The RobotSlang Benchmark:
Dialog-guided Robot Localization and Navigation

Shurjo Banerjee[1]   **Jesse Thomason**[2]   **Jason J. Corso**[1]
[1]The University of Michigan   [2]University of Washington
{shurjo, jjcorso}@umich.edu   jdtho@cs.washington.edu

Abstract: Autonomous robot systems for applications from search and rescue to assistive guidance should be able to engage in natural language dialog with people. To study such cooperative communication, we introduce Robot Simultaneous Localization and Mapping with Natural Language (RobotSlang), a benchmark of 169 natural language dialogs between a human DRIVER controlling a robot and a human COMMANDER providing guidance towards navigation goals. In each trial, the pair first cooperates to localize the robot on a global map visible to the COMMANDER, then the DRIVER follows COMMANDER instructions to move the robot to a sequence of target objects. We introduce a Localization from Dialog History (LDH) and a Navigation from Dialog History (NDH) task where a learned agent is given dialog and visual observations from the robot platform as input and must localize in the global map or navigate towards the next target object, respectively. RobotSlang is comprised of nearly 5k utterances and over 1k minutes of robot camera and control streams. We present an initial model for the NDH task, and show that an agent trained in simulation can follow the RobotSlang dialog-based navigation instructions for controlling a physical robot platform. Code and data are available at https://umrobotslang.github.io/.

Keywords: Benchmark, Robot, Visual Navigation, Natural Language, Dialog

1   Introduction

Language is a natural medium for people to communicate with and direct robots. Research on language-guided robots lets users specify high-level goals like *Push the full barrel* [1, 2] and lower-level instructions like *After the blue bale fly to the right towards the small white bush* [3, 4]. Meanwhile, commercial dialog-enabled smart assistants are imbued with expanded language capabilities, but have limited interaction with the real world. Dialog-enabled robots combine these strengths, facilitating task completion [5], task learning [6, 7], and language learning [8].

Two related skills are needed across potential robot applications, from search and rescue missions to assistive guidance in an office building: localization and navigation. For a non-expert user, dialog facilitates both. The user can ascertain *where* the robot is by making requests like *Describe your surroundings* and give instructions for where to go next, such as *go toward the edge of the maze*. There is an array of benchmarks for vision-and-language navigation (VLN), where an agent learns to follow such language directions in simulated environments [9, 10, 11, 12].

Models optimizing performance on simulation-only benchmarks do not consider physical robot navigation limitations, employing unrealistic strategies like beam search [13], repeatedly creating panoramas from egocentric cameras [14], or assuming pre-exploration of the environment [15]. Where simulation allows the collection of such large-scale benchmarks, efforts in learning-based robotics are limited by expensive data collection and require more robust, sample-efficient models.

To address this gap, we introduce Robot Simultaneous Localization and Mapping with Natural Language (RobotSlang), a benchmark of human-human cooperative trials for controlling a physical robot to visit object goals while communicating in natural language (Figure 1). RobotSlang data is collected on a physical robot platform and is directly applicable for training a language-guided robot to follow human language instructions, which can otherwise require careful transfer techniques [16].
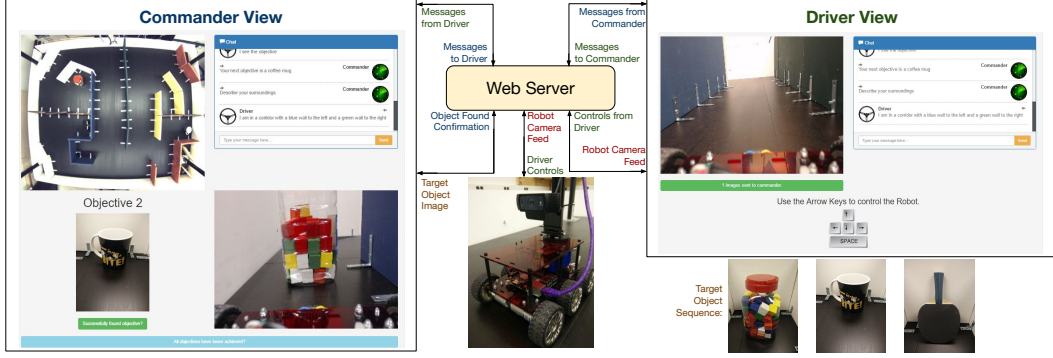
Figure 1: The COMMANDER and DRIVER communicate through a web application. The COMMANDER sees a static map and an image of the next target object (left). The DRIVER sees the first-person camera feed from the robot (a small rover controlled by a raspberry pi with an attached HD webcam) and can send the robot movement commands (right). Participants cooperate through a chat interface to guide the robot to a sequence of three target objects (bottom right). We synchronize, capture, and annotate the raw dialog text, sensor observations, and map data to create the RobotSlang benchmark.

2   Related Work

A learning agent reacting to human language instructions should be *embodied* in the real world and be able to carry on *dialog* with human users [17]. To this end, RobotSlang combines aspects of two research efforts: language-guided virtual agents in simulation and situated, human-robot dialog for guiding robot behavior. Table 1 summarizes the key differences between RobotSlang and comparable benchmarks across the language, vision, and robotics communities. RobotSlang provides a resource for studying how humans use language to cooperatively control a physical robot. As with initiatives like the Duckietown [18] and MuSHR [19], our physical robot setup can be recreated in other labs to make use of RobotSlang data for training and evaluation.

Language-Guided Virtual Agents.   Given a natural language instruction, an agent attempts to *ground* [20] this language to its visual surroundings. Early environments used simple texture renderings [21, 9], but advances in simulation and scene capture have lead to photorealistic indoor [10] and outdoor [12] spaces for VLN and general video understanding [22]. Some benchmarks include object manipulation and state changes, creating a more task-oriented setting [23].

Moving beyond static instructions, some benchmarks are created via human-human dialog, where a DRIVER moves the agent and asks questions that a COMMANDER answers. By enabling an agent to ask questions and gather more information during navigation [24, 25], ambiguous and underspecified commands can be clarified by a human interlocutor [26, 11, 27]. Some of these benchmarks require global localization [26], where the COMMANDER does not initially know enough about the location of the DRIVER to provide instructions. These benchmarks are all limited to simulations, while deployed robots operate in noisier, real-world environments. Our RobotSlang benchmark, by contrast, was gathered by pairing a human DRIVER controlling a physical robot and asking questions of a human COMMANDER, where the pair need to perform cooperative global localization while carrying out a navigation task to multiple object targets.

Language-Guided Physical Agents.   Physical robots engaged in task-oriented behaviors with human collaborators benefit from a natural language interface [29]. For example, human language commands can be combined with visual sensory input [28] to learn a controller for a quadcopter platform in an end-to-end fashion [30, 3]. Using natural language and mixed reality can allow users to control quadcopters with little training data [31].

Collaborative dialog can enable a robot to acquire new skills [6] and refine perceptual understanding [8] through language interaction. However, generating grounded language requests is often achieved through carefully controlled dialog managers [32] and generation semantics [5]. Learning to generate language from grounded, human-human dialogs [26, 33] such as those in the RobotSlang benchmark is an ambitious alternative for future work.

Comparable Efforts	Agent Type	Localization Task	Navigation Task	Questions Allowed	Human Answers	Sensor Observations
MARCO [21, 9]	**Virtual**	✗	✓	✗	-	**Render**
VLN [10, 12]	**Virtual**	✗	✓	✗	-	**Photoreal**
DRIF [28, 3]	**Physical**	✗	✓	✗	-	**Camera RGB**
DUIL [4]	**Physical**	✗	✓	✗	-	**Full Map**
VLNA [24, 25]	**Virtual**	✗	✓	✓	✗	**Photoreal**
MRDwH [11]	**Virtual**	✗	✓	✓	✓	**Render**
CVDN[27]	**Virtual**	✗	✓	✓	✓	**Photoreal**
Talk the Walk [26]	**Virtual**	✓	✓	✓	✓	**Photoreal**
RobotSlang	**Physical**	✓	✓	✓	✓	**Camera RGB**

Table 1: Compared to existing efforts involving vision and language input for controlling navigation agents, RobotSlang is the first to be built from human-human dialogs between a COMMANDER and DRIVER piloting a physical robot. RobotSlang also involves both a global localization and navigation task, where many previous efforts involve only navigation.

3 Robot Simultaneous Localization and Mapping with Natural Language

In Robot Simultaneous Localization and Mapping with Natural Language (RobotSlang), humans take on the roles of COMMANDER and DRIVER to guide a robot through a 2.5 by 2 meter tabletop maze to visit a series of three target landmark objects (Figure 1). In total, RobotSlang is comprised of 169 trials with associated dialogs and robot sensor streams carried out in 116 unique mazes.

Information and control asymmetry drives natural language communication to finish tasks, and a leaderboard was used to rank human-human teams and encourage efficient task completion (Section 3.1). The COMMANDER viewed a top-down, static map of the environment that included the landmarks to be visited, but could not see the robot. The DRIVER had access to the robot's sensor streams, including its front-facing camera, but did not have access to a global map. The participants communicated only over a text chat interface. First, the DRIVER needed to provide information about their surroundings to the COMMANDER to perform Localization from Dialog History (LDH). Then, the COMMANDER provided directions to the DRIVER. The DRIVER could ask questions throughout. Previous dialog-based navigation work forced strict turn taking and involved only one target per dialog [26, 27]. In RobotSlang, communication is at-will, and each trial involves visiting a sequence of three target objects. These differences lead to complex dialog phenomena (Section 3.2).

3.1 Data Collection

Each dialog in RobotSlang facilitated control of a small robotic car through a tabletop maze to visit a sequence of landmarks.[1] Throughout each trial, the robot collected front-facing camera stream data. The COMMANDER and DRIVER, although physically separated in different rooms, were connected via a web interface, shown in Figure 1. The DRIVER viewed a real-time stream from the robot camera, while the COMMANDER viewed a static, top-down RGB image of the maze and a photo of the next landmark object to visit. Team members communicated via a text chat interface. A central server hosted the web application and saved synchronized sensor and text dialog data streams.[2]

Sixteen participants were recruited to create RobotSlang. Participants were not familiar with the robot platform and were students from the university at which this research was conducted. The participants formed 41 teams, with each participant involved in 5 teams and 21 trials on average. To motivate participant teams to do well on the task, each trial was scored, and teams competed on a leaderboard. Teams were scored by overall time taken to visit all landmarks, with a small penalty for each motion command the DRIVER sent to the robot. The same maze was never presented to a team twice, and each team performed an average of 4 trials, resulting in 169 total trials. Dialogs lasted an average of six and a half minutes.

---

[1]The infrastructure to collect this data, including software and details of robot hardware and materials to construct physical mazes, is available at `https://umrobotslang.github.io/`.

[2]See Supplementary Material for trial videos and details about the web application and tabletop maze setup.

	Freq (%)	Examples
Needs History	53.0	COM: Go back to your initial position
		COM: now, go back to the location where you found the jar at first
		COM: go back to the location where you turn right first
Communication Repair	26.5	COM: Turn right before you go straight DRI: what? | Turn right where?
		COM: are you sure the blue walls are on your right? DRI: left | sorry
Re-localization Repair	14.7	COM: i asked you to turn right at the brown wall, and take the right at the white edge of the maxe COM: where are you right now?
		DRI: after the red wall or before the red wall? COM: before | Give me your position
Surrender Control	11.8	COM: your third objective is the ping-pong-paddle COM: can you go there on your own? DRI: yes
		COM: ok let me know if you get lost [end of dialog]
Perceptual Differences	2.9	COM: then move forward and make a right uturn around the orange walls DRI: it is red COM: correct, sorry I keep forgetting

Table 2: Dialog phenomena we annotated, along with the estimated frequency and examples from a 34 dialogs randomly sampled for annotation (20% of the total trials). We use the symbol | to separate short, sequential messages sent from the same speaker.

3.2 Dialog Data Analysis

The dialogs in RobotSlang are long. An average of 28 messages was sent per dialog, and dialogs averaged 200 words. By contrast, dialogs in the closely related, simulation-only CVDN task [27] contain an average of only 81.6 words per dialog, less than half the verbosity in our trial data.

Table 2 gives examples of dialog snippets from RobotSlang that exhibit complex phenomena. Because each trial involves visiting a sequence of objects, over 50% of dialogs involve COMMANDERs referring to past locations in their instructions. Successful modeling may require explicit memory structures. Humans make mistakes, both when giving instructions and when following them, but communication also facilitates repairing those mistakes. Over 25% of the dialogs in Robot-Slang exhibit mistakes and repair. Some of these mistakes require a new localization step, with the COMMANDER requesting location information again during navigation in 15% of dialogs. However, Humans also trust one another to follow "commonsense" routes and directions, with COMMANDERs surrendering control to DRIVERs in around 10% of dialogs. Finally, because the COMMANDER and DRIVER see different views of the environment, participants also need to mediate their perceptual differences explicitly in some dialogs, as studied in human-robot communication [34].

3.3 Replay-Simulation Environment

In lieu of training on the physical tabletop environment, we design a custom simulator (Figure 2). There are two aspects of the real world we simplify when creating the replay simulation environment: image observations and robot position. We annotate trial maps as occupancy grids where grid values represent corresponding, discretized pixel colors from maze walls. Grid points are 0.07 meters apart. The simulated agent travels between adjacent nodes using `forward` actions. The agent can change its heading with `left` and `right` actions of 45 degrees each. We precompute all shortest paths between navigable points for use during model training (Section 5.2). For training, we treat the shortest path between the robot's starting position and each target visitation object in turn as the human navigation path. We use the Floyd-Warshall algorithm [35] for shortest path planning.

We represent front-facing camera observations as a set of rays returning color and depth information. In particular, the agent's observation at every timestep consists of, for each ray, a discrete wall color, a distance to the wall, and whether the wall is too close to move towards (Figure 2). Thirteen equally
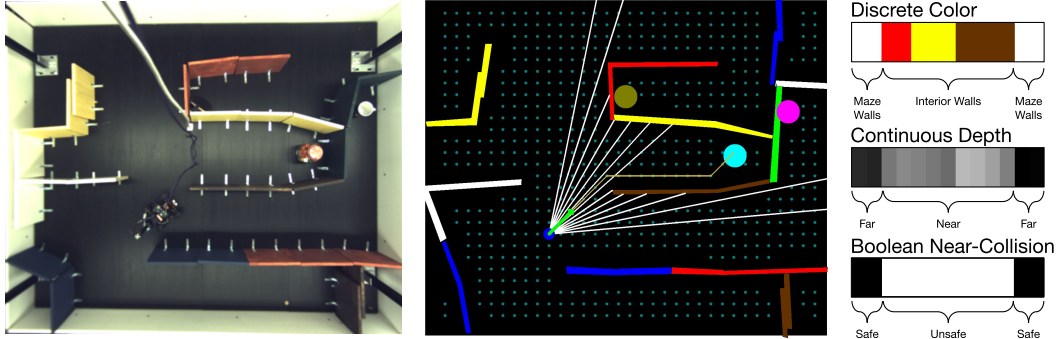
Figure 2: **Left:** The real-world, top down view of a RobotSlang maze. **Center:** The replay-simulation view of the same maze, with discretized navigable points and vision based on ray tracing. The three objects are are given unique colors to differentiate them from the rest of the maze, and rays are shown here in white. **Right:** Sample visual input in the replay simulator, showing the colors, depths, and collision possibilities for 13 rays out of the front-facing camera.

Fold	# Mazes	# Trials	# LDH	# NDH	% Dataset by Trial
Train	79	120	120	360	69
Val	19	26	26	78	15
Test	20	28	28	84	16

Table 3: Fold summaries in the RobotSlang benchmark, including the number of mazes, trials, and task instances. We split by mazes, such that no two folds contain trials with the same maze.

spaced rays are cast from the front of the agent, encompassing a field-of-view of 78 degrees—the same as the physical robot. We use Brensenham's line algorithm [36] to perform ray tracing.

The replay simulation allows training and testing models for tasks derived from RobotSlang. These models may be transferable to the physical platform for either fine-tuning with real-world camera input or preprocessed image observations using the simulator-familiar ray tracing method.[3]

4 Tasks

RobotSlang provides a resource for approaching numerous problems in human-robot communication, from query generation—learning how to ask good questions—to language-driven dynamics prediction—learning to predict the DRIVER's actions in response to language. While some work attempts to create agents that mimic both the COMMANDER and DRIVER, these methods are applied exclusively on data gathered in simulation and without regard to the need for sample efficiency with physical limitations on deployment [26, 33]. We instead focus on two core problems for any physical robot collaborating with a human giving language directions: Localization from Dialog History (LDH) and Navigation from Dialog History(NDH). We split the data into training, validation, and test folds (Table 3). Figure 3 give example LDH and NDH instances.

Localization from Dialog History (LDH)   We create a benchmark for learning models that perform Localization from Dialog History, emulating the human COMMANDER. We create one LDH instance per trial. For each trial, we annotated when initial localization was complete, before navigation began. A model receives as input the dialog history between the COMMANDER and DRIVER from the trial start to that hand-annotated end of localization. Given this information, the model must predict the location of the robot on the global map, just as the human COMMANDER does before giving navigation instructions.

---

[3]Due to COVID-19, we were not able to validate this transfer on the physical platform. As noted in the Supplementary Material, we validate the effectiveness of our ray-tracing measurements by utilizing them to successfully localize the robot in the RobotSlangtrials from the COMMANDER perspective.
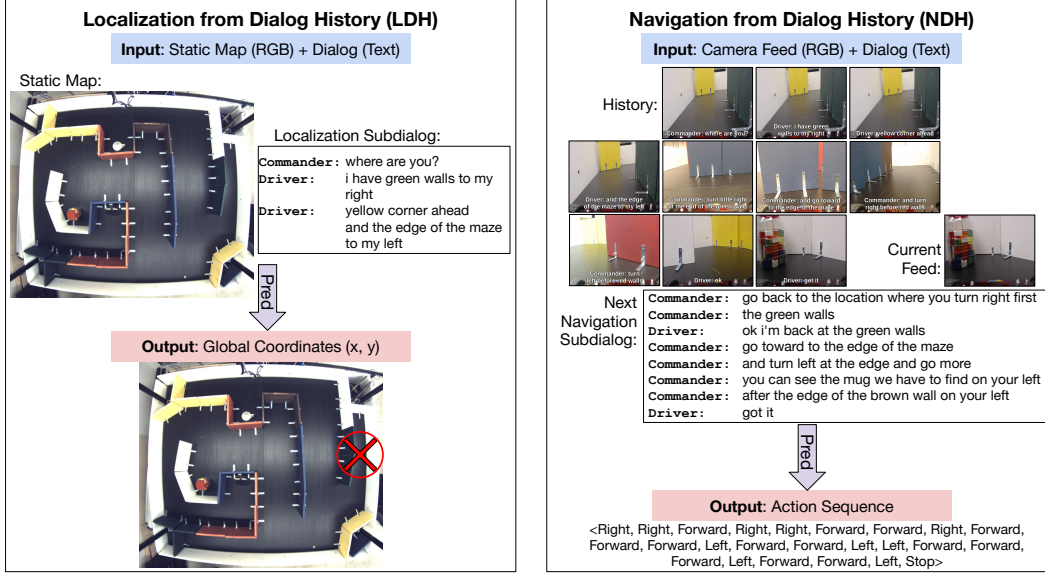
Figure 3: Instances of the LDH task (left) and NDH task (right). These tasks represent key skills of the COMMANDER and DRIVER: localization and navigation given natural language context. In both tasks, models have access to the dialog history so far as input. In LDH, models also see the global static map, and must predict where the robot is based on the dialog. In NDH, models also see the navigation history so far and the dialog snippet that guided the human-human pair to the next object, and must predict the navigation actions to reach that object.

As exhibited in Table 2, over 10% of trials require explicitly re-localizing when participants realize they have mismatched assumptions. In general, localization continues to happen in "soft" ways through the dialog, with the COMMANDER and DRIVER offering frequent sanity checks like *You should then have blue on your left and white in front of you* and *i have red walls ahead*. Thus, models may need to explicitly continue performing localization after this initial LDH step, which we leave for future work. After initial localization, navigation can begin.

Navigation from Dialog History (NDH)   We create a benchmark for learning models that perform Navigation from Dialog History, emulating the human DRIVER. For each trial, we create an NDH instance for each of the three objects to be visited. For each NDH instance, the robot begins at the trial initial position or in front of the last object visited, and must navigate to the next object in the visitation sequence. A model receives as input the robot navigation history, the dialog history between the COMMANDER and the DRIVER so far, and the dialog that guided the DRIVER from the starting location to the goal location. Given this information, the model must predict the controller actions taken by the DRIVER to move the robot according to the given language instructions. Actions are discretized to: `forward` (0.07 meters), `left` (45 degrees), `right` (45 degrees), and `stop`. We evaluate NDH performance based on the final position $(\hat{x}, \hat{y})$ of the robot.

Metrics.   Both the LDH and NDH tasks involve predicting a location $(\hat{x}, \hat{y})$ on the map, either directly for localizing the robot or indirectly as the result of a sequence of navigation actions. We evaluate performance with a Topological Distance (TD) metric that measures the shortest navigable path between the true position (x^*, y^*) and the predicted position $(\hat{x}, \hat{y})$. Concretely, $TD = |\text{shortestpath}((\hat{x}, \hat{y}), (x^*, y^*))|$. To analyze how much error comes form predicting when to `stop`, we also measure agent performance under *Oracle Stopping* [37], how well the agent would have done if it had stopped at the optimal time along its predicted path P. Oracle stopping distance OTD is calculated as $OTD = \min_{(x,y) \in P} |\text{shortestpath}((x, y), (x^*, y^*))|$ for P the predicted path.
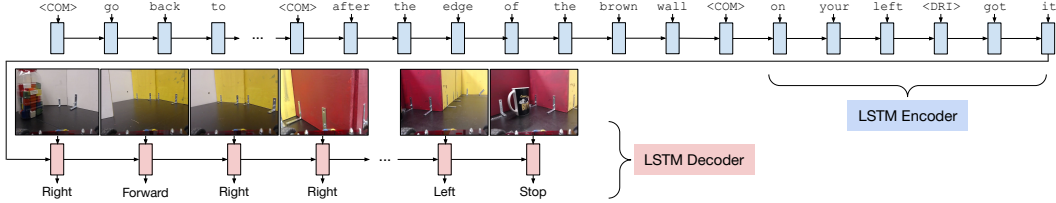
Figure 4: Our initial Sequence-to-Sequence model for the NDH task. An LSTM encoder takes as input sequence of GloVe-embedded language tokens representing the dialog history. The encoder LSTM initializes the hidden state of an LSTM decoder, which takes in an encoded visual observation from the robot's (or replay simulation agent's) front-facing camera and predicts an action.

5 Experiments

We evaluate human performance on the LDH task and create an initial, learned model for the NDH task which we evaluate in the replay simulation environment. We find that humans perform well on the LDH from reading dialog histories. Our initial model for the NDH outperforms simple baselines and ablations, indicating that the RobotSlang can be used to learn language navigation policies.

5.1 Localization from Dialog History.

The LDH task involves making a discrete prediction for the location of the robot given the COM-MANDER's static map view and the dialog history. However, human COMMANDERs do not necessarily need the precise location of the robot, and may give instructions that are valid for a distribution around their belief of where the robot *probably* is.

To assess the difference between humans' ability to make these discrete predictions and the physical locations of the robots, we conducted a human study with 8 participants who were not involved in the collection of RobotSlang. Human predictions for the robot location were, on average over all LDH instances, only 0.446 meters away from the true robot center by TD. This performance establishes an upper bound for future modeling attempts. We note that while humans are not perfect at pinpointing the robot's location, their guesses are often within two robot body lengths of the true center, since the robot is about 0.254 meters long.

5.2 Navigation from Dialog History.

Following closely related work in simulation-only, human-human dialogs for navigation [27], we focus our initial modeling efforts primarily on the NDH task. Given a dialog between a human COMMANDER and DRIVER, the robot agent must infer the next navigation actions to take, using the DRIVER's next actions as supervision during training. Because the RobotSlang data was gathered on a physical robot platform, the RobotSlang benchmark can be trained and evaluated in both our replay simulation (Section 3.3) and in the real world.

Initial Sequence-to-Sequence Model. We develop an initial, Sequence-to-Sequence (S2S) model, summarized in Figure 4. This model mirrors initial S2S models used to evaluate prior work in vision-and-language navigation [10, 27]. The dialog history is tokenized and words are embedded using GloVe-300.[4] Special tokens, <COM> and <DRI>, with random initial embeddings, are added to the sequence to indicate the speaker. An LSTM model is used to encode this token sequence, and its final hidden state initializes an LSTM decoder model. The LSTM decoder observes an image I from the robot's front-facing camera, and predicts an action $\hat{a}$. Under a *teacher forcing* curriculum, at training time the ground truth action $a*$ from the human trial is taken. Under a *student forcing* curriculum, at training time an action $\hat{a}$ is taken after sampling from the predicted logits [10]. All parameters are learned end-to-end by minimizing a cross entropy loss between predicted actions and the true action. At inference time, predicted action $\hat{a}$ is taken in the simulation environment. We train and evaluate NDH models using the replay simulation environment, approximating camera image observations with simulated ray tracing (Section 3.3).

---

[4] https://nlp.stanford.edu/projects/glove/

	Inputs		Validation TD ($\downarrow$)		Test TD ($\downarrow$)	
Training	Vis	Lang	True	Oracle	True	Oracle
Teacher	✓		3.77 ± 1.28	3.27 ± 1.21	3.67 ± 1.56	3.16 ± 1.50
Teacher		✓	4.01 ± 1.09	3.86 ± 1.06	4.03 ± 1.35	3.91 ± 1.31
Teacher	✓	✓	$*^L 3.68 \pm 1.28$	3.20 ± 1.27	3.76 ± 1.61	3.24 ± 1.51
Student	✓		3.42 ± 1.06	3.26 ± 1.11	3.32 ± 1.12	3.17 ± 1.10
Student		✓	4.04 ± 1.04	4.00 ± 1.03	4.10 ± 1.27	4.04 ± 1.26
Student	✓	✓	$*^{V,L} 3.27 \pm 1.12$	3.12 ± 1.13	$*^L 3.38 \pm 1.15$	3.20 ± 1.14
Random Action			4.32 ± 1.02	4.31 ± 1.02	4.31 ± 1.28	4.30 ± 1.28
Immediate Stop			4.31 ± 1.02	4.31 ± 1.02	4.33 ± 1.28	4.33 ± 1.28

Table 4: Sequence-to-Sequence model results on the NDH task under *student-* versus *teacher-* forcing, as well as under unimodal ablations [38]. The Topological Distance (TD) is the primary measure of performance—the length of the shortest path between the predicted final location for the robot and the true location. We also show performance under *Oracle Stopping*—the shortest distance achieved along the inferred navigation path. All metrics are in meters, and for all metrics lower is better. $*^{V,L}$ indicate when a model with access to both vision and language input statistically significantly outperforms the corresponding *V*ision-only or *L*anguage-only ablation.

Results. Table 4 summarizes our results on the NDH task. Our initial, sequence to sequence model outperforms baselines as well as unimodal ablations. Consistent with prior work [10], training with a student forcing regime makes the full model significantly[5] more robust to errors at inference time. The random baseline selects an action (except stop) at random up to a maximum number of steps, while the immediate stop baseline selects stop as the first action. These baselines provide context for the average distance agents need to cover to reach target objects in each NDH instance. In a unimodal ablation, the model is trained and tested with one modality (language or vision) empty. Our initial model does not exhibit substantial unimodal bias pathology [38]. The test fold performance of the vision-only model is not statistically significantly different from the full model.

6 Conclusions and Future Work

We introduce Robot Simultaneous Localization and Mapping with Natural Language (RobotSlang), a dataset of human-human dialogs for controlling a physical robot, and create associated localization and navigation task benchmarks. We present an initial model for Navigation from Dialog History that improves over simple baselines, demonstrating that language-driven instruction following can be achieved from real-robot, human-human dialog data.

Limitations and Future Work While RobotSlang data was collected on a physical robot platform, our Navigation from Dialog History evaluations were limited to our replay simulation environment. In the future, we will evaluate agents trained in the replay simulation on the physical robot platform. We are also interested in the possibilities of training on large scale vision-and-language navigation benchmarks and fine-tuning on RobotSlang as in-domain data for physical robot control. While the underlying visual and action data distributions differ, similarities in natural language can be leveraged to create more robust models.

Currently, we focus only on localization and navigation tasks, but dialog can involve training two agents to entirely fill the roles of COMMANDER and DRIVER as tried in simulation-only VLN datasets [26, 33]. Our initial model for NDH does not utilize explicit memory structures or historical visual features, but our analysis of dialog phenomena revealed that both are needed in over 50% of dialogs to resolve commands like *Go back to your initial position*. Further modeling attempts on the RobotSlang navigation task can make use of historical sensory context beyond language. We are also interested in exploring semi-supervised reinforcement learning for model training, which has been been successfully applied to multi-modal navigation tasks [39, 40].

---

[5]See the Supplementary Material for details on the statistical comparison between models.

References

[1] S. Patki, E. Fahnestock, T. M. Howard, and M. R. Walter. Language-guided semantic mapping and mobile manipulation in partially observable environments. In *CoRL*, Osaka, Japan, 2019.

[2] J. Arkin, D. Park, S. Roy, M. R. Walter, N. Roy, T. M. Howard, and R. Paul. Multimodal estimation and communication of latent semantic knowledge for robust execution of robot instructions. *International Journal of Robotics Research (IJRR)*, 2020.

[3] V. Blukis, Y. Terme, E. Niklasson, R. A. Knepper, and Y. Artzi. Learning to map natural language instructions to physical quadcopter control using simulated flight. In *CoRL*, 2019.

[4] D. P. Barrett, S. A. Bronikowski, H. Yu, and J. M. Siskind. Driving under the influence (of language). *Transactions on Neural Networks and Learning Systems*, 29(7):2668–2683, 2017.

[5] S. Tellex, R. Knepper, A. Li, D. Rus, and N. Roy. Asking for help using inverse semantics. In *RSS*, 2014.

[6] J. Y. Chai, Q. Gao, L. She, S. Yang, S. Saba-Sadiya, and G. Xu. Language to action: Towards interactive task learning with physical agents. In *IJCAI*, 2018.

[7] K. Bullard, S. Chernova, and A. L. Thomaz. Human-driven feature selection for a robotic agent learning classification tasks from demonstration. In *ICRA*, 2018.

[8] J. Thomason, A. Padmakumar, J. Sinapov, N. Walker, Y. Jiang, H. Yedidsion, J. Hart, P. Stone, and R. J. Mooney. Jointly improving parsing and perception for natural language commands through human-robot dialog. *The Journal of Artificial Intelligence Research (JAIR)*, 67, 2020.

[9] D. L. Chen and R. J. Mooney. Learning to interpret natural language navigation instructions from observations. In *AAAI*, 2011.

[10] P. Anderson, Q. Wu, D. Teney, J. Bruce, M. Johnson, N. Sünderhauf, I. Reid, S. Gould, and A. van den Hengel. Vision-and-Language Navigation: Interpreting visually-grounded navigation instructions in real environments. In *CVPR*, 2018.

[11] M. Marge, S. Nogar, C. Hayes, S. Lukin, J. Bloecker, E. Holder, and C. Voss. A research platform for multi-robot dialogue with humans. In *NAACL*, 2019.

[12] H. Chen, A. Suhr, D. Misra, N. Snavely, and Y. Artzi. Touchdown: Natural language navigation and spatial reasoning in visual street environments. In *CVPR*, 2019.

[13] L. Ke, X. Li, Y. Bisk, A. Holtzman, Z. Gan, J. Liu, J. Gao, Y. Choi, and S. Srinivasa. Tactical rewind: Self-correction via backtracking in vision-and-language navigation. In *CVPR*, 2019.

[14] D. Fried, R. Hu, V. Cirik, A. Rohrbach, J. Andreas, L.-P. Morency, T. Berg-Kirkpatrick, K. Saenko, D. Klein, and T. Darrell. Speaker-follower models for vision-and-language navigation. In *NeurIPS*, 2018.

[15] H. Tan, L. Yu, and M. Bansal. Learning to navigate unseen environments: Back translation with environmental dropout. In *NAACL*, 2019.

[16] P. Anderson, A. Shrivastava, J. Truong, A. Majumdar, D. Parikh, D. Batra, and S. Lee. Sim-to-real transfer for vision-and-language navigation. In *Conference on Robot Learning*, 2020.

[17] Y. Bisk, A. Holtzman, J. Thomason, J. Andreas, Y. Bengio, J. Chai, M. Lapata, A. Lazaridou, J. May, A. Nisnevich, N. Pinto, and J. Turian. Experience grounds language. In *EMNLP*, 2020.

[18] L. Paull, J. Tani, H. Ahn, J. Alonso-Mora, L. Carlone, M. Cap, Y. F. Chen, C. Choi, J. Dusek, Y. Fang, et al. Duckietown: an open, inexpensive and flexible platform for autonomy education and research. In *ICRA*, 2017.

[19] S. S. Srinivasa, P. Lancaster, J. Michalove, M. Schmittle, C. Summers, M. Rockett, J. R. Smith, S. Chouhury, C. Mavrogiannis, and F. Sadeghi. MuSHR: A low-cost, open-source robotic racecar for education and research. *arxiv*, 2019.

[20] S. Harnad. The symbol grounding problem. *Physica D*, 42:335–346, 1990.

[21] M. MacMahon, B. Stankiewicz, and B. Kuipers. Walk the talk: Connecting language, knowledge, and action in route instructions. In *AAAI*, 2006.

[22] L. Zhou, Y. Kalantidis, X. Chen, J. J. Corso, and M. Rohrbach. Grounded video description. In *CVPR*, 2019.

[23] M. Shridhar, J. Thomason, D. Gordon, Y. Bisk, W. Han, R. Mottaghi, L. Zettlemoyer, and D. Fox. ALFRED: A benchmark for interpreting grounded instructions for everyday tasks. In *CVPR*, 2020.

[24] K. Nguyen, D. Dey, C. Brockett, and B. Dolan. Vision-based navigation with language-based assistance via imitation learning with indirect intervention. In *CVPR*, 2019.

[25] K. Nguyen and H. Daumé III. Help, Anna! Visual Navigation with Natural Multimodal Assistance via Retrospective Curiosity-Encouraging Imitation Learning. In *EMNLP*, 2019.

[26] H. de Vries, K. Shuster, D. Batra, D. Parikh, J. Weston, and D. Kiela. Talk the walk: Navigating new york city through grounded dialogue. *arXiv*, 2018.

[27] J. Thomason, M. Murray, M. Cakmak, and L. Zettlemoyer. Vision-and-dialog navigation. In *CoRL*, 2019.

[28] D. Misra, A. Bennett, V. Blukis, E. Niklasson, M. Shatkhin, and Y. Artzi. Mapping instructions to actions in 3D environments with visual goal prediction. In *EMNLP*, 2018.

[29] S. Tellex, N. Gopalan, H. Kress-Gazit, and C. Matuszek. Robots that use language. *The Annual Review of Control, Robotics, and Autonomous Systems*, 15, 2020.

[30] V. Blukis, D. Misra, R. A. Knepper, and Y. Artzi. Mapping navigation instructions to continuous control actions with position visitation prediction. In *CoRL*, 2018.

[31] B. Huang, D. Bayazit, D. Ullman, N. Gopalan, and S. Tellex. Flight, Camera, Action! Using Natural Language and Mixed Reality to Control a Drone. In *ICRA*, 2019.

[32] N. Danas, C. Finkelstein, T. Nelson, S. Krishnamurthi, and S. Tellex. Formal dialogue model for language grounding error recovery. In *Combining Learning and Reasoning @ RSS*, 2019.

[33] H. R. Roman, Y. Bisk, J. Thomason, A. Celikyilmaz, and J. Gao. RMM: A recursive mental model for dialog navigation. In *Findings of EMNLP*, 2020.

[34] C. Liu and J. Y. Chai. Learning to mediate perceptual differences in situated human-robot dialogue. In *AAAI*, 2015.

[35] R. W. Floyd. Algorithm 97: shortest path. *Communications of the ACM*, 5(6):345, 1962.

[36] J. E. Bresenham. Algorithm for computer control of a digital plotter. *IBM Systems journal*, 4 (1):25–30, 1965.

[37] D. Misra, J. Langford, and Y. Artzi. Mapping instructions and visual observations to actions with reinforcement learning. In *EMNLP*, 2017.

[38] J. Thomason, D. Gordon, and Y. Bisk. Shifting the Baseline: Single Modality Performance on Visual Navigation & QA. In *NAACL*, 2019.

[39] P. Goyal, S. Niekum, and R. J. Mooney. Using natural language for reward shaping in reinforcement learning. In *IJCAI*, 2019.

[40] V. Dhiman, S. Banerjee, B. Griffin, J. M. Siskind, and J. J. Corso. A critical investigation of deep reinforcement learning for navigation. *arXiv*, 2018.