Vision-DIALOG Navigation by Exploring Cross-modal Memory

Yi Zhu\textsuperscript{1}, Fengda Zhu\textsuperscript{2}, Zhaohuan Zhan\textsuperscript{3}, Bingqian Lin\textsuperscript{3}, Jianbin Jiao\textsuperscript{1}, Xiaojun Chang\textsuperscript{2}, Xiaodan Liang\textsuperscript{3,4}
\textsuperscript{1}University of Chinese Academy of Sciences \textsuperscript{2}Monash University \textsuperscript{3}Sun Yat-sen University \textsuperscript{4}Dark Matter AI Inc.

Abstract

Vision-dialog navigation posed as a new holy-grail task in vision-language disciplinary targets at learning an agent endowed with the capability of constant conversation for help with natural language and navigating according to human responses. Besides the common challenges faced in visual language navigation, vision-dialog navigation also requires to handle well with the language intentions of a series of questions about the temporal context from dialogue history and co-reasoning both dialogues and visual scenes. In this paper, we propose the Cross-modal Memory Network (CMN) for remembering and understanding the rich information relevant to historical navigation actions. Our CMN consists of two memory modules, the language memory module (L-mem) and the visual memory module (V-mem). Specifically, L-mem learns latent relationships between the current language interaction and a dialog history by employing a multi-head attention mechanism. V-mem learns to associate the current visual views and the cross-modal memory about the previous navigation actions. The cross-modal memory is generated via a vision-to-language attention and a language-to-vision attention. Benefiting from the collaborative learning of the L-mem and the V-mem, our CMN is able to explore the memory about the decision making of historical navigation actions which is for the current step. Experiments on the CVDN dataset show that our CMN outperforms the previous state-of-the-art model by a significant margin on both seen and unseen environments.\textsuperscript{1}

1. Introduction

Powered by the recent progress in natural language processing and visual scene understanding, vision-language tasks such as Visual Question Answering (VQA) \cite{3,1,10} and Vision-Language Navigation (VLN) \cite{9,2,23,13} have been extensively explored. Recent works aims at developing a cognitive agent that jointly understands the natural language and visual scenes. However, such an agent is still far from being used in real-world applications (e.g., health care, intelligent tutoring) since it does not consider the continuous interaction with the outer environment over time. Specifically, the interaction in VQA is that the agent takes a question as input, and is required to answer a single question about a given image. The agent in VLN moves to the goal in a 3D environment following a natural language instruction. In contrast to the VQA and VLN, vision-dialog navigation \cite{31} is more challenging, where an agent is placed in a realistic environment and is required to find a target object by cooperating with the human using natural language dialogue. To achieve the navigation goal (i.e., find the target), the agent asks questions (e.g., Left or right from...
of dialogues to resolve. Previous work [31] resolves current dialog based on dialog history without considering visual information. To better understand the instruction implied in the current dialog, the agent utilizes not only contextual information from the previous dialogue but also historical visually grounded information.

In this paper, we propose a Cross-modal Memory Network (CMN) to exploit the agent memory about both the linguistic interaction with the human and the visual perception from the environment in the task of NDH. The CMN consists of two kinds of memory modules. The first module is the language memory module (L-mem), in which the dialogue histories between the navigator and the oracle are utilized to resolve the question and response at the current round. The goal of L-mem module is to give a better understanding of the instructions from the oracle who knows the best next step. The second module, the vision memory module (V-mem), aims at restoring the memory of visual scene during navigation. The contextualized representations generated by the L-mem are used to call back the visual memory about the places where the navigator has passed. In CMN, the L-mem and the V-mem are used collaboratively to explore the memory about the decision making of historical navigation actions at each step, which provides a cross-modal context for the understanding of the navigation instruction indicated by the current response.

Our method has the following merits: 1) Different from the existing vision-dialog navigation method that predicts each action individually, our CMN aims to restore the memory about the previous actions. 2) CMN learns to capture the cross-modal correlations between the language and visual information and generalizes well to unseen environments. 3) CMN is simple yet effective to tackle the challenging task of NDH and significantly outperforms the previous state-of-the-art method on both seen and unseen environments on the CVDN dataset.

2. Related Works

Visual Dialogue: The NDH agent is required to resolve the current dialog at each round based on dialog histories. Our work is related to some visual dialogue methods [8, 19, 25, 29, 12, 37, 16] for visual coreference resolution. These works learn to resolve the current sentence by exploring language attentions at word level or sentence level. [19] calculates the correlation between the pronoun words and the object labels that appeared in previous dialogues. [25, 16, 12] learn to contextualize the current question based on the attention on previous dialogs. Different from visual dialog where the dialogs share the same visual context, the historical information about the temporal visual views is important for the NDH agent. Our method restores cross-modal memory about both the dialog history and previous visual scenes.

Embodied Navigation: The problem of navigating in an embodied environment in vision and robotics has long been studied [32, 7, 11]. Despite of extensive research, embodied navigation problems remains challenging. A number of simulated 3D environments have been proposed to study navigation, such as Doom [17], AI2-THOR [18] and House3D [36]. Recently, deep reinforcement learning [24, 20, 28] shows its advantages in robust sequential decision making in noisy environments. Thus it is widely applied in embodied navigation. A number of works with deep reinforcement learning have achieve state-of-the-art results in many navigation benchmarks. [15, 22] However, the lack of photorealism and natural language instruction limits the application of these environments. Armeni et al. propose Stanford 2D-3DS [4], an embodied environment with realistic RGB-D and semantic information input. Anderson et al. [2] propose Room-to-Room (R2R) dataset, the first Vision-Language Navigation (VLN) benchmark based on real imagery [6].

The VLN task has attracted widespread attention since it is both widely applicable and challenging. Earlier work [35] combined model-free [24] and model-based [27] reinforcement learning to solve VLN. Fried et al. propose a speaker-follower framework for data augmentation and reasoning in supervised learning. In addition, a concept named “panoramic action space” is proposed to facilitate optimization. Later work [34] has found it beneficial to combine imitation learning [5, 14] and reinforcement learning [24, 28]. The self-monitoring method [21] is proposed to estimate progress made towards the goal. Researchers have identified the existence of the domain gap between training and testing data. Unsupervised pre-exploration [34] and Environmental dropout [30] are proposed to improve the ability of generalization. Rich information is explored by several self-supervised auxiliary reasoning tasks [38] to improve the visual grounding during navigation. The challenge of NDH task compared to VLN lies in two aspects: 1) The language instruction of VLN clearly describes the steps necessary to reach the goal while the NDH agent is given an ambiguous hint requiring exploration and dialog to resolve. 2) The trajectory of VLN is sequential, while the NDH trajectory which consists of sub-trajectories of each dialog is hierarchical. CMN captures the hierarchical correlation between and within sub-trajectories and explores cross-modal memory about historical actions to help better resolve the
dialog. However, current VLN methods seek to perceive the language and the visual scene sequentially.

**Vision-Dialog Navigation:** The task of Navigation by Dialog History (NDH) was recently proposed by [31], enabling the smart assistants with continuous communication and cooperation with the users via natural language and finally achieve their goal. An existing approach to this task follows the classical sequence-to-sequence formulation, beginning with the initial work introducing the task [31]. The actions of each step are predicted independently, this approach failed to explore the relevant information about the previous action decision based on the attention. By exploring the cross-modal memory of the agent interaction with the human and the environment, our method improves the navigation performance and makes it explainable for the agent decision making procedure.

### 3. Method

In this section, we briefly describe the Navigation by Dialog History (NDH) task and define the variables that will be used in the paper in Sec. 3.1. We introduce the feature representation for language and image in Sec. 3.2. We present the Vision Memory modules (V-mem) and the Language Memory modules (L-mem) of the proposed Cross-modal Memory Network (CMN) in Sec. 3.3 and Sec. 3.4.

### 3.1. Problem Setup

According to the NDH task, the dialogs often begin with an underspecified, ambiguous instruction (e.g., Go to the table), which requires further clarification. A dialog prompt is a tuple \((S, t_0, p_0, G_j)\) contains a house scan \(S\), a target object \(t_0\) to be found, a starting position \(p_0\), and a goal region \(G_j\). At each round of communication, the navigator asks a question \(Q\) and get a response \(R\) from the oracle, then predict the navigation action \(A\). Each sample of VDN consists of a repeating sequence \(\langle A_0, Q_1, R_1, A_1, ..., Q_k, R_k, A_k \rangle\) for \(k\) rounds of interaction. For each dialog with prompt \((S, t_0, p_0, G_j)\), a vision-dialog navigation instance is created for each of \(0 \leq i \leq k\). The input is a hint about the target \(t_0\) and a dialog history \(H_t = \{D_1, ..., D_{t-1}\}\) at the \(t\)-th round of dialog, where \(D_t = (Q_t, R_t)\).

Given the problem setup, the proposed CMN for NDH can be framed as an encoder-decoder architecture: (1) an encoder that explores the language memory about the historical communication \(H_t\) between the navigator and the oracle, generating contextualized representation for \(D_t\), (2) a decoder that first looks back to previous views of the navigator to help resolve the current dialog, and then converts the representation enhanced by the cross-modal memory into the navigation action space \(A_t\). Fig. 2 presents an overview of the architecture of CMN, which consists of L-mem and V-mem module. The L-mem learns to attend relevant previous dialogs to explore context information in a given dialog \(D_t = (Q_t, A_t)\). The V-mem learns to recall the cross-modal memory at the previous step for the visual perception of the current scene.

### 3.2. Feature Representation

**Language Features:** We first embed each word in the current dialog \(D_t\) to \(\{w_{t,1}, ..., w_{t,T}\}\) by using pre-trained GloVe [26] embeddings, where \(N\) denotes the sum of the number of tokens in \(Q_t\) and \(R_t\). Then a two-layer LSTM is employed to generate a sequence of hidden states \(\{h_{t,1}, ..., h_{t,T}\}\). The feature of each dialog \(D_t\) is the last hidden state of the LSTM \(h_{t,T}\), denoted as \(d_t \in \mathbb{R}^L\):

\[
\{h_{t,1}, ..., h_{t,N}\} = \text{LSTM}\{w_{t,1}, ..., w_{t,N}\}
\]

where \(L\) is the maximal length of the dialog sentence contains a question and an answer. Likewise, the dialog history \(H_t\) is embedded following Eq. 1, yielding \(\{d_t\}_{t=0} \in \mathbb{R}^{t \times L}\).

**Image Features:** For each visual frame, we use the panoramic representation for navigation. The panoramic view is split into image patches of 36 different views, resulting in panoramic features \(V_{t,s} = \{v_{t,s,i}\}, v_{t,s,i} \in \mathbb{R}^{2048}\) at the \(s\)-th step of round \(t\), where \(v_{t,s,i}\) denotes the pretrained CNN feature of the image patch at viewpoint \(i\).

### 3.3. Visual Memory

We expect the navigator to make the current decision by remembering the previous cross-modal memory about the environment. Here we introduce V-mem to restore the previous cross-modal memory during navigation to help generate memory-aware representations for the current vision perception. First, we used the final cross-modal encoding \(v_{t,s}^{vlm}\) of the previous step \(s - 1\) to attend on the panoramic features \(V_{t,s}\) in step \(s\), the resulted memory-aware features \(V_{t,s}^{vis}\) depicts the correlation between previous decision and current views. We first project the \(v_{t,s}^{vlm}\) and \(V_{t,s}\) to \(c\) dimensions and compute soft attention \(A^{vis}\) as follows:

\[
X = f_v(v_{t,s}^{vlm}) \odot f_{vlm}(v_{t,s})
\]

\[
A^{vis}(v_{t,s}^{vlm}, v_{t,s,i}) = \sigma(X) / \sqrt{c},
\]

where \(f_v(\cdot)\) and \(f_{vlm}(\cdot)\) denote the two-layer multi-layer perceptrons which convert the input to \(c\) dimensions. \(\sigma\) represent softmax function. \(\odot\) denotes hadamard product (i.e., element-wise multiplication). Then we compute the memory-aware representation which contains the information about the previous action decision based on the attention \(A^{vis}\) as:

\[
u_{t,s}^{mem} = \sum_{i=1}^{s} A^{vis}_{s,i} v_{t,s,i}.
\]
3.4. Language Memory

In this section, we formally describe the Language Memory (L-mem) module. Given the current question-answer \( D_t \) and the dialog history features, the L-mem module aims to attend to the memory about the most relevant dialogs in history with respect to the dialog at the current round. Specifically, we first compute scaled dot product attention (Attention) [33] in multi-head settings which are called multi-head attention. Let \( d_t \) and \( M_t = \{ h_t \}_{t=0}^{t-1} \) be the current dialog and the dialog history feature vectors, respectively. The trainable weights \( W_n^Q, W_n^K \) and \( W_n^V \in \mathbb{R}^L \times c \) are used to project the \( d_t \) and \( M_t \) into features of \( c \) dimension. In our experiment the dimension \( c \) is set to 512. Then we calculate the attention \( A_n^\text{lan} \) of \( d_t \) to each element of the dialog memory \( M_t \) as:

\[
A_n^\text{lan}(d_t, h_t) = \text{softmax}\left((d_t W_n^Q) (h_t W_n^K)^T / \sqrt{c}\right),
\]

\[
d_t = \text{concat}_{n=1}^N \left\{ \sum_{i=0}^{t} A_n^\text{lan}(d_t, h_i) W_n^V h_i \right\},
\]

\[
\hat{d}_t = \text{LayerNorm}(d_t + d_t),
\]

where the outputs of each of the \( N \) attention heads are concatenated and the contextualized representation of current dialog \( \hat{d}_t \) is computed by applying a residual connection, followed by layer normalization. Next, the \( \hat{d}_t \) is fed into a two-layer nonlinear multi-layer perceptron (\( f_{\text{lan}} \)), followed by a layer normalization and residual connection as:

\[
\hat{d}_t = \text{LayerNorm}(f_{\text{lan}}(d_t + \hat{d}_t)),
\]

\[
d_t^{\text{lex}} = \text{concat}(d_t, \hat{d}_t).
\]  

We then obtain the memory-aware representations by concatenating the contextual representation \( d_t \) and the original dialog representation \( d_t \), denoted as \( d_t^{\text{lex}} \in \mathbb{R}^{2L} \). Building on the multi-head attention mechanism, the L-mem can be stacked in multiple layers to get a high-level abstraction of the context of dialog history.

3.5. Cross-modal Memory

After exploring the memory of visual perception and language interactions with attention modules respectively, we further introduce cross-modal attention to explore the semantic correlation between the language and visual memory. We first perform language-to-vision attention by leveraging the memory-aware representation of the last dialog \( d_t^{\text{lex}} \) to attend the visual memory \( V_{t,s} \) via the scaled dot product attention.

\[
\begin{align*}
A_n^{\text{lan}}(d_t^{\text{lex}}, V_{t,s}) & = \text{softmax}\left((d_t^{\text{lex}} W_n^Q) (V_{t,s} W_n^K)^T / \sqrt{c}\right), \\
\hat{d}_{t,s}^{\text{lex}} & = \text{LayerNorm}(d_t^{\text{lex}} + d_t^{\text{lex}}), \\
\hat{e}_{t,s}^{\text{vln}} & = \text{concat}(\hat{d}_{t,s}^{\text{lex}}, V_{t,s}), \\
A_n^{\text{vln}}(\hat{e}_{t,s}^{\text{vln}}, \hat{d}_{t,s}^{\text{lex}}) & = \text{softmax}\left((\hat{e}_{t,s}^{\text{vln}} W_n^Q) (\hat{d}_{t,s}^{\text{lex}} W_n^K)^T / \sqrt{c}\right), \\
\hat{d}_{t,s}^{\text{lex}} & = \text{LayerNorm}(\hat{d}_{t,s}^{\text{lex}} + \hat{d}_{t,s}^{\text{lex}}).
\end{align*}
\]
product attention as:

\[ e_{t,s}^{vlm} = \text{Attention}(d_t^{ctx}, \{v_{t,0}^m, ..., v_{t,s}^m\}). \] (6)

The visual memory provides supplement information about previous views, which enables better scene understanding for the navigator. Then we calculate vision-to-language attention to generate the final cross-modal memory encoding \( e_{t,s}^{vlm} \) as:

\[ e_{t,s}^{vlm} = \text{Attention}(e_{t,s}^{vlm}, \{d_t^m, ..., d_t^m\}). \] (7)

Here the language memory is incorporated twice. The first time is in Eq. 5 and the second time is in Eq. 7. The differences between the two incorporation lie in three folds. Firstly, \( d_t^{ctx} \) is the concatenation of the last dialog feature \( d_t \) and the attention weighted feature of previous dialog history \( H_t \). So the dominate semantics derives from the last dialog \( d_t \), namely, \( d_t^{ctx} \) provides the contextualized information for \( d_t \). In contrast, the cross-modal memory-aware representation \( e_{t,s}^{vlm} \) exploit to discover the correlation between visual memory and all the existing dialogues. Secondly, the goal of calculating \( d_t^{ctx} \) is to help the navigator better understand the current response from oracle, while the \( e_{t,s}^{vlm} \) aims to learn the alignment between visual memory and language instructions, capturing the temporal correlations for better visual grounding. Lastly, the language-to-vision attention in Eq. 6 and the vision-to-language attention in Eq. 7 construct a closed reasoning path between visual and language context, providing rich information of cross-modal memory for the action prediction.

In Fig. 4 we describe the cooperation of the language memory and visual memory of the navigator. The language memory is maintained for each instance of the NDH task, resolving the ambiguous instruction of oracle. In contrast, the visual memory is collected within each round to capture temporal visual cues, which could benefit visual grounding. As is shown in Fig. 2, the action for each step is predicted based on the encoding produced by exploring the cross-modal attention between the visual and language memory.

### 3.6. Action Decoder

By performing memory-aware reasoning with the cooperation of both language instructions and visual views, the navigator is able to better understand the historical decisions from temporal alignment between dialog history and previous views, which provide rich context information for the action prediction for the current step \( s \) as:

\[ \hat{a}_{t,s} = \sigma(f_m(e_{t,s}^{vlm})), \]

\[ a_{t,s} = \text{softmax}(f_a(\hat{a}_{t,s})) \] (8)

where \( f_m(\cdot) \) and \( f_a(\cdot) \) are single-layer linear transformations to project the \( e_{t,s}^{vlm} \) from \( K + L \) dimension to \( K \), and project the \( \hat{a}_{t,s} \) from \( K \) dimension to \( M \) dimension which is the number of actions. Following [9], we employ the panoramic action space with panoramic features of images. The agent is required to choose a candidate from the panoramic features of the visual views in the next step.

### 4. Experiments

In this section, we first introduce the experimental settings, including the CVDN dataset, evaluation metrics, and implementation details, Sec. 4.1. Then we compare the proposed Cross-modal Memory Networks (CMN) with previous state-of-the-art methods and several baseline models in Sec. 4.2 and present ablation studies in Sec. 4.3. Finally, we show the quantitative results in Sec. 4.4.

#### 4.1. Settings

**Datasets:** We evaluate our model on the CVDN dataset which collects 2050 human-human navigation dialogs and over 7k trajectories in 83 MatterPort houses [2]. Each trajectory corresponds to several question-answer exchanges. The dataset contains 81 unique types of household objects, each type appears in at least 5 houses and appear between 2 and 4 times per such house. Each dialog begins with an ambiguous instruction, and the subsequent question-answer
interaction between the navigator and oracle will lead the navigator to find the target.

Evaluation Metrics: Following the previous works in visual language navigation and visual dialog navigation, we use four popular metrics to evaluate the proposed method from different aspects: (1) Success Rate (SR), the percentage of the final positions less than 3m away from the goal location. (2) Oracle Success Rate (OSR), the success rate if the agent can stop at the closest point to the goal along its trajectory. (3) Goal Progress (GP), the average agent progress towards the goal location. (4) Oracle Path Success Rate (OPSR), the success rate if the agent can stop at the closest point to goal along the shortest path. Note that this could be different from the OSR if the shortest path is not be used for supervision (i.e., mixed path or navigator path).

Different Supervision: The navigator paths in the CVDN dataset are collected from humans playing roles as the navigators, while the oracle paths are simultaneously generated by the shortest path planner. The typical supervision for the agent in the navigation task is defined by the shortest path, which is the same as the oracle path given in the CVDN dataset. However, even human demonstrations could be imperfect compared to the oracle path in realistic situations. Thus, the CVDN dataset also provides a new form of supervision called the mixed supervision path. The mixed supervision path is defined as the navigator path when the end nodes of the navigator and the oracle are the same, and the oracle path otherwise.

Implementation Details: We train in PyTorch using the Adam optimizer and set the learning rate to 0.0001. We train all agents with student-forcing for 20000 iterations of batch size 60, and evaluate validation performance every 100 iterations. The best performance across all epochs is reported for validation folds. The navigator moves its predicted action \( \hat{a} \) at each time step. Then cross-entropy loss is applied to \( \hat{a} \) and \( a^* \) which is the next action along the shortest path to the target. The total training process costs 2 GPU days on a single Titan 1080Ti device.

4.2. Quantitative Results

Compared Models: We compare our proposed CMN with several baselines and the state-of-the-art method: (1) The Shortest Path Agent takes the shortest path to the supervision goal at inference time and represents the upper bound navigation performance for an agent. (2) The Random Agent chooses a random heading and walks up to 5 steps forward (as in [2]). (3) The Vision Only baseline where the agent considers visual input with empty language inputs. (4) The Dialog Only baseline where the agent considers language input with zeroed visual features. (5) The sequence to sequence model proposed in [31] where the historical dialogs are concatenated to form a single instruction as in visual language navigation models [2].

Comparison to previous methods: As is shown in Tab. 1, our proposed CMN outperforms the previous state-of-the-art method [31] on the Goal Progress (m) with different supervisions (e.g., planner path (Oracle), player path (Navigator) and trusted path (Mixed)), demonstrating the ability of our method to grounding visual elements in the explored environments. When evaluated on the Val Unseen

<table>
<thead>
<tr>
<th>Method</th>
<th>Val Seen</th>
<th>Val Unseen</th>
<th>Test Unseen</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Oracle</td>
<td>Navigator</td>
<td>Mixed</td>
</tr>
<tr>
<td>Baseline (Shortest Path Agent)</td>
<td>8.29</td>
<td>7.63</td>
<td>9.52</td>
</tr>
<tr>
<td>Baseline (Random Agent)</td>
<td>0.42</td>
<td>0.42</td>
<td>0.42</td>
</tr>
<tr>
<td>Baseline (Vision Only)</td>
<td>4.12</td>
<td>5.58</td>
<td>5.72</td>
</tr>
<tr>
<td>Baseline (Dialog Only)</td>
<td>1.41</td>
<td>1.43</td>
<td>1.58</td>
</tr>
<tr>
<td>Sequence-to-sequence model [31]</td>
<td>4.48</td>
<td>5.67</td>
<td>5.92</td>
</tr>
<tr>
<td>CMN (Ours)</td>
<td>5.47</td>
<td>6.14</td>
<td>7.05</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Method</th>
<th>Val Seen</th>
<th>Val Unseen</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>GP (m)</td>
<td>OSR (%)</td>
</tr>
<tr>
<td>Seq-to-seq [31]</td>
<td>5.92</td>
<td>63.8</td>
</tr>
<tr>
<td>VLN Baseline [9]</td>
<td>6.15</td>
<td>58.9</td>
</tr>
<tr>
<td>CMN w/o V-mem</td>
<td>6.33</td>
<td>61.3</td>
</tr>
<tr>
<td>CMN w/o L-mem</td>
<td>6.47</td>
<td>58.6</td>
</tr>
<tr>
<td>CMN (Ours)</td>
<td>7.05</td>
<td>65.2</td>
</tr>
</tbody>
</table>

Table 1: Comparison of the performance on Goal Progress (m). Different supervisions of end path are used in training. Oracle indicates planner path, Navigator indicates player path, Mixed indicates trusted path.

Table 2: Comparison of the performance on several popular benchmarks of NDH. We train a vision-language navigation method on the CVDN dataset, the performance is reported as VLN baseline. We also show the ablation studies on the L-mem and V-mem modules of our CMN.
**Table 3:** Ablation studies on different types of memory information for our proposed CMN, including visual memory, language memory, and cross-modal memory.

<table>
<thead>
<tr>
<th>V-mem</th>
<th>L-mem</th>
<th>VL-mem</th>
<th>Goal Process (m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>2.95</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>2.74</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>2.04</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>2.54</td>
</tr>
</tbody>
</table>

and Test Unseen data, the gap between our CMN and the seq-to-seq method is also significant, showing that CMN generalizes well for unexplored environments.

### 4.3. Ablation Study

We ablate the V-mem and L-mem module on the Val sets in Tab. 2 and the Test set in Tab 3.

**Baselines:** In the first row, we conduct a baseline from VLN by directly using the concatenated dialog history as the language input. The difference between the VLN baseline and the Sequence-to-sequence model are two folds. First, the Seq-to-seq model uses a $1 \times 2048$ feature vector to represent each panoramic image, while the dimension of the visual feature used in baseline VLN is $1 \times 36 \times 2048$ for all 36 views of the panoramic image. Second, the action space in the Seq-to-seq model is the low-level visuomotor space, where the predictions of actions are 3-d logits. In contrast, the baseline VLN uses panoramic action space, where the agent has a complete perception of the scene and directly performs high-level actions. Our framework is built based on the baseline VLN with both the panoramic vision features and the panoramic action space.

**Effect of different memory module:** We disable the V-mem module by directly averaging the visual features of each panoramic view. In Tab. 2 and Tab. 3, we can see that the performance dropped when the model lost visual memory about previous navigation steps. The reason why we use averaged feature here is that the averaged features are more confused and useless than the last memory features since closer memory is more correlated to the current state than earlier memory, which helps us disable most of the functionality of memory modules in ablation studies.

To remove the L-mem module, we replace the memory-aware representation of the language interactions by the word-level context within the last question-answer sentences. It can be seen in Tab. 2 and Tab. 3 that the performance of our method dramatically decreases when discarding the language memory module (L-mem), indicating that the language memory is crucial for the understanding of the oracle instructions. In Tab. 3, we also set the output of our encoder ($e_{l,s-1}^{vlm}$, Fig. 2) to zero values to eliminate the cross-modal memory context (VL-mem) from the previous navigation step. The VL-mem can be regarded as a high-level abstraction of historical navigation and represents rich information about the previous action decision made by the agent. The results indicate that the performance of our model would drop when the cross-modal memory is lost.

**Discussion about the temporal order in memory:** As is shown in Fig. 2, CMN predicts the action at step $s$ by restoring the cross-modal memory $e_{l,s-1}^{vlm}$, which represents the memory about the decision making of the previous navigation action at step $s-1$. From this point of view, the information about the temporal order of the cross-modal memory is implicitly embedded in our CMN. We further consider a more explicit way that directly concatenates the embedding of the order and the memory features. The performance is comparable to the implicit way.

### 4.4. Qualitative Results

To see how our proposed CMN performs the visual dialogue navigation task, we visualize two qualitative examples in Fig. 5. The first example contains one round of dialogue with eight steps of navigation. We can see that the agent successfully reaches the target with a comprehensive understanding of the natural language response from oracle, indicating that our method is also compatible with the VLN task. In the second example, there are five rounds of dialogue between the navigator and the oracle. In the first round of communication, the navigator moves two steps down to the hallway, which requires the navigator to correctly recognize visual elements, including bed, hallway, and stairs while understanding language instructions. In the third round of communication, the oracle suggests the navigator to “go back”. Our CMN explores to better understand this instruction by referring it to dialog history to resolve the specific meaning, that is, the inverse navigation operation of the previous steps. In step 1 of the third round, the navigator returns back to the hall. Finally, it finds the target “towel”. Since our proposed Cross-modal Memory could help restore the visual and language memory of previous steps and interactions, the navigator can resolve the ambiguous instruction “go back” and thus return to the previous location.

### 5. Conclusion

In this work, we propose the Cross-modal Memory Network (CMN) to tackle the challenging task of visual dialogue navigation by exploring cross-modal memory of the agent. The language memory can help the agent better understand the responses from the oracle based on the communication context. The visual memory aims to explore visually grounded information on the previous navigation path, providing temporal correlations for the views. Benefiting from the collaboration of both visual and language memory, CMN is proved to achieve constant improvement over popular benchmarks on visual dialogue navigation, especially when generalizing to the unseen environments.
Target: lamp

Q1: Should I go across this area with the table or up the stairs to the left?
A1: Looks like we actually want to move to your right, and go straight pass the table tennis table and then just keep heading straight.

Target: towel

Q1: To the left of the bed, or down the hallway?
A1: Down hall, through door and down stairs.

Q2: should I continue down this hallway?
A2: follow hall, turn left, down stairs.

Q3: This is not the room, can you guide me?
A3: Go back, past stairs and down hall.

Q4: Said not this room. continue down the hall?
A4: Yes to end, bathroom on left.

Figure 5: Examples of vision dialogue navigation using our proposed Cross-modal Memory Network. The red arrows indicate the predicted actions and the yellow boxes indicate the targets. Best viewed in color.

Acknowledgement. This work was supported in part by National Key R&D Program of China under Grant No.2018AAA0100300, National Natural Science Foundation of China under Grant No.U19A2073, No.61976233, No.61836012, and No.61771447 and Nature Science Foundation of Shenzhen Under Grant No.2019191361.
References


