Robot Dialog Optimization via Modeling of Human Belief Updates

Taylor Kessler Faulkner, Scott Niekum, Andrea L. Thomaz
Departments of Computer Science and Electrical and Computer Engineering
University of Texas
Austin, Texas 78712
{taylor, sniekum}@cs.utexas.edu, athomaz@ece.utexas.edu

***Abstract*—Future autonomous robots deployed in homes and workplaces must be able to competently ask for help when situations arise that fall outside of their knowledge or capabilities. When asking for help, these robots must provide enough information to solve the problem without giving overwhelming amounts of information. We present a novel algorithm that learns to predict how natural language utterances affect a classifier model of human task knowledge. We show how these predictions can be used to generate language that is both minimal and sufficiently communicative.**

I. Introduction

One of the most significant obstacles to the widespread use of autonomous robots is the potential for unreliable performance in human environments. Robots that fail too often may be more of a hindrance than a help to the people around them. Our goal in this work is to enable imperfect autonomous robots to ask humans for help and learn from the feedback they receive. While there are many ways that robots can generically signal that they need assistance (e.g. flashing lights, error messages, or preprogrammed broad requests for help), our work is aimed at allowing robots to use language to communicate specific problems and possible solutions in the face of error conditions. Importantly, our approach does not assume that a person knows anything specific about the robot's capabilities or the task it is trying to complete.

In this work, we focus on balancing two important criteria of asking for help:

1) Avoiding overwhelming users with too much information.
2) Relaying adequate information about robot capabilities or task goals.

The first criterion is important because if people are given too much unnecessary information at once, they may provide worse assistance than if they are given less information that is more relevant [7]. The second criterion is necessary because we want the robot and human to have a shared mental model [5] of potential solutions so that the robot can receive assistance that is actually helpful. Consider the following scenario. A robot, **R**, needs to hammer a nail, but it can't find a hammer. **R** has one arm, no regrasping capabilities, a limited range of arm motion, and needs to hold the hammer very close to the head to use it. **R** asks a nearby human, **H**, for help by simply asking "Can you hand me a hammer?" Note that this utterance conveys a potential solution to a problem, but does not relay any specific information about the robot's abilities or its plans to use the hammer. In Figure 1 we see three possible outcomes showing different ways that **H** could hand the hammer to **R**. The leftmost outcome may look acceptable as this is the position in which most people hold hammers. However, since the gripper is located far from the head of the hammer, **R** cannot support the weight of the hammer. At best, **R** will be unable to use the hammer; at worst, **R** might malfunction or drop the hammer. In the middle outcome, **R** can successfully hold the hammer, but since the head is pointing into the gripper there is no way to effectively use it on a nail as the robot cannot regrasp. Finally, the rightmost outcome will be successful, as **R** can support the weight of the hammer and it is facing away from the gripper.

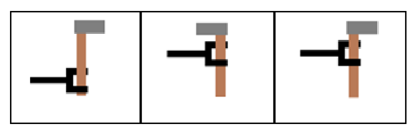

Fig. 1. Three different ways a person could hand the robot a hammer

We propose a three-part method to minimize the length of robot utterances about a problem while assuring that sufficient information is communicated to the person giving assistance. The pipeline for this method is shown in Figure 2. The first section of this pipeline consists of two parts: the *Utterance Generator* and the *Feature Selector*. The *Feature Selector* determines what features of the robot and its environment are relevant to asking for help. The *Utterance Generator* creates a set of proposed utterances based on the chosen features. The second part of this pipeline, the *Belief Update Learner*, allows the algorithm to learn how utterances affect a person's mental model of the actions needed to solve a problem. That is, given a human's current beliefs, we estimate their new beliefs given an utterance. We do so by representing a human's mental model as a classifier for potential outcomes such as those shown in Figure 1, in which people can classify an outcome o as positive if they think the robot can successfully complete its tasks proceeding from state o and negative if it cannot. The robot then makes queries that allow it to approximate a person's outcome classifier after stating subsets of the utterances from the *Utterance Generator*. The third part of the pipeline, the *Utterance Planner*, uses this information to create an utterance that balances the two previously mentioned criteria. Given a human's current beliefs and the beliefs we desire them to have, we choose a set of utterances such that each statement shifts the human's mental model in expectation so that it is sufficiently close to the robot's model.

In this work, we assume that the *Utterance Generator* and the *Feature Selector* exist and can give us appropriate utterances and features. Future work will address algorithms for these parts of the pipeline. We also assume that the robot knows what error has occurred and some possible way that a human could help; this process can be automated, but is not the focus of this work [21, 13, 3, 9, 12, 2, 10, 16]. We focus on the algorithms needed to build the *Belief Update Learner* and the *Utterance Planner*. While these algorithms may prove to be user specific or task specific, we consider the case in which, once learned, they can be applied to general users and tasks.

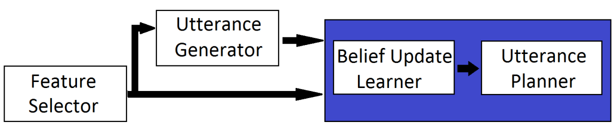

Fig. 2. Example pipeline containing our proposed method, with the focus of this work in blue

II. Related Work

Prior work has studied how to ask humans for help without assuming they are familiar with the robot. Tellex et al. present an approach that asks for help using inverse semantics to map objects to words that are likely to be least ambiguous to the human, in terms of what they refer to [22]. This approach conveys useful information and produces fairly short utterances, but it does not attempt to learn people's mental models of tasks and robot capabilities or handle cases where the human's help could potentially block future robot actions or damage the robot. Dey et al. show that, when a robot knows what error has occurred, it is best to give a minimal amount of information to a person to receive accurate help [7]. However, they do not provide an algorithm for choosing a good amount of information to relay.

Other research proposes methods that create shared mental models between robots and humans. Scheutz et al. provide a structure for creating shared mental models [19]. Several works use build common ground by changing the robot's mental model of the world or task to more closely fit the human's [14, 4, 23]. Kiesler studies how robot appearance and vocal sound can bring humans' mental models closer to the true capabilities of a robot [11]. These works all assume the robot and human to have a history of interactions or require hand-designed robot actions and appearance for different scenarios.

There is work on collaborative human-robot tasks that bases robot behavior on modeled human knowledge. Several works use the human model to modify robot actions rather than convince the human to take actions preferable to the robot [18, 17, 1, 20]. Fong et al. choose questions to ask people based on three models of human knowledge and ability to elicit helpful behavior [8]. Nikolaidis et al. present a game-theoretic approach to choose the most informative robot actions to take during a shared task that inform a person of the robot's capabilities [15]. Devin and Alami propose a method that enables a robot to adapt its plans and communicate about misinformation with a user [6]. These works assume that the robot and human can interact over an extended period of time or use hand-designed rather than learned models.

III. Algorithms

We propose two algorithms: one that learns a human's outcome classifier after stating subsets of utterances in order to learn a model of how human beliefs, i.e. their outcome classifiers, change with utterances, and one that selects the best set of utterances given these beliefs. Throughout this section, we assume that we have the following variables:

- X: a set of N utterances that are designed to convey information about features of the robot and its environment, including object poses, geometric and visual features of important objects, robot capabilities, and task goals.
- S : a fixed set of outcomes represented with images. We choose S to be a small set of positive and negative examples. Figure 1 represents some images that might be shown for a hammer task that requires a human handover.
- $\Psi(S)$: a mapping from outcomes in S to features in $\mathbb{R}^M$, where M is the number of features.
- $\phi(X)$: a mapping from utterances in X to features in $\mathbb{R}^M$. $\phi(X)$ maps to the same feature space as $\Psi(S)$.

Example utterances from X could include

- "My arm is weak": describes robot capabilities.
- "I need to go through this doorway": describes the task goal.
- "My gripper must be close to the heavy end": describes both robot capabilities and features of an object.

A. Belief Update Learner

We are taking a data driven approach to learn how human outcome classifiers change with robot utterances. We define a person's outcome classifier as $C_{\mathbf{H}} : \mathbb{R}^M \rightarrow \{0, 1\}$. That is, $C_{\mathbf{H}}$ is a map from M real-valued features to binary classes that denote if an outcome is acceptable or not. Our method learns many classifiers for each person participating in our data collection, each one representing a classifier after the robot has said some subset of utterances in X. We learn these classifiers over a variety of tasks and encountered problems. Each problem has D dimensions that each represent a feature that can be changed to produce positive and negative outcomes. For every participant, we state each utterance $x_i \in X$. The first utterance stated in this algorithm will be a vague description of a solution, e.g. "Please hand me the hammer," while the following utterances will be said in a different order for every trial. After each utterance, we present each $s_j \in S$ to the participant and request a positive or negative label. We then learn the Kernel Logistic Regression weights over these labels. Logistic Regression provides a probability over labels, rather than a hard label, allowing us to predict what outcomes a user is most likely to produce.

Consider again the hammer handover problem with $D = 1$. The single dimension is gripper placement on the handle. We would like the user to classify outcomes as positive if the gripper is between one and two inches from the hammer head. In Figures 3 and 4 we can see a possible result of one user's outcome classifier changing after a single utterance, where the outcomes the user thinks are positive marked in blue. Before the utterance, the user believes that the robot should hold the hammer low on the handle. After the utterance "My gripper needs to be close to the heavy end", the user's positively classified outcomes move closer to the desired region.

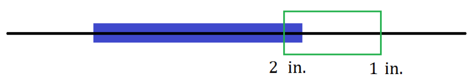

Fig. 3. Human's outcome classifier of gripper locations before any utterances. The green area marks truly positive grips one to two inches from the hammer head. The blue area marks the classifier's positive grips.

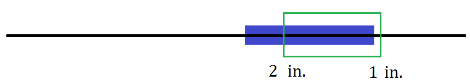

Fig. 4. Human's outcome classifier after the utterance "My gripper needs to be close to the heavy end"

We now generalize our collected data to form beliefs over how all humans' outcome classifiers change after utterances. For every participant's outcome classifier, we regress the features in $\phi(x_{0\ldots i})$ to the classifier learned for the subset of utterances $x_{0\ldots i}$ in order to generalize the classifiers to different sets of features. This allows us to predict what a person's classifier looks like for a set of utterances not previously seen.

B. Utterance Planner

We now propose an algorithm to choose an optimal set of utterances when interacting with a new user. When selecting these utterances, we attempt to maximize the intersection between the outcomes that we believe the user will positively classify and the outcomes that should be positively classified. We only care about the positive outcomes, as we assume the user is not an adversary and thus will never try to create an outcome that they believe will hinder the robot. Thus we just need their set of positive outcomes to closely match the ground truth positive outcomes. This method is shown in Algorithm 1.

C. Combined System

We present the separate parts of our method as a complete algorithm. We define the following variables:

- $J(X)$: cost of the utterances in X. This cost is yet to be determined but will depend on the word length of the utterance as well as other features.
- a and b: weights that balance the information conveyed by a set of utterances with the cost of the set. We will set these weights through a human study that measures people's responses to our produced utterances.

```
//Learn human outcome classifiers;
for each participant p do
    for each utterance x_i do
        speak(x_i);
        V = Query(S) //human classification values;
        W_{x_{0...i}} =
          learnKernelLogisticRegressionWeights(Ψ(S), V);
    end
end
//Regress from features to weights;
targets = features = [];
for each participant p do
    for i = 1 : N do
        features.append(φ(x_{0...i}));
        targets.append(W_{x_{0...i}});
    end
    f = Regress(features, targets);
end
//Choose an optimal set of utterances;
W_{GT} = ground truth classifier weights;
pos_{W_{GT}} = outcomes classified as positive by W_{GT};
pos_{f(φ(X))} = outcomes classified as positive by f(φ(X));
utterances =
  max_{x∈X} Σ a * (|pos_{W_{GT}} ∩ pos_{f(φ(X))}|) − b * J(x);
```

Algorithm 1: Method for robot dialog optimization

IV. Experiments

We ran a preliminary experiment to test the feasibility of our approach to learning a person's outcome classifier. We used the previously mentioned hammer gripping problem with $D = 1$. Consider a hammer with 100 possible discrete grip locations on its handle, with the first grip at the very bottom of the handle and the hundredth grip at the very top of the head. Thus there are 100 possible outcomes with the gripper positioned at each location. The training set for Kernel Logistic Regression is the labels a person assigns to the outcomes with the gripper positioned at every ten grasp locations, from zero to 100, and the testing set consists of the gripper placement at every other possible location. We learn from these testing sets using Kernel Logistic Regression with $\gamma = 0.01$.

V. Results and Discussion

In order to test our approach in an automated way, before collecting data from human subjects, we generated 1000 random models to represent what a human might initially believe about the task/robot in variety of ways. Then we analyze the extent to which our approach can uncover these various human models correctly.

Over this set of potential human outcome classifiers, our learned models achieve an average class-weighted F1 score of 0.86. We show the classifier that Kernel Logistic Regression learns from our training data for three potential users in Figures 5, 6 and 7. We plot the learned classifiers over the data from both the training and testing sets so that we see the full picture of the predicted classifier. The green area of this plot marks the region in the human's outcome classifier that corresponds to "true", and the blue line marks our learned probability of classifying each outcome as positive. We see that the peaks of the probability distribution of the learned classifier align well with the humans' ground truth classifiers, which allows our method to predict what outcomes are most likely to occur given the human's model.

We are in the process of obtaining IRB approval, and our next step in this work will be to collect data with human subjects. As stated in Section III-A, we will collect data over different tasks and problem scenarios. For each scenario, we will state utterances in X to each participant. We use a different order of utterances for each participant to learn more information about how combinations of utterances update human outcome classifiers. For example, stating utterance x_i and x_j may bring a person's outcome classifier closer to the truth, but inserting x_k in between the two may confuse them and make their classifier worse. After each individual utterance, we will show people images of a selection of possible outcome states, including outcomes that will be successful and outcomes that will not, in a random order and ask if they think the robot will be able to succeed at the task given each start state. Our algorithm will then learn a classifier over all possible outcome states based on the human's responses. After learning these classifiers for every participant, we will regress from our features to the classifier weights.

We expect to find sets of utterances that will push outcome classifiers towards our desired classifier. We will analyze the extent to which these sets of utterances can be user general versus user specific. If we find that different initial outcome classifiers and different people have nearly disjoint sets of utterances needed to bring them close to the ground truth classifier, then we will develop a test to estimate each person's initial classifier when asking for help. However, if the sets of utterances are nearly identical for every person, the robot can choose its dialogue to ask for help based on the current task and problem without having to know anything about the assisting human.

VI. Conclusion

In this work, we present an algorithm that will improve the ability of robots to effectively ask humans for help. This work is currently in progress, and we will soon begin collecting data to test our proposed algorithms. If these algorithms perform well experimentally, we will test them on a robotic system in a human environment in order to run human studies and receive subjective feedback on their performance.

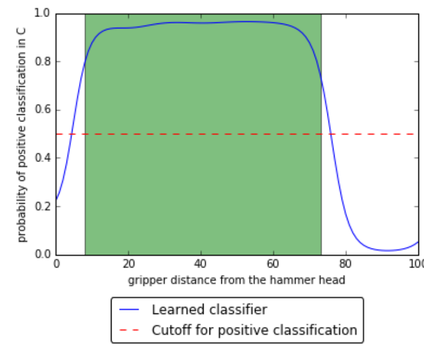

Fig. 5. Classifier: robot can grip the hammer between the 8^{th} and 73^{th} gripper locations

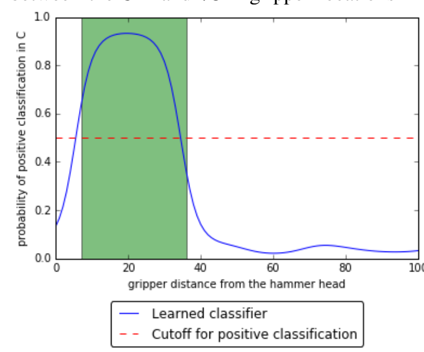

Fig. 6. Classifier: robot can grip the hammer between the 7^{th} and 36^{th} gripper locations

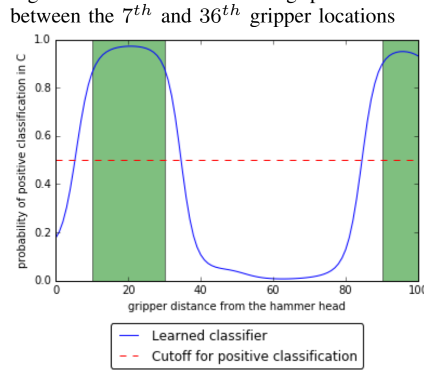

Fig. 7. Classifier: robot can grip the hammer between the 10^{th} and 30^{th} or 90^{th} and 100^{th} gripper locations

VII. Acknowledgements

This work has taken place in the Socially Intelligent Machines lab and the Personal Autonomous Robotics Lab at UT Austin. PeARL research is supported in part by the National Science Foundation (IIS-1638107, IIS-1617639). This material is based upon work supported by the National Science Foundation Graduate Research Fellowship Program under Grant No. DGE-1610403. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

[1] Samir Alili, Matthieu Warnier, Muhammad Ali, and Rachid Alami. Planning and plan-execution for human-robot cooperative task achievement. In *19th International Conference on Automated Planning and Scheduling*, pages 19–23, 2009.

[2] Dogan Altan and Sanem Sariel Talay. Probabilistic Failure Isolation for Cognitive Robots. In *FLAIRS Conference*, 2014. URL https://www.aaai.org/ocs/index.php/FLAIRS/FLAIRS14/paper/view/7878.

[3] Abdelbaki Bouguerra, Lars Karlsson, and Alessandro Saffiotti. Handling uncertainty in semantic-knowledge based execution monitoring. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 437–443. IEEE, 2007. URL http://ieeexplore.ieee.org/abstract/document/4399317/.

[4] Joyce Y Chai, Lanbo She, Rui Fang, Spencer Ottarson, Cody Littley, Changsong Liu, and Kenneth Hanson. Collaborative effort towards common ground in situated human-robot dialogue. In *Proceedings of the 2014 ACM/IEEE international conference on Human-robot interaction*, pages 33–40. ACM, 2014. URL http://dl.acm.org/citation.cfm?id=2559677.

[5] Sharolyn Converse. Shared mental models in expert team decision making. *Individual and group decision making: Current issues*, 221, 1993.

[6] Sandra Devin and Rachid Alami. An implemented theory of mind to improve human-robot shared plans execution. In *Human-Robot Interaction (HRI), 2016 11th ACM/IEEE International Conference on*, pages 319–326. IEEE, 2016. URL http://ieeexplore.ieee.org/abstract/document/7451768/.

[7] Anind K Dey, Stephanie Rosenthal, and Manuela Veloso. Using interaction to improve intelligence: how intelligent systems should ask users for input. In *Workshop on Intelligence and Interaction: IJCAI*, 2009.

[8] Terrence Fong, Charles Thorpe, and Charles Baur. Collaboration, dialogue, human-robot interaction. In *Robotics Research*, pages 255–266. Springer, 2003. URL https://link.springer.com/chapter/10.1007/3-540-36460-9_17.

[9] PM Frank, SX Ding, and T Marcu. Model-based fault diagnosis in technical processes. *Transactions of the Institute of Measurement and Control*, 22(1):57–101, 2000. URL http://journals.sagepub.com/doi/abs/10.1177/014233120002200104.

[10] Sertac Karapinar and Sanem Sariel. Cognitive robots learning failure contexts through real-world experimentation. *Autonomous Robots*, 39(4):469–485, 2015. URL https://link.springer.com/article/10.1007/s10514-015-9471-y.

[11] Sara Kiesler. Fostering common ground in human-robot interaction. In *Robot and Human Interactive Communication, 2005. ROMAN 2005. IEEE International Workshop on*, pages 729–734. IEEE, 2005. URL http://ieeexplore.ieee.org/abstract/document/1513866/.

[12] Jong Min Lee, Seung-Jong Kim, Yoha Hwang, and Chang-Seop Song. Diagnosis of mechanical fault signals using continuous hidden Markov model. *Journal of Sound and Vibration*, 276(3):1065–1080, 2004. URL http://www.sciencedirect.com/science/article/pii/S0022460X03009921.

[13] Cen Nan, Faisal Khan, and M Tariq Iqbal. Real-time fault diagnosis using knowledge-based expert system. *process safety and environmental protection*, 86(1):55–71, 2008. URL http://www.sciencedirect.com/science/article/pii/S0957582007000171.

[14] Stefanos Nikolaidis and Julie Shah. Human-robot teaming using shared mental models. *ACM/IEEE HRI*, 2012.

[15] Stefanos Nikolaidis, Swaprava Nath, Ariel D Procaccia, and Siddhartha Srinivasa. Game-Theoretic Modeling of Human Adaptation in Human-Robot Collaboration. In *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*, pages 323–331. ACM, 2017. URL http://dl.acm.org/citation.cfm?id=3020253.

[16] Peter Pastor, Mrinal Kalakrishnan, Sachin Chitta, Evangelos Theodorou, and Stefan Schaal. Skill learning and task outcome prediction for manipulation. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 3828–3834. IEEE, 2011. URL http://ieeexplore.ieee.org/abstract/document/5980200/.

[17] Dorsa Sadigh, S Shankar Sastry, Sanjit A Seshia, and Anca Dragan. Information gathering actions over human internal state. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pages 66–73. IEEE, 2016. URL http://ieeexplore.ieee.org/abstract/document/7759036/.

[18] Dorsa Sadigh, Shankar Sastry, Sanjit A Seshia, and Anca D Dragan. Planning for autonomous cars that leverages effects on human actions. In *Proceedings of the Robotics: Science and Systems Conference (RSS)*, 2016.

[19] Matthias Scheutz, Scott A DeLoach, and Julie A Adams. A Framework for Developing and Using Shared Mental Models in Human-Agent Teams. *Journal of Cognitive Engineering and Decision Making*, page 1555343416682891, 2017. URL http://journals.sagepub.com/doi/abs/10.1177/1555343416682891.

[20] E Sebastiani, R Lallement, R Alami, and L Iocchi. Dealing with on-line human-robot negotiations in hierarchical agent-based task planner. 2016.

[21] Gerald Steinbauer and Franz Wotawa. Robust plan execution using model-based reasoning. *Advanced Robotics*, 23(10):1315–1326, 2009. URL http://www.tandfonline.com/doi/abs/10.1163/156855309X462600.

[22] Stefanie Tellex, Ross A Knepper, Adrian Li, Daniela Rus, and Nicholas Roy. Asking for Help Using Inverse Semantics. In *Robotics: Science and systems*, volume 2, 2014. URL http://www.roboticsproceedings.org/rss10/p24.html.

[23] Elin A Topp, Helge Huettenrauch, Henrik I Christensen,

and Kerstin Severinson Eklundh. Bringing together human and robotic environment representations-a pilot study. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pages 4946–4952. IEEE, 2006. URL http://ieeexplore.ieee.org/abstract/document/4059204/.