

RMM: A Recursive Mental Model for Dialogue Navigation

Homero Roman Roman¹ Yonatan Bisk^{1,2}

Jesse Thomason³ Asli Celikyilmaz¹ Jianfeng Gao¹

¹Microsoft Research ²Carnegie Mellon University ³University of Washington

horomanr@microsoft.com

ybisk@cs.cmu.edu

Abstract

Language-guided robots must be able to both ask humans questions and understand answers. Much existing work focuses only on the latter. In this paper, we go beyond instruction following and introduce a two-agent task where one agent navigates and asks questions that a second, guiding agent answers. Inspired by theory of mind, we propose the Recursive Mental Model (**RMM**). The navigating agent models the guiding agent to simulate answers given candidate generated questions. The guiding agent in turn models the navigating agent to simulate navigation steps it would take to generate answers. We use the progress agents make towards the goal as a reinforcement learning reward signal to directly inform not only navigation actions, but also both question and answer generation. We demonstrate that **RMM** enables better generalization to novel environments. Interlocutor modelling may be a way forward for human-agent dialogue where robots need to both ask and answer questions.

1 Introduction

A key challenge for embodied language is moving beyond instruction following to instruction generation, which can require understanding the listener. The turn-based dialogue paradigm raises a myriad of new research questions, from grounded versions of traditional problems like co-reference resolution (Das et al., 2017a) to explicitly modeling theory of mind in order to consider the listener’s ability to understand generated instructions (Bisk et al., 2020). In this paper, we develop end-to-end dialogue agents to navigate photorealistic, indoor scenes to reach goal rooms. We train agents using the human-human Collaborative Vision-and-Dialogue Navigation (CVDN) (Thomason et al., 2019) dataset. CVDN dialogues are turn-based, with a *navigator* following *guide* instructions and asking questions when needed.

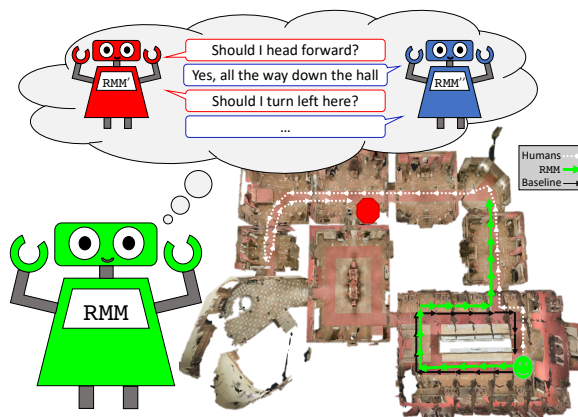


Figure 1: The **RMM** agent recursively models conversations with instances of itself to choose the right questions to ask (and answers to give) to reach the goal.

Modeling turn-based dialogues involves four core challenges:

- C1** A *navigator* deciding when to ask a question.
- C2** Generating *navigator* questions.
- C3** Generating *guide* question answers.
- C4** Generating *navigator* actions.

Prior work has addressed individual components of turn-based dialogue modeling. This work is the first to train *navigator* and *guide* agents to perform end-to-end, collaborative dialogues with question generation (**C2**), question answering (**C3**), and navigation (**C4**) conditioned on dialogue history.

Theory of mind (Gopnik and Wellman, 1992) posits that efficient questions and answers build on a shared world of experiences and referents. To communicate efficiently, people model both a listener’s mental state and the effects of their actions on the world. Modeling future worlds in navigation (Anderson et al., 2019) and control (Paxton et al., 2019) are open research questions, and we approximate solutions through a Recursive Mental Model (**RMM**) of a conversational partner. Our agent spawns instances of *itself* to simulate the ef-

fects of dialogue acts *before* asking a question or generating an answer to estimate their effects on navigation. Viewed as a single system, the agents cooperatively search through the space of dialogues to efficiently perform embodied navigation.

2 Related Work and Background

We build on research in multimodal navigation and the wider literature involving goal oriented dialogue. Table 1 summarizes how our work differs from existing work in vision-and-language navigation and task-oriented dialogue modelling.

Instruction Following tasks an embodied agent with interpreting natural language instructions and visual observations to reach a goal (Jayannavar et al., 2020; Wang et al., 2019; Ma et al., 2019; Anderson et al., 2018; Chen and Mooney, 2011). These instructions describe step-by-step actions the agent needs to take, and can involve the creation of speaker models for data augmentation that provide additional instructions (Fried et al., 2018). This paradigm has been extended to longer trajectories and outdoor environments (Chen et al., 2019), as well as to agents in the real world (Chai et al., 2018; Tellex et al., 2014). In this work, we focus on the the simulated, photorealistic indoor environments of the MatterPort dataset (Chang et al., 2017), and go beyond instruction following alone to a two-agent dialogue setting.

Navigation Dialogues task a *navigator* and a *guide* to cooperate to find a destination. Previous work includes substantial information asymmetry between the navigator and guide (de Vries et al., 2018; Narayan-Chen et al., 2019). Information asymmetry can take the form of the *navigator* seeing a bird’s eye, abstract semantic map while the *guide* sees egocentric simulation frames (de Vries et al., 2018), affecting the kind of dialog possible when low-level visual cues cannot be grounded by the *navigator*. Other work only investigates the *navigating* portion of the dialogue without considering text question generation and answering (Thomason et al., 2019). Going beyond models that perform navigation from dialogue history alone (Wang et al., 2020; Zhu et al., 2020; Hao et al., 2020), or decide when to ask *navigator* questions but do so as a simple “help” flag with oracle responses (Chi et al., 2020; Nguyen et al., 2019), in this work we train two agents: a navigator agent that asks questions, and a guide agent that answers those questions.

Representative Work	C1	C2	C3	C4
Anderson et al. (2018)				✓
Fried et al. (2018)			✓	✓
Narayan-Chen et al. (2019)			✓	
Nguyen and Daumé III (2019)	✓		✓	✓
Chi et al. (2020)	✓			✓
Thomason et al. (2019)				✓
RMM		✓	✓	✓

Table 1: Previous work has addressed subsets of the four key challenges for turn-based navigation dialogues by training single-turn agents. No prior work has tackled generating *navigator* questions (C2); by doing so, our work becomes the first to train two agents jointly on multi-turn dialogues where agents both produce and consume task-relevant language. We eschew only the challenge of deciding *when* to ask questions (C1), using a fixed heuristic instead.

Multimodal Dialogue takes several forms. In Visual Dialogue (Das et al., 2017a), an agent answers a series of questions about an image that may require dialogue context. Reinforcement learning gives strong performance on this task (Das et al., 2017b), and such paradigms have been extended to producing multi-domain visual dialogue agents (Ju et al., 2019). GuessWhat (de Vries et al., 2017) presents a similar paradigm, where agents use visual properties of objects to reason about which referent meets various constraints. Identifying visual attributes can also lead to emergent communication between pairs of learning agents (Cao et al., 2018).

Goal Oriented Dialogue systems can help a user achieve a predefined goal, from booking flights to learning kitchen tasks (Gao et al., 2019; Vlad Serban et al., 2015; Bordes and Weston, 2017; Chai et al., 2018). Modeling goal-oriented dialogue requires skills that go beyond language modeling, such as asking questions to clearly define a user request, querying knowledge bases, and interpreting results from queries as options to complete a transaction. Many recent task oriented systems are data-driven and trained end-to-end using semi-supervised or transfer learning methods (Ham et al., 2020; Mrksic et al., 2017). However, these data-driven approaches may lack grounding between the text and the environment state. Reinforcement learning-based dialogue modeling (Su et al., 2016; Peng et al., 2017; Liu et al., 2017) can improve completion rate and user experience by helping ground conversational data to environments.

3 Task and Data

Our work creates a two-agent dialogue task, building on the CVDN dataset (Thomason et al., 2019) of human-human dialogues. In that dataset, a human \mathcal{N} avigator and \mathcal{G} uide collaborate to find a goal room containing a target object. The \mathcal{N} avigator moves through the environment, and the \mathcal{G} uide views this navigation until the \mathcal{N} avigator asks a question in natural language (C1, C2). Then, the \mathcal{G} uide can see the next few steps a shortest path planner would take towards the goal, and produces a natural language response (C3). Dialogue continues until the \mathcal{N} avigator arrives at the goal (C4).

We model this dialogue between two agents:

1. Questioner (\mathcal{Q}) & Navigator (\mathcal{N})
2. Guide (\mathcal{G})

We split the first agent into its two roles: question asking (C2) and navigation (C4). As input, the agents receive the same data as their human counterparts in CVDN. Specifically, both agents (and all three roles) have access to the entire dialogue and visual navigation histories, in addition to a textual description of the target object (e.g., *a plant*). The \mathcal{N} avigator uses this information to execute on a sequence of actions composed of: `forward`, `left`, `right`, `look up`, `look down`, and `stop`. The Questioner asks for specific guidance from the Guide. The Guide is presented with the navigation and dialogue histories as well as the next five shortest path steps to the goal, given as a sequence of image observations those steps produce.

Agents are trained on human-human dialogues of natural language questions and answers from CVDN. Individual question-answer exchanges in that dataset are underspecified and rarely provide simple step-by-step instructions like “straight, straight, right, ...”. Instead, exchanges rely on assumptions of world knowledge and shared context (Frank and Goodman, 2012; Grice et al., 1975), which manifest as instructions rich with visual-linguistic co-references such as *should I go back to the room I just passed or continue on?*

The CVDN release does not provide any baselines or evaluations for the interactive dialogue setting we present, and instead focuses solely on navigation (C4). We use the same metric as that work, “Goal Progress” in meters—the distance reduction between the \mathcal{N} avigator’s starting position and ending position with respect to the goal location.

Dialogue navigation proceeds by iterating through the three roles until either the \mathcal{N} avigator

Algorithm 1: Dialogue Navigation

```
loc = p0;
hist = t0;
 $\vec{a} \sim \mathcal{N}(\text{hist});$ 
loc, hist = update( $\vec{a}$ , loc, hist);
while  $\vec{a} \neq \text{STOP}$  and  $\text{len}(\text{hist}) < 20$  do
   $q \sim \mathcal{Q}(\text{hist}, \text{loc});$  // Question
   $\vec{s} = \text{path}(\text{loc}, \text{goal}, \text{horizon} = 5);$ 
   $o \sim \mathcal{O}(\text{hist}, \text{loc}, q, \vec{s});$  // Answer
   $\text{hist} \leftarrow \text{hist} + (q, o);$ 
  for  $a \in \mathcal{N}(\text{hist})$  do
     $\text{loc} \leftarrow \text{loc} + a;$  // Move
     $\text{hist} \leftarrow \text{hist} + a;$ 
  end
end
return  $(\text{goal} - t_0) - (\text{loc} - t_0)$ 
```

chooses to stop or a maximum number of turns are played (Algorithm 1). In addition to “Goal Progress”, we report BLEU scores (Papineni et al., 2002) for evaluating the generation of questions and answers by comparing against human questions and answers. Note, in our dialogue setting, Goal Progress also implicitly measures the utility of generated language and is therefore complementary to BLEU when evaluating utility versus fluency.

4 Models

We introduce the Recursive Mental Model (RMM) as an initial approach to our new full dialogue CVDN task formulation. Key to this approach is allowing component models (\mathcal{N} avigator, Questioner, and Guide) to learn from each other and roll out possible dialogues and trajectories. We compare our model to a traditional sequence-to-sequence baseline, and we explore data augmentation using the Speaker-Follower method (Fried et al., 2018).

4.1 Sequence-to-Sequence Architecture

The underlying architecture, shown in Figure 2, is shared across all approaches. The core dialogue tasks are navigation action decoding and language generation for asking and answering questions. We present three sequence-to-sequence (Bahdanau et al., 2015) models to perform as \mathcal{N} avigator, Questioner, and Guide. The models rely on an LSTM (Hochreiter and Schmidhuber, 1997) encoder for the dialogue history. To encode visual observations, our models take the penultimate ResNet (He et al., 2015) layer as the image observation. Future work may explore different and more nuanced encoding architectures.

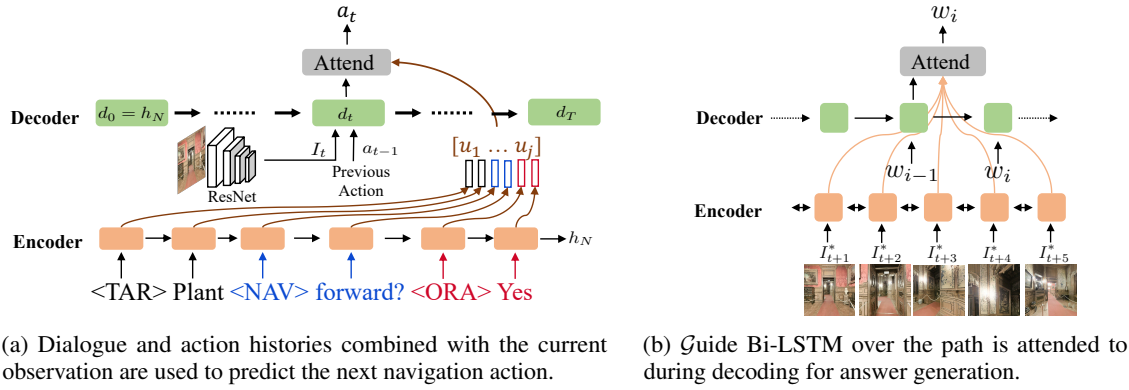


Figure 2: Our backbone Seq2Seq architectures are provided visual observations and portions of the dialogue history when taking actions or asking/answering questions.

Navigation Action Decoding (C4) Initially, the dialogue context is just a target object t_O category, for example “plant.” The goal room contains an instance of that category. As questions are asked and answered, the dialogue context grows. Following prior work (Anderson et al., 2018; Thomason et al., 2019), dialogue history words \vec{w} words are embedded as 256 dimensional vectors and passed through an LSTM to produce \vec{u} context vectors and a final hidden state h_N . The hidden state h_N is used to initialize the LSTM decoder. At every timestep the decoder is updated with the previous action a_{t-1} and current image observation I_t . The hidden state is used to attend over the language \vec{u} and predict the next action a_t (Figure 2a).

We pretrain the Navigator on the navigation task alone before fine-tuning in the full dialogue setting that we introduce. The next action is sampled from the model’s predicted logits, and the episode ends when either a stop action is sampled or 80 actions are taken (Thomason et al., 2019).

Speaker Models (C2 & C3) To generate questions and answers, we train sequence-to-sequence models (Figure 2b) where an encoder takes in a sequence of images and a decoder produces a sequence of word tokens. At each decoding timestep, the decoder attends over the input images to predict the next word of the question or answer. This model is also initialized via training on CVDN dialogues. In particular, question asking (Questioner) encodes the images of the current viewpoint where a question is asked, and then decodes the question tokens produced by the human Navigator. Question answering (Guide) encodes images of the next five steps the shortest path planner would take towards the goal, then decodes the language tokens pro-

duced by the human Guide. Pretraining initializes the lexical embeddings and attention alignments before fine-tuning in the collaborative, turn-taking setting we introduce in this paper.

Conditioning Context We define three levels of dialogue context given as input to our Navigator agents in order to evaluate how well they utilize the generated conversations. We compare agents’ ability to navigate to the goal room given:

- t_O the target object present in the goal room;
- QA_{i-1} additionally the previous question-and-answer exchange;
- $QA_{1:i-1}$ additionally the entire dialogue history.

We constrain the Questioner and Guide speaker models to condition on fixed contexts. The Questioner model takes as input the current visual observation I_t and the target object t_O . The Guide model takes the visual observations $I_{t+1}^* \dots I_{t+5}^*$ of the next five steps of navigation according to a shortest path planner, the target object t_O , and the last question Q_{i-1} generated by the Questioner.¹

4.2 Recursive Mental Model

We introduce the Recursive Mental Model agent (RMM),² which is trained with reinforcement learning to propagate feedback from navigation error through all three component models: Navigator, Questioner, and Guide. In this way, the training signal for question generation includes the training

¹This limits phenomena like co-reference, but dramatically reduces the model’s input space. Handling arbitrarily long contexts with limited training data is left to future work.

²<https://github.com/HomeroRR/rmm>

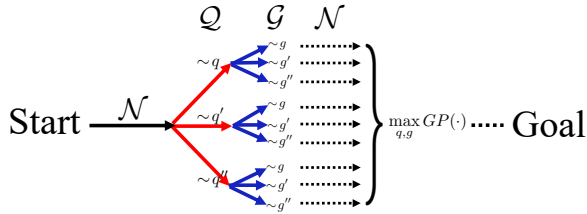


Figure 3: The Recursive Mental Model allows for each sampled generation to spawn a new dialogue and corresponding trajectory to the goal. The dialogue that leads to the most goal progress is followed by the agent.

signal for answer generation, which in turn is derived from the training signal from navigation error. The agent’s progress towards the goal in the environment informs the dialogue itself; each model educates the others (Figure 3).

Each model among the \mathcal{N} avigator, \mathcal{Q} uestioner, and \mathcal{G} uide may sample N trajectories or generations of max length L . These samples in turn are considered recursively by the **RMM** agent, leading to N^T possible dialogue trajectories, where T is at most the maximum trajectory length. To prevent unbounded exponential growth during training, each model is limited to a maximum number of total recursive calls per run. Search techniques, such as frontiers (Ke et al., 2019), could be employed in future work to guide the agent.

Training In the dialogue task we introduce, the agents begin only knowing the name of the target object. The \mathcal{N} avigator agent must move towards the goal room containing the target object, and can ask questions using the \mathcal{Q} uestioner model. The \mathcal{G} uide agent answers those questions given a privileged view of the next steps in the shortest path to the goal rendered as visual observations.

We define two different loss functions to learn the parameters θ of the \mathcal{N} avigator agent. We learn a policy $\pi_\theta(\tau|t_O)$ which maximizes the log-likelihood of the shortest path trajectory τ given target object t_O present in the goal room (Eq. 1). The action decoder $\mathbf{a}_t = f_{\theta_D}(\mathbf{z}_t, I_t)$ takes language encoder $\mathbf{z}_t = f_{\theta_E}(w_{1:t})$ as input along with the image observations I_t at time t . Dialogue context at time t , $w_{1:t}$ is input to the language encoder. The cross entropy loss is defined as:

$$J_{CE}(\theta) = - \sum_{t=1}^T \log \pi_\theta(\mathbf{a}_t | I_t, t_O, w_{1:t}) \quad (1)$$

Our second \mathcal{N} avigator RL agent loss is standard policy gradient based Advantage Actor Critic (Sut-

ton and Barto, 1998) minimizing a k-step TD³ error of the critic, $J_{RL}(\theta)$:

$$= - \sum_{t=1}^T \mathbf{A}^\pi \log \pi_\theta(\mathbf{a}_t | I_t, t_O, w_{1:t}) + \frac{1}{2} \sum_{t=1}^T (\mathbf{A}^\pi)^2 \quad (2)$$

$\mathbf{A}^\pi = r_{t+1} + V^\pi(I_{t+1}) - V^\pi(I_t)$ is the advantage function in Eq. 2, where r_{t+1} is the reward measured by the goal progress and the V^π denotes the state-value (critic) model. The first term in Eq. 2 is the actor loss, while the second term is the critic (value) loss of the advantage actor critic loss function. The overall system is trained end-to-end using sum of the RL agent loss of the navigator agent $J_{RL}(\theta)$ and the cross entropy loss between the ground truth and the generated trajectories, $J_{CE}(\theta)$. The speaker model parameters are also updated via the sum of the standard question/answer generation cross entropy and the composite \mathcal{N} avigator agent loss from the branch with the max goal progress.

Inference During training, exact environmental feedback—the remaining distance to the goal—can be used to evaluate samples and trajectories. This information is not available during inference, so we instead rely on the navigator’s confidence to determine which of several sampled paths should be explored. For every question-answer pair sampled, the agent rolls forward five navigation actions per sequence, and the trajectory sequence with the highest probability is used for the next timestep. This heuristic does not guarantee that the model is progressing towards the goal, but empirically confidence-based estimation enables progress.

4.3 Dialogue Gameplay

As is common in dialogue settings, there are several moving pieces and a growing notion of state throughout training and evaluation. In addition to the \mathcal{N} avigator, \mathcal{Q} uestioner, and \mathcal{G} uide, the \mathcal{N} avigator agent also needs to determine when to invoke the \mathcal{Q} uestioner model to get supervision from the \mathcal{G} uide (**C1**). We leave this component—when to ask questions—for future work and set a fixed number of steps before asking a question. We invoke the \mathcal{Q} uestioner model after every 4 navigation steps based on the human average of 4.5 steps between questions in CVDN.

Setting a maximum trajectory length is required due to computational constraints as the the lan-

³Temporal Difference

guage context $w_{1:j}$ grows. Following Thomason et al. (2019), we use a maximum navigation length of 80 steps, leading to a maximum of $\frac{80}{4} = 20$ question-answer exchanges per dialogue.

We use a single model for question and answer generation, and indicate the role of spans of text by prepending <NAV> (Questioner navigation questions) or <ORA> (Guide answers based on oracle views) tags (Figure 2a) to condition the generation task. During roll outs the model is reinitialized to prevent information sharing via the hidden units.

4.4 Training Details

We initialize the Navigator, Questioner, and Guide agents as encoder-decoder LSTM models with 512 hidden dimensions. The Navigator encoder is a forward LSTM, while the Questioner and Guide speaker models use bi-LSTM encoders. We use the 512 dimensional penultimate ResNet layer for image observations I_t , embed words w in 256 dimensions, and embed actions in 32 dimensions. The models observe a word history up to 160 tokens, and can decode up to 80 actions per episode. The value/critic module is a linear layer with relu and dropout on top of the hidden state.

We optimize the Navigator models with the Adam optimizer (Kingma and Ba, 2015) with a learning rate of 0.0001 with weight decay 0.0005. For the Questioner and Guide models, we use an RMSProp optimizer with learning rate 0.0001.

Models are pretrained on CVDN data with batches of size 100 for 20,000 iterations. During self-play, models are trained with batches of 10, for **RMM** with $N = 3$, or 100 else for 5,000 iterations. A dropout rate of 0.5 is used during all training. All Navigator models are trained using student sampling (Anderson et al., 2018). In **RMM** $_{N=3}$, one action sequence is produced via argmax decoding, while the other two via sampling (no temperature). The same is true for language decoding but with a temperature of 0.6. Exploration of how sampler strategies effect performance is left for future work.

Data Augmentation (DA) Navigation agents can benefit from generated language instructions (Fried et al., 2018). We augment the baseline model’s navigation training data in a fashion similar to the rollouts of **RMM** $_{N=3}$ to create a more direct comparison between the baseline and **RMM**. We choose a CVDN conversation and sample three action trajectory rollouts, two by sampling an action at each timestep, and one by taking the argmax

Model	Goal Progress (m) \uparrow				BLEU \uparrow	
	t_0	QA_{i-1}	$QA_{1:i-1}$	+Oracle Stopping	QA_{i-1}	$QA_{1:i-1}$
Seq2Seq	20.1	10.5	15.0	22.9	0.9	0.8
Seq2Seq + DA	20.1	10.5	10.0	14.2	1.3	1.3
RMM $_{N=1}$	18.7	10.0	13.3	20.4	3.3	3.0
RMM $_{N=3}$	18.9	11.5	14.0	16.8	3.4	3.6
Shortest Path	—————		32.8	—————		
<hr/>						
Seq2Seq	6.8	4.7	4.6	6.3	0.5	0.5
Seq2Seq + DA	6.8	5.6	4.4	6.5	1.3	1.1
RMM $_{N=1}$	6.1	6.1	5.1	6.0	2.6	2.8
RMM $_{N=3}$	7.3	5.5	5.6	8.9	2.9	2.9
Shortest Path	—————		29.3	—————		

Table 2: Dialogue results on CVDN. Data Augmentation adds noisy training data for the model. Goal progress evaluates the quality of the inferred navigation trajectory, while BLEU scores estimate the quality of the generated questions and answers. Evaluations conditioning on the entire dialogue history are highlighted in gray with the best results in blue.

action at each timestep. We evaluate those trajectories’ progress towards the conversation goal location and keep the best for augmentation. We give the visual observations of the chosen path to the pretrained Questioner model to produce a relevant instruction. This trajectory paired with a generated language instruction is added to the training data, and we downweight the contributions of these noisier pairs to the overall loss, so $loss = \lambda * generations + (1 - \lambda) * human$. The choice of λ affects the fluency of the language generated; we use $\lambda = 0.1$.

5 Results

In Table 2 we present dialogue results for our **RMM** agent and competitive baselines. We report two main results and four ablations for *seen* and *unseen* house evaluations; the former are novel dialogues in houses seen at training time, while the latter are novel dialogues in novel houses.

Full Evaluation The full evaluation paradigm conditions navigation on the entire dialogue history $QA_{1:i-1}$ in addition to the original target object t_0 . We present two conditions for **RMM** ($N = 1$ and $N = 3$). Recall that N indicates the number of trajectories (Navigator) or generations (Questioner, Guide) explored in our recursive calls. $N = 1$ corresponds to taking the single maximum prediction while $N = 3$ allows the agent to sample alternatives (Section 4.2). While low, the BLEU scores are better for **RMM**-based agents across settings.

A challenge for navigation agents is knowing when to `stop`. Following previous work (Anderson et al., 2018), we additionally report Oracle Success Rates measuring the best goal progress the agents achieve along the trajectory.

In *unseen* environments, the **RMM**-based agents make the most progress towards the goal and benefit from exploration at during inference ($N = 3$), and this result holds when considering Oracle Success. In *seen* environments, by contrast, the **RMM**-based agents perform slightly less well than the baseline sequence-to-sequence models on goal progress. This effect may be a consequence of *environment bias* in navigation simulations where houses are seen at both training and inference time with overlapping paths (Zhang et al., 2020).

Ablations We also include two simpler results: t_O , where the agent is only provided the target object and explores based on this simple goal, and QA_{i-1} where the agent is only provided the previous question-answer pair. Both of these settings simplify the learning and evaluation by focusing the agent on search and less ambiguous language, respectively. There are two results to note.

First, given only t_O the **RMM** trained model with sampling generalizes best to *unseen* environments. In this setting, during inference all models have the same limited information, so the RL loss and exploration have better equipped **RMM** to generalize.

Second, several trends invert between the *seen* and *unseen* scenarios. Specifically, the simplest model with the least information performs best overall in *seen* houses. This high performance coupled with weak language appears to indicate the models are learning a different (perhaps search based) strategy rather than how to utilize dialogue. In the QA_{i-1} and $QA_{1:i-1}$ settings, the agent generates a question-answer pair before navigating, so the relative strength of the **RMM** model’s communication becomes clear. We analyze the generated language and navigation behavior of our models.

6 Analysis

We analyze the lexical diversity and effectiveness of generated questions by the **RMM**.

6.1 Lexical Diversity

Both **RMM** and Data Augmentation introduce new language by exploring and the environment and generating dialogues. In the case of **RMM**, an RL loss is used to update the models based on the most

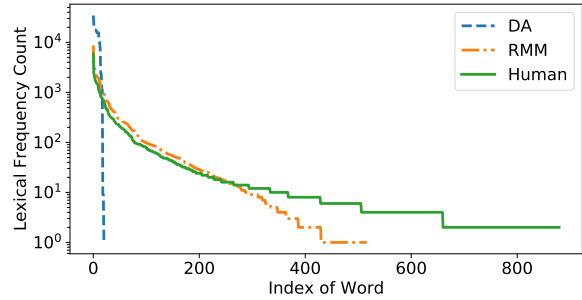


Figure 4: Log-frequency of words generated by human speakers as compared to the Data Augmentation (DA) and our Recursive Mental Model (RMM) models.

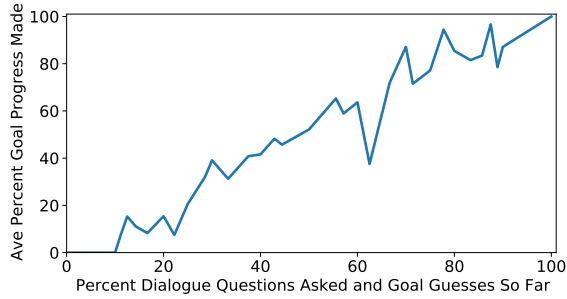
successful dialogue. Using Data Augmentation, the best generations are simply appended to the dataset for one epoch and weighted appropriately for standard, supervised training. The augmentation strategy leads to small boost in BLEU performance and goal progress in several settings (Table 2), but the language appears to collapse to repetitive and generic interactions. We see this manifest rather dramatically in Figure 4, where the DA is limited to only 22 lexical types. In contrast, **RMM** continues to produce over 500 unique lexical types, much closer to the nearly 900 used by humans.

Human Evaluation We collected human judgments comparing human dialogs with generated dialogs from the baseline and RMM agents on 254 randomly selected episodes from the *unseen* validation set. While RMM uses an RL objective to inform its language generation and achieves higher progress towards the goal in this setting (Table 2), it is rated as equally or more grammatical (57%) and as equally or more fluent (60%) than the baseline agent, suggesting that RMMs generated language has not devolved into a neuralese to achieve better task performance. Human dialogs were rated as equally or more grammatical and fluent than RMM (89%/83%) and the baseline (88%/80%).

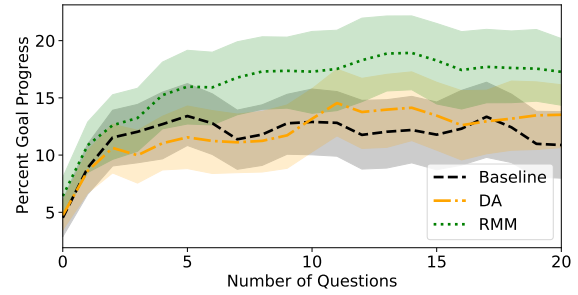
6.2 Effective Questions

The dialogue paradigm allows us to assess the efficacy of speech acts in accomplishing goals. In a sense, the best question elicits the answer that maximizes the progress towards the goal room. If agents are truly effective at modeling one other, we expect the number of dialogue acts to be minimal.

Human conversations in CVDN always reach the goal location, and usually with only 3-4 questions, as shown in Figure 5a. We see that the relationship between questions and progress is roughly linear,



(a) Human goal progress as dialogues unfold. As humans ask questions and make goal guesses, they roughly linearly make progress towards the goal location.



(b) Model goal progress against the number of questions. DA and RMM generated dialogues make slower but consistent progress (ending below 25% of total goal progress).

Figure 5: Effectiveness of human dialogues (left) versus our models (right) at reaching the goal location. The slopes indicate the effectiveness of each dialogue exchange in reaching the target.

excusing the occasional lost and confused human teams. The final human-human question is often simply confirmation that navigation has arrived successfully to the goal room.

In Figure 5b, we plot dialogues for the Baseline, Data Augmentation, and **RMM** agents against percent goal progress. The **RMM** consistently outperforms the other two agents in terms of goal progress for each dialogue act. We see an increase in progress for the first 10 to 15 questions before **RMM** levels off. By contrast, the Baseline and Data Augmentation agents exhibit shallower curves and fail to reach the same level of performance.

6.3 Example Dialogue

While Figure 1 shows a cherry-picked **RMM** trajectory from an *unseen* validation house, Figure 6 gives a lemon-picked **RMM** trajectory. We discuss the successes and failures of a lemon-picked—showcasing model failure—trajectory in Figure 6. As with all CVDN instances, there are multiple target object candidates (here, “fire extinguisher”) but only one valid goal room. Agents can become distracted by objects of the target instance in non-goal rooms. When the \mathcal{G} uide is shown the next few shortest path steps to communicate, those steps are towards the goal room. As can be seen in Figure 6, the learned agents have difficulty in deciding when to stop and begin retracing their steps, and in this case never arrived to the correct goal room.

The learned models’ generated language is of different levels of quality, with **RMM** language much more coherent and verbose than Data Augmentation language. Figure 7 shows generated conversations along with the Goal Progress (GP) at each point when a question was asked. Note that the generation procedure for all models use the same sam-

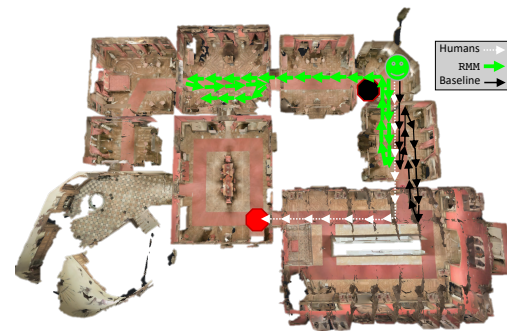


Figure 6: Trajectories in an unseen environment attempting to find a target “fire extinguisher.” The red stop-sign is the goal room, while the black stop-sign is a non-goal room containing fire extinguishers. The white trajectory is the human path from CVDN, black is the Baseline model, and green is our **RMM**_{N=3}.

pler, and they start training from the same checkpoint, so the relatively coherent nature of the **RMM** as compared to the simple repetitiveness of the Data Augmentation is entirely due to the recursive calls and RL loss. No model uses length penalties or other generation tricks to avoid degeneration.

7 Conclusions and Future Work

We present a two-agent task paradigm for cooperative vision-and-dialogue navigation (CVDN). Existing work in vision-and-language navigation is largely limited to navigation only (C4), sometimes with limited additional instructions (C4,C3). By contrast, this work requires navigation (C4), question asking (C2), and question answering (C3) components for learned, end-to-end dialogue. We find that simple speaker models are insufficient for the dialogue setting, and demonstrate promising results from a recursive RL formulation with turn taking informed by theory of mind.

	Conversation	GP
Human	👤 Do I go in between the ropes to my right or straight forward?	🗨️ straight forward through the next room 0
	👤 Should I proceed down the hall to the left of turn right?	🗨️ head down the hall to your right into the next room 13.3
	👤 Should I go through the open doors that are the closest to me?	🗨️ You are in the goal room 29.1
DA	👤 should i go into the room?	🗨️ you are in the goal room. 5.7
	👤 should i go into the room?	🗨️ you are in the goal room. 0.0
RMM _{N=3}	👤 should i head forward or bedroom the next hallway in front of me?	🗨️ yes, all the way down the small hall. 4.0
	👤 should i turn left here?	🗨️ head into the house, then you will find a doorway at the goal staircase. go through the doors before those two small exit chairs, about half way down the hall. 5.7
	👤 lots of sink in this house, or wrong did. ok which way do i go	🗨️ go down the hallway, take a left and go down the next hallway and up the stairs on the right. 8.8

Figure 7: Dialogue samples for Figure 6 with corresponding Goal Progress – see appendix for complete outputs.

There are several limitations to the models presented in this paper. We consider only agent-agent models, while the long-term goal of human-agent communication will require both human-in-the-loop training and evaluation. Future work using **RMM**-style modelling inspired by theory of mind will likely need to explicitly model the human interlocutor due to perceptual and communication differences (Liu and Chai, 2015), rather than assuming the interlocutor can be modeled as a copy of oneself as in this paper. Such modeling may incorporate world knowledge for richer notions of common ground, for example by explicitly detecting scene objects rather than using a fixed visual embedding (Zhang et al., 2020). Additionally, we currently require the *N*avigator agent to ask questions after a fixed number of steps, while determining when to ask questions is a complex problem in itself (C1) (Chi et al., 2020). Furthermore, we use a fixed branching factor, while a dynamic branching factor in non-parametric learning setting can incorporate the uncertainty of the policy model.

We hope this task paradigm will inspire future research on learning agent-agent, task-oriented communication with an eye towards human-agent cooperation and language-guided robots.

Acknowledgements

We thank the anonymous reviewers and the AC for the questions they raised and their helpful commentary, which strengthened the presentation of our task and models. This work was funded in part by ARO grant (W911NF-16-1-0121).

References

- Peter Anderson, Ayush Shrivastava, Devi Parikh, Dhruv Batra, and Stefan Lee. 2019. Chasing ghosts: Instruction following as bayesian state tracking. In *Thirty-third Conference on Neural Information Processing Systems (NeurIPS)*.
- Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sunderhauf, Ian Reid, Stephen Gould, and Anton van den Hengel. 2018. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *Computer Vision and Pattern Recognition (CVPR)*.
- Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. [Neural machine translation by jointly learning to align and translate](#). In *International Conference on Learning Representations (ICLR)*.
- 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](#). *ArXiv*.
- Antoine Bordes and Jason Weston. 2017. [Learning end-to-end goal-oriented dialog](#). In *International Conference on Learning Representations (ICLR)*.
- Kris Cao, Angeliki Lazaridou, Marc Lanctot, Joel Z Leibo, Karl Tuyls, and Stephen Clark. 2018. [Emergent communication through negotiation](#). In *International Conference on Learning Representations (ICLR)*.
- Joyce Y Chai, Qiaozhi Gao, Lanbo She, Shaohua Yang, Sari Saba-Sadiya, and Guangyue Xu. 2018. [Language to action: Towards interactive task learning with physical agents](#). In *International Joint Conference on Artificial Intelligence (IJCAI)*.
- Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niener, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. 2017. Matterport3d: Learning from rgb-d data in indoor environments. In *International Conference on 3D Vision*.

- David Chen and Raymond J Mooney. 2011. Learning to interpret natural language instructions from observations. In *Conference on Artificial Intelligence (AAAI)*.
- Howard Chen, Alane Suhr, Dipendra Misra, Noah Snaveley, and Yoav Artzi. 2019. Touchdown: Natural language navigation and spatial reasoning in visual street environments. In *Computer Vision and Pattern Recognition (CVPR)*.
- Ta-Chung Chi, Mihail Eric, Seokhwan Kim, Minmin Shen, and Dilek Hakkani-tur. 2020. Just ask: An interactive learning framework for vision and language navigation. In *Conference on Artificial Intelligence (AAAI)*.
- Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, José M.F. Moura, Devi Parikh, and Dhruv Batra. 2017a. Visual Dialog. In *Computer Vision and Pattern Recognition (CVPR)*.
- Abhishek Das, Satwik Kottur, José M.F. Moura, Stefan Lee, and Dhruv Batra. 2017b. Learning cooperative visual dialog agents with deep reinforcement learning. In *International Conference on Computer Vision (ICCV)*.
- Michael C. Frank and Noah D. Goodman. 2012. [Predicting pragmatic reasoning in language games](#). *Science*, 336(6084):998–998.
- 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 *Neural Information Processing Systems (NeurIPS)*.
- Jianfeng Gao, Michel Galley, and Lihong Li. 2019. [Neural approaches to conversational AI](#). *Foundations and Trends in Information Retrieval*.
- Alison Gopnik and Henry M Wellman. 1992. Why the child's theory of mind really is a theory. *Mind 'I&' Language*, 7 (1-2):145171.
- Herbert Paul Grice, P Cole, and J J Morgan. 1975. [Logic and conversation](#). *Syntax and Semantics, volume 3: Speech Acts*, pages 41–58.
- Donghoon Ham, Jeong-Gwan Lee, and Youngsoo Jang and Kee Eung Kim. 2020. End-to-end neural pipeline for goal-oriented dialogue system using gpt-2. In *Conference on Association for the Advancement of Artificial Intelligence (AAAI)*.
- Weituo Hao, Chunyuan Li, Xiujun Li, Lawrence Carin, and Jianfeng Gao. 2020. [Towards Learning a Generic Agent for Vision-and-Language Navigation via Pre-training](#). In *Computer Vision and Pattern Recognition (CVPR)*.
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. [Deep residual learning for image recognition](#). *arxiv:1512.03385*.
- Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.
- Prashant Jayannavar, Anjali Narayan-Chen, and Julia Hockenmaier. 2020. Learning to execute instructions in a Minecraft dialogue. In *Association for Computational Linguistics (ACL)*.
- Da Young Ju, Kurt Shuster, Y-Lan Boureau, and Jason Weston. 2019. All-in-one image-grounded conversational agents. *ArXiv*, abs/1912.12394.
- Liyiming Ke, Xiujun 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 *Computer Vision and Pattern Recognition (CVPR)*.
- Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*.
- Bing Liu, Gökhan Tür, Dilek Hakkani-Tür, Pararth Shah, and Larry P. Heck. 2017. [End-to-end optimization of task-oriented dialogue model with deep reinforcement learning](#). *CoRR*, abs/1711.10712.
- Changsong Liu and Joyce Yue Chai. 2015. Learning to mediate perceptual differences in situated human-robot dialogue. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, pages 2288–2294.
- Chih-Yao Ma, Zuxuan Wu, Ghassan AlRegib, Caiming Xiong, and Zolt Kira. 2019. The regretful agent: Heuristic-aided navigation through progress estimation. In *Computer Vision and Pattern Recognition (CVPR)*.
- Nikola Mrksic, Diarmuid Ó Séaghdha, Tsung-Hsien Wen, Blaise Thomson, and Steve J. Young. 2017. Neural belief tracker: Data-driven dialogue state tracking. In *Association for Computational Linguistics (ACL)*.
- Anjali Narayan-Chen, Prashant Jayannavar, and Julia Hockenmaier. 2019. [Collaborative dialogue in Minecraft](#). In *Association for Computational Linguistics (ACL)*.
- Khanh Nguyen and Hal Daumé III. 2019. [Help, anna! visual navigation with natural multimodal assistance via retrospective curiosity-encouraging imitation learning](#). In *Empirical Methods in Natural Language Processing (EMNLP)*.
- Khanh Nguyen, Debadepta Dey, Chris Brockett, and Bill Dolan. 2019. [Vision-based navigation with language-based assistance via imitation learning with indirect intervention](#). In *Computer Vision and Pattern Recognition (CVPR)*.

- Kishore Papineni, Salim Roukos, Todd Ward, and Weijing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Association for Computational Linguistics (ACL)*.
- Chris Paxton, Yonatan Bisk, Jesse Thomason, Arunkumar Byravan, and Dieter Fox. 2019. Propection: Interpretable plans from language by predicting the future. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*.
- Baolin Peng, Xiujun Li, Lihong Li, Jianfeng Gao, Asli Celikyilmaz, Sungjin Lee, and Kam-Fai Wong. 2017. Composite task-completion dialogue system via hierarchical deep reinforcement learning. In *Empirical Methods in Natural Language Processing (EMNLP)*.
- Pei-Hao Su, Milica Gasic, Nikola Mrksic, Lina Maria Rojas-Barahona, Stefan Ultes, David Vandyke, Tsung-Hsien Wen, and Steve J. Young. 2016. Online active reward learning for policy optimisation in spoken dialogue systems. In *Proceedings of the Association for Computational Linguistics (ACL)*.
- R. S. Sutton and A. G. Barto. 1998. Reinforcement learning: An introduction. *MIT press Cambridge*.
- Stefanie Tellex, Ross A Knepper, Adrian Li, Thomas M. Howard, Daniela Rus, and Nicholas Roy. 2014. Asking for help using inverse semantics. In *Robots: Science and Systems (RSS)*.
- Jesse Thomason, Michael Murray, Maya Cakmak, and Luke Zettlemoyer. 2019. Vision-and-dialog navigation. *Conference on Robot Learning (CoRL)*.
- I. Vlad Serban, R. Lowe, P. Henderson, L. Charlin, and J. Pineau. 2015. A Survey of Available Corpora for Building Data-Driven Dialogue Systems. *ArXiv*.
- Harm de Vries, Kurt Shuster, Dhruv Batra, Devi Parikh, Jason Weston, and Douwe Kiela. 2018. Talk the walk: Navigating new york city through grounded dialogue. *arXiv:1807.03367*.
- Harm de Vries, Florian Strub, Sarath Chandar, Olivier Pietquin, Hugo Larochelle, and Aaron C. Courville. 2017. Guesswhat?! visual object discovery through multi-modal dialogue. In *Computer Vision and Pattern Recognition (CVPR)*.
- Xin Wang, Qiuyuan Huang, Asli Celikyilmaz, Jianfeng Gao, Dinghan Shen, Yuan-Fang Wang, William Yang Wang, and Lei Zhang. 2019. Reinforced Cross-Modal Matching and Self-Supervised Imitation Learning for Vision-Language Navigation. In *Computer Vision and Pattern Recognition (CVPR)*.
- Xin Wang, Vihan Jain, Eugene Ie, William Yang Wang, Zornitsa Kozareva, and Sujith Ravi. 2020. Environment-agnostic Multitask Learning for Natural Language Grounded Navigation. *arXiv:2003.00443*.
- Yubo Zhang, Hao Tan, and Mohit Bansal. 2020. Diagnosing the environment bias in vision-and-language navigation. In *International Joint Conference on Artificial Intelligence (IJCAI)*.
- Yi Zhu, Fengda Zhu, Zhaohuan Zhan, Bingqian Lin, Jianbin Jiao, Xiaojun Chang, and Xiaodan Liang. 2020. Vision-Dialog Navigation by Exploring Cross-modal Memory. In *Computer Vision and Pattern Recognition (CVPR)*.

A Appendix

A.1 Additional Reproducibility details

1. Hardware: Single NVIDIA P100 GPU

2. Training times:

Setting	Iterations	Average time
Pretraining	20K	273m
Baseline	5k	1,622m
Data Aug	5k	1,161m
RMM1	5k	4,205m
RMM3	5k	6,590m

3. Model parameters:

Speaker 3.5M

Action Decoding 4.7M

4. Hyperparameters:

Bounds for temperature sampling [0.1-2.0]

Bounds for lambda DA contribution [0.1-1.0]

Trials for temperature sampling [0.1, 1.0, 2.0]

Trials for lambda DA contribution [0.1, 0.25, 0.5, 0.75, 1.0]

Method for choice Grid search

A.2 Human Evaluation details

The table below shows the full results of the human evaluation on a randomly selected subset of 245 *unseen* environment dialogues. Questions asked:

Q1 Which dialog is more grammatically correct?

Q2 Which dialog is more fluent?

Q3 Do the two dialogues seem to have the same goal?

Model	% Favored		% Same Goal	
	Q1	Q2		Q3
RMM	35.0	37.0	Same	49.6
Baseline	42.5	39.8	Diff	50.4
No Difference	22.4	23.2		
RMM	11.0	17.3	Same	44.9
Human	76.8	67.7	Diff	55.1
No Difference	12.2	15.0		
Baseline	12.2	20.1	Same	45.7
Human	78.0	63.8	Diff	54.3
No Difference	9.8	16.1		

A.3 Conversation Examples

Full human and baseline conversations for Figure 6.

	Conversation	GP	
Human	<ul style="list-style-type: none"> 👤 Do I go in between the ropes to my right or straight forward? 👤 straight forward through the next room 👤 Should I proceed down the hall to the left of turn right? 👤 head down the hall to your right into the next room 👤 Should I go through the open doors that are the closest to me? 👤 You are in the goal room 	<p>0</p> <p>13.31</p> <p>29.05</p>	
	Baseline	<ul style="list-style-type: none"> 👤 ok now which way ? 👤 keep going forward and then into a bedroom on your left . it ' s to the left of the bed . 👤 do i continue going to the right into the living room ? 👤 go into the door in front of you , then an right , the bathroom ' s there . 👤 do i keep going to the right into the living room ? 👤 it looks like that is the goal room . 👤 should i go into the laundry room to my right , or do i need to go in ? 👤 i see many ways to tell . you just go around . you ' re in the right room . 👤 ok now which way ? 👤 go into the room near you . that should be the goal 👤 okay i see any set of stairs . should i go up them ? 👤 turn right , then another right , the bathroom ' s there . 👤 ok . can you go more directions . i see many rooms on this floor . 👤 turn right , then another right , the bathroom ' s there . 👤 should i go the room ? or into the door to my left ? 👤 turn right , then another right , the bathroom ' s there . 👤 should i stay in this room or go out ? 👤 turn right , then another right , the bathroom ' s there . 👤 ok , should i go in this direction ? 👤 turn right , go past the pool table , into the room just towards the stairs . that should be the goal . 👤 should i go left toward the room , or past the painting and straight ? or to my right ? 👤 turn right , then another right , the bathroom ' s there . 👤 should i take a left here in the room ? 👤 that is the goal room 👤 should i stay in this room or go out ? 👤 you see the front , that is the room . its goal room 👤 the room ? or into the door to my left ? 👤 go up the stairs in front of you , then to your left , you ' ll see a bathroom . i think that bathroom is the goal 👤 should i go left toward the room , or past the painting and straight ? or to my right ? 👤 go towards the table in front of me of the hallway after take a right , then make a left into the sitting room . 👤 should i go to the right towards the blue painting ? 👤 continue going up , go to the bedroom 👤 should i go into the door on the left , or keep going forward ? 👤 go into the bedroom near you . that should be the goal 👤 do i need to go up ? 👤 go out in the room , make a right past the blue painting and go up the stairs 👤 ok now which way ? 👤 go up the spiral stairs again when you ' go into that bedroom straight ahead . 👤 should i go outside ? 👤 a bathroom on pretty left , go in it and then into the bathroom in that bedroom 	<p>4.0</p> <p>5.7</p> <p>8.8</p> <p>7.5</p> <p>4.0</p> <p>0.4</p> <p>0</p> <p>4.0</p> <p>8.8</p> <p>9.9</p> <p>7.5</p> <p>5.7</p> <p>8.8</p> <p>13.3</p> <p>9.3</p> <p>9.3</p> <p>11.1</p> <p>8.8</p> <p>7.5</p> <p>8.8</p>

Data Augmentation and **RMM** conversations for Figure 6.

	Conversation	GP
DA	<ul style="list-style-type: none"> 👤 should i go into the room? 👤 you are in the goal room. 👤 should i go into the room? 👤 you are in the goal room. 	5.66
	<ul style="list-style-type: none"> ... (repeats until maximum trajectory length is reached) 	0.03
	<hr/>	
RMM	<ul style="list-style-type: none"> 👤 should i head forward or bedroom the next hallway in front of me ? 👤 yes , all the way down the small hall . 👤 should i turn left here ? 👤 head into the house , then you will find a doorway at the goal staircase . go through the doors before those two small exit chairs , about half way down the hall . 👤 lots of sink in this house , or wrong did . ok which way do i go 👤 go down the hallway , take a left and go down the next hallway and up the stairs on the right . 👤 right place now ? 👤 turn left and when you reach to the bathroom . before these go up the stairs on your left , and walk passed the mirror and chairs on the left into hallway across from the stairs 👤 do i go left , right , or up the stairs ? 👤 turn to your left and keep going straight 👤 do i make a right turn or shall i enter the room to my left ? 👤 turn left to go down the hall , then that fire bed has that 👤 left or continue ? 👤 yes , go past the dining table and take an immediate right . head through the small door to the left of the window before those two way doors behind you , go up the second small set of stairs . 👤 should i go downstairs or turn left ? 👤 go to the right and go to the front doorway . 👤 should i go down the stairs or stay on this floor ? 👤 take a left towards the bathroom , then take a left before it and go all the way down the hall 👤 do i go up these is to the right or right of the steps ? 👤 go to the left side of the staircase and turn left in the doorway before the two small office chairs , about half way down the hall . 👤 should i turn left , go straight into the living room , or up the stairs ? 👤 turn to your right and go straight down the hall 👤 do i go out into the hallway ? 👤 go left down the hall where the office floor . and pass the second door after a right and table . 👤 ok , should i go right or left next ? 👤 go back to the staircase . go through the doorway you and before the hallway on the right . 👤 do i make a left turn or shall i enter the room to my left ? 👤 go down the hall and turn right into the bedroom 👤 should i go to the left or the right ? 👤 yes , go out of this room , turn right and go down the white hall before the staircase stairs , then go down the way down that way you get . 👤 ok i was a in by this office painting , or i just in the second hallway in front of me ? 👤 okay . 👤 which way do i go in , or do i head up the stairs ? 👤 go all the way to the one of the staircase . turn left in the doorway before the two two office chairs , about half way down the hall . 👤 ok wrong far which way do i go 👤 right then at the top of the stairs . 👤 left or continue ? 👤 yes . go down the hall and stop at the landing of the stairs . 	4.0
		5.7
		8.8
		7.5
		4.0
		0.4
		0
		4.0
		8.8
		9.9
		7.5
		5.7
		8.8
		13.3
		9.3
		9.3
		11.1
		8.8
		7.5