Coherence, Symbol Grounding and Interactive Task Learning

Mattias Appelgren
University of Edinburgh
M.R.Appelgren@sms.ed.ac.uk

Alex Lascarides
University of Edinburgh
alex@inf.ed.ac.uk

Abstract

To teach agents through natural language interaction, we need methods for updating the agent's knowledge, given a teacher's feedback. But natural language is ambiguous at many levels and so a major challenge is for the agent to disambiguate the intended message, given the signal and the context in which it's uttered. In this paper we look at how coherence relations can be used to help disambiguate the teachers' feedback and so contribute to the agent's reasoning about how to solve their domain-level task. We conduct experiments where the agent must learn to build towers that comply with a set of rules, which the agent starts out ignorant of. It is also unaware of the concepts used to express the rules. We extend a model for learning these tasks which is based on coherence and show experimentally that our extensions can improve how fast the agent learns.

1 Introduction

Many commercial scenarios create planning problems consisting of goal conditions which are complex and vaguely specified. An example is problems created by Standard Operating Procedures (SOPs)—large manuals containing instructions and rules which workers must follow. In companies such as Amazon or Ocado these feature rules such as "make sure the box is properly sealed" or "never put frozen items in the same bag as meat products".

Building a precise formal representation of such problems which supports inference and planning is a challenging task for two reasons. Firstly, the array of contingencies where SOPs apply may be so extensive that it is untenable for a domain expert to communicate all these possibilities to a software developer; and secondly, the SOPs often change in unforeseen ways (such as in bespoke manufacturing or large online retail where product lines are highly dynamic), making previously irrelevant concepts become relevant. For example, a company that starts to sell batteries must ensure the labels are put to the left rather than right of the package (this is a SOP in Amazon (Personal Communication)). This spatial relation may not have been part of the original domain specification, but an agent that had to follow this rule would now have to refine their domain model to include it, and learn what the word "left" means.

Since communicating the current SOPs is difficult and they change periodically, it would be useful for the domain expert to be able to teach the agent personally, after the agent has been deployed. A natural way to do so is through a teacher-apprentice interaction where the teacher observes the apprentice attempting to complete the task, reacting when the apprentice performs actions inconsistent with the SOPs. This way of teaching is simpler on the teacher since it is easier to react to a situation than predicting all contingencies in advance. The apprentice, in this situation, must have the capacity to learn the constraints as well as new concepts which were not previously a part of their domain model.

In this paper we tackle a task which is analogous to, but simpler than, SOP compliant packing. Instead of rules referring to weight or fragility ("don't put heavy things above eggs" or "protect the vase with bubble wrap because it is fragile"), in our task the agent must learn and reason about constraints in a blocks world where colour is a proxy for these concepts (e.g. "put red blocks on blue blocks"). The agent starts out with a domain model with no colour concepts, nor does it have any colour terms within its natural language vocabulary. It must learn from a teacher both the rules that constrain the task, and how to ground the previously unknown colour terms (which pop-

Proceedings of the 23rd Workshop on the Semantics and Pragmatics of Dialogue, September 4-6, 2019, London, U.K.

ulate the rules).

This work extends the task and agent in Appelgren and Lascarides (2019) where an agent learns from a simulated teacher's corrective feedback. They build a graphical model that captures the semantics of correction. This allows the agent to learn to recognise colours and learn which constraints are a part of the goal. We address two shortcomings of their paper by: 1) utilising the evidence that the teacher has *not* corrected the agent's latest action, and 2) extending the model to capture extended dialogue, allowing us to deal with anaphoric expressions, which are ubiquitous in spontaneous natural language interactions.

2 Related Work

Teaching agents through interaction is a central theme in areas such as Learning through Demonstration (Argall et al., 2009), advice giving (Maclin and Shavlik, 1996; Kuhlmann et al., 2004; Benavent and Zanuttini, 2018), and learning reward functions in Reinforcement Learning (Christiano et al., 2017; Hadfield-Menell et al., 2016). However, the area that shares our goals most is Interactive Task Learning (ITL) (Laird et al., 2017).

ITL focuses on teaching agents the parameters or rules which govern a task, rather than optimising a known task (such as in Reinforcement Learning), through interaction with a teacher (e.g. (Scheutz et al., 2017; Lindes et al., 2017; She et al., 2014; Chai, 2018))). The main contribution of our work and of Appelgren and Lascarides (2019) is to extend the types of interaction which teachers perform beyond instructions and definitions, with a focus in this paper on correction and elaboration. Correction has only been studied with use of very simple language; e.g. "no" (Nicolescu and Mataric, 2003).

The goal in our task is to learn to identify valid sequential plans autonomously, as opposed to learning how to perform new actions by combining primitive actions (Chai, 2018; Scheutz et al., 2017) or learning low level motor control directly (Knox and Stone, 2009). The agent must also refine its domain model with unforeseen concepts that are discoverd through interaction, as opposed to having a full domain conceptualisation and needing only to learn to map language onto these known concepts (contra Wang et al. (2016); Kuhlmann et al. (2004)). To do this language grounding we follow an approach where individual classifiers are trained for each concept (Matuszek, 2018).

Figure 1: The shades used for blocks within each colour category.

3 Task

Agents must learn a tower building task in the blocks world. Each scenario consists of 10 coloured blocks that must be placed into a tower. The resulting tower must conform to a set of constraints, or rules, which are part of the task's goal description, G. In this paper we consider rules of two forms:

$$r_1^{c_1,c_2} = \forall x.c_1(x) \rightarrow \exists y.c_2(y) \wedge on(x,y) \quad (1)$$
$$r_2^{c_1,c_2} = \forall y.c_2(y) \rightarrow \exists x.c_1(x) \wedge on(x,y) \quad (2)$$

where c_1 and c_2 are colours (e.g., red, blue, maroon).

The task is implemented in a virtual environment, where each scenario is defined in the Planning Domain Definition Language (PDDL). Agents interact with the world through the action $put(x,y)$, which simply places object x on object y. In each scenario, the agent must build a tower consistent with G. However, it begins ignorant of the specific constraints that define G. Further, the agent can see what blocks exist and their spatial relation to each other, but it is unaware of what colour terms are used to describe them. Instead, it only observes the RGB values of each block (henceforth referred to as $F(x)$). Additionally, the agent begins with no knowledge of what colour terms exist or what parts of the RGB spectrum divide into different colour terms (ie, it is unaware of the terms in Figure 1 and what disparate RGB values map to a particular concept). As such, the agent faces the problem of jointly learning: (a) the vocabulary of colour terms; (b) how to ground those terms in the embodied environment (i.e. finding a mapping from colour term

to the range of RGB values it denotes); and (c) the constraints on the goal G, in terms of those colours.

A teacher observes the agent attempting to build the towers. Every time the agent takes an action which breaks one of the rules in G (or which leads to a situation where a rule will inevitably be broken) the teacher provides verbal feedback. The feedback serves to correct the agents mistake by providing an explanation as to why the action was incorrect. However, the verbal component may be ambiguous between several rules (see Section 4 for details). Thus, the agent must disambiguate the teacher's intended message while simultaneously learning to ground new terms in the embodied environment by learning the partition of RGB values into the concepts in Figure 1.

4 Coherence

The agent must learn the task by exploiting evidence supplied by the teacher's dialogue actions. It does this by reasoning about how the teacher's utterance coherently connects to the context in which it was uttered. To simplify matters we assume that all the teacher's dialogue moves are coherent, sincere (i.e. she believes what she says) and competent (i.e. what she believes is true).

The basic dialogue move the teacher makes is a correction of the form $u =$ "no, put red blocks on blue blocks" (or any other pair of colours). This utterance is ambiguous between rules $r_1^{red,blue}$ and $r_2^{red,blue}$ (henceforth shortened to $r_1^{r,b}$ and $r_2^{r,b}$). The semantics of correction stipulate that the content of the correction must negate some part of the corrected action (Asher and Lascarides, 2003). In our planning domain, this means that the teacher will utter u if the agent's latest action $a = put(x, y)$ violates the rule that she intended u to express, as stipulated in (3), where $V(r, a)$ represents that rule r was violated by action a:

$$Corr(a, u) \leftrightarrow (r_1^{r,b} \in G \wedge V(r_1^{r,b}, a)) \vee (r_2^{r,b} \in G \wedge V(r_2^{r,b}, a)) \quad (3)$$

The action a can violate a rule in the goal in two ways. For the first case, consider S_1 in Figure 2. If $r_1^{r,b} \in G$, then an action resulting in S_1 would directly violate the rule since $r_1^{r,b}$ requires each red block to be on a blue block, but here a red block was put on a non-blue block. Where $a = put(o_1, o_2)$, this *D*irect violation is expressed as (4), and similarly S_2 directly violates $r_2^{r,b}$ because of (5):

$$V_D(r_1^{r,b}, a) \leftrightarrow red(o_1) \wedge \neg blue(o_2) \wedge on(o_1, o_2) \quad (4)$$

$$V_D(r_2^{r,b}, a) \leftrightarrow \neg red(o_1) \wedge blue(o_2) \wedge on(o_1, o_2) \quad (5)$$

$r_1^{r,b}$ is not directly violated in S_2 and $r_2^{r,b}$ is not directly violated in S_1. However, these rules are respectively *I*ndirectly violated: it is impossible to complete a rule-compliant tower without first removing the top block from it. That is, an indirect violation of $r_1^{r,b}$ means that there are more red blocks on the table than blue ones, and furthermore (given that it violates the latest action $put(o_1, o_2)$), this was not the case before this action, and therefore o_2 must be blue and o_2 not red. Formally, indirect violations of the rule $r_1^{r,b}$ (which is satisfied by S_2) and $r_2^{r,b}$ (which is satisfied by S_1) are respectively defined by (6) and (7):

$$V_I(r_1^{r,b}, a) \leftrightarrow \neg red(o_1) \wedge blue(o_2) \wedge on(o_1, o_2) \wedge |\{o_3 : red(o_3) \wedge on(o_3, table)\}| > |\{o_4 : blue(o_4) \wedge on(o_4, table)\}| \quad (6)$$

$$V_I(r_2^{r,b}, a) \leftrightarrow red(o_1) \wedge \neg blue(o_2) \wedge on(o_1, o_2) \wedge |\{o_3 : blue(o_3) \wedge on(o_3, table)\}| > |\{o_4 : blue(o_4) \wedge on(o_4, table)\}| \quad (7)$$

When uttering u, our teacher helps the agent to determine which type of violation has happened by pointing at the tower if it's a *D*irect violation V_D or pointing at the block which can no longer be placed in the tower if it's an *I*ndirect violation V_I.

If the agent can ground either the colour term "red" and/or "blue" to blocks of those colours, then it can use the coherence equations (4)–(7) to infer whether the teacher's utterance u was intended to convey $r_1^{r,b}$, or $r_2^{r,b}$. Conversely, if an agent knows the intended meaning of u, then it can use these equations to make inferences about the colours of the blocks. However, our agent may know neither how to ground the colour terms (i.e., it can observe the RGB values but doesn't know what colour terms denote them) nor know how to

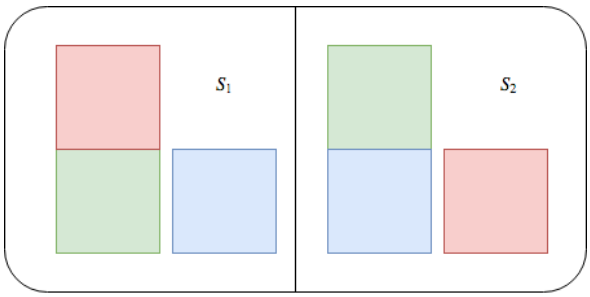

Figure 2: These two states would both be corrected if either $r_1^{(r,b)}$ or $r_2^{(r,b)}$ were in the goal.

disambiguate u. Therefore, in a context where the agent is sufficiently unsure as to the correct interpretation of the correction, because it is also unsure about how to ground the colour terms in u in the current visual scene, we allow the agent to utter a query, all of whose possible answers resolve the ambiguity. Due to the constraints expressed in (4)–(7), finding out the colour of just one of the blocks is sufficient to infer both the intended meaning of u and whether the situation is like S_1 or S_2 above. So the agent will simply ask such a yes/no question: for example, "is the top block red?". The teacher's answer provides the agent with an anchoring point, from which it can make further inferences via (4)–(7).

Additionally, when the teacher *doesn't* correct the agent's latest action a, then via the above equations, together with the agent's current beliefs about which blocks are which colours, the agent can infer beliefs about which rules are *not* a part of the goal (on the grounds that if that rule had been in the goal, a correction of a would have been uttered).

Interpreting u only requires knowledge of the action a it corrects. However, certain utterances are only interpretable through their coherent connection to previous dialogue. In this paper, our teacher uses two such utterances: u_2 = "no, that is wrong for the same reason" and u_3 = "no, that is not red either". u_2 presupposes a prior (identical) reason (in our task, a rule violation) is a part of the multimodal context; u_3 presupposes that something else (in the context) is not red.

In line with existing coherence-based theories of discourse (eg., Hobbs (1985); Kehler (2002); Asher and Lascarides (2003)) we assume that any utterance containing an anaphor or presupposition must be coherently connected to the unit that contains its antecedent. Thus u_2 (or u_3) must coherently attach to more than just the agent's latest action a; it must also attach to a prior utterance—this is why starting a dialogue with u_2 or u_3 sounds anomalous. Constraints on which parts of an *embodied* dialogue context the current utterance can coherently connect to are not yet fully understood (though see (Hunter et al., 2018) for initial work). We therefore take a very permissive approach: in principle, u_2 (or u_3) can coherently attach to any prior dialogue move. However, in line with existing theories of discourse interpretation, we adopt a preference for attaching to the most recent utterance u that supports a coherent interpretation, and in particular resolves the anaphor. In other words, an utterance of the form u_2 or u_3 attaches with *correction* to the latest agent's action a, but also to the most recent prior utterance u where a coherence relation $R(u, u_2)$ (or $R(u, u_3)$) can be established and an antecedent identified.

The utterance u_2 can be interpreted as an *elaboration* of any prior correction u: even if u were simply the expression "no", thanks to (3) a violation can be accommodated as part of the content of u precisely because it corrects an agent's (prior) action. Thus in embodied dialogue (1), u_2 attaches to a_2 with *correction* and also to u_1 with *elaboration* (because u_1 is more recent than u_0):

(1) a. a_0: $put(o_1, o_2)$
b. u_0: "No, put green blocks on orange blocks"
c. a_1: $put(o_3, o_4)$
d. u_1: "No, put red blocks on blue blocks"
e. a_2: $put(o_5, o_6)$
f. u_2: "No, that is wrong for the same reason"

The relation $elaboration(u_1, u_2)$ entails that however u_1 is disambiguated—ie, $r_1^{r,b}$, or $r_2^{r,b}$—"the reason" in u_2 refers to the same rule. So a_1 and a_2 both violate the same rule, and so impose joint constraints on the colours of the four blocks o_3, o_4, o_5 and o_6. This differs from the interpretation of a similar dialogue where the agent says u_2' below:

(2) a. a_1: $put(o_3, o_4)$
b. u_1: "No, put red blocks on blue blocks"
c. a_2: $put(o_5, o_6)$
d. u_2': "No, put red blocks on blue blocks"

u_2' doesn't feature any anaphoric expression, and so coherence does *not* demand that it be related to u_1. Thus the ambiguities in u_1 and u_2 may resolve in different ways. This illustrates how anaphora can impose additional constraints on interpretation of both the linguistic and non-linguistic moves. Our model (Section 5) and experiments (Section 6) show that exploiting anaphora in the interaction helps the agent to learn faster.

The utterance u_3 = "that is not red either" requires an antecedent individual that's not red. With this in mind, consider dialogue (3):

(3)
a. a_0: $put(o_1, o_2)$
b. u_0: "No, put orange blocks on red blocks"
c. a_1: $put(o_3, o_4)$
d. u_1: "No, put red blocks on blue blocks"
e. a_2: $put(o_5, o_6)$
f. u_2: "No, put purple blocks on pink blocks"
g. a_3: $put(o_7, o_8)$
h. u_3: "No, that is not red either"

The utterance u_3 corrects a_3, and coherence demands that it also attach to a prior utterance that entails that something isn't red. It cannot attach to u_2 with elaboration or with any other relation: in particular, it cannot elaborate either of the rules that u_2 might express while at the same time violating a rule that's expressed in terms of red, which it must do given that u_3 corrects an action (i.e., a_3). On the other hand, if the agent's beliefs about the colours of o_3 and o_4 are consistent with resolving the ambiguity in u_1 to $r_2^{r,b}$, then by (5) this interpretation provides an antecedent that's not red—namely o_3—and moreover it supports an elaboration relation between u_1 and u_3. Thus discourse coherence results in u_3 attaching to u_1 with *elaboration*, u_1 gets resolved to mean $r_2^{r,b}$, and hence (via equation (5)) o_3 and o_7 are not red and o_4 and o_8 are blue.

5 Method

We build an agent which utilises coherence to learn from the teacher's feedback. The agent architecture is the same as in Appelgren and Lascarides (2019) except the model for learning from correction is replaced. Figure 3 shows an overview of the system. The main components are the action selection system, which makes use of a symbolic planner (Hoffmann and Nebel, 2001), a search strategy to find the most likely plan consistent with what has been learned so far, and the correction handling system, which learns from the dialogue.

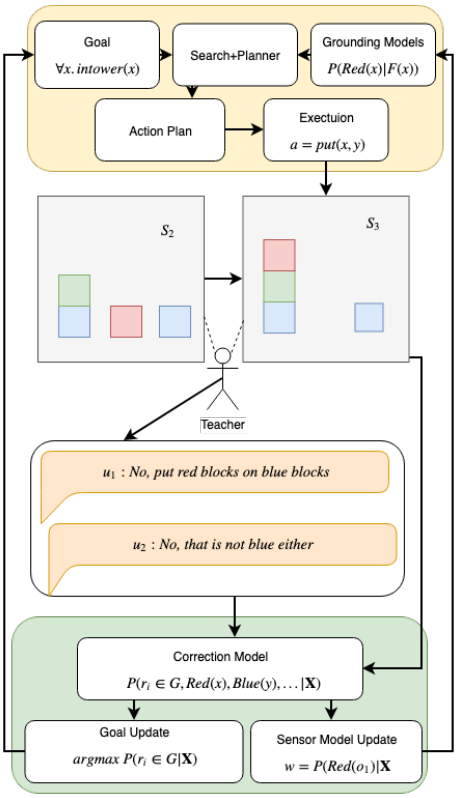

Figure 3: The agent consists of an action selection system (yellow) and a learning system (green). Action selection uses a symbolic planner to find a plan given the most likely goal and grounding of colour terms. The learning system uses coherence to build a probability model, used to learn what rules are in the goal and how to ground colour terms.

The agent learns to find the most likely rules that are part of the goal G and learns classifiers for colour terms. The classifiers (which we call the grounding models) are binary classifiers for every relevant colour, such as $P(Red(x)|F(x))$ and $P(Blue(x)|F(x))$. These are estimated using Bayes Rule, utilising Weighted Kernel Density Estimation (KDE) (Gisbert, 2003) for estimating $P(F(x)|Red(x))$.

5.1 Learning from Dialogue moves

The agent must learn what the rules are and how to ground colour terms. To learn the rules the agent must resolve the ambiguity in the teacher's messages. To learn the colours the agent must associate the RGB values to colour words, thus creat-

ing training data. Both outcomes are linked, since disambiguation of messages leads to learning the rules and associates colour words to blocks, but resolving the ambiguity requires knowledge of colours.

To resolve the ambiguity and use the available evidence to learn, we build a probability model which captures the semantics of the dialogue, and how it links to the environment, by capturing equations (3)–(7) in a probabilistic graphical model. The model is updated dynamically each time a new dialogue move is performed, adding new factors and new evidence, rather than creating a new model for every correction as is done by Appelgren and Lascarides (2019).

Specifically, when a correction is uttered a set of nodes are added to the current model of the dialogue. As an example we shall use u = "no, put red blocks on blue blocks" being directly violated. The nodes added from this correction can be seen in Figure 4. Here we know that equation (3) must hold, with rules $r_1^{r,b}$ or $r_2^{r,b}$. This is captured by adding a node $Corr(a, u)$, which is binary and observed to be $True$. Connected to this node are nodes $V_D(r_1^{r,b}, a)$, $V_D(r_2^{r,b}, a)$, $r_1^{r,b} \in G$, and $r_2^{r,b} \in G$. These are also binary variables, but they are latent. In the probability model this creates a factor

$$P(Corr_i(a, u)|V_D(r_1^{r,b}, a), V_D(r_2^{r,b}, a), r_1^{r,b} \in G, r_2^{r,b} \in G) \quad (8)$$

Which gives probability 1 to any outcome which satisfies equation (3).

For each $V_D(r_i^{r,b})$, additional nodes are created to capture equations (4) and (5). The nodes capture the colour of the relevant objects: $Red(o_1)$ and $Blue(o_2)$. The probability distribution (9) is 1 whenever the values of the variables satisfy those in equations (4) and (5).

$$P(V_D(r_1^{r,b}, a)|Red(o_1), Blue(o_2)) \quad (9)$$

Since $Red(o_1)$ and $Blue(o_2)$ aren't observable, nodes are also added for the observable RGB values of the objects: $F(o_1)$ and $F(o_2)$. $P(Red(o_1)|F(o_1))$ and $P(Blue(o_2)|F(o_2))$, which are the aforementioned grounding models, are learned using a weighted KDE. We also add priors for $P(r_i^{r,b} \in G)$ which are set to 0.01 and for $P(Red(o_1))$ and $P(Blue(o_2))$ which is set to 0.5.

The difference from Appelgren and Lascarides (2019) comes from the fact that when further corrections are given the model is updated by adding new nodes for the new correction and the possible violations it denotes. These nodes will be linked together if, for example, the same rule or the same blocks appear in the several corrections. This allows the agent to make inferences which change a belief from a previous correction given the new evidence. However, the biggest strength comes from modelling the interpretation of the anaphoric utterances, as discussed in Section 4.

5.1.1 Updating when no correction occurs

When a correction is given the agent adds nodes for the rules which are entailed by the content of the correction and observes $Corr(u, a) = True$. When no correction is given the agent instead adds nodes for all known rules and observes $Corr(u, a) = False$. That is, the agent adds a correction node which captures the fact that no rule which is in the goal was violated (through the negative case of equation (3)) as well as the nodes for direct violation of rules, capturing equations (4) and (5). Thus, the only non-zero probability interpretations of a non-corrected action are those which ensure these equations hold.

5.1.2 Handling Anaphoric Utterances

A discussed in Section 4, when an elaboration, such as $Elaboration(u_1, u_2)$ from dialogue (1), is given, the agent knows that the content of u_1 applies to the current action (a_2) and that the same rule must be violated by both a_1 and a_2. Thus, the nodes which were added for u_1 and a_1 are also added for the action a_2, as seen in Section 5.1. Further, an additional factor is added to capture that the same rule must be violated. This factor depends on $V(r_i, a_i)$ for the relevant rules and actions:

$$\phi(V(r_1^{r,b}, a_1), V(r_2^{r,b}, a_1), V(r_1^{r,b}, a_2), V(r_2^{r,b}, a_2)) \quad (10)$$

The factor gives a score of 0 to any situation where one of the rules, $r_i^{r,b}$, is violated for one of a_1 or a_2 but not the other, thus enforcing the constraint that the same rule must be violated in both situations.

When it comes to "no, that is not red either" (see dialogue (3)) the same applies. Further, we know that o_3 and o_7 are $\neg red$. The effect of this in our model is to add $\neg red(o_3)$ and $\neg red(o_7)$ as observed variables, whereas they would be latent

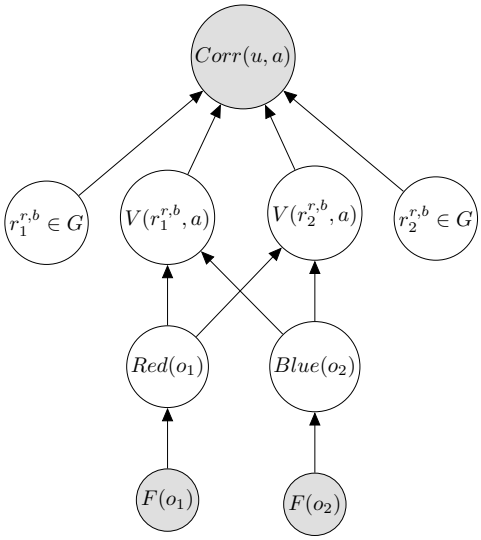

Figure 4: The nodes added to the probabilistic graphical model after a correction $u =$ "no, put red blocks on blue blocks". Grey nodes are observed and white ones are latent.

otherwise. Similarly, after a question is answered, the agent adds the 'colour' node's value to the observed variables.

As we see, the structure of our model makes it straight forward to add new logical constraints, as imposed by symbolic semantic models of coherent discourse, by adding factors that force probabilities to zero when an interpretation is deemed by the symbolic semantic model to be incoherent.

5.2 Updating the Goal

The graphical model is used by the agent to estimate which rules are most likely to be in the goal. This is done by finding the rules which have the highest probability of being in the goal:

$$argmax\ P(r_i \in G|\mathbf{X}) \tag{11}$$

where $\mathbf{X}$ represents the available observations (including the RGB values, correction variables, and observed colour variables). Since r_i being in G is a binary decision, this means all rules which have a probability higher than 0.5 of being in the goal are added and the rest are not.

5.3 Updating the Grounding Models

To update the grounding models we seek labels for individual colours. Since our graphical model creates constraints on what colours blocks may have, we use the probability estimate as a soft label to update our model. For example let

$$w = P(Red(o_1) = True|\mathbf{X}) \tag{12}$$

Figure 5: Cumulative regret averaged over 50 different trials on a planning problem containing three rules.

then w is used as a weighted label for o_1 which is used to update the probability density using weighted KDE.

6 Experiments

Our experiments test two hypotheses: exploiting anaphora when repeating a prior correction will lead to the agent learning to make fewer planning errors; and exploiting the evidence given by uncorrected moves will also lead to faster learning.

To test this we run four different set ups. First, we vary the teacher's dialogue strategy between **anaphor**, in which she uses utterances like u_2 and u_3 in dialogues (1) and (4) when the agent repeats an error, and **no-anaphor**, in which even for the agent's repeated errors, she repeats an utterance of the form $u =$ "no, put red blocks on blue blocks". Second, we vary the agent's reasoning about the task, between **simple**, in which it updates its probabilistic model only when the teacher says something, vs. **full**, in which it updates its model every time it performs an action $put(x, y)$, taking the teacher's silence to be a meaningful act as described in Section 5.1.1. The two types of teachers and agents gives four different combinations.

Each of these four combinations is run on 50 different planning problems—that is, we experiment with 50 different goals G, where each G is defined by two or three rules drawn from pairs of different colour categories and shades, such as red, purple, or maroon. Each planning problem (or G) is learned via a sequence of 50 scenarios: in each scenario the agent is presented with a (distinct) initial state of 10 blocks on the table, and the agent has to complete a tower that satisfies G, aided by the teacher's feedback. The colour of the 10

Figure 6: Cumulative regret averaged over 50 different trials on a planning problem containing two rules.

blocks in each scenario is randomly generated by either selecting a completely random colour with probability 0.2 or (with probability 0.8) selecting a pair of colours present in one of the rules (e.g. red and blue for $r_1^{r,b}$), selecting randomly from the hues of those colours, which biases states to include many of the constrained colours. We filter out any scenarios for which no correct tower exists. To measure the effectiveness of the agent we measure regret, accumulated over the 50 scenarios. Regret is simply the number of mistakes the agent makes, i.e. the number of corrected actions it performs.

6.1 Results

We present results for for experiments where each goal consists of two rules (Figure 6) and three rules (Figure 5).

Our hypothesis was that anaphors would help the agent make fewer mistakes; similarly for using the full evidence. Both of these results can be observed in the Figures 6 and 5. To test the significance of these results we performed a paired t-test on the total regret of each agent. The tests are made pairwise between agents using simple vs full, but keeping anaphor fixed, and between anaphor and no-anaphor, keeping simple vs full fixed. These significance tests are in Table 1.

These tests confirm that learning from the teacher's silence, as well as from corrective moves, speeds up learning significantly. These benefits stem mainly from the ability to observe more training examples of colours, colour learning being the major bottleneck in this problem. The effects of anaphora on learning is more nuanced, however. The fact that exploiting anaphora significantly improves performance for the three-rule case, but in the two-rule case it is not quite significant, suggests that the more complex the (latent) goal, the more useful anaphora will be. A further issue concerning the utility of anaphora could also be linked to the way we constructed the 50 initial states for each planning problem (see earlier discussion), which does *not* guarantee that if $r_1^{r,b}$, say, is a rule in G, then the initial state contains at least two red blocks of a different hue and/or two blue blocks of a different hue.

	Two Rules	**Three Rules**
s/**s+a**	t=1.5, p=0.14	t=2.6, p=0.012
s/**f**	t=4.39, p=6.1e-5	t=4.4, p=6.1e-5
s+a/**f+a**	t=2.1, p=0.046	2.1, p=0.043
f/**f+a**	t=2.3, p=0.024	t=3.9, p=3.1e-4

Table 1: Results of t-test between combinations of simple (s) with and without anaphora (a) and full (f) with and without anaphora (the superior system in bold).

7 Conclusion

We presented a novel graphical model which exploits the semantics of coherent discourse to jointly learn three tasks via natural language interaction with a teacher: how to refine the domain model to include new concepts; how to ground novel natural language terms to those concepts; and how to infer the correct goal description, so as to construct valid sequential plans. The graphical model extends on previous work by allowing it to learn from uncorrected moves in the dialogue as well as from utterances containing anaphoric expressions. Our experiments show that these extensions can help reduce the number of mistakes made by the agent while learning the task. In the future we intend to tackle more complex planning problems, featuring goal constraints with more complex structure, that are expressed in terms of unforeseen concepts other than colour. Additionally we intend to drop assumptions about the infallibility of the teacher.

Acknowledgements: We thank EPSRC for funding Mattias Appelgren, Ram Ramamoorthy and Yordan Hristov for helpful advice and discussions, and two anonymous reviewers for helpful feedback. All remaining errors are our own.

References

Mattias Appelgren and Alex Lascarides. 2019. Learning plans by acquiring grounded linguistic meanings from corrections. In *In Proc. of the 18th Inter- national Conference on Autonomous Agents and Multiagent Systems (AAMAS 2019), Montreal, Canada, May 1317, 2019, IFAAMAS*, page 9 pages.

Brenna Argall, Sonia Chernova, Manuela Veloso, and Brett Browning. 2009. A survey of robot learning from demonstration. 57:469–483.

Nicholas Asher and Alex Lascarides. 2003. *Logics of conversation*. Cambridge University Press.

Florian Benavent and Bruno Zanuttini. 2018. An experimental study of advice in sequential decision-making under uncertainty. In *AAAI*.

Joyce Yue Chai. 2018. Language to action: Towards interactive task learning with physical agents. In *AAMAS*.

Paul F. Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 4302–4310.

Francisco J Goerlich Gisbert. 2003. Weighted samples, kernel density estimators and convergence. *Empirical Economics*, 28(2):335–351.

Dylan Hadfield-Menell, Anca D. Dragan, Pieter Abbeel, and Stuart J. Russell. 2016. Cooperative inverse reinforcement learning. In *NIPS*.

J. R. Hobbs. 1985. On the coherence and structure of discourse. Technical Report CSLI-85-37, Center for the Study of Language and Information, Stanford University.

Jörg Hoffmann and Bernhard Nebel. 2001. The FF planning system: Fast plan generation through heuristic search. 14:253–302.

Julia Hunter, Nicholas Asher, and Alex Lascarides. 2018. A formal semantics for situated conversation. *Semantics and Pragmatics*.

A. Kehler. 2002. *Coherence, Reference and the Theory of Grammar*. CSLI Publications, Cambridge University Press.

W. Bradley Knox and Peter Stone. 2009. Interactively shaping agents via human reinforcement: the TAMER framework. In *Proceedings of the 5th International Conference on Knowledge Capture (K-CAP 2009), September 1-4, 2009, Redondo Beach, California, USA*, pages 9–16.

Gregory Kuhlmann, Peter Stone, Raymond J. Mooney, and Jude W. Shavlik. 2004. Guiding a reinforcement learner with natural language advice: Initial results in robocup soccer.

John E. Laird, Kevin A. Gluck, John R. Anderson, Kenneth D. Forbus, Odest Chadwicke Jenkins, Christian Lebiere, Dario D. Salvucci, Matthias Scheutz, Andrea Lockerd Thomaz, J. Gregory Trafton, Robert E. Wray, Shiwali Mohan, and James R. Kirk. 2017. Interactive task learning. *IEEE Intelligent Systems*, 32:6–21.

Peter Lindes, Aaron Mininger, James R. Kirk, and John E. Laird. 2017. Grounding language for interactive task learning. In *RoboNLP@ACL*.

Richard Maclin and Jude W. Shavlik. 1996. Creating advice-taking reinforcement learners. *Machine Learning*, 22:251–281.

Cynthia Matuszek. 2018. Grounded language learning: Where robotics and nlp meet. In *IJCAI*.

Monica N. Nicolescu and Maja J. Mataric. 2003. Natural methods for robot task learning: instructive demonstrations, generalization and practice. In *The Second International Joint Conference on Autonomous Agents & Multiagent Systems, AAMAS 2003, July 14-18, 2003, Melbourne, Victoria, Australia, Proceedings*, pages 241–248.

Matthias Scheutz, Evan A. Krause, Bradley Oosterveld, Tyler M. Frasca, and Robert Platt. 2017. Spoken instruction-based one-shot object and action learning in a cognitive robotic architecture. In *AAMAS*.

Lanbo She, Shaohua Yang, Yu Cheng, Yunyi Jia, Joyce Yue Chai, and Ning Xi. 2014. Back to the blocks world: Learning new actions through situated human-robot dialogue. In *SIGDIAL Conference*.

Sida I. Wang, Percy Liang, and Christopher D. Manning. 2016. Learning language games through interaction. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*.