Translating English to Reward Functions

James MacGlashan[*], **Monica Babeş-Vroman**[+], **Marie desJardins**[#], **Michael Littman**[*]

Smaranda Muresan[§] , **Shawn Squire**[#]

[*] Computer Science Department, Brown University
[+] Computer Science Department, Rutgers University
[#] Department of Computer Science and Electrical Engineering, University of Maryland, Baltimore County
[§] Department of Computer Science, Columbia University

Abstract

For intelligent agents and robots to be useful to the general public, people will need to able to communicate the tasks they want the agents to complete without having any technical knowledge or programming ability. Communicating tasks to agents via natural language is an especially appealing way to accomplish this goal. Similarly, providing agents with demonstrations of what they should do given a natural language command is an appealing low-effort way to train agents. In this work, we present a novel generative task model, in which tasks are defined by MDP reward functions. This generative task model can be combined with different language models to produce a complete system that can learn the meaning of individual words from demonstrations. Because these meanings are grounded in reward functions, goals can be executed that require complex multiple-step behavior. We present two language models that can be used with our task model and empirically validate them on a dataset with natural language commands gathered from a user study.

1 Introduction

For robots and other intelligent agents to be useful to the general public, they need to be able to autonomously carry out complex tasks. However, it is equally important for humans to be able to communicate a desired complex task to an agent, ideally using natural language instructions instead of a formal machine-oriented task representation. We describe an approach for an agent to map natural language commands into task descriptions and execute them. Specifically, we learn mappings from training data that consist of pairs of natural language commands and demonstrations of the command being executed. Learning command interpretations from demonstrations is useful because it requires little or no knowledge of the internal task representations, thereby allowing training datasets to be created by users with a limited technical background.

This work provides two main contributions: (1) a generative model of tasks and behavior that can be used to weakly supervise a semantic language model and (2) an investigation of multiple language models for use with the task model and that do not require any background grammar knowledge. The task model represents task descriptions as abstract *Markov decision process reward functions* that are defined using propositional features of the world. Defining tasks as abstract reward functions is useful because reward functions can induce complex multi-step behavior without having to specify the steps of that behavior. As a result, the natural language commands can be quite general. For instance, "bring me a cup of coffee" would give rise to a reward function that motivates the agent to choose steps for getting the coffee and bringing it to the user, without the user specifying any of those details in the command. Since tasks are represented with reward functions, the demonstrations can provide weak supervision to the language model by using inverse reinforcement learning (Abbeel and Ng 2004) to infer the probability that each possible reward function was the intended one.

We investigate two language models: a Bag-of-Words model and the IBM Model 2 machine translation model (Brown et al. 1990; 1993). Neither of these models require or use any background grammatical knowledge. While we suspect that language models that incorporate grammar models might yield higher performance, the additional requirement of a grammar model would likely make it more difficult to adapt this approach to new applications.

We empirically validate our approach on a simulated "Cleanup World" domain (based on the game of Sokoban); natural language commands for the domain were gathered from Amazon Mechanical Turk users. A typical task is to take a star block into an orange painted room; one data instance for this task is the command "go into tan room and push star into orange room" with a sequence of states and actions demonstrating the completion of the task. Note that this demonstration consists of many primitive movement actions, but the provided command does not describe each action that the agent needs to take, only the high-level goal. Using this dataset, we show that the Bag-of-Words model is effective but for only certain kinds of simple commands, and that IBM Model 2 is capable of learning the meaning of commands that specify multiple constraints.

2 Background

To represent tasks and perform planning, we make use of the *Object-oriented Markov Decision Process* (OO-MDP) formalism (Diuk, Cohen, and Littman 2008). OO-MDPs extend the conventional MDP formalism by providing a rich representation of states. MDPs are defined as a four-tuple: ($\mathcal{S}$, $\mathcal{A}$, $\mathcal{T}$, $\mathcal{R}$), where $\mathcal{S}$ is a set of states of the world; $\mathcal{A}$ is a set of actions that the agent can take; $\mathcal{T}$ describes the transition dynamics, which specify the probability of the agent transitioning to each state given the current state and the action taken; and $\mathcal{R}$ is a reward function specifying the reward received by the agent for each transition. MDPs may also have terminal states that cause action to cease once they are reached. Terminal states may indicate a task execution failure or may correspond to goal conditions.

OO-MDPs add to the MDP formalism a set of attributes $\mathcal{X}$; a set of object classes $\mathcal{C}$, each class of which ($c \in \mathcal{C}$) has an associated subset of attributes ($\mathcal{X}_c \subseteq \mathcal{X}$); and a set of propositional functions $\mathcal{P}$. States in an OO-MDP are defined as a set of object instances, each of which belongs to an object class and has a value assignment to each of the attributes associated with the object class. The value assignment of each object instance defines the object's state. An MDP's state ($s \in \mathcal{S}$) is fully defined by the collective states (attribute values) of the object instances. The propositional functions ($\mathcal{P}$) operate on the object instance states to provide additional high-level information about the MDP's state. For instance, consider a propositional function called `touching` that takes two object instances of class `BLOCK` as parameters (`touching`(`BLOCK`, `BLOCK`)). When `touching` is applied to block object instances b_1 and b_2, it will evaluate to true if the value of the position attributes of b_1 and b_2 are adjacent and false otherwise. Propositional functions can be evaluated in this way on any combination of object instances that satisfy the class type of the function's parameters.

In the original OO-MDP work, the high-level information provided by propositional functions was used to model and learn the transition dynamics of the MDP. In this work, we use propositional functions to define factored task descriptions that can map to and from natural language commands.

3 The Task Model

We assume that any task that a person may request the agent to perform can be represented as a reward function and corresponding terminal states (if it is a terminating task) in an OO-MDP. For brevity, we will refer to a reward function and set of terminal states as a *task*. Everything about the OO-MDP, except the task, is assumed to be known by the agent; the task, however, must be inferred from the natural language command given to the agent. Once the agent has inferred the task, it can plan and carry out the task using any "off-the-shelf" MDP planning algorithm. Our learning process uses a generative model in which action-level behavior and commands are generated from tasks and the current state. The generative model (shown in Figure 1) consists of an input current state (S), a set of lifted tasks (L), a set of grounded tasks (T), a set of object binding constraints (C), a set of behavior trajectories (B), and a language model that produces natural language commands and is dependent on the lifted task and binding constraints.

A *lifted task* is a reward function that is defined in terms of logical expressions of OO-MDP propositional functions, but with the parameters of the propositional functions left as free variables. Leaving the parameters as free variables makes lifted tasks define *classes* of tasks that could be executed by the agent. For example, consider an OO-MDP that can

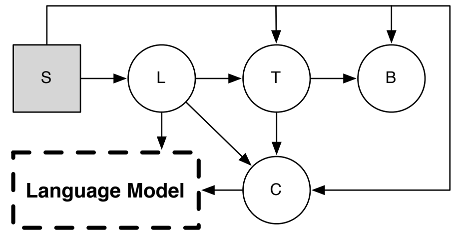

Figure 1: The generative task model, with arrows indicating conditional probabilities.

contain various block objects in different rooms. A reward function for a task to take a block to a room could be:

$$\mathcal{R}(s,a,s') = \begin{cases} 1 & \text{if } \texttt{blockInRoom}^{s'}(?b, ?r) \\ 0 & \text{otherwise} \end{cases}, \quad (1)$$

where s' is the state to which the agent transitions when it takes action a in state s, $\texttt{blockInRoom}^{s'}$ indicates an evaluation of the propositional function `blockInRoom` in state s', and $?b$ and $?r$ are two different free variables. Note that because the parameters of a propositional function are typed to specific object classes, the free variables are also typed. In this case, any object assigned to $?b$ must be a `BLOCK` object and any value assigned to $?r$ must be a `ROOM` object. For this goal-directed task, the terminal states are the states that yield a reward of 1.

The set of possible lifted tasks is specified by the designer. The prior probability distribution over lifted tasks, given a current state, is a uniform distribution over the subset of lifted tasks that are *permissible* in the input state. A lifted task is permissible in a state if the state has at least one unique object instance belonging to the class for each of the free variables. In the previous example, that means there must be at least one `BLOCK` object and one `ROOM` object.

A *grounded task* depends on the input state and lifted task and is defined by an object assignment to the free variables in the lifted task. The prior probability distribution over the set of grounded tasks is uniform over the set of possible object assignments for the lifted task, given the input state. If the lifted task is a goal-directed task, then the prior probability distribution is uniform over the set of object assignments that do not cause the goal to be satisfied in the input state.

A *behavior trajectory* is a sequence of state–action pairs that the agent can experience from an input state and is conditionally dependent on the grounded task and input state. The conditional probability of the behavior, given the task, is formulated by treating each action selection in each state of the trajectory as an independent event and by defining the action-selection probability distribution as a Boltzmann distribution over the optimal Q-values for the task (similar to the likelihood of a trajectory in maximum likelihood inverse reinforcement learning (Babeş-Vroman et al. 2011)). Formally, the probability of any behavior trajectory b (of length N) given task t is defined as

$$\Pr(b|t) = \prod_i^N \pi_t(s_i, a_i), \quad (2)$$

where (s_i, a_i) is the ith state–action pair in behavior trajectory b, and $\pi_t(s_i, a_i)$ is the Boltzmann policy distribution (the probability of taking action a_i in state s_i). Given the optimal Q-values for an MDP, the Boltzmann policy distribution is defined as

$$\pi(s,a) = \frac{e^{\frac{Q(s,a)}{\tau}}}{\sum_{a'} e^{\frac{Q(s,a')}{\tau}}}, \quad (3)$$

where τ is a temperature parameter. Higher temperatures make the model less sensitive to suboptimal actions taken in the demonstration, but less discriminative between different tasks. Lower temperature values are more discriminative, but are also more sensitive to error when the demonstration contains suboptimal actions.

If the task is a terminating goal-directed task, this conditional probability formalism may inappropriately assign equal probability to a trajectory under two different tasks if the optimal policy for one is a subset of the other. For example, consider a home in which going to the dining room requires going through the living room. A demonstration of a task for going to the living room will be assigned the same probability under both the living room and dining room task, because the policy for the states in the demonstration would be identical. To address this limitation, we augment the MDP action set to include a special terminating action that must be executed for the agent to end the task and receive the goal reward. The demonstration is then augmented to include the agent executing the terminate action at the end. In our previous example, a demonstration of going to the living room would not produce equal probability for the dining room task because it would be suboptimal for the agent to terminate the task before it accomplishes its goal.

If all an agent could directly infer from a command was the lifted task, it would be impossible for the agent to determine which of the possible grounded tasks for an input state was the intended task. The inclusion of *object binding constraints* that natural language can reference enables the agent to resolve this ambiguity. In the task model, object binding constraints are dependent on the grounded task, lifted task, and input state, and consist of logical expressions for a given object assignment that are true in the input state. For instance, imagine a state with a box block (b_0), a sphere block (b_1), and three rooms (r_0, r_1, r_2) that were painted different colors (red, green, and blue). If the intended task was to take b_1 into room r_0, a natural way to describe the task without knowing the agent's internal identifiers for each block and room would be to say "take the spherical block to the red room." In such a command, the qualifiers "spherical" and "red" specify binding constraints that disambiguate which object assignment to use. These binding constraints are specified as qualifications to the free variables in the lifted task: $\beta = \texttt{isSphere}(?b) \wedge \texttt{isRed}(?r)$. Constraints are specified using the free variables of the lifted task to keep them invariant of the identifiers in the input state.

The conditional probability distribution of object constraints given a grounded task, lifted task, and input state, is uniform across the set of *permissible* object constraints. An object constraint is permissible if it is a logical expression that is true in the initial state for the object assignment specified in the grounded task and if the logical expression is limited to the variables introduced in the lifted task, or variables that can be related to variables in the lifted task. For instance, the previously specified binding constraint is permissible because the grounded task assigns lifted task variables $?b$ to b_1 and $?r$ to r_0, and the expression $\texttt{isSphere}(b_1) \wedge \texttt{isRed}(r_0)$ is true in the initial state. In contrast, the constraint $\texttt{isSphere}(?b) \wedge \texttt{isBlue}(?r)$ is not permissible because it is not true for the given object assignment, nor is the constraint $\texttt{isSphere}(?b) \wedge \texttt{isBlue}(?r2)$, because $?r2$ is not a free variable in the lifted task, and is not related to any of the free variables in the lifted task. However, if b_1 is in room r_1, which is green, in the input state, then the constraint $\texttt{isRed}(?r) \wedge \texttt{blockInRoom}(?b, ?r2) \wedge \texttt{isGreen}(?r2)$ would be permissible, because the variable $?r2$ is related to variable $?b$ from the lifted task, and the statement is true in the input state when r_1 is assigned to $?r2$. Such relational extensions allow the object constraints to model natural language commands like "take the block in the green room to the red room."

The task model presented above can be combined with any language model as long as its probability distributions can be made dependent in some way on the lifted task and object binding constraints. The notation $\Pr(e|l,c)$ is used to represent the probability of a natural language command e given lifted task l and object binding constraints c; this probability is assumed to be defined by the specific language model used. Given the language model, we have a complete generative model from tasks to behavior and language, and can perform learning from a dataset of demonstrations and natural language command pairs using any parameter inference algorithm, such as Expectation Maximization (EM) (Dempster, Laird, and Rubin 1977). The agent can then infer the most likely task given a natural language command, plan behavior to complete it, and execute the plan. Specifically, the probability of task t given natural language command e and input state s is:

$$\Pr(t|e,s) \propto \sum_l \Pr(l|s) \Pr(t|s,l) \sum_c \Pr(c|s,l,t) \Pr(e|l,c). \quad (4)$$

4 Language Models

We investigated two different grammar-free language models for use with our task model: a Bag-of-Words model and IBM Model 2, a word-based machine-translation model. In this section, we describe each model and how it integrates with our task model.

Bag-of-Words

The Bag-of-Words (BoW) language model is similar to the topic modeling approach used by McCallum (1999), in which words are distributed according to a mixture of topics and a document is generated by selecting a topic with some mixture weight, generating a word from the topic's word distribution, and then repeating the process. In our BoW model (Figure 2), we replace the role of topics with semantic components of the lifted task and object binding

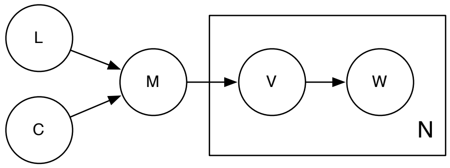

Figure 2: The Bag-of-Words model.

constraints, generating words as a mixture model of those semantic components.

Random variable M is assigned a frequency vector of the semantic components that appear in the lifted task and object binding constraints, consisting of the names of propositional functions and the object class names of the parameters of the propositional functions; a special constant referred to by the symbol '#' is also always assumed to appear once. For instance, for the lifted task in Equation 1 and object binding constraint $\texttt{isRed}(?r)$, the corresponding value assigned to variable M would be:

$$\langle \# = 1, \texttt{blockInRoom} = 1, \texttt{isRed} = 1, \texttt{BLOCK} = 1, \texttt{ROOM} = 2 \rangle.$$

Normalizing the frequency of each semantic component defines the conditional probability that a semantic component will be selected ($\Pr(v|m)$); any semantic component not in the semantic component frequency vector m is assigned zero probability. For a command of length n, a semantic component will be selected n times, each time generating a natural language word according to a multinomial word distribution for each semantic component. Therefore, the conditional probability of any natural language command e is:

$$\Pr(e|m) = \prod_{w \in e} \left[\sum_v \Pr(v|m)\theta_{vw} \right]^{K(w,e)}, \qquad (5)$$

where θ_{vw} is a parameter specifying the conditional probability that the natural language word w is generated given semantic component v, and $K(w, e)$ is the number of times that word w appears in command e.

For this model, the parameters θ can be learned using EM. At each iteration, the parameters are updated using the formula $\theta_{vw} = \frac{\hat{N}(v,w)}{\hat{N}(v)}$, where $\hat{N}(x)$ is the expected number of times that x appears in some dataset. For a dataset consisting of pairs of demonstration behavior and natural language commands ($D = \{(b_1, e_1)...(b_{|D|}, e_{|D|})\}$), these values are computed as follows:

$$\hat{N}(v,w) = \sum_i \frac{K(w,e_i)\sum_m \Pr(v|m)\theta_{vw}\Pr(e_i - w|m)U(m,b_i|s_i)}{\sum_m \Pr(e_i|m)U(m,b_i|s_i)} \qquad (6)$$

$$\hat{N}(v) = \sum_i \frac{\sum_u K(u,e_i)\sum_m \Pr(v|m)\theta_{vu}\Pr(e_i - u|m)U(m,b_i|s_i)}{\sum_i \sum_m \Pr(e_i|m)U(m,b_i|s_i)}, \qquad (7)$$

where $\Pr(e_i - w|m)$ is the probability of the command e_i given the semantic component frequency vector m if one occurrence of the word w were removed from the command e_i, and

$$U(m,b|s) = \sum_{l,t,c} \Pr(l|s)\Pr(t|l,s)\Pr(b|s,t)\Pr(c|s,l,t)\Pr(m|l,c). \qquad (8)$$

IBM Model 2

IBM Model 2 (Brown et al. 1990; 1993) is a word-based statistical machine-translation model. In statistical machine translation, the task is to translate a sentence from a source language f (e.g., French) to a target language e (e.g., English). Our task corresponds to the problem of translating from an English command e into its corresponding machine-language command m.

We adapt IBM Model 2 to our commands model by using the lifted task and object binding constraints to deterministically generate a machine-language command m, then generating the natural language command e in the standard IBM Model 2 fashion. Each semantic component of the lifted task is added to the machine-language command in the order that they appear, and then the same is done for the binding constraints. For instance, the lifted task in Equation 1 and object binding constraint $\texttt{isRed}(?r)$ would generate the machine-language command "# blockInRoom block room isRed room." The probability of a natural language command given the machine-language command is defined as:

$$\Pr(\mathbf{e}|\mathbf{m}) = \eta(n_e|n_m)\sum_{\mathbf{a}}\prod_j^{n_e} q(a_j|j,n_m,n_e)r(e_j|m_{a_j}), \qquad (9)$$

where $\eta(n_e|n_m)$ is the parameter specifying the probability that a machine-language command of length $n_m$would generate a natural language command of length n_e; $\mathbf{a}$ is a possible alignment from natural language words to machine-language words; $q(a_j|j,n_m,n_e)$ is the alignment parameter specifying the probability that the natural language word in position j would be aligned with the machine-language word in position a_j for a machine-language command of length n_m and natural language command of length n_e; and $r(e_j|m_{a_j})$ is the translation parameter specifying the probability that natural language word e_j would be generated from machine-language word m_{a_j}. The number of alignments ($\mathbf{a}$) is typically very large, so in practice we estimate the value using sampling.

The probability of any possible machine-language command can be computed from the demonstrations in the data:

$$\Pr(m|s,b) = \frac{\sum_{l,t,c}\Pr(l|s)\Pr(t|s,l)\Pr(b|s,t)\Pr(c|s,l,t)\Pr(m|l,c)}{\sum_{l,t}\Pr(l|s)\Pr(t|s,l)\Pr(b|s,t)}. \qquad (10)$$

We use a modified EM algorithm in which each natural language command in the dataset is paired with all machine-language commands with non-zero probability given the behavior trajectory, and the counts computed by the EM algorithm are weighted by that probability. The resulting weakly supervised learning algorithm is shown as pseudocode in the the supplementary document. The method re-estimates the q and r parameters iteratively. In each iteration, it loops through each non-zero probability machine-language command for each data instance in the dataset. For each of those machine-language commands, it matches each natural language word with each machine-language word (including the constant word), updating the relevant expected counts for the q and r parameters by a value δ, where δ represents the machine-language-command-weighted probability that a given natural language word in a given position of the natural language command would have been generated by a given machine-language word in a given position of a machine-language command. Formally:

$$\delta(i,j,u,m) = \Pr(m|s_i,b_i)\frac{q(u|j,n_m,n_i)r(e_j|m_u)}{\sum_v q(v|j,n_m,n_i)r(e_j|m_v)}. \qquad (11)$$

After the expected counts have been estimated for the whole dataset, the q and r parameters are updated according to them and the process repeats. Note that this algorithm does not update the η parameter, which specifies the probability that a natural language command of certain length is generated by a machine-language command of a certain length. Because this parameter does not depend on the q or r parameters it is set once according to:

$$\eta(n_e|n_m) = \frac{\sum_i I(e_i, n_e) \sum_m I(m, n_m) \Pr(m|s_i, b_i)}{\sum_i \sum_m I(m, n_m) \Pr(m|s_i, b_i)}, \tag{12}$$

where $I(x, y)$ is an indicator function that returns one when the length of command x is y and zero otherwise.

5 Experimental Results

To empirically validate our approach, we collected natural language commands from users on Amazon Mechanical Turk (AMT) for a domain we created called *Cleanup World*. Cleanup World (like Sokoban) is a 2D grid world of various rooms connected by open doors. Rooms can also contain items that can be moved around. The agent moves using north-south-east-west actions. Moving into a location containing an item causes the item to move in the direction the agent is moving, as long as a wall or other item is not in its path. Cleanup World is represented as an OO-MDP consisting of four object classes: `AGENT`, `ITEM`, `ROOM`, and `DOORWAY`. The agent is defined by x and y position attributes. Items are defined by position attributes, a color attribute (which can be orange, teal, tan, yellow, or magenta), and a type attribute (star, moon, box, circle). The room and door objects are defined by attributes describing their rectangular bounding box (top, left, bottom, right), and the room object also has a color attribute. The propositional functions defined for the OO-MDP include `agentInRoom(AGENT, ROOM)`, and `itemInRoom(ITEM, ROOM)`, as well as propositional functions to indicate the color and type of rooms and items (e.g., `roomIsOrange(ROOM)`, `itemIsStar(ITEM)`).

We considered two different lifted tasks: the agent going to a specific room and the agent moving a block to a specific room. For these tasks, the behavior parameter τ was set to 0.005. To collect natural language instructions for different grounded versions of these tasks, we presented Turkers on AMT animated images showing either the agent moving to a room of a specific color or moving a star block to a room of a specific color. An example image from the animation is shown in Figure 3. We chose to use ambiguous colors for the rooms in our visual representation to elicit different verbalizations for the different colors. To prevent contamination of the commands we received, we never provided users with any example commands.

After removing labels that did not follow the instructions or which did not describe the task in any capacity, we obtained a dataset of 240 instances. The goal of our work is to be able to give autonomous agents high-level commands that leave the details of how to complete the task as a problem for the agent to solve. However, most of the natural language commands we received included details of the path

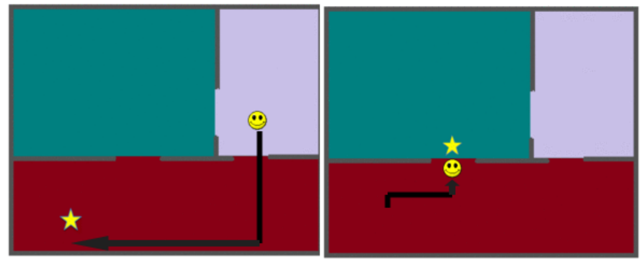

Figure 3: An example task to take a star to a room shown to users on AMT. The left frame shows the initial state and the path to the star; the right frame shows the path from there to the end state. (In the actual study, users were shown a single animated image without arrows.)

the agent followed, rather than only describing the task goal. Although this data is interesting because it tests our model's performance on language that it wasn't intended to model, a dataset that better reflects the problem we were trying to solve is also useful for comparison. Therefore, we created a simplified version of this dataset that omits the text that is extraneous to the task description. The average command lengths (in words) were 13.57 and 8.87 in the original and simplified datasets, respectively.

Performance was measured using leave-one-out (LOO) cross validation; a prediction was considered correct if the most likely grounded task inferred from a natural language command was the actual grounded task.

The LOO accuracy for each language model on each variant of the dataset is shown in Table 1. Both the BoW model and IBM Model 2 (IBM2) performed well on the simplified dataset. Although BoW had a slightly higher accuracy, the differences were not found to be statistically significant ($p > 0.5$ in a chi-squared test). The performance of both models was lower on the original full dataset than on the simplified dataset, which was expected since the original dataset had language that our task model was not designed to reflect. However, IBM2's performance only dropped by 7.15% (which was not a statistically significant drop; $p > 0.07$), whereas BoW dropped by 30.75% (which was a statistically significant drop; $p < 10^{-11}$). The difference in performance between the models on the original dataset was highly statistically significant ($p < 0.00001$). IBM2's superior performance to BoW on the original dataset was expected, because the original dataset often included instructions with many qualifiers. For instance, users described the agent's progress through various rooms and often remarked on the color of that room. For the BoW model, having multiple colors specified in the command made it impossible to disambiguate which color was associated with each semantic component, since the BoW model does not reason over any structure of the sentence. In contrast, in IBM2, the word position influences its association with each semantic word, which enabled IBM2 to pull out the parts of the text that were relevant to the task description. These results indicate that while BoW is effective for simple commands in which the words themselves are fully representative of the

	Simplified Dataset	Original Dataset
BoW	83.75%	53.00%
IBM2	81.25%	74.10%

Table 1: LOO accuracy for the AMT experiments.

task, IBM2 should be preferred for commands in which the structure of the sentence matters.

To demonstrate that IBM2 successfully learned the meaning of English words, we extracted IBM2's word parameters after it had finished training. The most likely English words generated from the semantic word "agentInRoom" (which was associated with the lifted "go to room" task), were "walk," "through," "move," "go," and "from." "From" and "through" occurred because in "go to room" tasks, users often described *from* which room to leave and often commanded the robot to go *through* a door to the goal room. For example, one of the commands provided was "walk through doorway from orange room to beige room."

Since the color of rooms was typically used to describe the goal room of both lifted tasks, the words associated with it are especially relevant. The "roomIsOrange" semantic word was mostly likely to generate the words "red" and "orange;" "roomIsTan" was most likely to generate the words "tan" and "beige;" and "roomIsTeal" was most likely to generate "green" and "blue."

6 Related Work

Our work relates to the broad class of methods for grounded language learning that aim to learn word meanings from a situated context. Instead of using annotated training data consisting of sentences and their corresponding semantic representations, (Kate and Mooney 2006; Wong and Mooney 2007; Zettlemoyer and Collins 2005; 2009), most of these approaches leverage non-linguistic information from a situated context as their primary source of supervision.

These approaches have been applied to various tasks, the one closest to ours being interpreting verbal commands in the context of navigational instructions (Vogel and Jurafsky 2010; Chen and Mooney 2011; Grubb et al. 2011), robot manipulation (Duvallet, Kollar, and Stentz 2013; Tellex et al. 2014), and puzzle solving and software control (Branavan, Zettlemoyer, and Barzilay 2010). Reinforcement learning has been applied to train a policy to follow natural language instructions for software control and map navigation (Vogel and Jurafsky 2010). However, our goal is to move away from low-level instructions that correspond directly to actions in the environment to high-level task descriptions expressed using complex language. Unlike previous approaches that learn the meanings of words from pairs of natural language instructions and demonstrations of corresponding high level actions (Tellex et al. 2014; Duvallet, Kollar, and Stentz 2013), we learn mappings of natural language instructions to task descriptions represented as MDP reward functions. In addition, our generative model allows an investigation of multiple language models that can be used with the task model. Besides the generally used Bag-of-Words model (Branavan et al. 2009; Vogel and Jurafsky 2010), we showed that a word-based statistical machine translation model provides better results. The idea of using statistical machine translation approaches for semantic parsing was introduced by Wong and Mooney (2007) (in a supervised learning setting). In future work, we plan to move beyond word-based SMT models such as IBM Model 2 to grammar-based SMT models such as Synchronous Context-Free Grammars (Wu 1997).

7 Conclusions and Future Work

We presented a novel generative task model that expresses tasks as factored MDP reward functions and terminal states, and to which natural language commands can be mapped. Mapping natural language commands to MDP tasks is a powerful approach because it allows people to provide agents with high-level commands without specifying the details of how to complete the tasks. This generative task model is flexible and can be combined with different language models. We also presented two grammar–free language models that can be combined with our task model: a Bag-of-Words mixture model and a variant of IBM Model 2 that treats semantic inference as a machine-translation problem. These models were empirically tested on a dataset in which commands were provided by users of the Amazon Mechanical Turk. We found that when commands are simple and can be fully represented by the words that are present in the command, both the BoW model and IBM Model 2 are effective at learning the meaning of words. However, when the meaning of the commands depends on the structure of the sentences, IBM Model 2 is more effective than BoW.

In the future, we will extend this approach to operate on a wider variety of task types. For instance, logical quantifiers such as "for all" and "there exists" can be used with the OO-MDP propositional functions to define more expressive tasks and can be incorporated into our language models by treating quantifiers as another semantic component (for the BoW model) or as a machine-language word (for IBM Model 2). In complex environments (especially in robotics applications), planning for a task can be difficult and expensive, so another future direction is to incorporate into our model the ability for a command to specify subtasks, which would make the overall planning problem more tractable. Finally, while we chose to use grammar-free language models because they require little background knowledge and can be easily trained from simple datasets of commands and demonstration, one could also combine a grammar-based model with our task model. Comparing the performance of grammar-based language models to our current grammar-free models will give important insights into when each approach may be appropriate.

References

Abbeel, P., and Ng, A. Y. 2004. Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the International Conference on Machine Learning*.

Babeş-Vroman, M.; Marivate, V.; Subramanian, K.; and Littman, M. 2011. Apprenticeship learning about multiple intentions. In *Proceedings of the Twenty Eighth International Conference on Machine Learning (ICML 2011)*.

Branavan, S. R. K.; Chen, H.; Zettlemoyer, L. S.; and Barzilay, R. 2009. Reinforcement learning for mapping instructions to actions. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1 - Volume 1*, ACL '09.

Branavan, S.; Zettlemoyer, L. S.; and Barzilay, R. 2010. Reading between the lines: Learning to map high-level instructions to commands. In *Association for Computational Linguistics (ACL 2010)*.

Brown, P. F.; Cocke, J.; Pietra, S. A. D.; Pietra, V. J. D.; Jelinek, F.; Lafferty, J. D.; Mercer, R. L.; and Roossin, P. S. 1990. A statistical approach to machine translation. *Comput. Linguist.* 16(2):79–85.

Brown, P. F.; Pietra, V. J. D.; Pietra, S. A. D.; and Mercer, R. L. 1993. The mathematics of statistical machine translation: Parameter estimation. *Comput. Linguist.* 19(2):263–311.

Chen, D. L., and Mooney, R. J. 2011. Learning to interpret natural language navigation instructions from observations. In *Proceedings of the 25th AAAI Conference on Artificial Intelligence (AAAI-2011).*, 859–865.

Dempster, A. P.; Laird, N. M.; and Rubin, D. B. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society* 39(1):1–38.

Diuk, C.; Cohen, A.; and Littman, M. 2008. An object-oriented representation for efficient reinforcement learning. In *Proceedings of the Twenty-Fifth International Conference on Machine Learning (ICML-08)*.

Duvallet, F.; Kollar, T.; and Stentz, A. T. 2013. Imitation learning for natural language direction following through unknown environments. In *IEEE International Conference on Robotics and Automation (ICRA)*.

Grubb, A.; Duvallet, F.; Tellex, S.; Kollar, T.; Roy, N.; Stentz, A.; and Bagnel, J. A. 2011. Imitation learning for natural language direction following. In *Proceedings of the ICML Workshop on New Developments in Imitation Learning*.

Kate, R. J., and Mooney, R. J. 2006. Using string-kernels for learning semantic parsers. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, ACL-44.

McCallum, A. 1999. Multi-label text classification with a mixture model trained by em. In *AAAI'99 Workshop on Text Learning*, 1–7.

Tellex, S.; Thaker, P.; Joseph, J.; and Roy, N. 2014. Learning perceptually grounded word meanings from unaligned parallel data. *Machine Learning* 94(2):205–232.

Vogel, A., and Jurafsky, D. 2010. Learning to follow navigational directions. In *Association for Computational Linguistics (ACL 2010)*.

Wong, Y. W., and Mooney, R. 2007. Learning synchronous grammars for semantic parsing with lambda calculus. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL-2007)*.

Wu, D. 1997. Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. *Computational Linguistics* 23(3):377–403.

Zettlemoyer, L. S., and Collins, M. 2005. Learning to map sentences to logical form: Structured classification with probabilistic categorial grammars. In *Proceedings of UAI-05*.

Zettlemoyer, L., and Collins, M. 2009. Learning context-dependent mappings from sentences to logical form. In *Proceedings of the Association for Computational Linguistics (ACL'09)*.

A Weakly Supervised IBM Model 2 Learning

Algorithm 1 Weakly Supervised IBM Model 2 Learning

Input: Dataset $D = \{(e^i, n_i, b_i, s_i)\}$ where each instance tuple is an english command, command length, demonstration, and input state, respectively. e^i_j is the jth word of command e^i.

for $s = 1$ to S **do** ▷ Run EM for S iterations
  $c(...) \leftarrow 0$ ▷ Set all counts to 0
  for $i = 1$ to $|D|$ **do**
    for $m \in M$ **do** ▷ Each machine command
      for $j = 1$ to n_i **do**
        for $u = 0$ to n_m **do**
          $c(m_u, e_j^{(i)})$ += $\delta(i, j, u, m)$
          $c(m_u)$ += $\delta(i, j, u, m)$
          $c(u|j, n_m, n_i)$ += $\delta(i, j, u, m)$
          $c(j, n_m, n_i)$ += $\delta(i, j, u, m)$
        end for
      end for
    end for
  end for
  $r(e_j|m_u) := \frac{c(m_u, e_j)}{c(m_u)}$
  $q(u|j, k, n) := \frac{c(u|j, n_m, n_i)}{c(j, n_m, n_i)}$
end for

B Amazon Mechanical Turk Datasets

In this section we present the AMT datasets that were used, both the original (Table 2) and the simplified version (Table 3). Note that the actual datasets were paris of english commands and trajectories. Since we cannot easily provide the trajectories for the commands, we instead list what the overall task description was using the OO-MDP propositional functions, similar what our task model would infer from the trajectory.

The Original AMT Dataset

cross path from green room to orange room.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
enter room with orange carpet.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go from room with green floor to one with orange floor	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go from teal room to orange room	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go from teal room to red room	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go through door into red room.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go through door leading into orange room and stop.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go through right door.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go through second door on left.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go to orange block through space on right side of blue block	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
move from blue color to red color	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
move from blue to orange, without going into tan.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go forward and to right into orange room.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
move from green tile to orange tile.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
move from green to orange	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
move from blue area by going up 4 times to peach area	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
move from blue room to orange room.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
move from green area to orange area.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
move from green area to orange area.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
move from green-floored room to orange-floored room.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
move in beige area first n go into orange area	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
proceed from blue room directly to orange room, and don't forget to smile !	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
start around middle left hand side of green-floored room. go through doorway to orange-floored room. stop just inside orange-floored room.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go from blue to orange diagonally upward.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
start at green and go to pink	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
start at right of blue rectangle, move diagonally towards right door, stand in threshold, n move through door into peach square and stand outside door	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
start from blue retangle and transfer to orange one.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
start in blue room and go through door to orange room	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
start in blue section, go up and to right, and n through door to orange section.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$

start in green room and move into orange space.	agentInRoom$(?a, ?r) \wedge$ roomIsOrange$(?r)$
start in greenish room. walk into orange room. go just past doorway. don't go all way to middle of room. n stop.	agentInRoom$(?a, ?r) \wedge$ roomIsOrange$(?r)$
start on blue area n go into orange area	agentInRoom$(?a, ?r) \wedge$ roomIsOrange$(?r)$
start on green spot and move to orange spot.	agentInRoom$(?a, ?r) \wedge$ roomIsOrange$(?r)$
start on turquoise rectangle and n walk up to orangey rectangle.	agentInRoom$(?a, ?r) \wedge$ roomIsOrange$(?r)$
go from blue room to orange room.	agentInRoom$(?a, ?r) \wedge$ roomIsOrange$(?r)$
go from bluish green room to orange room.	agentInRoom$(?a, ?r) \wedge$ roomIsOrange$(?r)$
go from cyan room at bottom, to top right orange room. do not go through flesh colored room.	agentInRoom$(?a, ?r) \wedge$ roomIsOrange$(?r)$
go from green room into orange room using doorway	agentInRoom$(?a, ?r) \wedge$ roomIsOrange$(?r)$
go from green room to orange room	agentInRoom$(?a, ?r) \wedge$ roomIsOrange$(?r)$
go from green room to orange room	agentInRoom$(?a, ?r) \wedge$ roomIsOrange$(?r)$
walk from orange colored room where you start at to left into beige colored room.	agentInRoom$(?a, ?r) \wedge$ roomIsTan$(?r)$
walk from orange room to brown room, in anor words take door on right to brown room.	agentInRoom$(?a, ?r) \wedge$ roomIsTan$(?r)$
start in red section and go through door to tan section.	agentInRoom$(?a, ?r) \wedge$ roomIsTan$(?r)$
go from orange room to olive room	agentInRoom$(?a, ?r) \wedge$ roomIsTan$(?r)$
take a couple steps into tan room and stop.	agentInRoom$(?a, ?r) \wedge$ roomIsTan$(?r)$
go through left door	agentInRoom$(?a, ?r) \wedge$ roomIsTan$(?r)$
please walk to white background area to left, from orange background area you are presently staying at.	agentInRoom$(?a, ?r) \wedge$ roomIsTan$(?r)$
walk from orange room to left through entryway to tan room	agentInRoom$(?a, ?r) \wedge$ roomIsTan$(?r)$
walk from orange room into beige room	agentInRoom$(?a, ?r) \wedge$ roomIsTan$(?r)$
move from orange to yellow	agentInRoom$(?a, ?r) \wedge$ roomIsTan$(?r)$
walk from orange square to beige square by walking a straight line to left.	agentInRoom$(?a, ?r) \wedge$ roomIsTan$(?r)$
start at orange and go to beige.	agentInRoom$(?a, ?r) \wedge$ roomIsTan$(?r)$
you would start in bottom left of orange room. you would n walk slightly into tan room. you would not go into green room.	agentInRoom$(?a, ?r) \wedge$ roomIsTan$(?r)$
walk into tan room	agentInRoom$(?a, ?r) \wedge$ roomIsTan$(?r)$
re is a doorway to left of you and below you. take doorway to left. floor should be a pale peach color.	agentInRoom$(?a, ?r) \wedge$ roomIsTan$(?r)$
walk straight into tan room.	agentInRoom$(?a, ?r) \wedge$ roomIsTan$(?r)$

go in orange room n walk left to beige room	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTan}(?r)$
move from orange area into tan area.	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTan}(?r)$
walk from orange room to yellow room	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTan}(?r)$
start in orange room walk left through door a few steps into tan room	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTan}(?r)$
move from orange to tan	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTan}(?r)$
start in orange-floored room. go through doorway to beige-floored room. stop when you are just a little bit inside.	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTan}(?r)$
from where you stand, take three steps towards and into yellow room.	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTan}(?r)$
walk through door in longer side of room, possibly to your left, and stand in tan room.	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTan}(?r)$
walk through doorway from orange room to beige room.	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTan}(?r)$
move from orange room to beige room.	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTan}(?r)$
go through door into white room.	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTan}(?r)$
walk from red room to beige room.	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTan}(?r)$
start in orange room. walk into beige room. n stop.	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTan}(?r)$
walk from orange area to brown area.	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTan}(?r)$
move from orange to tan	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTan}(?r)$
from pink to whlite	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTan}(?r)$
go from orange room to skin coloured room	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTan}(?r)$
walk into beige room	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTan}(?r)$
from orange to salmon	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTan}(?r)$
in orange rectangle facing door, walk straight until you are inside tan rectangle.	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTan}(?r)$
walk on orange n beige tile to your left.	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTan}(?r)$
start on orange and move to beige	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTan}(?r)$
move straight to white color	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTan}(?r)$
go west from orange to cream	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTan}(?r)$
brown to grey	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTeal}(?\texttt{r})$
by walking from yellow to green	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTeal}(?\texttt{r})$
go from flesh colored room down to cyan room. do not cut through orange room.	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTeal}(?\texttt{r})$
go from olive room to green room	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTeal}(?\texttt{r})$
go from white room to blue room. door is on bottom.	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTeal}(?\texttt{r})$
go through door below on left	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTeal}(?\texttt{r})$
go through door into blue room and stop.	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTeal}(?\texttt{r})$
go through door on right into blue room.	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTeal}(?\texttt{r})$
go through door on south wall.	$\texttt{agentInRoom}(?a, ?r) \land \texttt{roomIsTeal}(?\texttt{r})$

go through door straight ahead and enter room with green floor.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
go to yellow color first and n green	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
in tan rectangle, face door that leads into blue rectangle. walk straight, taking shortest path that leads through door into blue rectangle.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
creme n green	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
move diagonaly from beige block to deep blue block	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
move down diagonally from tan area to green area.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
move downwards to next box	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
move from cream tile to green tile.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
move from tan to blue	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
move from cream-floored room to green-floored room.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
move from white to blue color	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
move from yellow to green	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
move south through first passageway in wall below you. once you go through passage, look down, floor should be blue.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
move to lower rectangle	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
go diagonally towards south-east from cream to blue	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
please go from beige area to green area without passing through orange area	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
please walk through doorway into bluegreen room.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
start at beige and go to green.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
start from beige place, n go to green place.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
start from beige retangle and tranfser to blue retangle.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
start in beige area and move into green area and stop.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
start in beige room. walk into greenish room. go just past doorway. don't go all way to middle of room. n stop.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
start in beige square and n move downwards through space into blue square and stop.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
start in tan room and n go through door into blue room.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
start in tan section, and n go down through door to blue section.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
go forward and to left into blue room	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
go from brown to green	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
go from beige area to green one	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
go from beige carpeted room to teal carpeted room.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$

go from beige room directly into blue room	agentInRoom$(?a, ?r) \wedge$ roomIsTeal(?r)
go from blue room to tan room and n stop.	agentInRoom$(?a, ?r) \wedge$ roomIsTeal(?r)
enter flesh colored room from cyan room and push star into orange room. be sure not to go into orange room yourself, just star.	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
exit room with blue floor through left door, and enter room with white / cream floor. you will see a large gold star near a door in right hand wall. push star through that door. it should enter a room with a red floor. do not enter room yourself once you have pushed star into it.	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
go through door into tan room and push star just inside orange room and stop.	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
go through door on left into white room, round up star and push it into red room.	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
go through far left door into white room. get star and move it through door into red room.	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
go through left door and carry object into or room.	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
go through left opening and get star, push to right through opening, but stop. do not enter.	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
go through left opening into beige room and left of star, and push it straight into red room	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
go thru left door into tan room and push star into orange room	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
go to 'beige' room, and push star into 'red' room.	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
go up from green room through entry to tan room and approach star. push star right through entry to orange room	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
in blue rectangle, face door that leads into tan rectangle. walk straight into tan rectangle and stop to left of star, on opposite side of star as orange rectangle. n, push star into orange rectangle to point that star is in orange but you are still in tan.	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
facing doors, walk into beige room (which is door on left side, if you're facing doors). when you see star, push it through or door (orange room) but do not step into room !	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
move from green tile to cream tile and push star to orange tile.	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$

move from blue room to tan room. move star into orange room without setting foot into orange room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
move from blue space to tan space and push star into red space.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
move on from blue color to white where way is, and n go along with star to reach orange.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
move star from blue section to grey. n move it into orage section.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
move up from green area to left side of star in tan area. push star into orange area.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
push star from olive room to orange room	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
push star into orange section in least moves possible.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
push star to brown color box	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
start around middle of green-floored room. go towards doorway to beige-floored room. push star directly into orange-floored room, but do not enter.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
from blue room, walk into tan room up and to left. push star into orange room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
start from blue retangle and transfer to beige one. while doing this, drag star to ornge box ; while you're still in beige retangle.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
start in blue room, go into tan room and push star into orange room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
start in blue section, go up to tan and push star into orange / red section.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
start in greenish room. go into beige room. pick up star. take it to doorway to orange room. keep your body in beige room, but put star just inside orange room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
start on blue area n go into tan area till you get star in orange area	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
start on right hand side of blue square and move up and to left in a diagonal direction, into beige square. next, go to right, pushing star until it passes into red square and n stop while still in beige one.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
start with blue.... go to left to beige.... put star in front of you.... move him to orange.... dont cross over to orange..... just star	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
starting in green room, walk into orange room and push star into beige room without actually entering it.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$

take doorway on your left. when you enter room re should be a big star. push star through doorway to your right. you do not have to completely enter that room.	`blockInRoom`$(?b, ?r)$ $\wedge$ `itemIsStar`$(?b)$ $\wedge$ `roomIsOrange`$(?r)$
walk from blue room to beige room. push star from beige room into pink room.	`blockInRoom`$(?b, ?r)$ $\wedge$ `itemIsStar`$(?b)$ $\wedge$ `roomIsOrange`$(?r)$
from green-floored room, take star from cream-floored room to orange room, but do not enter orange room.	`blockInRoom`$(?b, ?r)$ $\wedge$ `itemIsStar`$(?b)$ $\wedge$ `roomIsOrange`$(?r)$
go forward and left into beige room, take star and put it in orange room while remaining in beige room.	`blockInRoom`$(?b, ?r)$ $\wedge$ `itemIsStar`$(?b)$ $\wedge$ `roomIsOrange`$(?r)$
go from blue room to beige room. move star into orange room without going into it yourself.	`blockInRoom`$(?b, ?r)$ $\wedge$ `itemIsStar`$(?b)$ $\wedge$ `roomIsOrange`$(?r)$
go into beige room and push star into orange room.	`blockInRoom`$(?b, ?r)$ $\wedge$ `itemIsStar`$(?b)$ $\wedge$ `roomIsOrange`$(?r)$
go into white room throught door on left and push star into red room.	`blockInRoom`$(?b, ?r)$ $\wedge$ `itemIsStar`$(?b)$ $\wedge$ `roomIsOrange`$(?r)$
go through center opening into beige enclosure and get behind star and push it into opening of orange enclosure	`blockInRoom`$(?b, ?r)$ $\wedge$ `itemIsStar`$(?b)$ $\wedge$ `roomIsOrange`$(?r)$
bring star to beige room	`blockInRoom`$(?b, ?r)$ $\wedge$ `itemIsStar`$(?b)$ $\wedge$ `roomIsTan`$(?r)$
enter green room, pick up star and go to doorway to tan room.	`blockInRoom`$(?b, ?r)$ $\wedge$ `itemIsStar`$(?b)$ $\wedge$ `roomIsTan`$(?r)$
go from orange room to blue room, get star and take it into light room.	`blockInRoom`$(?b, ?r)$ $\wedge$ `itemIsStar`$(?b)$ $\wedge$ `roomIsTan`$(?r)$
go from orange room to blue room, pick up star, and move it into cream colored room.	`blockInRoom`$(?b, ?r)$ $\wedge$ `itemIsStar`$(?b)$ $\wedge$ `roomIsTan`$(?r)$
go from orange room to green room, pick up gold star, and walk it into beige room.	`blockInRoom`$(?b, ?r)$ $\wedge$ `itemIsStar`$(?b)$ $\wedge$ `roomIsTan`$(?r)$
go from orange room, n to green room, n pick up star, and bring it to tan room	`blockInRoom`$(?b, ?r)$ $\wedge$ `itemIsStar`$(?b)$ $\wedge$ `roomIsTan`$(?r)$
go get that star in blue room and bring it back to orange room by beige room.	`blockInRoom`$(?b, ?r)$ $\wedge$ `itemIsStar`$(?b)$ $\wedge$ `roomIsTan`$(?r)$
go get star and bring it to peach colored section.	`blockInRoom`$(?b, ?r)$ $\wedge$ `itemIsStar`$(?b)$ $\wedge$ `roomIsTan`$(?r)$
go into blue room and pick up star n carry it into beige room.	`blockInRoom`$(?b, ?r)$ $\wedge$ `itemIsStar`$(?b)$ $\wedge$ `roomIsTan`$(?r)$
go into blue room, pick up star, and take star just through door to tan room.	`blockInRoom`$(?b, ?r)$ $\wedge$ `itemIsStar`$(?b)$ $\wedge$ `roomIsTan`$(?r)$
go into green room (hallway ?), take star re, and go to beige room.	`blockInRoom`$(?b, ?r)$ $\wedge$ `itemIsStar`$(?b)$ $\wedge$ `roomIsTan`$(?r)$
go into green room and bring star into tan room.	`blockInRoom`$(?b, ?r)$ $\wedge$ `itemIsStar`$(?b)$ $\wedge$ `roomIsTan`$(?r)$

exit door to left, enter room with teal floor, walk around room to star and take that star to or room with cream floor.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
go into green room and get star, n take star to cream room	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
go into green room and push star into white room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
go straight down and through open door. go all way to left of star. push star up and a little to right through door. stop when you are just inside door.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
go straight, enter green room, turn right, pick up star, and push it into yellow room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
go to blue room and bring star to door of tan room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
go to blue room and move star to orange room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
go to grey room and move star to orange room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
leave orange room and enter turquoise one. grab star and stand in doorway of turquoise and cream colored rooms while still holding star.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
leave red-floored room and enter blue-floored room. approach star and push it through doorway of tan-floored room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
leave room you're in, go to green room, get star, and go to beige room with star	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
exit orange colored room and enter teal room. pick up star and carry it to doorway of white room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
move from peach block to blue block, grab star and move it just through entrance of tan block.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
move from red area to turquoise area, pick up star, and move it into tan area.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
move from tangerine space to teal space, n push star into beige space.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
move into blue room. take star into tan room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
move into blue room. push star into white room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
move star up from blue area to tan area through opening between those two colors.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
move through rooms clockwise, push star through door into tan room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
move to star and n bump it into tan-colored area.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$

pick star inside blue room and bring it directly to beige room	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
proceed to green room. grab star and push it just past door to yellow room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
exit red room into blue room. turn right. go forward and pick up star. take star out into yellow room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
from orange, move south into green room. position yourself to bottom left of yellow star to grab it, n move north into doorway of tan room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
get star from green room and bring it to beige room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
go down to blue-green block, get star and take star to beige block above.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
go down towards opening at bottom of reddish box, go through opening. n go to left towards star, going around it and pushing it upwards towards opening to tan section. push star all way through door, and stop in doorway between teal and tan sections.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
go from orange room to blue room where you push star into beige room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
enter room to west and puch star into room to south	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
from orange room, walk into tan room on left. push star into blue room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
go from orange spot to beige spot push star in to green spot but do not enter green spot stay in beige spot	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
go into beige room and push star into green room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
go into beige room from orange room. put star in center of beige room into teal room	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
go move star into bottom box	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
go out exit to your left and push star straight out of exit to dark blue room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
go through door, take star, and put it in blue room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
go through left door and knock star down through door below	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
go to beige room and push star into blue room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
go to left n push star to door of blue room	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
go to top right room, and push star down to cyan room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$

from orange room, walk to tan room and push star into teal room without walking into teal room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
in orange rectangle, face door that leads into tan rectangle. walk straight through door until you are above star, in direction opposite blue rectangle. n, push star down, towards blue rectangle, until star goes into blue but you are still standing in tan.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
leave orange room and go into beige room. push star into teal room without crossing doorway into room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
move from orange tile to cream tile and push star to green tile.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
move from orange to tan, push star into blue without entering blue	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
move from right to left from red into beige. approach star from right so you end up just above it. push star directly down until star reaches blue.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
move from orange area to yellow area and push star onto green area	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
move from orange to above star, n push it down into green.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
move happy in a straight line from red room on top right, through door, to beige room on top left, so that happy is positioned directly above star. use happy to push star downwards just until star crosses through door into red room on bottom.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
move into white room. get star and take it into blue room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
move out of orange room through opening in left and go into tan room. move behind star and n push star down through opening into blue room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
from orange-floored room, take star from cream-floored room to green room, but do not enter green room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
move star from beige room to blue room	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
push star from olive room to green room	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
push star through door on left, n door directly below	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
shift to your right box and move star down to next box	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
start at orange, move to tan - bring star to green.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
start in orange box, go into peach box, n push star into blue box.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$

start in orange room. go into beige room. pick up star. take it to doorway of greenish room. put star just inside greenish room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
start in orange-floored room. go through doorway to beige-floored room. push star directly down through to green-floored room. do not enter green-floored room ; stop just at doorway.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
start in red section. go through door to right and go into red section. push star down into blue section.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
starting in orange walk left and slightly up, into tan just above star and stop. push star straight down from tan into green and stop just as star passes through door.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
go forward into orange room, take star and put it in blue room while you remain in beige room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
go from orange to cream towards star diagonally north - west direction and push star downwards south till blue	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
go from orange area into beige area and push star to blue area.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
go from orange room to cream room. go up to star. push it into green room, but do not actually enter green room. stop at doorway.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
go from orange room to skin coloured room and n push star to teal room whilst still in skin coloured room	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
go from orange room to star in beige room. move star into blue room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$

Table 2: The original AMT Dataset.

Simplified Dataset

cross path to orange room.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
enter room with orange carpet.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go to one with orange floor	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go to orange room	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go to red room	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go through door into red room.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go through door leading into orange room and stop.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go through right door.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go through second door on left.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go to orange block through space	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
move to red color	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
move to orange, without going into tan.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$

go forward and to right into orange room.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
move to orange tile.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
move to orange	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
move by going up 4 times to peach area	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
move to orange room.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
move to orange area.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
move to orange area.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
move to orange-floored room.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go into orange area	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
proceed to orange room, and don't forget to smile !	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go through doorway to orange-floored room. stop just inside orange-floored room.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go to orange diagonally upward.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go to pink	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
move diagonally towards right door, stand in threshold, n move through door into peach square and stand outside door	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
transfer to orange one.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go through door to orange room	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go up and to right, and n through door to orange section.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
move into orange space.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
walk into orange room. go just past doorway. don't go all way to middle of room. n stop.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go into orange area	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
move to orange spot.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
walk up to orangey rectangle.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go to orange room.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go to orange room.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go to top right orange room. do not go through flesh colored room.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go into orange room using doorway	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go to orange room	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
go to orange room	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsOrange}(?r)$
walk to left into beige colored room.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTan}(?r)$
walk to brown room, in anor words take door on right to brown room.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTan}(?r)$
go through door to tan section.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTan}(?r)$
go to olive room	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTan}(?r)$
take a couple steps into tan room and stop.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTan}(?r)$
go through left door	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTan}(?r)$
please walk to white background area to left.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTan}(?r)$
walk to left through entryway to tan room	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTan}(?r)$
walk into beige room	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTan}(?r)$
move to yellow	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTan}(?r)$

walk to beige square by walking a straight line to left.	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
go to beige.	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
you would n walk slightly into tan room.	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
walk into tan room	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
re is a doorway to left of you and below you. take doorway to left. floor should be a pale peach color.	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
walk straight into tan room.	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
walk left to beige room	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
move into tan area.	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
walk to yellow room	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
walk left through door a few steps into tan room	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
move to tan	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
go through doorway to beige-floored room. stop when you are just a little bit inside.	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
from where you stand, take three steps towards and into yellow room.	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
walk through door in longer side of room, possibly to your left, and stand in tan room.	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
walk through doorway to beige room.	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
move to beige room.	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
go through door into white room.	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
walk to beige room.	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
walk into beige room. n stop.	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
walk to brown area.	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
move to tan	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
from to white	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
go to skin coloured room	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
walk into beige room	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
to salmon	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
walk straight until you are inside tan rectangle.	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
walk on orange n beige tile to your left.	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
move to beige	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
move straight to white color	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
go west to cream	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTan}(?r)$
to grey	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTeal(?r)}$
by walking to green	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTeal(?r)}$
go down to cyan room.	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTeal(?r)}$
go to green room	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTeal(?r)}$
go to blue room. door is on bottom.	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTeal(?r)}$
go through door below on left	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTeal(?r)}$
go through door into blue room and stop.	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTeal(?r)}$
go through door on right into blue room.	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTeal(?r)}$
go through door on south wall.	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTeal(?r)}$
go through door straight ahead and enter room with green floor.	$\mathtt{agentInRoom}(?a, ?r) \wedge \mathtt{roomIsTeal(?r)}$

go to green	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
face door that leads into blue rectangle. walk straight, taking shortest path that leads through door into blue rectangle.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
creme n green	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
move diagonaly to deep blue block	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
move down diagonally to green area.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
move downwards to next box	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
move to green tile.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
move to blue	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
move to green-floored room.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
move to blue color	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
move to green	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
move south through first passageway in wall below you. once you go through passage, look down, floor should be blue.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
move to lower rectangle	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
go diagonally towards south-east to blue	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
please go to green area	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
please walk through doorway into bluegreen room.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
go to green.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
go to green place.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
tranfser to blue retangle.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
move into green area and stop.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
walk into greenish room. go just past doorway. don't go all way to middle of room. n stop.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
move downwards through space into blue square and stop.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
go through door into blue room.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
go down through door to blue section.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
go forward and to left into blue room	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
go to green	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
go to green one	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
go to teal carpeted room.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
go directly into blue room	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
go to tan room and n stop.	$\texttt{agentInRoom}(?a, ?r) \wedge \texttt{roomIsTeal}(?\texttt{r})$
enter flesh colored room and push star into orange room. be sure not to go into orange room yourself, just star.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
exit through left door. you will see a large gold star near a door in right hand wall. push star through that door. it should enter a room with a red floor. do not enter room yourself once you have pushed star into it.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
go through door and push star just inside orange room and stop.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$

go through door on left, round up star and push it into red room.	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
go through far left door. get star and move it through door into red room.	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
go through left door and carry object into or room.	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
go through left opening and get star, push to right through opening, but stop. do not enter.	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
go through left opening and left of star, and push it straight into red room	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
go thru left door and push star into orange room	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
push star into red room.	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
approach star. push star right through entry to orange room	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
walk straight and stop to left of star, on opposite side of star as orange rectangle. n, push star into orange rectangle to point that star is in orange but you are still in tan.	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
facing doors, walk into room (which is door on left side, if you're facing doors). when you see star, push it through or door (orange room) but do not step into room !	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
push star to orange tile.	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
move star into orange room without setting foot into orange room.	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
push star into red space.	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
go along with star to reach orange.	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
move star into orange section.	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
push star into orange area.	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
push star to orange room	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
push star into orange section in least moves possible.	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
push star to brown color box	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
push star directly into orange-floored room, but do not enter.	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
push star into orange room.	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
drag star to orange box ; while you're still in beige rectangle.	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
push star into orange room.	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$
push star into orange / red section.	blockInRoom$(?b, ?r)$ $\wedge$ itemIsStar$(?b)$ $\wedge$ roomIsOrange$(?r)$

pick up star. take it to doorway to orange room. keep your body in beige room, but put star just inside orange room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
go till you get star in orange area	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
go to right, pushing star until it passes into red square.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
put star in front of you.... move him to orange.... dont cross over to orange..... just star	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
push star into beige room without actually entering it.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
take doorway on your left. when you enter room re should be a big star. push star through doorway to your right. you do not have to completely enter that room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
push star into pink room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
take star to orange room, but do not enter orange room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
go forward and left, take star and put it in orange room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
move star into orange room without going into it yourself.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
push star into orange room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
go into throught door on left and push star into red room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
go through center opening and get behind star and push it into opening of orange enclosure	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsOrange}(?r)$
bring star to beige room	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
pick up star and go to doorway to tan room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
get star and take it into light room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
pick up star, and move it into cream colored room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
pick up gold star, and walk it into beige room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
pick up star, and bring it to tan room	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
go get that star and bring it back to orange room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
go get star and bring it to peach colored section.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
pick up star n carry it into beige room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
pick up star, and take star just through door to tan room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
take star re, and go to beige room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
bring star into tan room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$

exit door to left, walk around room to star and take that star to or room with cream floor.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
get star, n take star to cream room	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
push star into white room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
go straight down and through open door. go all way to left of star. push star up and a little to right through door. stop when you are just inside door.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
go straight, turn right, pick up star, and push it into yellow room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
bring star to door of tan room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
move star to orange room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
move star to orange room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
grab star and stand in doorway of turquoise and cream colored rooms while still holding star.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
approach star and push it through doorway of tan-floored room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
leave room you're in, get star, and go to beige room with star	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
pick up star and carry it to doorway of white room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
grab star and move it just through entrance of tan block.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
pick up star, and move it into tan area.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
push star into beige space.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
take star into tan room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
push star into white room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
move star up to tan area through opening between those two colors.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
move through rooms clockwise, push star through door into tan room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
move to star and n bump it into tan-colored area.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
pick star and bring it directly to beige room	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
grab star and push it just past door to yellow room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
go forward and pick up star. take star out into yellow room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
position yourself to bottom left of yellow star to grab it, n move north into doorway of tan room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
get star and bring it to beige room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$

get star and take star to beige block above.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
go through opening. n go to left towards star, going around it and pushing it upwards towards opening to tan section. push star all way through door, and stop in doorway between sections.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
push star into beige room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTan}(?r)$
enter room to west and push star into room to south	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
push star into blue room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
push star in to green spot but do not enter green spot	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
push star into green room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
put star into teal room	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
go move star into bottom box	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
go out exit to your left and push star straight out of exit to dark blue room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
go through door, take star, and put it in blue room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
go through left door and knock star down through door below	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
push star into blue room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
go to left n push star to door of blue room	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
go to top right room, and push star down to cyan room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
push star into teal room without walking into teal room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
walk straight through door until you are above star, push star down, towards blue rectangle, until star goes into blue.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
push star into teal room without crossing doorway into room.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
push star to green tile.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
push star into blue without entering blue	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
approach star from right so you end up just above it. push star directly down until star reaches blue.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
push star onto green area	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$
move to above star, n push it down into green.	$\texttt{blockInRoom}(?b, ?r) \wedge \texttt{itemIsStar}(?b) \wedge \texttt{roomIsTeal}(?r)$

move happy so that happy is positioned directly above star. use happy to push star downwards just until star crosses through door into red room on bottom.	$\texttt{blockInRoom}(?b, ?r) \land \texttt{itemIsStar}(?b) \land \texttt{roomIsTeal}(?r)$
get star and take it into blue room.	$\texttt{blockInRoom}(?b, ?r) \land \texttt{itemIsStar}(?b) \land \texttt{roomIsTeal}(?r)$
move behind star and n push star down through opening into blue room.	$\texttt{blockInRoom}(?b, ?r) \land \texttt{itemIsStar}(?b) \land \texttt{roomIsTeal}(?r)$
take star to green room, but do not enter green room.	$\texttt{blockInRoom}(?b, ?r) \land \texttt{itemIsStar}(?b) \land \texttt{roomIsTeal}(?r)$
move star to blue room	$\texttt{blockInRoom}(?b, ?r) \land \texttt{itemIsStar}(?b) \land \texttt{roomIsTeal}(?r)$
push star to green room	$\texttt{blockInRoom}(?b, ?r) \land \texttt{itemIsStar}(?b) \land \texttt{roomIsTeal}(?r)$
push star through door on left, n door directly below	$\texttt{blockInRoom}(?b, ?r) \land \texttt{itemIsStar}(?b) \land \texttt{roomIsTeal}(?r)$
shift to your right box and move star down to next box	$\texttt{blockInRoom}(?b, ?r) \land \texttt{itemIsStar}(?b) \land \texttt{roomIsTeal}(?r)$
bring star to green.	$\texttt{blockInRoom}(?b, ?r) \land \texttt{itemIsStar}(?b) \land \texttt{roomIsTeal}(?r)$
push star into blue box.	$\texttt{blockInRoom}(?b, ?r) \land \texttt{itemIsStar}(?b) \land \texttt{roomIsTeal}(?r)$
pick up star. put star just inside greenish room.	$\texttt{blockInRoom}(?b, ?r) \land \texttt{itemIsStar}(?b) \land \texttt{roomIsTeal}(?r)$
push star directly down through to green-floored room. do not enter green-floored room ; stop just at doorway.	$\texttt{blockInRoom}(?b, ?r) \land \texttt{itemIsStar}(?b) \land \texttt{roomIsTeal}(?r)$
go through door to right. push star down into blue section.	$\texttt{blockInRoom}(?b, ?r) \land \texttt{itemIsStar}(?b) \land \texttt{roomIsTeal}(?r)$
walk left and slightly up, just above star and stop. push star straight down into green and stop just as star passes through door.	$\texttt{blockInRoom}(?b, ?r) \land \texttt{itemIsStar}(?b) \land \texttt{roomIsTeal}(?r)$
take star and put it in blue room.	$\texttt{blockInRoom}(?b, ?r) \land \texttt{itemIsStar}(?b) \land \texttt{roomIsTeal}(?r)$
push star downwards south till blue	$\texttt{blockInRoom}(?b, ?r) \land \texttt{itemIsStar}(?b) \land \texttt{roomIsTeal}(?r)$
push star to blue area.	$\texttt{blockInRoom}(?b, ?r) \land \texttt{itemIsStar}(?b) \land \texttt{roomIsTeal}(?r)$
go up to star. push it into green room, but do not actually enter green room. stop at doorway.	$\texttt{blockInRoom}(?b, ?r) \land \texttt{itemIsStar}(?b) \land \texttt{roomIsTeal}(?r)$
push star to teal room whilst still in skin coloured room	$\texttt{blockInRoom}(?b, ?r) \land \texttt{itemIsStar}(?b) \land \texttt{roomIsTeal}(?r)$
go to star. move star into blue room.	$\texttt{blockInRoom}(?b, ?r) \land \texttt{itemIsStar}(?b) \land \texttt{roomIsTeal}(?r)$

Table 3: The simplified AMT Dataset.