COMMON SENSE AND SEMANTIC-GUIDED NAVIGATION VIA LANGUAGE IN EMBODIED ENVIRONMENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

One key element which differentiates humans from artificial agents in performing various tasks is that humans have access to common sense and semantic understanding, learnt from past experiences. In this work, we evaluate whether common sense and semantic understanding benefit an artificial agent when completing a room navigation task, wherein we ask the agent to navigate to a target room (e.g. "go to the kitchen"), in a realistic 3D environment. We leverage semantic information and patterns observed during training to build the common sense which guides the agent to reach the target. We encourage semantic understanding within the agent by introducing grounding as an auxiliary task. We train and evaluate the agent in three settings: (i) imitation learning using expert trajectories (ii) reinforcement learning using Proximal Policy Optimization and (iii) self-supervised imitation learning for fine-tuning the agent on unseen environments using auxiliary tasks. From our experiments, we observed that common sense helps the agent in long-term planning, while semantic understanding helps in short-term and local planning (such as guiding the agent when to stop). When combined, the agent generalizes better. Further, incorporating common sense and semantic understanding leads to 40% improvement in task success and 112% improvement in success per length (*SPL*) over the baseline during imitation learning. Moreover, initial evidence suggests that the cross-modal embeddings learnt during training capture structural and positional patterns of the environment, implying that the agent inherently learns a map of the environment. It also suggests that navigation in multi-modal tasks leads to better semantic understanding.

1 INTRODUCTION

Humans utilize semantic information and common sense knowledge when exploring unseen environments. For instance, we do not need to ask for detailed instructions to navigate to a restroom in a new restaurant. On the other hand, artificial agents find it challenging to perform grounded language navigation tasks (e.g."go to the kitchen") or embodied question answering (e.g."what color is the car?") in realistic 3D environments and, especially in unfamiliar environments, tend to fail miserably (Tangiuchi et al., 2019). We hypothesize that common sense and semantic understanding can benefit artificial agents in the same way that they benefit humans. For example, in the task of navigating around the house, common sense understanding can be helpful in long-term planning and setting the general course of the trajectory. When trying to go to the kitchen, it is useful to know that a dining room is likely close to the kitchen, and that a hallway is likely close to the dining room. On the other hand, semantic understanding (i.e. a deeper understanding of the layout of each room, objects that are usually in it, etc.) should help in choosing better local actions (such as "stop" when the target room is identified or "go forward" when the target room is in the vicinity of the current view). In this work, we evaluate the role of common sense and semantic understanding in embodied agents, using concept-driven navigation ($RoomNav$ (Wu et al., 2018b)) as a testbed.

With advancements in 3D simulated environments such as Habitat AI (Manolis Savva et al., 2019) or MatterPort3D (Chang et al., 2017), it is possible to train agents that can interact with these multi-modal environments and perform a variety of embodied tasks such as following instructions, answering questions, or navigating. Figure 1 depicts the $RoomNav$ task, wherein an agent is put in a realistic 3D house environment and is given an instruction to navigate to a target room by performing a sequence of actions: turn left, turn right, move forward, or stop. We focus on the *RoomNav* task

Figure 1: Panoramic view of the agent (left, right, and front views concatenated).

as it allows us to investigate several research questions related to grounded language understanding, such as (1) does common sense modeling help navigation? (2) does semantic understanding facilitate navigation and does navigation lead to better semantic representations? (3) can auxiliary tasks help in hard exploration problems? (4) is the agent able to approximate the layout of the environments?

We leverage the semantic information and patterns observed during training on the $RoomNav$ task (such as next room, and sequence of rooms observed along the trajectories) to build the common sense which guides the agent to reach the target room. We enforce semantic understanding by performing two auxiliary tasks: (i) grounding during navigation (asking the agent to predict the current and nearby rooms from current view as depicted in figure 1) (ii) grounding after navigation (asking questions such as “did you see a bathroom on your way). The common sense module, therefore, captures sequential and structural information of rooms and is used to guide the agent in a hierarchical fashion (Kulkarni et al., 2016; Wernsdorfer & Schmid, 2014). Inspired by Wang et al. (2018), we leverage the idea of self-supervised imitation learning (*SIL*), which is fine-tuning the agent on unseen environments using a cycle-reconstruction loss obtained by reversing the original problem and using it as a critic. Unlike previous works, we perform *SIL* by introducing auxiliary tasks related to semantic understanding. We teach the agent how to perform semantic understanding and then use that knowledge to make the agent get familiar with unseen environment. We also address the challenge of multiple targets (for example, a house can have multiple bedrooms) by modifying the loss and reward function (section 4), a problem which has been widely ignored in previous works by filtering out such scenarios to avoid ambiguity. lastly, we showcase that cross-modal embeddings trained with semantic and common sense understanding mimic structural and positional patterns, which helps in effective planning and overall navigation tasks.

2 Related Work

Embodied Tasks: Significant research has been done around visual and video question answering (Agrawal et al., 2017; Das et al., 2016; 2017b; Le et al., 2019; de Vries et al., 2017, among others). Recently, some work has been done around grounded language understanding (Harnad, 1999; Hermann et al., 2017), wherein an agent interacts with and navigates through a simulated 3D environment to complete some tasks such as room navigation, finding an object (Wijmans et al., 2019; Wu et al., 2018b), embodied question answering (Das et al., 2017a; 2018; Gordon et al., 2017; Manolis Savva et al., 2019; Mirowski et al., 2018), and following instructions (Anderson et al., 2017; Fried et al., 2018; Shah et al., 2018; Wang et al., 2018). Although researchers have significantly advanced the state of the art in these tasks, the fundamental question of how language and semantics facilitate navigation and how navigation helps facilitate semantic understanding is yet to be fully addressed.

Robustification through Reinforcement Learning: While Wu et al. (2018b) used Deep Deterministic Policy Gradient (DDPG, Heess et al. (2015)) and Asynchronous Advantage Actor Critic (A3C, Mnih et al. (2016)) to evaluate generalizability aspects of the agents for the $RoomNav$ task on the semantically rich House3D environment, their policies did not leverage any common sense or knowledge-grounded semantic information available in the environment.

Understanding and Common Sense: Hermann et al. (2017) analyzed the problem of how agents learn to interpret instructions and how they generalize in one-shot and multi-task settings through reinforcement and unsupervised learning using the DeepMind Lab framework. Their study show-

cased the importance of prior semantic knowledge and curriculum learning for improved generalization and faster task completion. These experiments were performed in simple one or two room settings (Beattie et al., 2016) with few objects in the environment. On the other hand, there are some environments which are realistic such as House3D (Wu et al., 2018a) or MatterPort3D (Chang et al., 2017) which constitutes of indoor houses with multiple rooms containing a wide variety of objects. However, limited amount of research has been done in these environments around understanding or common sense. Work done by Yang et al. (2019) is the closest research which has been done around the ideas that we propose. The authors used an external knowledge graph and fed in a Graph Convolutional Network (Kipf & Welling, 2017) to encode the knowledge which is then fed as priors for object navigation. In our work, in addition to implicitly incorporating common sense understanding, we teach the agent how to perform semantic understanding itself via auxiliary tasks, which helps the agent in two ways: (i) guiding the agent to help navigate throughout the trajectory (unlike target alone in case of prior work) and (ii) further fine-tune the agent on unseen environments through self-supervised imitation learning.

Our objective is not to build an agent which outperforms state-of-the-art in existing embodied task. The primary focus of our research is to evaluate if common sense and semantic understanding facilitate navigation and if navigation helps in better semantic understanding. We propose a variety of ways in which common sense and semantic understanding can be fed to the agent in order to perform exhaustive experiments to interpret how common sense and semantic understanding impact the navigation task. We choose *RoomNav* task because it involves planning and semantic understanding. *RoomNav* task has been addressed on Habitat environment Manolis Savva et al. (2019) previously, however, the data used is not accessible anymore. Moreover, we select MatterPort3D environment for our experiments which unlike Habitat is realistic in nature and is harder to address. To the best of our knowledge, we do not know of any other work related to $RoomNav$ on MatterPort3D environment and therefore we could not directly compare against previous work.

3 Agent Architecture

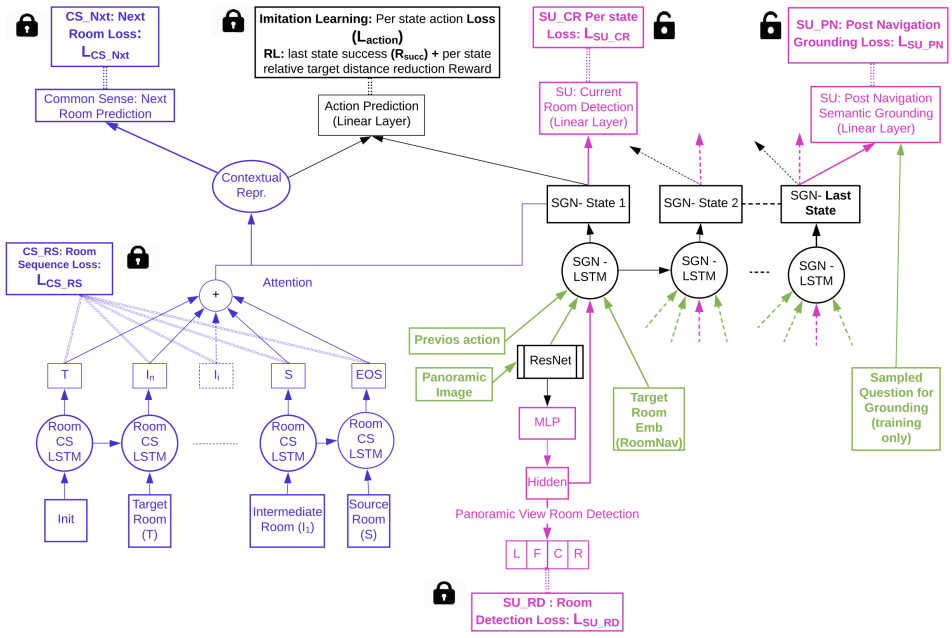


Figure 2: Common Sense and Semantic-Guided Navigation for *RoomNav* Task. Input to the agent is represented in green. Black components correspond to the baseline navigator model. Purple components are introduced to incorporate common sense planning while pink components are for Semantic Understanding.

Input: We have four kinds of inputs (shown in green in Figure 2): (i) task specific instruction (e.g. "go to the kitchen") (ii) RGB values for each state, (iii) previous action, and (iv) semantic

information such as room annotations or generating grounding questions. Semantic information, either for obtaining loss in semantic predictions or for generating questions for grounding, is used during training only as we are not aware of semantics on unseen environments. Following previous work on embodied navigation (Wang et al., 2018)), we extract panoramic image features using a fixed pre-trained ResNet-152 (He et al., 2015). The difference is that we turn the agent 90 degrees to the left and right, respectively, to get a 90-degree view for each direction (in a total of 270 degrees) at each timestep. We extract and concatenate images features and pass them through a single layer Feed Forward Neural Network to obtain the visual representation. We use RGB features only as opposed to using other sensors such as semantic masking features (Wu et al., 2018b) or depth features so that the agent will learn to perform understanding of the environment rather than memorizing segments or avoiding obstructions.

Baseline Model: We use an *LSTM* baseline following the work in Das et al. (2017a) for our experiments. The components shown in black color in figure 2 depict the baseline, which consists of an *LSTM* navigator. The navigator receives previous action, visual representation obtained from *ResNet*, and target room embedding as input, and predicts one of the four possible actions at each step. Rationale behind choosing a straightforward *LSTM* as baseline against a more state-of-the-art architecture is to evaluate how much common sense and semantic understanding contributes in task completion and to evaluate a variety of ways in which *CS* and *SU* can be incorporated within the agent.

3.1 Common sense and Semantically-Grounded Agent

Our architecture consists of the following three modules as depicted in Figure 2:

a. Semantically-Grounded Navigator (*SGN*): The navigator (shown in black in Figure 2) is an *LSTM* model which generates a state at each step that is used for multiple tasks: (i) action prediction to perform one of four possible actions at each step: go forward (0.25m), turn left (10 degrees), turn right (10 degrees), and stop (ii) semantic grounding via detecting the current room (*SU_CR*), and (iii) semantic grounding via post navigation grounding (*SU_PN*), that is generating a response in the final state when the agent predicts "stop". The *SGN* takes the $RoomNav$ instruction at each step (which remains static throughout the task), visual representation (which changes at each step) as input. We add a linear layer for each of the three tasks, however, for action prediction we also obtain the contextual representation by attending to the current *SGN* state over the generic sequence of rooms (between source and target room) generated by the common sense planning module (*CS_RS*). The attention helps the navigator move towards rooms which are closer to the target. Both the contextual representation and *SGN* state are used at each step for action prediction.

b. Common Sense Planning Module (*CS*): We argue that realistic house environments follow structural patterns (such as a refrigerator is usually placed in the kitchen) and sequential patterns (such as the kitchen is usually near the dining room). We incorporate this information via *CS* modules. For distant destinations, the agent may not be able to utilize a static instruction for route planning. Instead, we design a next room prediction module (*CS_Nxt*) to help the agent navigate to an intermediate target along the route. Next room prediction is trained during the imitation learning phase since during the *RL* phase the agent might deviate from optimal trajectories (the sequence of rooms in sub-optimal trajectories will not reflect common sense). Next room prediction is a function of the current state (obtained from *SGN*) and the sequence of rooms between the source and the target room. To incorporate this, the room sequence common sense model (*CS_RS*) is trained simultaneously to capture generic room sequence patterns (between source and target rooms) across the houses through sequences observed in training trajectories. However, such an intermediate target prediction model does not always have explicit information about what rooms are near the target room. Therefore, we design a backward room sequence model using the *LSTM* to generate sequences starting from the target room. We get the contextual representation by attending to the output of the backward room sequence and predict the next intermediate room.

c. Semantic Understanding (*SU*): To incorporate *SU*, apart from the two auxiliary tasks for the *SGN* described previously, we introduce another task for detecting the current and nearby rooms, using a separate Multi-layer Perceptron (*MLP*) room detection model (*SU_RD*). The *MLP* takes the representation of the panoramic view from *ResNet* as input and generates a hidden state before performing classification over four rooms (left (L), front (F), current (C), right (R)). We also pass

the hidden state of the *MLP* to the *SGN* for better action prediction, especially in cases where the panoramic image contains multiple rooms (as shown in figure 1). Regarding the two auxiliary tasks performed by *SGN*, we add separate output heads on top of *SGN*. We add a linear layer to perform current room detection (*SU_CR*) using the state obtained from *SGN* at each step and another linear layer to produce a "yes" or "no" response to the grounded question (e.g. "did you see a bathroom on your way?"). The questions - generated by sampling from the annotations and using templates for positive and negative cases - are used to incorporate semantic understanding (*SU_PN*) within the agent. The rationale behind having two models for current room detection (*SGN* and SU_RD) is that we can use the latter to fine-tune *SGN* on unseen environments as explained in section 4.

3.2 LEARNING

a. Training and Fine-tuning on Unseen Environments: We train the agent in three ways: (i) imitation learning with shortest path trajectories available during the training (ii) reinforcement learning, to further robustify the agent after imitation learning, using Proximal Policy Optimization (PPO) on the training environments, and (iii) self-supervised imitation learning on unseen environments, inspired by the work from Wang et al. (2018). Self-supervision is the reason we have two room detection models (*SU_RD* and *SU_CR*): we use the rooms detected from *SU_RD* as the ground truth for performing self-supervised imitation learning on unseen environments. We let *SU_CR* get further fine-tuned on unseen environments by obtaining the loss based on *SU_RD*'s output. Further, we sample grounded questions using the rooms detected by *SU_RD* on the trajectory to perform *SU_PN* on unseen environments. Losses obtained from these auxiliary tasks on unseen environments update the *SGN* which is also used for navigation, hence continues to update the semantic understanding on newer environments similar to what humans do.

b. Loss and Rewards: During imitation learning, apart from the main action prediction task, we perform five auxiliary tasks: (i) next room detection (*CS_Nxt*) (ii) target to source room sequence prediction (*CS_RS*) (iii) current room detection using *SU_CR* (iv) post navigation response generation (*SU_PN*) and (v) current and nearby room predictions (*SU_RD*). In total, we have six losses during imitation learning including action prediction. Equation 1 shows the sum of all the losses during imitation learning. Equation 2 corresponds to per-step loss for each task as a function of state and input. State in equation 2, corresponds to the *SGN* state of the *LSTM* once all the four inputs (target room embedding, image representation from *ResNet*, previous action and hidden state of room detection *MLP*) are passed through it, as represented in equation 3. During self-supervised imitation learning, we fine-tune the agent using two auxiliary tasks on unseen environments using the labels obtained from *SU_RD MLP* model. Equation 4 corresponds to self-supervised imitation learning loss, with losses (L') coming from the models. During the reinforcement learning phase, since we fine-tune the policy on the already seen environment, we freeze the *CS* and *SU* modules to avoid introducing noise. We only update the action layer and the *SGN* with per step reward shown in equation 5. We have a reward for agent getting closer to the target, a success reward, and a discounted future reward. We set the discount factor to be 0.99 in our experiments and success reward to be 10. All the λ's corresponds to hyper-parameters across the equations, c_t correspond to contextual representation as shown in Figure 2 and z_t correspond to state in *CS_RS LSTM*.

$$L_{imitation-learning} = \lambda_a * L_{action} + \lambda_{cs\_nxt} * L_{CS\_Nxt} + \lambda_{cs\_rs} * L_{CS\_RS} + \lambda_{su\_rd} * L_{SU\_RD} + \lambda_{su\_cr} * L_{SU\_CR} + \lambda_{su\_pn} * L_{SU\_PN} \tag{1}$$

$$\begin{gathered} l_{action} = l_{P(action_t|s_t)};\ \ l_{CS\_Nxt} = l_{P(CS\_Nxt_t|s_t,c_t)}; \\ l_{CS\_RS} = l_{P(CS\_RS_t|target,z_t)};\ \ l_{SU\_RD} = l_{P(SU\_RD_t|ResNet(pan\_img_t))}; \\ l_{SU\_CR} = l_{P(SU\_CR_t|s_t)};\ \ l_{SU\_PN} = l_{P(SU\_PN_t|s_{last})} \end{gathered} \tag{2}$$

$$s_t = SGN(action_{t-1},\ ResNet(pan\_img_t),\ MLP(pan\_img),\ target\_room\_emb) \tag{3}$$

$$L_{self-supervised-IL} = \lambda_{su\_cr} * L^{'}_{SU\_CR}\ + \lambda_{su\_pn} * L^{'}_{SU\_PN} \tag{4}$$

$$R(s_t, a_t) = \lambda\_td * (d_{t-1} - d_t)\ + \lambda\_suc * (succ\_reward) + \sum_{t'=t+1}^{T} \gamma^{t'-1} r(s_{t'}, a_{t'}) \tag{5}$$

4 EXPERIMENTS AND RESULTS

Data and Environment: We use the Habitat environment (Manolis Savva et al., 2019) with the MatterPort3D dataset for all of our tasks. Habitat's task is Point Navigation, where an agent needs to navigate from a source coordinate to a target coordinate. We adapt this task to form *RoomNav* by replacing the target coordinates with the corresponding 27 room types annotated in the dataset (excluding "other room"). After removing games where the target is in the same room as the source room and where the target is in the border of several rooms, we generated 53 houses (5020 games) for training, 11 houses for validation (168 games), and 15 houses (324 games) for testing. We use the same measure as Point Navigation task to define the complexity of each game by the geodisic distance of the shortest path. The average number of rooms between the source and target room is 2.41, 3.01, and 4.06, respectively for easy, medium, and hard games in the training data.

Evaluation Metrics: We use two metrics: (i) Success rate: rate of the games in which agent enters the target room. (ii) Success Per Length (*SPL*) which is a success metric normalized with respect to the shortest path (Wang et al., 2018). *SPL* requires that the agent chooses to terminate in the target room for the game to be considered successful. In our evaluation, we also calculate *non-stop SPL* where we relax this requirement of termination.

Model	SPL	non-stop SPL	average steps	succ. rate	easy succ. rate	medium succ. rate	hard succ. rate
baseline	0.067	0.239	119	0.253	0.412	0.246	0.175
CS_Nxt	0.120	**0.264**	137	0.303	0.441	0.285	**0.246**
CS_RS	0.078	0.236	114	0.256	0.426	0.2301	0.191
CS_Nxt + CS_RS	0.079	0.260	127	0.306	0.485	**0.300**	0.214
SU_CR	0.085	0.227	120	0.244	0.471	0.223	0.143
SU_RD	0.103	0.245	**114**	0.269	0.485	0.231	0.191
SU_PN	0.077	0.254	154	0.293	0.427	0.292	0.222
SU_PN + SU_RD (SIL baseline)	0.056	0.223	158	0.252	0.48	0.21	0.17
SU_CR + SU_RD (SIL baseline)	**0.141**	0.248	118	0.278	0.529	0.238	0.19
CS_Nxt + SU_CR	0.094	0.259	119	0.284	0.50	0.277	0.175
CS_Nxt + SU_PN	0.068	0.260	146	**0.315**	**0.574**	0.269	0.222
CS_Nxt + CS_RS + SU_CR	0.051	0.223	135	0.253	0.441	0.215	0.191
SIL: SU_PN + SU_RD	0.072	0.239	154	0.275	0.529	0.246	0.167
SIL: SU_CR + SU_RD	**0.142**	0.253	116	0.296	0.559	0.262	0.191

Table 1: Results on Imitation Learning and Self-supervised IL on Test Environments. Metrics: SPL - Shortest path normalized by length; *non-stop SPL* - Best SPL anywhere in trajectory.

4.1 IMITATION LEARNING

From Table 1, it can be observed that common sense planning and semantic understanding helps in Imitation Learning experiments across the board when compared to the *LSTM* only baseline.

Common Sense Planning: We incorporate common sense in two ways: (i) next room guidance (*CS_Nxt*) and (ii) generic room sequence between target and source room (*CS_RS*) as described in section 3. As anticipated, *CS* modules help in long-term planning hence helps in medium and harder games more than the easy games. Next room guidance alone leads to the second best SPL (80% improvement over baseline) and hard game success rate (40% improvement over baseline). Room sequence module (*CS_RS*) alone helps in harder games. When combined with (*CS_Nxt*), the performance on easy game improves significantly, while performance on medium games outperforms all other settings. Although, performance on harder games degrades relatively, which implies that it might be hard for the room sequence module to learn longer patterns.

Semantic Understanding: *SU* is performed in three ways: (i) *SGN* predicting current room (*SU_CR*) (ii) *SGN* performing grounding post navigation (*SU_PN*) and (iii) MLP predicting nearby rooms (*SU_RD*). As expected, *SU* generally helps in short-term planning through better semantic detection leading to higher SPL scores with least average number of steps required to complete

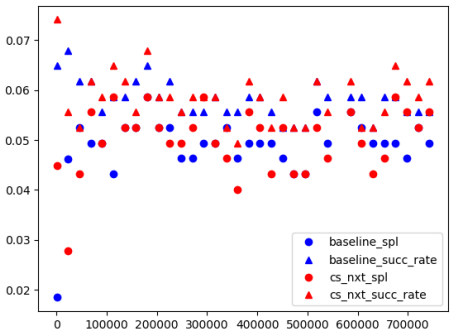
the task. However, (*SU_PN*) do not follow similar pattern, since grounding in case of (*SU_PN*) is performed in terminal state, hence it directly does not impact turn-level action prediction. In fact, it performs significantly better than most settings on medium and hard games because it lets the *SGN* focus more on action prediction during intermediate steps, while ensuring semantic understanding at terminal state. Early stopping could also be the reason why (*SU_RD*) and (*SU_CR*) individually might perform worse on medium and harder games. When combined together, we get the best *SPL* score (0.141, 110% better than the baseline. However improvement can be seen only on easy games. *SGN* in these settings might tend to focus more on auxiliary task than the action prediction task. We hypothesize that hyperparameter tuning is important in combining different modules.

Common Sense Planning and Semantic Understanding: *CS* when combined with *SU* leads to best overall success (25% relative improvement overall) when *SGN* is trained with (*CS_Nxt*) and (*SU_PN*) because former helps in long-term planning and the latter leads to better semantic detection guides the agent in effective stopping. As expected, most improvement is observed on easy games (40% relative improvement over baseline), performance improvement on medium games is intermediate, while that on hard games is second best. Adding (*SU_CR*) to (*CS_Nxt*) or (*CS_Nxt*) and (*CS_RS*) degrades the performance caused by early stopping. Moreover, it can be observed that when several auxiliary tasks are introduced to the agent, the performance generally degrades as the agent tends to focus more on the auxiliary tasks than the action prediction task.

4.2 Reinforcement learning: Preliminary Analysis

After bootstrapping the policy with imitation learning, we fine-tune the policy using *PPO*. We compare baseline model with the best performing model (*CS_Nxt*) on hard games. Figure 3 depicts the performance of both the models on hard games of test environment. From our initial analysis, we observed that agent trained with common sense generalizes better on unseen environments on success and *SPL* metrics. Although, performance of both the models degrade initially on the success metrics and improves on *SPL*, the results obtained with existing *RL* experiments should be considered as preliminary findings.

Figure 3: SPL and Success Rate vs. Number of frames used for training on hard games of test environment



4.3 Self-supervised Imitation learning (*SIL*)

To perform self-supervised imitation learning (*SIL*), we either use *SU_CR* or *SU_PN* as the auxiliary task to fine-tune *SGN* by assuming the output of *SU_RD* as the ground truth. We do not use *CS_Nxt* model for performing *SIL* because performance of *CS* modules degrades significantly when step level auxiliary tasks are introduced, as depicted earlier. After performing *SIL* for the first 20 steps on unseen environments, using the *SU_CR* for fine-tuning helps the overall accuracy by 6.5% with respect to *SU_CR* + *SU_RD* baseline. When *SU_PN* is used for fine-tuning instead, we achieve an improvement of 9.1% with respect to *SU_PN* + *SU_RD* baseline. With more *SIL* steps, performance degrades due to introduction of noise, as the current room prediction from the *SU_RD* model is not perfect/ground truth. Through *SIL*, we showcase that the agent can be made to update the semantic understanding through navigation.

4.4 Cross-modal Embeddings

Traditional word embeddings (e.g. Pennington et al. (2014)), are functions of words or semantic entities appearing in similar contexts and may not capture the visual and structural properties of entities in a realistic 3D world. Therefore, we use cross-modal embeddings, which are randomly initialized

Figure 4: Top view of house 17DR with dark areas as obstacles

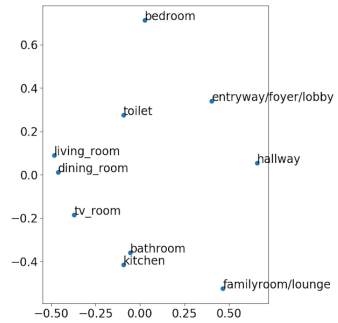


Figure 5: Embeddings of *SU_CR + SU_RD* model before fine-tuning on 17DR house

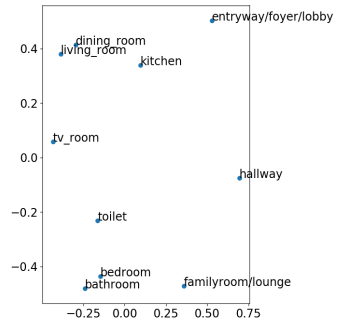


Figure 6: Embeddings of *SU_CR + SU_RD* model after fine-tuning using *SIL*

and then trained with common sense and semantic understanding across multiple games. We qualitatively analyze these embeddings to see if they reflect the structural and visual characteristics of the environment. Figure 4 represents the top-view of a training environment and Figure 5 visualizes the embeddings trained using the (*SU_CR + SU_RD*) model in two dimensions after aligning. We fine-tuned the agent (and embeddings) on the house shown in figure 4 using self-supervised imitation learning and visualize it in figure 6, after aligning it with respect to the original map for comparison. It can be observed that fine-tuned embeddings tend to mimic the structural and positional patterns of the house. Embeddings obtained before fine-tuning mimic the average structural pattern of rooms across all the houses, such as the dining room is close to the living room. Apart from positional characteristics, the cross-modal embeddings also mimic visual properties, for example the bathroom and the TV room are usually separated before and after fine-tuning, because these rooms are visually distinct from other rooms.

Cross-modal Embeddings as an alternative to SLAM: From this observation, we can conclude that after fine-tuning the agent on a given house using *SIL*, embeddings tend to mimic the layout of the house. These approximate maps obtained from embeddings can be an alternative to SLAM algorithm (Durrant-Whyte & Bailey, 2006). We leave detailed comparison to future work.

Navigation facilitates Semantic Understanding: The embeddings or semantic representations learnt by our approach contain information beyond what is captured in language, such as structural and visual characteristics, which is similar to how humans represent semantics in their mind. From the analysis performed above, we show that cross-modal navigation improves semantic understanding by capturing information from various modalities into the embeddings.

5 Conclusion

The only thing which humans have when they navigate in unseen environments is common sense and semantic understanding obtained through past experiences. In this work, we investigate if this also holds in artificial agents, by incorporating common sense and making the agent semantically aware while performing a room navigation task in realistic 3D environments. We showcased that the agent generalizes better if it is taught to perform common sense and semantic understanding. We introduced semantic grounding within the navigator through multiple auxiliary tasks, and showcased that the agent can be fine-tuned further to generalize better on unseen environments through auxiliary tasks using self-supervised imitation learning. We also showed that the cross-modal embeddings obtained during training tend to capture structural and positional patterns of the houses, implying that the agent learns better planning with common sense and semantic auxiliary tasks.

REFERENCES

Aishwarya Agrawal, Jiasen Lu, Stanislaw Antol, Margaret Mitchell, C. Lawrence Zitnick, Devi Parikh, and Dhruv Batra. VQA: visual question answering - www.visualqa.org. *International Journal of Computer Vision*, 123(1):4–31, 2017. doi: 10.1007/s11263-016-0966-6. URL https://doi.org/10.1007/s11263-016-0966-6.

Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian D. Reid, Stephen Gould, and Anton van den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. *CoRR*, abs/1711.07280, 2017. URL http://arxiv.org/abs/1711.07280.

Charles Beattie, Joel Z. Leibo, Denis Teplyashin, Tom Ward, Marcus Wainwright, Heinrich Küttler, Andrew Lefrancq, Simon Green, Víctor Valdés, Amir Sadik, Julian Schrittwieser, Keith Anderson, Sarah York, Max Cant, Adam Cain, Adrian Bolton, Stephen Gaffney, Helen King, Demis Hassabis, Shane Legg, and Stig Petersen. Deepmind lab. *CoRR*, abs/1612.03801, 2016. URL http://arxiv.org/abs/1612.03801.

Angel X. Chang, Angela Dai, Thomas A. Funkhouser, Maciej Halber, Matthias Nießner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from RGB-D data in indoor environments. In *2017 International Conference on 3D Vision, 3DV 2017, Qingdao, China, October 10-12, 2017*, pp. 667–676, 2017. doi: 10.1109/3DV.2017.00081. URL https://doi.org/10.1109/3DV.2017.00081.

Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, José M. F. Moura, Devi Parikh, and Dhruv Batra. Visual dialog. *CoRR*, abs/1611.08669, 2016. URL http://arxiv.org/abs/1611.08669.

Abhishek Das, Samyak Datta, Georgia Gkioxari, Stefan Lee, Devi Parikh, and Dhruv Batra. Embodied question answering. *CoRR*, abs/1711.11543, 2017a. URL http://arxiv.org/abs/1711.11543.

Abhishek Das, Satwik Kottur, José M. F. Moura, Stefan Lee, and Dhruv Batra. Learning cooperative visual dialog agents with deep reinforcement learning. *CoRR*, abs/1703.06585, 2017b. URL http://arxiv.org/abs/1703.06585.

Abhishek Das, Georgia Gkioxari, Stefan Lee, Devi Parikh, and Dhruv Batra. Neural modular control for embodied question answering. *CoRR*, abs/1810.11181, 2018. URL http://arxiv.org/abs/1810.11181.

Harm de Vries, Florian Strub, Sarath Chandar, Olivier Pietquin, Hugo Larochelle, and Aaron C. Courville. Guesswhat?! visual object discovery through multi-modal dialogue. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pp. 4466–4475, 2017. doi: 10.1109/CVPR.2017.475. URL https://doi.org/10.1109/CVPR.2017.475.

Hugh F. Durrant-Whyte and Tim Bailey. Simultaneous localization and mapping: part I. *IEEE Robot. Automat. Mag.*, 13(2):99–110, 2006. doi: 10.1109/MRA.2006.1638022. URL https://doi.org/10.1109/MRA.2006.1638022.

Daniel Fried, Ronghang Hu, Volkan Cirik, Anna Rohrbach, Jacob Andreas, Louis-Philippe Morency, Taylor Berg-Kirkpatrick, Kate Saenko, Dan Klein, and Trevor Darrell. Speaker-follower models for vision-and-language navigation. *CoRR*, abs/1806.02724, 2018. URL http://arxiv.org/abs/1806.02724.

Daniel Gordon, Aniruddha Kembhavi, Mohammad Rastegari, Joseph Redmon, Dieter Fox, and Ali Farhadi. IQA: visual question answering in interactive environments. *CoRR*, abs/1712.03316, 2017. URL http://arxiv.org/abs/1712.03316.

Stevan Harnad. The symbol grounding problem. *CoRR*, cs.AI/9906002, 1999. URL http://arxiv.org/abs/cs.AI/9906002.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015. URL http://arxiv.org/abs/1512.03385.

Nicolas Heess, Greg Wayne, David Silver, Timothy P. Lillicrap, Yuval Tassa, and Tom Erez. Learning continuous control policies by stochastic value gradients. *CoRR*, abs/1510.09142, 2015. URL http://arxiv.org/abs/1510.09142.

Karl Moritz Hermann, Felix Hill, Simon Green, Fumin Wang, Ryan Faulkner, Hubert Soyer, David Szepesvari, Wojciech Marian Czarnecki, Max Jaderberg, Denis Teplyashin, Marcus Wainwright, Chris Apps, Demis Hassabis, and Phil Blunsom. Grounded language learning in a simulated 3d world. *CoRR*, abs/1706.06551, 2017. URL http://arxiv.org/abs/1706.06551.

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*, 2017.

Tejas D. Kulkarni, Karthik Narasimhan, Ardavan Saeedi, and Josh Tenenbaum. Hierarchical deep reinforcement learning: Integrating temporal abstraction and intrinsic motivation. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pp. 3675–3683, 2016. URL http://papers.nips.cc/paper/6233-hierarchical-deep-reinforcement-learning-integrating-temporal-abstraction-and

Hung Le, Doyen Sahoo, Nancy F. Chen, and Steven C. H. Hoi. Multimodal transformer networks for end-to-end video-grounded dialogue systems. *CoRR*, abs/1907.01166, 2019. URL http://arxiv.org/abs/1907.01166.

Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, Devi Parikh, and Dhruv Batra. Habitat: A Platform for Embodied AI Research. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

Piotr Mirowski, Matthew Koichi Grimes, Mateusz Malinowski, Karl Moritz Hermann, Keith Anderson, Denis Teplyashin, Karen Simonyan, Koray Kavukcuoglu, Andrew Zisserman, and Raia Hadsell. Learning to navigate in cities without a map. *CoRR*, abs/1804.00168, 2018. URL http://arxiv.org/abs/1804.00168.

Volodymyr Mnih, Adrià Puigdomènech Badia, Mehdi Mirza, Alex Graves, Timothy P. Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. *CoRR*, abs/1602.01783, 2016. URL http://arxiv.org/abs/1602.01783.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pp. 1532–1543, 2014. URL https://www.aclweb.org/anthology/D14-1162/.

Pararth Shah, Marek Fiser, Aleksandra Faust, J. Chase Kew, and Dilek Hakkani-Tür. Followne: Robot navigation by following natural language directions with deep reinforcement learning. *CoRR*, abs/1805.06150, 2018. URL http://arxiv.org/abs/1805.06150.

T. Tangiuchi, Daichi Mochihashi, T. Nagai, S. Uchida, N. Inoue, Ichiro Kobayashi, T. Nakamura, Yoshinobu Hagiwara, Naoto Iwahashi, and Tetsunari Inamura. Survey on frontiers of language and robotics. *Advanced Robotics*, pp. 1–31, 06 2019. doi: 10.1080/01691864.2019.1632223.

Xin Wang, Qiuyuan Huang, Asli Çelikyilmaz, Jianfeng Gao, Dinghan Shen, Yuan-Fang Wang, William Yang Wang, and Lei Zhang. Reinforced cross-modal matching and self-supervised imitation learning for vision-language navigation. *CoRR*, abs/1811.10092, 2018. URL http://arxiv.org/abs/1811.10092.

Mark Wernsdorfer and Ute Schmid. Grounding hierarchical reinforcement learning models for knowledge transfer. *CoRR*, abs/1412.6451, 2014. URL http://arxiv.org/abs/1412.6451.

Erik Wijmans, Samyak Datta, Oleksandr Maksymets, Abhishek Das, Georgia Gkioxari, Stefan Lee, Irfan Essa, Devi Parikh, and Dhruv Batra. Embodied question answering in photorealistic environments with point cloud perception. *CoRR*, abs/1904.03461, 2019. URL http://arxiv.org/abs/1904.03461.

Yi Wu, Yuxin Wu, Georgia Gkioxari, and Yuandong Tian. House3d: A rich and realistic 3d environment. *arXiv preprint arXiv:1801.02209*, 2018a.

Yi Wu, Yuxin Wu, Georgia Gkioxari, and Yuandong Tian. Building generalizable agents with a realistic and rich 3d environment. *arXiv preprint arXiv:1801.02209*, 2018b.

Wei Yang, Xiaolong Wang, Ali Farhadi, Abhinav Gupta, and Roozbeh Mottaghi. Visual semantic navigation using scene priors. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*, 2019. URL https://openreview.net/forum?id=HJeRkh05Km.