Embodied Question Answering

Abhishek Das[1], **Samyak Datta**[1], **Georgia Gkioxari**[2], **Stefan Lee**[1], **Devi Parikh**[2,1], **Dhruv Batra**[2,1]

[1]Georgia Institute of Technology, [2]Facebook AI Research

[1]{abhshkdz, samyak, steflee}@gatech.edu [2]{gkioxari, parikh, dbatra}@fb.com

embodiedqa.org

Abstract

We present a new AI task – **Embodied Question Answering** (EmbodiedQA) – where an agent is spawned at a random location in a 3D environment and asked a question (*'What color is the car?'*). In order to answer, the agent must first intelligently navigate to explore the environment, gather necessary visual information through first-person (egocentric) vision, and answer the question (*'orange'*).

EmbodiedQA requires a range of AI skills – language understanding, visual recognition, active perception, goal-driven navigation, commonsense reasoning, long-term memory, and grounding language into actions. In this work, we develop a dataset of questions and answers in House3D environments (Wu et al., 2018), evaluation metrics, and a hierarchical model trained with imitation and reinforcement learning for this task.

1 Introduction

> The embodiment hypothesis is the idea that intelligence emerges in the interaction of an agent with an environment and as a result of sensorimotor activity.
>
> (Smith and Gasser, 2005)

Our long-term goal is to build intelligent agents that can *perceive* their environment (through vision, audition, or other sensors), *communicate* (*i.e.*, hold a natural language dialog grounded in the environment), and *act* (*e.g.* aid humans by executing API calls or commands in a virtual or embodied environment). In addition to being a fundamental scientific goal in artificial intelligence (AI), even a small advance towards such intelligent systems can *fundamentally change our lives* – from assistive dialog agents for the visually impaired, to natural-language interaction with self-driving cars, in-home robots, and personal assistants.

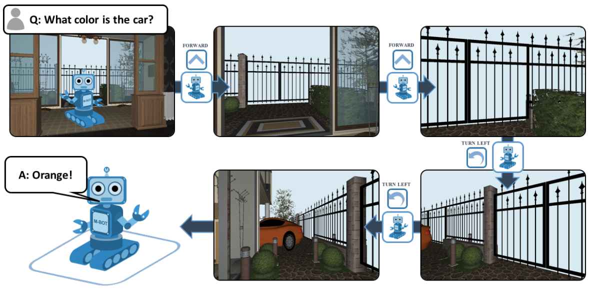

Figure 1: Embodied Question Answering tasks agents with navigating rich 3D environments in order to answer questions. These agents must jointly learn language understanding, visual reasoning, and goal-driven navigation to succeed.

As a step towards goal-driven agents that can perceive, communicate, and execute actions, we present a new AI task – ***Embodied Question Answering*** (EmbodiedQA) – along with a dataset of questions in virtual environments, evaluation metrics, and a reinforcement learning (RL) model.

Concretely, the EmbodiedQA task is illustrated in Fig. 1 – an agent is spawned at a random location in an environment (a house or building) and asked a question (*e.g. 'What color is the car?'*). The agent perceives its environment through first-person egocentric vision and can perform a few atomic actions (move-forward, turn, strafe, *etc.*). The goal of the agent is to intelligently navigate the environment and gather visual information necessary for answering the question.

EmbodiedQA is a challenging task that subsumes several fundamental problems as sub-tasks. Clearly, the agent must understand language (*what is the question asking?*) and vision (*what does a 'car' look like?*), but it must also learn:

Active Perception: The agent may be spawned anywhere in the environment and may not immediately 'see' the pixels containing the answer to the visual question (*i.e.* the car may not be visible). Thus, the agent *must* move to succeed – controlling the pixels that it perceives. The agent must learn to map its visual input to the correct actions based on its perception of the world, the underlying physical constraints, and its understanding of the question.

Commonsense Reasoning: The agent is not provided a floor-plan of the environment, and must navigate from egocentric views alone. Thus, it must learn common sense (*Where am I? Where are cars typically found? Where is the garage with respect to me?*) similar to how humans navigate in unfamiliar houses (*The car is probably in the garage, so I should find an exit*).

Language Grounding: One commonly noted shortcoming of modern vision-and-language models is their lack of grounding – these models often fail to associate entities in text with corresponding image pixels, relying instead on dataset biases to respond seemingly intelligently even when attending to irrelevant regions (Das et al., 2016). In EmbodiedQA, we take a goal-driven view of grounding – our agent grounds a question not into pixels but into a sequence of actions ('garage' *means* to navigate towards the exterior where the 'car' is parked).

As a first step in this challenging space, we judiciously scope out a problem space – environments, question types, and learning paradigm – that allows us to augment sparse RL rewards with imitation learning (showing the agent expert trajectories) and reward shaping (Ng et al., 1999) (giving intermediate 'closer/farther' navigation rewards). Specifically, our approach follows the recent paradigm from robotics and deep RL (Levine et al., 2016; Misra et al., 2017) – the training environments are sufficiently *instrumented*, and provide access to the agent location, RGB, depth & semantic annotations of the visual environment, and allow for computing obstacle-avoiding shortest navigable paths from the agent to any target.

At test time, our agents operate entirely from egocentric RGB vision alone – no structured representation of the environments, no access to a map, no explicit localization of the agent or mapping of the environment, no A* or any other heuristic planning, no hand-coded knowledge about the environment or task, and no pre-processing phase for the agent to build a map of the environment. The agent in its entirety – vision, language, navigation, answering – is trained from raw sensory input (pixels and words) to goal-driven multi-room navigation to visual question answering!

Contributions.

- We propose a new AI task: EmbodiedQA, where an agent is spawned in an environment and must intelligently navigate from egocentric vision to gather the necessary information to answer questions about the environment.

- We introduce a hierarchical navigation module that decomposes navigation into a 'planner' that selects actions, and a 'controller' that executes these primitive actions. When the agent decides it has seen the required visual information, it stops navigating and outputs an answer.

- We initialize our agents with imitation learning and show that agents can answer questions more accurately after fine-tuning with RL – that is, when allowed to control their own navigation *for the explicit purpose* of answering questions accurately. Unlike some prior work, we explicitly test *generalization to unseen environments*.

- We evaluate our agents in House3D (Wu et al., 2018), a rich, interactive 3D environment based on human-designed indoor scenes from SUNCG (Song et al., 2017). These diverse, simulated environments enable us to test generalization of our agent across floor-plans and object configurations – without safety, privacy, expense concerns inherent to real robotic platforms.

- We introduce EQA, a dataset of visual questions and answers grounded in House3D. The question types test a range of agent abilities – scene recognition (`location`), spatial reasoning (`preposition`), color recognition (`color`). While the EmbodiedQA task definition supports free-form natural language questions, we represent each question in EQA as a *functional program* that can be programmatically generated and executed on the environment to determine the answer. This enables us to control the distribution of question-types and answers, deter algorithms from exploiting dataset bias (Goyal et al., 2017), and provide fine-grained breakdown of performance by skill.

- We integrated House3D with Amazon Mechanical Turk (AMT), allowing humans to *remotely operate the agent in real time*, and collected expert demonstrations of question-guided navigation for EmbodiedQA that serve as a benchmark to compare proposed and future models.

2 EQA: Questions In Environments

2.1 House3D: Simulated 3D Environments

We instantiate EmbodiedQA in House3D (Wu et al., 2018), a recently introduced rich, simulated environment based on 3D indoor scenes from the SUNCG dataset (Song et al., 2017). We build EQA on a pruned subset of environments from House3D, across a total of 767 environments.

2.2 Question-Answer Generation

We draw inspiration from the CLEVR (Johnson et al., 2017) dataset, and programmatically generate a dataset (EQA) of grounded questions and answers. This gives us the ability to carefully control the distribution of question-types and answers in the dataset, and deter algorithms from exploiting dataset bias. Overall, we have the following question types in the EQA dataset:

EQA v1	`location:`	*'What room is the <OBJ> located in?'*
	`color:`	*'What color is the <OBJ>?'*
	`color_room:`	*'What color is the <OBJ> in the <ROOM>?'*
	`preposition:`	*'What is <on/above/below/next-to> the <OBJ> in the <ROOM>?'*
	`existence:`	*'Is there a <OBJ> in the <ROOM>?'*
	`logical:`	*'Is there a(n) <OBJ1> and a(n) <OBJ2> in the <ROOM>?'*
	`count:`	*'How many <OBJS> in the <ROOM>?'*
	`room_count:`	*'How many <ROOMS> in the house?'*
	`distance:`	*'Is the <OBJ1> closer to the <OBJ2> than to the <OBJ3> in the <ROOM>?'*

	Environments	Unique Questions	Total Questions
`train`	643	147	4246
`val`	67	104	506
`test`	57	105	529

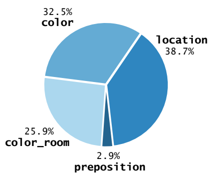

Figure 2: Overview of the EQA v1 dataset including dataset split statistics (left) and question type breakdown (right).

EQA v1 Statistics. The EQA v1 dataset consists of over 5000 question across over 750 environments, referring to a total of 45 unique objects in 7 unique room types. The dataset is split into `train`, `val`, `test` such that there is no overlap in environments across splits. Fig. 2 shows the dataset splits and question type distribution. Approximately 6 questions are asked per environment on average, 22 at most, and 1 at fewest. There are relatively few `preposition` questions as many frequently occurring spatial relations are too easy to resolve without exploration and fail the entropy filtering.

3 Hierarchical Model for EmbodiedQA

Vision. Our agent takes egocentric 224×224 RGB images from the House3D renderer as input, which we process with a CNN consisting of 4 {5×5 Conv, BatchNorm, ReLU, 2×2 MaxPool} blocks, producing a fixed-size representation.

Language. Questions are encoded with 2-layer LSTMs with 128-d hidden states. While LSTMs may be overkill for the simple questions in EQA v1, it gives us the flexibility to expand to human-written or more complex questions in future.

Navigation. Our planner-controller navigator (PACMAN) decomposes navigation into a 'planner', that selects actions (forward, turn-left, turn-right, stop), and a 'controller', that executes these primitive actions a variable number of times (1,2, . . .) before returning control back to the planner. Intuitively, this structure separates the intent of the agent (*i.e.* get to the other end of the room) from the series of primitive actions (*i.e. 'forward, forward, forward, ...'*), and is reminiscent of hierarchical RL (Andreas et al., 2017; Oh et al., 2017; Tessler et al., 2017). It also enables planning at shorter timescales, strengthening gradient flows.

Question Answering. After the agent decides to stop or a max number of actions (= 100) have been taken, the question answering module is executed to provide an answer based on attention over the last 5 frames the agent has observed.

Training. We employ a two-stage training process. First, the navigation and answering modules are independently trained using supervised learning on automatically generated expert demonstrations of navigation. Second, the navigation architecture is fine-tuned using policy gradients.

4 Experiments and Results

Question Answering Accuracy. Our agent (and all baselines) produces a probability distribution over 172 answers (colors, rooms, objects). We report the mean rank (**MR**) of the ground-truth answer in the answer list sorted by the agent's beliefs, computed over `test` questions from EQA.

Navigation Accuracy. We evaluate navigation performance by reporting the distance to the target object at navigation termination ($\mathbf{d_T}$), change in distance to target from initial to final position ($\mathbf{d_\Delta}$), and the smallest distance to the target at any point in the episode ($\mathbf{d_{min}}$). All distances are

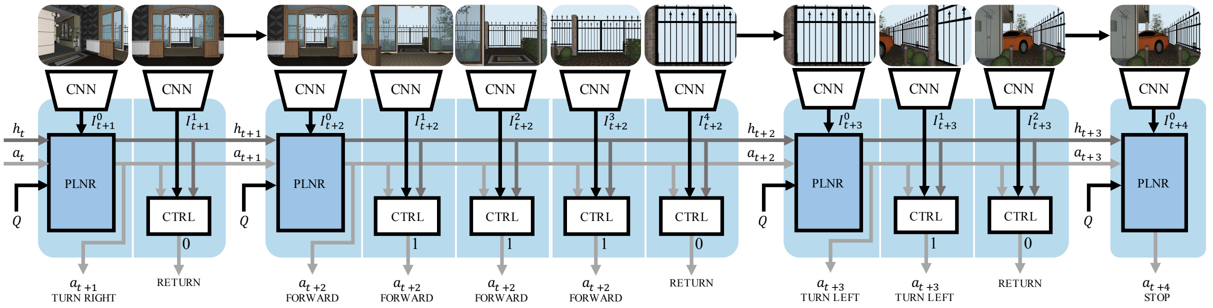

Figure 3: Our navigator decomposes navigation into a planner and a controller. The planner selects actions and the controller executes these actions for variable timesteps. Thus, the planner operates on shorter timescales, strengthening gradient flows.

		Navigation																			**QA**		
		$\mathbf{d_T}$			$\mathbf{d_\Delta}$			$\mathbf{d_{min}}$			$\%\mathbf{r_T}$			$\%\mathbf{r}_{\hookleftarrow}$			$\%\mathbf{stop}$			**MR**		
		T_{-10}	T_{-30}	T_{-50}	T_{-10}	T_{-30}	T_{-50}	T_{-10}	T_{-30}	T_{-50}	T_{-10}	T_{-30}	T_{-50}	T_{-10}	T_{-30}	T_{-50}	T_{-10}	T_{-30}	T_{-50}	T_{-10}	T_{-30}	T_{-50}
Baselines	Reactive	2.09	2.72	3.14	-1.44	-1.09	-0.31	0.29	1.01	1.82	50%	49%	**47%**	52%	53%	48%	-	-	-	3.18	3.56	3.31
	LSTM	1.75	2.37	2.90	-1.10	-0.74	-0.07	0.34	1.06	2.05	55%	53%	44%	59%	57%	50%	80%	75%	80%	3.35	3.07	3.55
	Reactive+Q	1.58	2.27	2.89	-0.94	-0.63	-0.06	0.31	1.09	1.96	52%	51%	45%	55%	57%	**54%**	-	-	-	3.17	3.54	3.37
	LSTM+Q	1.13	2.23	2.89	-0.48	-0.59	-0.06	0.28	0.97	1.91	**63%**	53%	45%	64%	59%	**54%**	80%	71%	68%	3.11	3.39	3.31
Us	PACMAN+Q	**0.46**	**1.50**	**2.74**	**0.16**	**0.15**	**0.12**	0.42	1.42	2.63	58%	54%	45%	60%	56%	46%	**100%**	**100%**	**100%**	**3.09**	3.13	3.25
	PACMAN-RL+Q	1.67	2.19	2.86	-1.05	-0.52	0.01	**0.24**	**0.93**	**1.94**	57%	**56%**	45%	**65%**	**62%**	52%	32%	32%	24%	3.13	**2.99**	**3.22**
Oracle	HumanNav*	0.81	0.81	0.81	0.44	1.62	2.85	0.33	0.33	0.33	86%	86%	86%	87%	89%	89%	-	-	-	-	-	-
	ShortestPath+VQA	-	-	-	0.85	2.78	4.86	-	-	-	-	-	-	-	-	-	-	-	-	3.21	3.21	3.21

measured in meters along the shortest path to the target. We also record the percentage of questions for which an agent either terminates in ($\%\mathbf{r_T}$) or ever enters ($\%\mathbf{r}_{\hookleftarrow}$) the room containing the target object(s). Finally, we also report the percentage of episodes in which agents choose to terminate navigation and answer before reaching the maximum episode length ($\%\mathbf{stop}$). To sweep over the difficulty of the task, we spawn the agent 10, 30, or 50 actions away from the target and report each metric for T_{-10}, T_{-30}, T_{-50}.

- **All baselines are poor navigators.** All baselines methods have *negative* $\mathbf{d_\Delta}$, *i.e.* they end up *farther* from the target than where they start. This confirms our intuition that EmbodiedQA is indeed a difficult problem.

- **Memory helps.** All models start equally far away from the target. Baselines augmented with memory (LSTM *vs.* Reactive and LSTM-Q *vs.* Reactive-Q) end closer to the target, *i.e.* achieve smaller $\mathbf{d_T}$, than those without.

- **PACMAN Navigator performs best.** Our proposed navigator (PACMAN+Q) achieves the smallest distance to target at termination ($\mathbf{d_T}$), and the RL-finetuned navigator (PACMAN-RL+Q) achieves highest answering accuracy.

- **RL agent overshoots.** We observe that while PACMAN-RL+Q gets closest to the target (least $\mathbf{d_{min}}$) and enters the target room most often (highest $\%\mathbf{r}_{\hookleftarrow}$), it does *not* end closest to the target (does not achieve lowest $\mathbf{d_T}$). These and our qualitative analysis suggests that this is because RL-finetuned agents learn to explore, with a lower stopping rate ($\%\mathbf{stop}$), and often overshoot the target. This is consistent with observations in literature (Misra et al., 2017). This does not hurt QA accuracy because the answerer can attend to the last 5 frames along the trajectory, and can potentially be corrected by including a small penalty for each action.

References

Jacob Andreas, Dan Klein, and Sergey Levine. 2017. Modular multitask reinforcement learning with policy sketches. In *ICML*.

Abhishek Das, Harsh Agrawal, C. Lawrence Zitnick, Devi Parikh, and Dhruv Batra. 2016. Human Attention in Visual Question Answering: Do Humans and Deep Networks Look at the Same Regions? In *EMNLP*.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *CVPR*.

Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C. Lawrence Zitnick, and Ross Girshick. 2017. CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. In *CVPR*.

Sergey Levine, Chelsea Finn, Trevor Darrell, and Pieter Abbeel. 2016. End-to-end training of deep visuomotor policies. *JMLR*, 17(1):1334–1373.

Dipendra Misra, John Langford, and Yoav Artzi. 2017. Mapping instructions and visual observations to actions with reinforcement learning. In *EMNLP*.

Andrew Y. Ng, Daishi Harada, and Stuart J. Russell. 1999. Policy invariance under reward transformations: Theory and application to reward shaping. In *ICML*.

Junhyuk Oh, Satinder Singh, Honglak Lee, and Pushmeet Kohli. 2017. Zero-shot task generalization with multi-task deep reinforcement learning. In *ICML*.

Linda Smith and Michael Gasser. 2005. The development of embodied cognition: six lessons from babies. *Artificial life*, 11(1-2).

Shuran Song, Fisher Yu, Andy Zeng, Angel X Chang, Manolis Savva, and Thomas Funkhouser. 2017. Semantic scene completion from a single depth image. In *CVPR*.

Chen Tessler, Shahar Givony, Tom Zahavy, Daniel J. Mankowitz, and Shie Mannor. 2017. A deep hierarchical approach to lifelong learning in minecraft. In *AAAI*.

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