural language questions to a guide when it needs assistance and the guide responds in natural language by using privileged knowledge of the environment accessible only to it, thus expanding beyond the traditional VLN task towards deployable interactive agents that are more robust and generalizable. But preliminary navigator modeling using CVDN is still VLN-style via the Navigation from Dialog History (NDH) task, treating the dia-

logue history as a static instruction.

In this paper, we present work on training VISITRON, a multi-modal Transformer-based navigator with a focus on tackling challenges unique to CVDN: i) moving beyond rote memorization to associative learning in order to learn to identify and acquire visio-linguistic concepts and semantics while interacting in new environments, and ii) learning when to ask questions (Chi et al., 2020). VISITRON builds off the recent cross-modal object-semantics aligned pre-training (OSCAR) strategy and uses object-tags as explicit anchor points during training to learn to associate the environment’s visual semantics with the textual dialogue history, thus allowing for interaction/experience-grounded (Bisk et al., 2020) visio-linguistic concepts and semantics identification and acquisition. VISITRON is trained in a data-driven fashion to identify when to engage in dialogue, i.e., ask questions, vs. when to navigate, thus providing the first known empirical baselines for this task. We also present empirical results from various first-principles modeling ablations performed with VISITRON. We demonstrate that for CVDN, panoramic viewpoint selection is a better formulation than discrete turn-based action prediction, akin to what has been seen on VLN with R2R (Fried et al., 2018). We observe that multi-task learning with long-trajectory VLN datasets leads to significant CVDN performance gains relative to training on CVDN alone. VISITRON is competitive with models on the leaderboard for the *static* NDH task on EvalAI (Yadav et al., 2019), attaining state-of-the-art performance on the Success weighted by Path Length (SPL) metric. Given VISITRON’s design and ability to identify when to engage in dialogue, we also propose a generalization of the game-play mode introduced by Roman et al. (2020) for jointly fine-tuning and evaluating VISITRON and future such models with pre-trained guides to help them easily adapt to their guides’ capabilities.

2 Background

2.1 Vision-and-Language Navigation

The Vision-and-Language Navigation (VLN) task requires an agent spawned in an indoor environment at a starting position s_0 to follow natural language instructions x and navigate to a target position s_{goal} . This can also be seen as a Partially Observable Markov Decision Process $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, P_s, r \rangle$ where \mathcal{S} is the visual state space, \mathcal{A}

is the discrete action space, P_s is the unknown environment distribution from which the next state is drawn and $r \in \mathbb{R}$ is the reward function (Hao et al., 2020). At a given time step t , the agent receives an RGB image observation $obs(s_t)$, where $s_t \in \mathcal{S}$. Based on the observation, the agent takes an action $a_t \in \mathcal{A}$, transitions into the next state s_{t+1} drawn as follows: $s_{t+1} \sim P_s(\cdot|s_t, a_t)$, and receives a new image observation $obs(s_{t+1})$. To end the episode, the agent must select the special STOP action. A T -step trajectory can be represented as $\tau = [s_0, a_0, s_1, a_1, \dots, s_T, a_T]$. The episode is considered successful if the agent stops within ϵ distance of the goal, i.e., $|s_T - s_{goal}| \leq \epsilon$. Using a training dataset $\mathcal{D} = \{(\tau, x)\}$ consisting of *expert* trajectory τ and instructions x pairs, the goal is to train a policy $\pi_\theta(\tau|x)$ with θ parameters that maximizes the log-likelihood of the target trajectory given instructions x :

$$\begin{aligned} \max_{(\tau, x) \sim \mathcal{D}} \mathcal{L}_\theta(\tau, x) &= \log \pi_\theta(\tau|x) \\ &= \sum_{t=0}^T \log \pi_\theta(a_t|s_t, x) \end{aligned} \quad (1)$$

Several datasets have been released for VLN based on Matterport3D (Chang et al., 2017), a large-scale RGB-D dataset containing ~ 10000 panoramic views from ~ 194000 RGB-D images of 90 building-scale scenes. The most popular VLN dataset based on Matterport3D is the Room-to-Room (R2R) dataset (Anderson et al., 2018b), containing ~ 7200 trajectories and 3 natural language instructions per trajectory. For validation and test sets, *seen* and *unseen* splits are created to easily evaluate how well an agent generalizes. Room-4-Room (R4R) (Jain et al., 2019) is an augmentation of R2R wherein existing short trajectories in R2R are joined to form longer, challenging trajectories. Room-across-Room (RxR) (Ku et al., 2020) is a newly introduced dataset with several properties, including but not limited to multilingual instructions, larger scale (for each language, ~ 14000 trajectories with 3 instructions per trajectory), fine-grained spatio-temporal grounding and follower demonstrations.

A navigating agent’s actions typically belong in a pre-defined discrete set comprising options such as FORWARD, LEFT, RIGHT, etc. Predicting the next best action from this low-level visuomotor space (Fried et al., 2018) of actions is referred to

as *turn-based action prediction*. Given the nature of the aforementioned VLN datasets, it is also possible to have a navigating agent’s actions belong in the panoramic space, wherein the agent selects the next best viewpoint in the navigation graph from the panoramic space visible to it at its current location. This is referred to as *viewpoint selection*.

2.2 Cooperative Vision-and-Dialog Navigation

Cooperative Vision-and-Dialog Navigation (CVDN) is a recently introduced dataset (Thomason et al., 2020) collected by partnering crowd-workers in simulated photo-realistic environments. One worker acts as a NAVIGATOR, seeking to navigate to a goal and interacting in natural language with a GUIDE along the way if it needs assistance. The other worker acts as a GUIDE, answering the NAVIGATOR’s questions while having privileged access to the best next steps the NAVIGATOR should take according to an ORACLE full-state shortest path planner. The collection of each CVDN instance begins with the state $(\mathcal{S}, T_O, s_0, G)$, where \mathcal{S} is the environment in which the agents are placed, s_0 is the start location of the NAVIGATOR, G is the goal region and T_O is the initial hint given to both agents about the goal region containing object O . At any time step t , the NAVIGATOR can make one of three choices: i) take a sequence of k_t navigation steps $N_t = [n_t^1, n_t^2, \dots, n_t^{k_t}]$, ii) ask a question Q_t to the GUIDE, iii) declare its current position as the goal region. If a question is asked, the GUIDE looks at l next steps along the shortest path to the goal and replies with an answer A_t . The instance ends when the NAVIGATOR reaches G . Thus, a CVDN instance comprises $[(\mathcal{S}, T_O, s_0, G), \langle N_0, Q_1, A_1, N_1, Q_2, A_2, N_2, \dots, Q_m, A_m, N_m \rangle]$, where m is the number of dialogue exchanges between the NAVIGATOR and GUIDE, and N_0 is the sequence of navigation steps before the 1st exchange.

2.2.1 Navigation from Dialog History (NDH)

With the CVDN dataset, the NDH task for the NAVIGATOR was introduced (Thomason et al., 2020), in which the NAVIGATOR needs to navigate towards a goal given a dialogue history. Specifically, the NAVIGATOR is spawned at the *terminal* position of N_{t-1} (or s_0 in the case of N_0) in environment \mathcal{S} and is given $(T_O, Q_{1:t}, A_{1:t})$. The task is to predict the navigation steps that bring the agent closer to the goal region G . To train a NAVIGATOR *agent*

for this task, the navigation steps needed for supervision from the dataset can be provided in any of the three forms: i) *human* NAVIGATOR steps, N_t : the navigation steps that were taken by the *human* NAVIGATOR after the dialogue exchange at time step t , ii) ORACLE steps, O_t : the shortest path steps accessible to the GUIDE when it gave the answer A_t , iii) MIXED: a mix of both *human* NAVIGATOR and ORACLE supervision where the supervision path is N_t when $e(O_t) \in N_t$, and O_t otherwise, where $e(\cdot)$ represents the *terminal* position of a sequence of navigation steps. The *agent* NAVIGATOR is trained VLN-style using Equation 1 on NDH instances extracted as described above from the CVDN instances, and evaluated on NDH instances using VLN metrics such as Goal Progress and Success weighted by Path Length (SPL), defined in Section 4.1. In the CVDN literature, it has been observed that MIXED supervision typically performs the best, followed by ORACLE and *human* NAVIGATOR supervision respectively. However, for the purposes of all our experiments, we pick the *human* NAVIGATOR supervision mode to establish a lower-bound on performance for VISITRON.

2.2.2 Gameplay Mode

In the CVDN dataset, a *human* NAVIGATOR cooperates with a *human* GUIDE to find a goal region G with target object O . Roman et al. (2020) introduced the game-play mode, which is essentially an *agent-agent* replica of this dynamic dataset creation process wherein the two trained agents consume each other’s outputs. This mode can be applied during both fine-tuning and evaluation and helps understand how well a pre-trained NAVIGATOR agent adapts to the capabilities of different GUIDE agents in a dynamic/interactive setting. For the sake of consistency with game-play mode notation introduced by Roman et al. (2020), we denote the role of asking questions that is intrinsic to the NAVIGATOR by QUESTIONER. Thus, in a game-play mode episode, at $t = 0$ (prior to the first QA exchange), the NAVIGATOR takes N_0 steps given the initial hint T_O . For time steps $t > 0$, the QUESTIONER generates a question Q_t , GUIDE generates an answer A_t having access to the next l steps in the shortest path, and then NAVIGATOR generates N_t navigation steps of length k_t . All agents have access to the entire visual navigation ($N_{0:t-1}$) and dialogue ($Q_{1:t-1}A_{1:t-1}$) histories in addition to the initial hint T_O . The QUESTIONER asks questions every 4th time-step, which is a hard-coded heuristic

by Roman et al. (2020) since their NAVIGATOR does not know *when* to ask questions. The episode ends when the NAVIGATOR declares that the current position is in the goal region G or a maximum number of turns (20) are played. NAVIGATOR’s performance in game-play mode is measured using Goal Progress (see Section 4.1). While the focus of our work is not to train a QUESTIONER, we ensure our NAVIGATOR is equipped with the ability to identify *when* to ask questions. This leads to our proposed **general game-play mode**, wherein the aforementioned description of a regular game-play mode episode still holds but the hard-coded heuristic of asking questions every 4th time-step is eliminated, i.e., the NAVIGATOR decides when a question must be asked to continue game-play.

2.3 OSCAR

The OSCAR pre-training strategy (Li et al., 2020) for cross-modal Transformers uses object tags detected in images as *anchor points* to ease the learning of semantic alignments between images and text. The input is represented as Word-Tag-Image $(\mathbf{w}, \mathbf{q}, \mathbf{v})$, where \mathbf{w} and \mathbf{q} are the sequence of word embeddings of the text and object tags respectively, and \mathbf{v} is the sequence of region features of the image. To generate \mathbf{v} , Faster R-CNN (Ren et al., 2015) is used to extract visual semantics of each region as (v', z) where $v' \in \mathbb{R}^P$ ($P = 2048$) is the region feature, $z \in \mathbb{R}^6$ is the region position represented by the coordinates of the top-right and bottom-left corners and the height & width. v' and z are concatenated to form a position-sensitive region feature, which is further transformed into v using a projection layer such that v has the same dimension as the input token embeddings. It is then pre-trained with a Masked Token Loss (MTL) and a Contrastive Loss (CL).

$$\begin{aligned} \mathcal{L}_{Pre-training} &= \mathcal{L}_{MTL} + \mathcal{L}_{CL} \\ &= -\mathbb{E}_{(v, h) \sim \mathcal{D}} \log p(h_i | \mathbf{h}_{\setminus i}, \mathbf{v}) \\ &\quad - \mathbb{E}_{(h', w) \sim \mathcal{D}} \log p(y | f(\mathbf{h}', \mathbf{w})) \end{aligned}$$

The MTL is akin to that in BERT (Devlin et al., 2019), masking the input tokens (\mathbf{w}, \mathbf{q}) with a probability of 15% and predicting them. The CL is computed by polluting the object tags \mathbf{q} with a probability of 50% with randomly chosen object tags from the dataset, and a feed-forward layer on top of [CLS] predicts whether the input contains the original image representation or a pol-

luted one. In the previous equation, $\mathbf{h} = [\mathbf{w}, \mathbf{q}]$, $\mathbf{h}' = [\mathbf{q}, \mathbf{v}]$, $\mathbf{h}_{\setminus i}$ are the surrounding tokens of masked token h_i , $f(\cdot)$ denotes the binary classifier where $y = 0$ if the object tags are polluted and 1 otherwise, and \mathcal{D} is the dataset. OSCAR uses a collection of popular image-text datasets for pre-training, including but not limited to Conceptual Captions (Sharma et al., 2018), MS-COCO (Lin et al., 2014), Flickr30K (Young et al., 2014) and GQA (Hudson and Manning, 2019). Such datasets typically have images of objects taken from perfect angles whereas a navigating agent will see objects from different vantage points, which also motivates augmenting OSCAR and performing an additional phase of navigation-specific pre-training.

3 Approach

The policy for NDH (and VLN) can be decomposed into an encoder-decoder setup, $\pi_\theta = f_{\theta_E} \circ f_{\theta_D}$:

- A vision-language encoder $f_{\theta_E} : \{s_{1:t}, \mathbf{x}\} \rightarrow z_t$, where $s_{1:t}$ are visual states, \mathbf{x} is the dialogue history (or instructions for VLN) and z_t is the joint latent representation at time step t .
- An action decoder $f_{\theta_D} : \{s_t, z_t, \mathbf{a}_{t-1}\} \rightarrow \mathbf{a}_t$, where \mathbf{a}_t is the next action.

We model π_θ by VISITRON, a visio-linguistic Transformer-based model. VISITRON’s encoder is structurally similar to OSCAR’s Transformer (Li et al., 2020). This is by design to enable easy transfer of visual semantics-aligned representations learned from disembodied image-text data. We make navigation-specific modifications to OSCAR, but they are all structured as augmentations of modules instead of removal of network components, thus enabling us to use the pre-trained weights of OSCAR’s Transformer to initialize large portions of our encoder. The augmentations are described in Section 3.1. As with OSCAR, the input to VISITRON’s encoder is represented as Word-Tag-Image $(\mathbf{w}, \mathbf{q}, \mathbf{v})$, where \mathbf{w} and \mathbf{q} are the sequence of word embeddings of the text and object tags respectively, and \mathbf{v} is the sequence of region features of the image. We represent the panorama in 36 views, extract Faster R-CNN (Ren et al., 2015) region features r' from each view and add positional vector p , $r = (r', p)$. To incorporate 3D direction, we add direction embedding d to the region features, $v = r + d$. d is a 128-dimensional orientation vector represented by repeating $[\sin \phi; \cos \phi; \sin \omega; \cos \omega]$

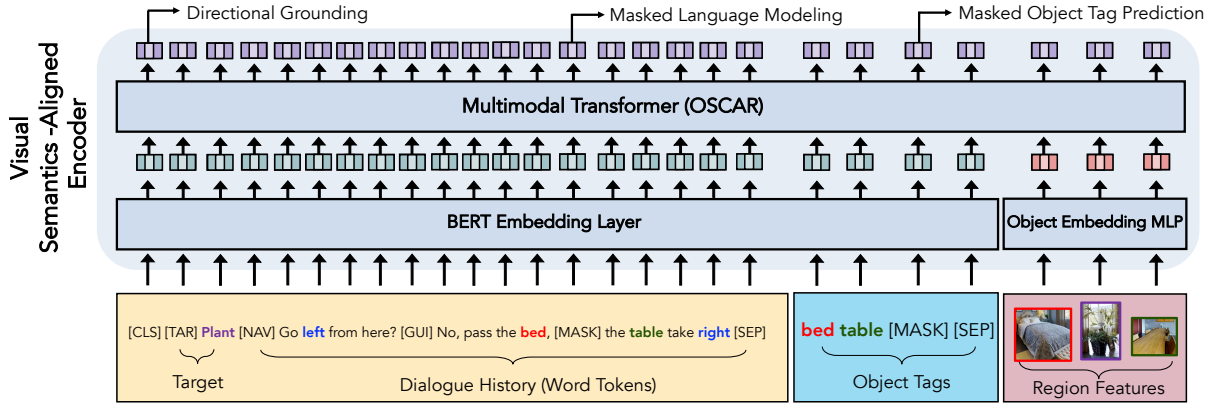


Figure 2: VISITRON’s Encoder Architecture and Semantics-Aligned Navigation Pre-Training Tasks

32 times where ϕ and ω are heading and elevation poses. In addition to the standard [CLS] and [SEP], we also use [TAR], [NAV], [GUI] as delimiter tokens for the initial target hint, NAVIGATOR’s questions and the GUIDE’s answers respectively. While this input structure is dialogue-specific, it is amenable to instructions-based datasets for multi-tasking.

3.1 VISITRON Pre-Training

We adopt a two-stage pre-training strategy, initializing VISITRON’s encoder with weights from OSCAR to begin with web-scale disembodied visiolinguistic representations, followed by facilitating a domain shift to navigation and actions by pre-training on navigation data. For each navigation trajectory, we extract (w, q, v, a) tuples where w is the dialogue history/instruction, q is the sequence of object tags from the current panorama, v is the sequence of region features and a is the direction in the 360° panoramic space where the next node in the trajectory is located (Fried et al., 2018). The pre-training objectives are:

1. **Masked Language Modeling:** Input word tokens are replaced with [MASK] with 15% probability and the masked token x_i is predicted conditioned on surrounding tokens $x_{\setminus i}$.
2. **Masked Object Tag Prediction:** Object tags are replaced with [MASK] with 15% probability. A feed-forward head on top of [MASK] is used to predict the tag from a distribution over Faster R-CNN semantic classes. This provides more fine-grained object supervision unlike OSCAR’s global masked token loss for tokens in both object tags and text, since this computes a distribution over the object detector’s semantic classes instead of over the

entire input vocabulary.

3. **Directional Grounding:** [CLS] hidden state goes into a feed-forward head to predict a .

Figure 2 illustrates VISITRON’s encoder architecture and the pre-training objectives we use, with an extracted tuple from a sample NDH instance.

3.2 VISITRON Fine-Tuning

After pre-training the encoder, we leverage it with an attention-based Long Short-Term Memory (LSTM) action decoder (Hochreiter and Schmidhuber, 1997), as shown in Figure 3. At time-step t , the decoder (cell state d_t) takes the previous action a_{t-1} , the panoramic ResNet features extracted from the current location/state and decodes the next action a_t , while attending to the VISITRON encoder’s cross-modal representation of its input. After this LSTM is fine-tuned, the same stack is frozen and a randomly initialized two-layer feed-forward head is added and trained with a binary cross-entropy loss to learn to classify when to ask a question. The supervision for this head comes from the elongated CVDN instances defined in Section 2.2, with time-steps when a question was asked serving as positive labels and the remaining time-steps during which navigation occurs serving as negative labels. Note that as described in Section 2.1, the decoder’s actions can belong in either the panoramic space or the low-level visuo-motor space (Fried et al., 2018), leading to independent formulations for *viewpoint selection* and *turn-based action prediction*.

4 Experiments

In this section, we first describe the evaluation metrics we adopt. We then describe and discuss

Table 1: Pre-Training Ablations (Fine-Tuning and Evaluating on NDH)

| Semantics-aligned Pre-Training Curriculum | | | | | | Val Seen | | | | Val Unseen | | | |
|-------------------------------------------|------------------------|-------------|---------------------|------------------------------|-----------------------|-------------|--------------|--------------|--------------|-------------|--------------|--------------|--------------|
| Stage 1: Web (OSCAR) | | | Stage 2: Navigation | | | | | | | | | | |
| # | Contrastive+ Masked LM | Object Tags | Masked LM | Masked Object Tag Prediction | Directional Grounding | GP (m) ↑ | SPL (%) ↑ | SR (%) ↑ | nDTW (%) ↑ | GP (m) ↑ | SPL (%) ↑ | SR (%) ↑ | nDTW (%) ↑ |
| 1 | | | | | | 4.76 | 36.56 | 46.07 | 30.97 | 2.09 | 9.96 | 22.49 | 6.50 |
| 2 | ✓ | | | | | 4.82 | 50.73 | 58.11 | 47.34 | 2.67 | 24.88 | 34.29 | 24.21 |
| 3 | ✓ | ✓ | | | | 4.38 | 45.15 | 52.09 | 41.14 | 2.30 | 13.03 | 24.81 | 8.63 |
| 4 | ✓ | ✓ | ✓ | | | 5.09 | 25.92 | 41.10 | 17.91 | 1.90 | 11.27 | 23.48 | 5.62 |
| 5 | ✓ | ✓ | ✓ | ✓ | | 4.83 | 48.22 | 56.02 | 47.01 | 2.70 | 24.04 | 32.86 | 23.46 |
| 6 | ✓ | ✓ | ✓ | ✓ | ✓ | 5.34 | 55.16 | 61.78 | 54.83 | 2.71 | 24.56 | 32.52 | 24.51 |

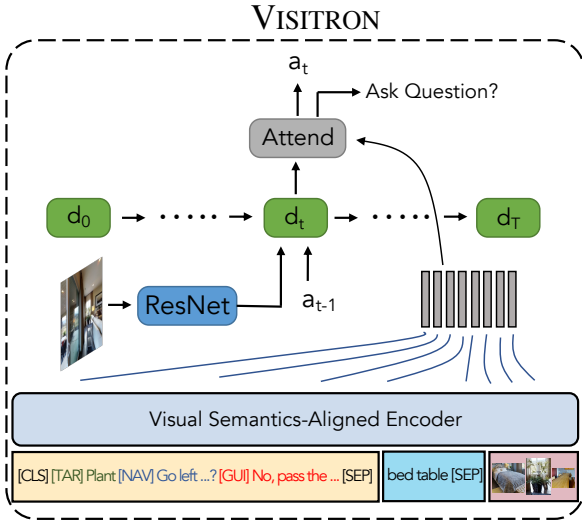


Figure 3: NAVIGATOR predicts navigation actions, given dialogue history and visual observations. The same stack decides when to ask the GUIDE a question. A similar setup can be used for question generation.

our experimental observations from performing ablations during VISITRON pre-training and fine-tuning respectively. We present our observations for question-asking classification for CVDN, establishing a strong baseline for future models. We finally present and discuss our observations from submitting our model checkpoints to the *static* EvalAI leaderboard for CVDN.

4.1 Evaluation Metrics

We evaluate VISITRON’s ability to navigate to the goal with the following metrics:

- *Goal Progress (GP)* measures the difference between the distance from the start position to the final goal and the distance from the end position to the final goal. It is used to determine how much progress in meters the agent has made towards the final goal.
- *Success weighted by (Normalized Inverse) Path Length (SPL)* introduced by Anderson

et al. (2018a) provides a measure of success normalized by the ratio between the length of the shortest path and the selected path.

- *Success Rate (SR)* measures the success of an episode. If the agent stops within 3 meters of the goal, it is considered a success.
- *Normalized Dynamic Time Warping (nDTW)* introduced by Ilharco et al. (2019) helps measure a navigator agent’s fidelity to the dialogue history/instruction by softly penalizing deviations from the reference path.

We evaluate the question-asking classification head by computing *accuracy* and *balanced accuracy* (Brodersen et al., 2010). The latter accounts for the natural class imbalance of more navigation time-steps than question-asking time-steps expected in dialogue-based navigation by computing the average of recall obtained on each class.

4.2 Pre-Training Ablations

Using NDH and R2R trajectories, we pre-train VISITRON as described in Section 3.1. We begin experimenting with cumulative addition of each pre-training stage and objective to obtain an ablative understanding of their effect on the downstream NDH task. Results are shown in Table 1. We see that **our pre-training strategy helps**: the best performance on Val Seen (as measured by all metrics) is obtained when using all pre-training stages and objectives. We also see that Goal Progress (GP) is highest on Val Unseen in this setting (an absolute increase of 0.62 relative to no pre-training). Rows 3-4 demonstrate the efficacy of our second-stage masked language modeling (MLM) task, helping improve Val Seen GP from 4.38 to 5.09. Rows 4-5 demonstrate the efficacy of our newly introduced masked object tag prediction task as a means towards experience-driven concepts and semantics

Table 2: Fine-Tuning Ablations

| # | Action Space | Multi-Task Fine-Tuning NDH+ | Val Seen | | | | Val Unseen | | | |
|----------|---------------------|-----------------------------|----------|-----------|----------|------------|------------|-----------|----------|------------|
| | | | GP (m) ↑ | SPL (%) ↑ | SR (%) ↑ | nDTW (%) ↑ | GP (m) ↑ | SPL (%) ↑ | SR (%) ↑ | nDTW (%) ↑ |
| VISITRON | 1 Turn-based | ✗ | 1.15 | 9.66 | 11.78 | 26.86 | 1.60 | 13.02 | 14.77 | 29.28 |
| | 2 Action Prediction | ✓(RxR) | 1.50 | 12.30 | 15.18 | 19.95 | 0.97 | 11.52 | 15.44 | 20.49 |
| | 3 Viewpoint | ✗ | 5.34 | 55.16 | 61.78 | 54.83 | 2.71 | 24.56 | 32.52 | 24.51 |
| | 4 Selection | ✓(RxR) | 5.11 | 12.33 | 25.65 | 4.66 | 3.25 | 10.74 | 27.34 | 3.78 |

identification and acquisition, with significant increases in all metrics across both validation seen and unseen splits. Rows 5-6 show that our directional grounding task for pre-training the encoder plays a particularly important role: the increase in both GP and nDTW suggest that this task improves VISITRON’s ability to navigate closer to the goal while ensuring that dialogue fidelity is maintained in the process by aligning encoder representations in the direction along the reference path.

4.3 Fine-Tuning Ablations

Next, we perform ablations during fine-tuning, leveraging all objectives from Table 1 since our previous analysis demonstrated their effectiveness. For VLN agents, it has been shown that viewpoint selection in the panoramic space is a better formulation than turn-based action prediction in the low-level visuomotor space (Fried et al., 2018). However, it is not immediately obvious or known whether this can be extrapolated to dialogue-based navigation as in CVDN. So we experiment with both formulations for our NAVIGATOR. Given the sparsity of NDH instances ($\sim 4k$) for fine-tuning, we also study if multi-task fine-tuning with the RxR dataset helps boost performance. Table 2 presents the fine-tuning ablation results. Row 1 and 3 demonstrate that **panoramic viewpoint selection is a better formulation than turn-based action prediction for CVDN**, with all metrics increasing significantly when switching to viewpoint selection. Further, we see in rows 3 and 4 that **multi-task fine-tuning leads to better CVDN generalization**, with Val Unseen GP increasing from 2.71 to 3.25 when multi-tasking with viewpoint selection. However, we see this increase in GP occurs alongside a decrease in nDTW, SPL and SR. This decrease can be attributed to the fact that the RxR dataset has very long trajectories, which prime the model to take long paths to the final CVDN goal (which GP cares about), well-beyond the next 5 GUIDE steps in the NDH instance that nDTW, SPL and SR

evaluate against.

4.4 Question-Asking Classification and Leaderboard Evaluation

We pick the VISITRON model checkpoint with the highest GP in Table 2 (row 4), and perform imitation learning of the question-asking classification head as described in Section 3.2. We evaluate the classification head by creating elongated CVDN instances from the validation sets as described in Section 2.2, akin to how supervision was provided during training: time-steps when a question was asked serve as positive instances and the remaining time-steps during which navigation occurs serve as negative instances. As seen in Table 3, our approach to identifying when to ask questions vs. when to navigate establishes a strong baseline for future work on identifying when to ask questions with CVDN, as measured by accuracy and balanced accuracy on Val Unseen. It is important to note that our design choice of adding and training a separate head for this task while keeping the navigator stack frozen ensures that there is no direct impact on navigation performance itself. This is unlike approaches that perform direct navigation action space augmentation with a special action for question-asking, where navigation actions themselves are affected by the presence of an additional competing variable for shared total probability mass.

Table 3: Question-Asking Classification Performance

| Metric (%) | Val Seen | Val Unseen |
|-------------------|----------|------------|
| Accuracy | 68.05 | 67.87 |
| Balanced Accuracy | 63.33 | 61.09 |

We submitted this model checkpoint to the CVDN leaderboard aimed at the *static* NDH task. We observe in Table 4 that this model checkpoint’s performance is competitive with state-of-the-art models with a hidden test GP of 3.11. However, the low hidden test SPL of 12 indicates the impact

that multi-task fine-tuning with long RxR paths had on this checkpoint’s ability to take short paths to the goal, like we discussed earlier in Section 4.3. Given this expected decrease in SPL when utilizing such long trajectories, we also created a model checkpoint by multi-task fine-tuning VISITRON on NDH, R2R and R4R. We observe that this model checkpoint obtains state-of-the-art SPL of 25 alongside an associated decrease in GP to 2.40.

Table 4: NDH Hidden Test Set Performance

| # | Method | GP (m) \uparrow | SPL (%) \uparrow |
|---|------------------------------------------------|-------------------|--------------------|
| 1 | MT-RCM + EnvAg (Wang et al., 2020) | 3.91 | 17 |
| 2 | BabyWalk (Zhu et al., 2020b) | 3.65 | 11 |
| 3 | VISITRON | 3.11 | 12 |
| 4 | Cross-modal Memory Network (Zhu et al., 2020c) | 2.95 | 14 |
| 5 | PREVALENT (Hao et al., 2020) | 2.44 | 24 |
| 6 | VISITRON (Best SPL) | 2.40 | 25 |

5 Related Work

Vision-and-language pre-training (Tan et al., 2019; Lu et al., 2019; Sun et al., 2019; Chen et al., 2020; Zhou et al., 2020) has grown to become a popular area of research, primarily aimed at solving downstream tasks such as image captioning, visual question answering and image retrieval. This line of work typically involves learning cross-modal representations using self-supervised objectives with a co-attention Transformer that fuses the two modalities represented by input token embeddings and visual region features, where the latter is typically sourced from Faster R-CNN (Ren et al., 2015).

Research in vision-and-language navigation (VLN) has also seen tremendous progress (Fried et al., 2018; Ke et al., 2019; Anderson et al., 2019; Tan et al., 2019; Zhu et al., 2020a) since the advent of the Room-to-Room (R2R) dataset (Anderson et al., 2018b) based on Matterport3D (Chang et al., 2017), with scope for further advances only increasing with the recent release of the much larger, densely annotated and multilingual Room-across-Room (RxR) dataset (Ku et al., 2020). As an extension to VLN, the recent Cooperative Vision-and-Dialog Navigation (CVDN) dataset (Thomason et al., 2020) allows for training interactive navigator and guide agents. The dominant focus of research with CVDN so far has been the Navigation from Dialog History (NDH) task introduced with CVDN, which is equivalent to treating the dialogue history as a VLN-style fixed instruction. The NDH formulation allows for easy transfer and multi-task learning (Hao et al., 2020; Wang et al., 2020; Zhang

et al., 2020) with VLN. However, state-of-the-art VLN models such as VLN-BERT (Majumdar et al., 2020) rely on the fully-observable setting when framing the task as *ahead-of-time* path selection, which is fundamentally at odds with the need for dialogue in CVDN: dialogue is aimed at enabling the navigating agent to succeed *while* it makes navigation decisions and decides it needs assistance. The recent Recursive Mental Model (RMM) (Roman et al., 2020) for CVDN attempts to address this by introducing a simulated dialogue game-play mode, where a trained navigator is fine-tuned jointly with a pre-trained guide and evaluated in this mode. However, the RMM navigator does not dynamically ask questions, instead relying on a data-driven heuristic of asking questions after every 4th navigation time-step. VISITRON’s design naturally leads to a generalization of this game-play mode which eliminates the aforementioned heuristic.

Our work is similar to recent work (Hao et al., 2020) on leveraging pre-trained cross-modal representations for the NDH task. However, our work takes on added goals of learning when to ask questions and associative learning of visio-linguistic concepts and semantics to ensure they can be identified and acquired when interacting in new environments, which are key requirements for full cooperative vision-and-dialogue navigation.

6 Conclusion and Future Work

We presented VISITRON, a Transformer-based navigator designed to identify and acquire visio-linguistic concepts and semantics and make decisions, all key traits for interactive navigation inherent to CVDN. We demonstrated the efficacy of our approach via experiments and ablations. We proposed generalizing the game-play regime introduced with RMM (Roman et al., 2020) to enable interactive fine-tuning and evaluation of VISITRON-like models with pre-trained guides. The trade-off between GP and SPL in dialogue-based navigation, Sim-to-Real transfer (Anderson et al., 2021) and robustness in dialogue-based navigation in presence of speech recognition errors (Gopalakrishnan et al., 2020) are all important problems that merit detailed investigation in future work.

7 Societal Impact

The primary dataset of interest for our work on interactive navigation in photo-realistic indoor environments: Cooperative Vision-and-Dialog Nav-

igation (CVDN), is an English-only dataset. We also multi-task with several other datasets, namely R2R, R4R and RxR, but RxR is the only multilingual dataset and covers English, Hindi and Telugu. Due to CVDN being English-only, we utilized the English-portion of the RxR data during multi-task fine-tuning. There are over 6500 known languages spoken in the world today and vision-and-dialog navigation research could, in principle, be deployed in every home in the world, but due to current data limitations, it can only be deployed in English-speaking homes. Our modeling methods should transfer to other languages given sufficient volume of data, but ensuring that might not be possible for low-resource or endangered languages. VISITRON may benefit from new training schemes and modeling improvements to account for such scenarios. When deployed in real homes, speech would be the primary modality for most humans to interact with such robots. While speech recognition research has advanced considerably, ensuring accurate speech recognition across various speaker populations and accents is still challenging. Errors in speech recognition could impact VISITRON’s ability to navigate accurately, so making VISITRON robust to speech recognition errors will be necessary, potentially via augmentation of the language component of its training data with synthetic and actual speech recognition errors (Gopalakrishnan et al., 2020).

During navigation, VISITRON needs access to neighboring viewpoints to select from. Each environment in CVDN contains an underlying navigation graph which provides this information, which might not be the case in real unseen environments. In its absence, additional modules can be added that generate a local navigation graph based on the surroundings (Anderson et al., 2021). Datasets in the vision-and-language navigation space such as R2R and CVDN typically consider the environment to be static. Obstacle avoidance methods need to be added to models built using these datasets to avoid hazardous collisions in a dynamic environment, such as with moving humans and pets.

Large language models are known to have a high carbon footprint associated with training them (Strubell et al., 2019). VISITRON is about the same size as BERT (Devlin et al., 2019), which is now ubiquitously used in both academic and industrial settings and can be trained reasonably fast. The carbon footprint of this work was maintained within permissible limits by using a maximum of 8

Tesla V100 GPUs for training.

Acknowledgments

Many thanks to Jesse Thomason and Aishwarya Padmakumar for useful technical discussions and actionable feedback on multiple versions of this paper. We would also like to thank the anonymous reviewers for their service and useful feedback.

References

- Peter Anderson, Angel Chang, Devendra Singh Chaplot, Alexey Dosovitskiy, Saurabh Gupta, Vladlen Koltun, Jana Kosecka, Jitendra Malik, Roozbeh Motlaghi, Manolis Savva, et al. 2018a. On evaluation of embodied navigation agents. *arXiv preprint arXiv:1807.06757*.
- Peter Anderson, Ayush Shrivastava, Devi Parikh, Dhruv Batra, and Stefan Lee. 2019. Chasing ghosts: Instruction following as bayesian state tracking. In *Advances in Neural Information Processing Systems*, pages 371–381.
- Peter Anderson, Ayush Shrivastava, Joanne Truong, Arjun Majumdar, Devi Parikh, Dhruv Batra, and Stefan Lee. 2021. Sim-to-real transfer for vision-and-language navigation. In *Conference on Robot Learning*, pages 671–681. PMLR.
- Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton van den Hengel. 2018b. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3674–3683.
- Yonatan Bisk, Ari Holtzman, Jesse Thomason, Jacob Andreas, Yoshua Bengio, Joyce Chai, Mirella Lapata, Angeliki Lazaridou, Jonathan May, Aleksandr Nisnevich, Nicolas Pinto, and Joseph Turian. 2020. [Experience grounds language](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8718–8735, Online. Association for Computational Linguistics.
- Kay Henning Brodersen, Cheng Soon Ong, Klaas Enno Stephan, and Joachim M. Buhmann. 2010. [The balanced accuracy and its posterior distribution](#). In *Proceedings of the 2010 20th International Conference on Pattern Recognition, ICPR '10*, page 3121–3124, USA. IEEE Computer Society.
- Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. 2017. Matterport3d: Learning from rgb-d data in indoor environments. *International Conference on 3D Vision (3DV)*.

- Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020. Uniter: Universal image-text representation learning. In *European conference on computer vision*, pages 104–120. Springer.
- Ta-Chung Chi, Minmin Shen, Mihail Eric, Seokhwan Kim, and Dilek Hakkani-tur. 2020. Just ask: An interactive learning framework for vision and language navigation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 2459–2466.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.
- Daniel Fried, Ronghang Hu, Volkan Cirik, Anna Rohrbach, Jacob Andreas, Louis-Philippe Morency, Taylor Berg-Kirkpatrick, Kate Saenko, Dan Klein, and Trevor Darrell. 2018. Speaker-follower models for vision-and-language navigation. In *Advances in Neural Information Processing Systems*, pages 3314–3325.
- Karthik Gopalakrishnan, Behnam Hedayatnia, Qinqiang Chen, Anna Gottardi, Sanjeev Kwatra, Anu Venkatesh, Rafer Gabriel, and Dilek Hakkani-Tür. 2019. Topical-chat: Towards knowledge-grounded open-domain conversations. In *INTERSPEECH*.
- Karthik Gopalakrishnan, Behnam Hedayatnia, Longshaokan Wang, Yang Liu, and Dilek Hakkani-Tur. 2020. Are neural open-domain dialog systems robust to speech recognition errors in the dialog history? an empirical study. In *INTERSPEECH*.
- Weituo Hao, Chunyuan Li, Xiujuan Li, Lawrence Carin, and Jianfeng Gao. 2020. Towards learning a generic agent for vision-and-language navigation via pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13137–13146.
- Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.
- Minlie Huang, Xiaoyan Zhu, and Jianfeng Gao. 2020. Challenges in building intelligent open-domain dialog systems. *ACM Transactions on Information Systems (TOIS)*, 38(3):1–32.
- Drew A Hudson and Christopher D Manning. 2019. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6700–6709.
- Gabriel Ilharco, Vihan Jain, Alexander Ku, Eugene Ie, and Jason Baldridge. 2019. General evaluation for instruction conditioned navigation using dynamic time warping. In *ViGIL@ NeurIPS*.
- Vihan Jain, Gabriel Magalhaes, Alexander Ku, Ashish Vaswani, Eugene Ie, and Jason Baldridge. 2019. Stay on the path: Instruction fidelity in vision-and-language navigation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1862–1872.
- Liyiming Ke, Xiujuan Li, Yonatan Bisk, Ari Holtzman, Zhe Gan, Jingjing Liu, Jianfeng Gao, Yejin Choi, and Siddhartha Srinivasa. 2019. Tactical rewind: Self-correction via backtracking in vision-and-language navigation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6741–6749.
- Seokhwan Kim, Mihail Eric, Karthik Gopalakrishnan, Behnam Hedayatnia, Yang Liu, and Dilek Hakkani-Tur. 2020. Beyond domain APIs: Task-oriented conversational modeling with unstructured knowledge access. In *Proceedings of the 21th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 278–289, 1st virtual meeting. Association for Computational Linguistics.
- Alexander Ku, Peter Anderson, Roma Patel, Eugene Ie, and Jason Baldridge. 2020. Room-across-room: Multilingual vision-and-language navigation with dense spatiotemporal grounding. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4392–4412.
- Xiujuan Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, et al. 2020. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *European Conference on Computer Vision*, pages 121–137. Springer.
- Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer.
- Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *Advances in Neural Information Processing Systems*, pages 13–23.
- Arjun Majumdar, Ayush Shrivastava, Stefan Lee, Peter Anderson, Devi Parikh, and Dhruv Batra. 2020. Improving vision-and-language navigation with image-text pairs from the web. In *European Conference on Computer Vision*, pages 259–274. Springer.
- Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28:91–99.

- Homero Roman Roman, Yonatan Bisk, Jesse Thomason, Asli Celikyilmaz, and Jianfeng Gao. 2020. Rmm: A recursive mental model for dialog navigation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 1732–1745.
- Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. 2018. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565.
- Emma Strubell, Ananya Ganesh, and Andrew McCallum. 2019. Energy and policy considerations for deep learning in nlp. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3645–3650.
- Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. 2019. Videobert: A joint model for video and language representation learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7464–7473.
- Hao Tan, Licheng Yu, and Mohit Bansal. 2019. Learning to navigate unseen environments: Back translation with environmental dropout. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2610–2621.
- Jesse Thomason, Michael Murray, Maya Cakmak, and Luke Zettlemoyer. 2020. Vision-and-dialog navigation. In *Conference on Robot Learning*, pages 394–406.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.
- Xin Eric Wang, Vihan Jain, Eugene Ie, William Yang Wang, Zornitsa Kozareva, and Sujith Ravi. 2020. 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 16*, pages 413–430. Springer.
- Deshraj Yadav, Rishabh Jain, Harsh Agrawal, Prithvijit Chattopadhyay, Taranjeet Singh, Akash Jain, Shiv Baran Singh, Stefan Lee, and Dhruv Batra. 2019. Evalai: Towards better evaluation systems for ai agents. *arXiv preprint arXiv:1902.03570*.
- Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. 2014. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78.
- Yubo Zhang, Hao Tan, and Mohit Bansal. 2020. Diagnosing the environment bias in vision-and-language navigation. *arXiv preprint arXiv:2005.03086*.
- Luowei Zhou, Hamid Palangi, Lei Zhang, Houdong Hu, Jason Corso, and Jianfeng Gao. 2020. Unified vision-language pre-training for image captioning and vqa. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 13041–13049.
- Fengda Zhu, Yi Zhu, Xiaojun Chang, and Xiaodan Liang. 2020a. Vision-language navigation with self-supervised auxiliary reasoning tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10012–10022.
- Wang Zhu, Hexiang Hu, Jiacheng Chen, Zhiwei Deng, Vihan Jain, Eugene Ie, and Fei Sha. 2020b. Babywalk: Going farther in vision-and-language navigation by taking baby steps. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2539–2556.
- 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10730–10739.