Modeling Interaction Structure for Robot Imitation Learning of Human Social Behavior

Malcolm Doering, Dylan F. Glas, *Member, IEEE*, and Hiroshi Ishiguro, *Member, IEEE*

***Abstract*—This study presents a learning-by-imitation technique that learns social robot interaction behaviors from natural human–human interaction data and requires minimum input from a designer. To solve the problem of responding to ambiguous human actions, a novel topic clustering algorithm based on action co-occurrence frequencies is introduced. The system learns human-readable rules that dictate which action the robot should take, based on the most recent human action and the current estimated topic of conversation. The technique is demonstrated in a scenario where the robot learns to play the role of a travel agent. The proposed technique outperformed several baseline techniques in qualitative and quantitative evaluations. It responded more accurately to ambiguous questions and participants found it was easier to understand, provided more information, and required less effort to interact with.**

***Index Terms*—Human–robot interaction, imitation learning, interaction structure, spoken dialog system, unsupervised learning.**

I. Introduction

PRESENTLY, social robots show a potential in the roles of elder care, personal companions, hotel concierges, and in day-to-day interaction [1]–[4], and with the cost-effectiveness of automation, this trend is likely to continue. However, one of the difficulties of introducing robots to new domains is the creation of social interaction logic, which dictates how the robot behaves and interacts with people. It is tedious for an interaction designer to create all the behaviors for a robot by hand, and it is an incredibly challenging task to anticipate all the varieties of ways that humans may behave in a social interaction.

All these problems can be addressed through an imitation learning based approach for the development of robot interaction logic, as demonstrated by Liu *et al*. [5]–[7]. Natural human–human interaction data can be collected with sensor networks in target environments, like stores and offices. Machine-learning algorithms can then be applied to this data to train a robot to take the place of one of the humans by imitating their behavior. For example, by passively collecting data from natural interactions in a travel agency, a robot could learn how to become a travel agent (Fig. 1).

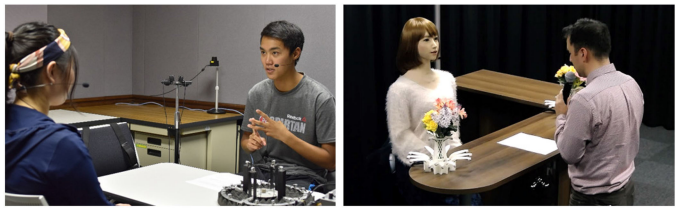

Fig. 1. (Left) Human–human data collection in travel agent domain. (Right) After training the system the robot replaces the human travel agent.

With an imitation-learning approach, it is possible for a robot to learn the correct responses to unambiguous human actions. However, it is much more difficult to learn how to respond correctly to context-dependent, ambiguous actions. For example, consider utterances like "How much is it?" which do not explicitly state the topic of conversation. In some special cases, it is possible to infer the relevant context from features directly observable through the robot's sensors, such as in a camera-shop scenario [5], [6], where the location of the robot and customer provided information about which camera was being discussed. More generally though, the relevant context is *hidden*, i.e., not directly observable.

To explore the problem of hidden context, in this work, a travel agency scenario is presented, where the topics of conversation (travel packages) are abstract entities, which are not directly observable from the sensor data. Since the topic of conversation is hidden, some additional information about the state of the interaction must be modeled in order for the robot to form correct responses to ambiguous customer questions.

The second important problem of learning social-robot interaction logic is interpretability [8], [9]. Previous approaches to data-driven human–robot interaction, such as [5] and [6], use models that are not understandable by a human designer. However, interpretability is a desirable characteristic for several reasons. For example, in the case that the machine learning algorithm makes a mistake, it is advantageous for a human behavior designer to fix any errors in the learned interaction logic by hand. Furthermore, in many fields, such as health, finance, and defense, where the decisions of a robot may have signifi-

Manuscript received May 9, 2018; revised October 15, 2018; accepted January 20, 2019. Date of publication February 26, 2019; date of current version May 15, 2019. The work was supported by JST, ERATO, ISHIGURO symbiotic Human-Robot Interaction Project, Grant Number JPMJER1401. This paper was recommended by Associate Editor L. L. Chen. *(Corresponding author: Malcolm Doering.)*

The authors are with the Department of Systems Innovation, Graduate School of Engineering Science, Osaka University, Osaka 565-0871, Japan, and also with the Advanced Telecommunications Research Institute International, Kyoto 619-0288, Japan (e-mail: malcolm.doering@atr.jp; dylan.f.glas@gmail.com; ishiguro@sys.es.osaka-u.ac.jp).

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org, provided by the author.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/THMS.2019.2895753

cant consequences, it is imperative to understand the reasoning behind those decisions [10].

In this study, the problems of hidden context and interpretability are solved by modeling the structure of the interaction itself and learning human-readable robot-interaction rules. With the proposed technique, the topics of conversation are automatically discovered by clustering speech actions observed in human–human training examples, based on action co-occurrence frequency. Then, a topic-state estimator is trained to infer the topic of conversation at each turn in a real-time interaction. Human-readable interaction rules are learned based on speech action transition probabilities and the inferred topic states in the training data. These rules dictate how the robot should respond in real-time based on the customer's previous action and the current topic state.

II. Related Work

A. Data-Driven Robot Interaction Logic

In data-driven approaches, to create a robot-interaction logic, the robot imitates the outward behaviors observed in human–human, or teleoperated example interactions, without requiring a model of the world or input from a human designer.

One popular data-driven approach is to crowd-source large amounts of human–human interaction data in virtual worlds from which to train a virtual agent or robot [11]–[13]. These works differ from ours since their data was collected through a virtual world, which was not susceptible to the sensor noise present in real-world interactions.

A second approach is to collect human–human interaction data and manually annotate it for supervised learning, such as [14] for a robot tutor. In comparison, our proposed approach does not rely on expensive manual annotation of the data.

A third approach is to learn from unlabeled human–human interaction data via unsupervised learning. Human action cluster IDs were used as labels in a supervised learning framework to train a robot shopkeeper, [5], [6]. Their system could only use information directly observable from the sensors, so it cannot deal with otherwise unresolvable human actions, e.g., ambiguous questions that depend on topic of conversation.

B. Topic in Dialog

The main focus of this study is to model the topic of conversation as it changes over the course of an interaction for the purpose of generating appropriate robotic actions.

Latent Dirichlet allocation (LDA) is a statistical approach to discovering the topic based on word frequency and co-occurrence [15]. Methods based on LDA are popular unsupervised techniques for discovering the topics in sets of large documents [16], [17]. However, in contrast to the topic of large documents, the topic of conversation changes dynamically over time, so these approaches are not directly applicable to our problem.

Linguistic studies of discourse introduced the concepts of focus of attention and attentional state, which are more closely related to the topic of conversation [18], [19]. However, these theories rely on syntactic parsing and identification of individual words, so they are not directly applicable to situations with many speech-recognition errors.

Probabilistic graphical models, such as hidden Markov models (HMM) have been the common methods for tracking states that evolve over time [20]. For example, [21] uses a dynamic Bayesian model to discover the underlying states of multiparty conversations, and [22] uses an HMM to discover interaction structure of tutoring sessions. These works mainly focus on descriptive analysis of data, and do not use their discoveries for generating robot or agent behaviors.

A topic tracking system for human–robot dialog is presented in [23] which discover the topics of conversation automatically from interaction data based on word co-occurrences; however, they did not formally evaluate it to demonstrate that it works.

A system for automatically estimating joint attention state from visual/auditory input data, comparable to our topic state estimation approach is presented in [24] (Section V-C).

C. Dialog Acts

Dialog acts are abstract representations of actions in dialog indicating intent, such as greeting, request, and yes–no question [25]. Unsupervised learning has been applied to automatically discover dialog acts from lexical and contextual features of utterances [26], [27]. A Markov random field based clustering method is introduced in [27] to discover dialog acts from transcripts of tutoring sessions. The goal of speech clustering (Section III-D) is similar to unsupervised dialog act learning, but finds actions at a lower level of abstraction than is typical of dialog acts.

Adjacency pairs are pairs of dialog acts from alternating speakers that frequently co-occur, such as question→statement [28]–[30]. The robot action predictor (Section IV-B) works on the basis of adjacency pairs of speech clusters that are extracted from human–human interaction examples.

D. Coreference Resolution

Modeling hidden context so the robot can answer ambiguous questions is related to coreference resolution (CR)—the process of determining whether two referring expressions refer to the same entity—and anaphora resolution—the process of determining how a previously mentioned term affects the meaning of a related term mentioned later [31].

The state-of-the-art CR system is an artificial neural network trained on manually annotated text and manually transcribed speech [32]. An approach adapted for online chat dialog is presented in [33]. However, CR systems have not been designed to be reliable against speech-recognition errors, typical in human–robot interaction. In contrast, our proposed approach to answering ambiguous questions does not rely on manually annotated data and is robust to some speech-recognition errors.

E. Dialog Systems

Some spoken dialog systems use data-driven approaches to learn dialog policies from unlabeled datasets, such as Twitter

[34], [35]. However, these approaches do not model the state, i.e., topic of conversation.

Frame-based dialog systems keep track of the dialog state by tracking a set of slots and values, representing the user's goals [36]. Many approaches to state tracking, including rule-based systems, graphical models, and artificial neural networks, have been evaluated on manually labeled benchmark datasets [37], [38]. The state of the art uses a recurrent neural network [39]. The most successful methods rely on hand-annotated corpora, so they are not suitable for rapid deployment into new domains.

In nonframe-based approaches, tree-structured ontologies [40] and Wikipedia article categories [41] have been used to represent topics of conversation. However, these approaches depend on humans to specify domain-specific knowledge or to annotate training data. Thus, they are also troublesome to apply to novel domains. Our objective is to discover the topic of conversation automatically, without human input.

III. Data Collection

The proposed method uses human–human interaction data to train a robot to perform socially appropriate behaviors in face-to-face interactions with a human.

A. Travel Agent Scenario

The proposed system is applicable to any domain in which the participants' actions are highly repeatable, but for demonstration a travel agent interaction scenario was chosen to train and test the system. In the scenario, a customer enters a room, approaches a table with the travel agent, and they proceed to converse about the available travel packages until the customer decides whether or not to purchase one.

Three travel packages were created for the travel agent and customer to talk about: A trip to the Sahara Desert, a trip to London, and a boat cruise along the coast of Antarctica. Additionally, each travel package had six attributes: Destination, duration, price, what is included in the package, what there is to do on the trip, and who else will be going on the trip (other tourists, a desert-survival guide, etc.).

B. Data Collection

In the data collection procedure, six fluent English-speaking participants (four male, two female, mean age 24.3, s.d. 3.1) took turns playing the roles of a travel agent and customer. The person in the travel agent role was provided with a listing of travel packages and their attributes. In order to collect a variety of utterances, the customer was instructed to take turns acting like three different customer types: A poor student looking for a budget vacation, an adventurous person looking for a crazy vacation, and an undecided person who simply wants to collect information. The travel agent was instructed to greet the customer, provide information about the travel packages, and refer the customer to checkout if a decision was made. An example data collection interaction is shown in the supplementary video.

C. Data Preprocessing

Interchannel noise suppression and speech segmentation are first applied to identify the segments of speech in the audio data. Next, the speech segments were passed through a voice-to-text module using the Google Speech API. Last, a simple turn-taking model was applied, in which consecutive utterances by the same speaker with no intervening utterances by the other speaker were concatenated together. These steps result in a series of alternating customer text/travel agent text utterances.

Audio was recorded from 192 interactions with 2481 customer / travel agent utterance pairs in total, which is a comparable quantity to other datasets for training robots [5], [13], [42], [43]. The average number of turns in each dialog was 12.9 (s.d. 3.5). The average number of words in each utterance was 9.3 (s.d. 5.8) for customers and 17.3 (s.d. 12.7) for travel agents. The automatic speech recognition (ASR) word error rate on a representative subset of 500 utterances was 23%. The average length of the interactions was about three minutes.

D. Speech Actions

In human–robot interaction there are two challenges that make textual-analysis-based dialog approaches impractical. First, the input is noisy: One HRI study reports that a speech recognition system that performed with 92.5% accuracy in 75 dBA noise achieved only 21.3% accuracy in a real-world environment [44]. Second, participants in our studies often speak casually, with sentence fragments and bad grammar, hindering extraction of features based on grammatical structure.

Clustering utterances into speech clusters based on simple lexical features (n-grams) aids in mitigating these challenges. By grouping noisy utterances together with other similar utterances, the intended meaning can be recovered by looking at the other members of the speech cluster.

The speech clusters represent the common speech actions in the human–human interactions.

E. Speech Vectorization

Before clustering the speech, each customer and travel agent utterance was vectorized. The utterance vectors consist of n-grams (uni-, bi-, and tri-grams) of word stems and keywords. Word stemming was done via Python NLTK's WordNet-based lemmatizer (http://www.nltk.org/) and keywords were extracted using AlchemyAPI (http://www.alchemyapi.com/). The customer utterance vectorization had 2577 dimensions and the travel agent utterance vectorization had 5457 dimensions.

F. Speech Clustering

The dynamic tree cut hierarchical clustering algorithm was used on the utterance vectorizations to find speech clusters [45]. Customer and travel agent utterances were clustered separately since their actions did not frequently overlap. A second pass of automatic processing was performed to remove noisy clusters, which had very high intracluster distances, and reassigned the utterances they contained the remaining clusters using a nearest neighbor-based approach. If an utterance was farther than an

TABLE I
EXAMPLES FROM THE SPACE OF LEARNED ACTIONS

Speech Cluster ID	Typical Utterance (Medoid)	Speech Cluster Size
2001	Hello.	61
3050	Hi there, how can I help you?	30
3004	Will ring you up at the next counter.	55
2006	Can you tell me more about the Sahara Desert?	40
2015	Okay, how much is that one?	40
3008	That is $3000.	49
2048	Okay, I guess I'll take the trip to London.	25

TABLE II
EXAMPLE OF A SPEECH CLUSTER

Travel Agent Speech Cluster 3047	
Contents (ASR results)	The camels that you would be traveling up in your camel caravan.
	Look up the camels.
	Camel selfies.
	I'll be crossing by camel caravan.
	It will last for 14 nights and you'll be traveling by camel caravan.
	Well you could you'll be traveling by camel caravan I should say.
	Well you'll be with a lot of camels and five four other two arrests.
	⋮
Typical Utterance: "You be traveling by camel Caravan."	

empirically set distance from the nearest centroid, it was not reassigned to a cluster.

In the end, the clustering procedure yielded 113 customer clusters and 58 travel agent clusters, containing 82% of the utterances in the training data. Due to the high word error rate and the presence of nonrepeatable utterances (e.g., off-topic utterances) 12% of the utterances were not clustered. Regardless, the speech-clustering procedure successfully revealed the highly repeatable actions which the proposed technique aims to learn. Some examples from the space of learned actions are shown in Table I. Furthermore, Table II shows an example of one customer speech cluster, illustrating that the members of speech clusters include similar utterances, but with some natural variation and speech-recognition errors. These speech clusters defined the "actions" of the system.

Note, this clustering procedure splits some clusters that should ideally be merged, namely clusters of functionally-equivalent but lexically-dissimilar utterances (e.g., "How much does it cost?" and "What is the price?"). The utterance features and clustering distance metric are arbitrary, thus they could be replaced with improved techniques in the future.

Since the robot needs to be able to speak an utterance corresponding to each travel agent speech cluster, a typical utterance (bottom of Table II) was selected for each by finding the medoid utterances, i.e., the most similar utterance to all other utterances in the speech cluster, using the Levenshtein distance metric normalized for utterance length. Since complete utterances with few ASR errors tend to share the most similarities with other utterances in the same cluster, typical utterances tend to be well formed and easy to understand.

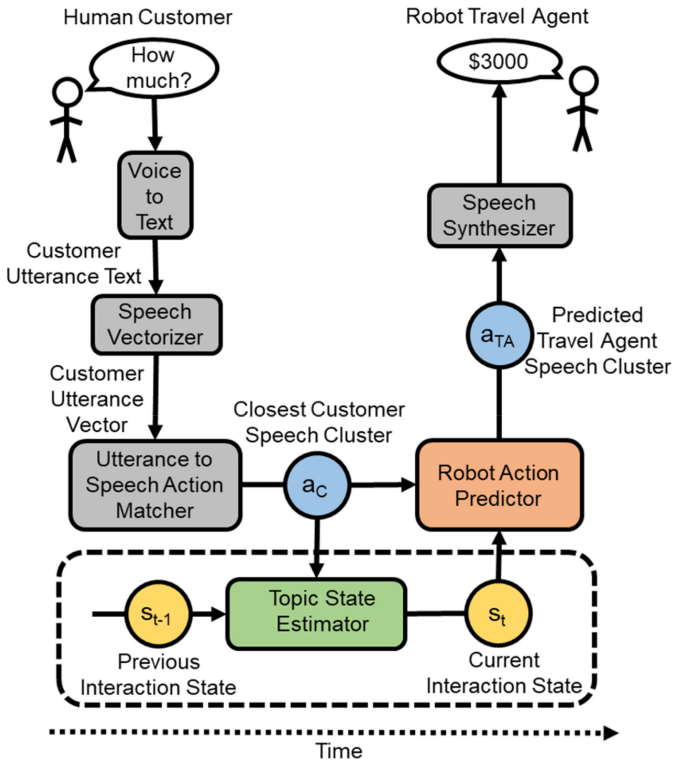

Fig. 2. System runtime diagram. The topic module is within dashed lines.

IV. ACTION PREDICTION

Two of the three main components of the proposed system are the utterance to speech action matcher and the robot action predictor (Fig. 2). Preliminary action prediction results using only these components demonstrate the necessity of incorporating topic of conversation to deal with ambiguity.

A. Utterance to Speech–Action Matching

In the proposed system, a customer's action is recognized by vectorizing their utterance text and matching the vectorization to one of the known customer speech actions. Matching using a nearest centroid classifier with the cosine distance metric performed well empirically, achieving 91% utterance-to-speech-action matching accuracy in a tenfold cross validation.

B. Robot Action Prediction

A robot system must decide which action to take next. In the proposed system, this robot action prediction is accomplished by learning a set of human-readable interaction rules from the sequences of speech action IDs in the human–human interaction training data. Action (speech cluster) IDs (Table I) are discrete, symbolic representations of abstract actions.

More specifically, the set of interaction rules consists of IF customer action, THEN robot action style rules based on customer action to travel agent action transition probabilities computed from the human–human data. A rule is created for each customer action by finding the most likely travel agent action

TABLE III
EXAMPLES OF RULES LEARNED BY THE SYSTEM

Input Customer Action	Customer Action Typical Utterance	Predicted Travel Agent Action
2001	Hello.	Hi there, how can I help you today?
2017	Okay, and how long is that trip?	That is 10 days. (Correct only for Antarctica)
2022	Okay, I think I'm going to go with the London trip.	Okay, that's a great choice.

to follow using (1), where a_C is the preceding customer action, A_{TA} is the set of all possible travel agent actions, a_{TA} is the robot's action (a travel agent action), and $P(a|a_C)$ is the transition probability from a_C to a.

$$a_{TA} = \underset{a \in A_{TA}}{\operatorname{argmax}} \; P(a|a_C). \tag{1}$$

Since the goal of the system is to learn repeatable behaviors, customer action to travel agent action pairs that occurred only once in the training data were pruned.

During operation of the system, after predicting the action a_{TA}, the robot speaks the typical utterance (bottom of Table II) of the corresponding speech cluster.

C. Preliminary Evaluation and Discussion

The system was evaluated on the 192 human–human trained interactions using hold-one-out cross-validation and a human evaluator categorized the system's responses to each customer utterance as either correct or incorrect. Most importantly, the system response correctness to ambiguous and unambiguous customer actions was examined separately.

The system learned many correct interaction rules, but some incorrect rules (Table III). In particular, rules with ambiguous customer speech actions were mostly incorrect. In Table III, The rule for the ambiguous customer action 2017 with typical utterance "Okay, and how long is that trip?" only contains a single predicted travel agent action, "That is ten days". This is the correct response for the Antarctica travel package, but it would be incorrect for either of the other two travel packages.

Many customer utterances are ambiguous with respect to which travel package is under discussion, so predicting the robot's action based on the customer's action alone is insufficient. In fact, only 39% of the system's responses to ambiguous customer actions were correct, compared to 66% percent of the responses to unambiguous customer actions.

V. TOPIC ESTIMATION

To address the problem of ambiguous utterances, a topic state estimator was incorporated to model topic of conversation (Fig. 2). Observations of the interaction data motivated the design of a novel topic clustering algorithm. The topic clusters are used to estimate the topic of conversation in real time so the robot can respond to ambiguous customer actions.

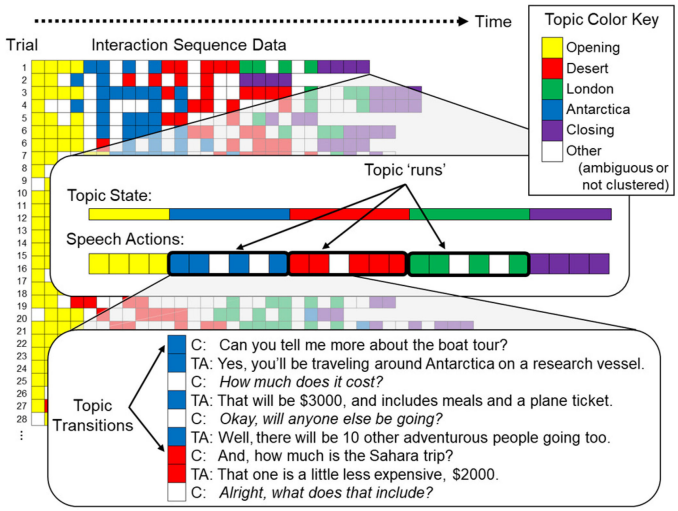

Fig. 3. (Background) The training data, with boxes representing speech actions annotated with topic. (Middle) A single interaction sequence, showing topic and topic state. (Foreground) A topic "run" containing ambiguous customer utterances (ambiguous utterances are italicized).

A. Interaction Structure

The core of each training interaction consisted of the customer asking questions about the travel packages. Customers tend to ask several questions about one specific travel package at a time and then ask about another travel package. They typically asked about at least two of the packages per interaction. Interactions usually opened with an exchange of greetings and introduction of the travel packages, and closed with the travel agent thanking the customer.

These observations indicate that there are topics which constitute phases of a conversation, e.g., London, desert, and Antarctica. In fact, the proposed algorithm (Section V-B) discovered two other conversational phases, corresponding to opening and closing, although this was not initially anticipated. Here, the term "topic" encompasses not only the target topics but also these additional two conversation phases.

Visualizing the interaction sequences helped us to understand their structure. The background of Fig. 3 shows the interaction structure of several interaction sequences, as described above. Each box represents a speech action, and those speech actions that are uniquely associated with a single topic are color-coded. Speech actions that are not associated with only a single topic (i.e., ambiguous) are white. Actions associated to the same topic typically occur together in "runs" where the topic of conversation remains the same. These topic runs are sometimes interspersed by ambiguous actions (foreground of Fig. 3, with the ambiguous utterances italicized).

B. Topic Clustering Algorithm

Motivated by the observation that speech actions uniquely associated with a single topic often occur together, a clustering algorithm was designed based on action co-occurrence to automatically discover the structural patterns, including topics of conversation, in the training data. The end goal of clustering

TABLE IV
TOPIC CLUSTERING ALGORITHM

Supp	Support matrix
T	Set of topic clusters $\tau \in$ T
τ	Set of actions a $\in \tau$
Input: Supp	
1.	Supp' ←filter(Supp, θ_1)
2.	T←initialize_topic_clusters(Supp', θ_2)
	While stop_condition_not_met():
3.	topicFits←compute_action_to_topic_fits(Supp', T)
4.	**Add** action α to topic τ that meets Fit criteria using θ_3 and θ_4.
5.	topicFits←compute_action_to_topic_fits(Supp', T)
6.	**Remove** all actions from topic clusters that do not meet Fit criteria using θ_3 and θ_4.
	Return T

actions into topic clusters is an action-to-topic mapping that can be used to disambiguate customer speech.

The objective is to find sets of actions (topic clusters), whose members frequently co-occur with each other but not with members of any other set. A topic cluster should contain actions associated with that topic, such as unambiguous actions (e.g., "Tell me about the London trip?", "It includes five nights in a hotel"), and exclude actions associated with other topics, such as ambiguous questions (e.g., "How long is that one?", "What's included in the price?") and backchannels (e.g., "Okay").

To quantify the level of co-occurrence between each pair of speech actions, subsequences of length 2–5 actions were generated from the original interaction sequences and the support [46] between pairs of actions was calculated using (2), where (a_i, a_j) is the action pair, H is the set of all interaction subsequences, and h is a generic interaction subsequence. Thus, the support of action pair (a_i, a_j) is equal to the proportion of interaction subsequences that contain that pair.

$$\mathrm{Supp}(a_i, a_j) \;= \frac{|\{h \in \mathrm{H};\ (a_i, a_j) \subseteq \mathrm{h}\}|}{|H|}\ . \tag{2}$$

With (2) a support matrix, Supp, was generated representing the support of all possible pairs of actions, $\mathrm{Supp} \in \mathbb{R}_{\geq 0}^{|A_C \cup A_{TA}| \times |A_C \cup A_{TA}|}$, where A_C is the set of all customer actions and A_{TA} is the set of all travel agent (robot) actions.

The proposed topic clustering algorithm (Table IV) takes the support matrix Supp as input and returns a set of topic clusters T. A topic cluster τ, $\tau \in$ T is defined as a subset of speech actions a, $a \in A_C \cup A_{TA}$. The topic clusters τ are mutually exclusive (an action can only be in one topic cluster).

The first step of the algorithm (Step 1 in Table IV) is to filter out action pairs with support below a threshold, θ_1, since actions that co-occur infrequently tend to only co-occur at all due to random noise or speech clustering errors.

Next, the algorithm initializes the topic clusters by creating seed clusters from the action pairs with very high support, using a higher threshold, θ_2 (Step 2). Action pairs with high support frequently co-occur, so they probably belong to the same topic. This step is accomplished by selecting pairs of actions with support above θ_2 and then performing connected components analysis to find groups of connected actions.

Subsequently, the topic clustering procedure, which was inspired by iterative and density based methods for graph clustering [47], consists of two subroutines that iterate until the algorithm converges (i.e., a cycle of repeating states is detected) or a maximum number of iterations is reached (since a cycle could be arbitrarily long). One subroutine adds actions to topic clusters and the other removes actions from topic clusters.

Before each of the subroutines is run, the Fit, defined in (3), of each action to each topic cluster is computed (Steps 3 and 5), which indicates how well an action a belongs in a topic cluster τ. This metric works by looking at the proportion of support of the action a with each of the actions in the topic cluster τ to support that action with all actions in all topic clusters T. If an action has a high degree of co-occurrence with most of the actions in a certain topic cluster but not with actions in any other topic clusters, then it probably uniquely belongs to that topic.

$$\mathrm{Fit}(\mathrm{a}, \tau) \;= \frac{\sum_{a_\tau \in \tau} \mathrm{Supp}\,(a, a_\tau)}{\sum_{\tau \in T} \sum_{a_\tau \in \tau} \mathrm{Supp}\,(a, a_\tau)}\ . \tag{3}$$

During the Add subroutine (Step 4), all actions a that meet two criteria are added to the topic cluster they best fit, τ_{best}. First, in order to assign actions to their best-fit clusters the action a should match to τ_{best} above a Fit threshold, θ_3. Second, to exclude ambiguous actions, the action a should fit τ_{best} at least θ_4 times better than any other cluster. The second subroutine, Remove (Step 6), removes all actions from topic clusters that do not meet the above criteria.

The values for θ_1 and θ_2 were chosen by looking at the support for each action pair, which has a long tail distribution. θ_1 was set to six in order to filter actions at the end of the long tail and θ_2 was set to 90, a value well above the elbow. θ_3 and θ_4 were chosen arbitrarily. Values of 0.4 and 3 were found to work well, respectively.

When run on the human–human interaction training data, the algorithm converged after 154 iterations and discovered five topic clusters. In an experiment with 108 runs with various parameter values, all of them converged. The minimum, maximum, and average number of iterations before the algorithm converged were 93, 251, and 153 (s.d. 28). An average run took 10.0 s (s.d 1.0). The maximum cycle length was 12 iterations.

Of the 171 total actions, the topic clustering algorithm clustered 53% (90) of them and left out the remaining 81. Based on manual topic annotations, 75% (129) of the 171 actions were clustered into the correct topic cluster, or correctly left out (actions labeled "Other"). Many actions that were incorrectly clustered into the wrong topic or excluded were not well-represented in the training data or contained speech clustering errors. However, among the 90 actions that fell into topic clusters, 90% (81) of them were in the correct topic cluster. Thus, the clusters successfully identified the relevant topics.

C. Topic State Estimation

In each interaction there is an underlying topic state (middle of Fig. 3), which is persistent over time and represents which of the topics discovered by the topic clustering algorithm the conversation is about at each speaker's turn. This topic state is

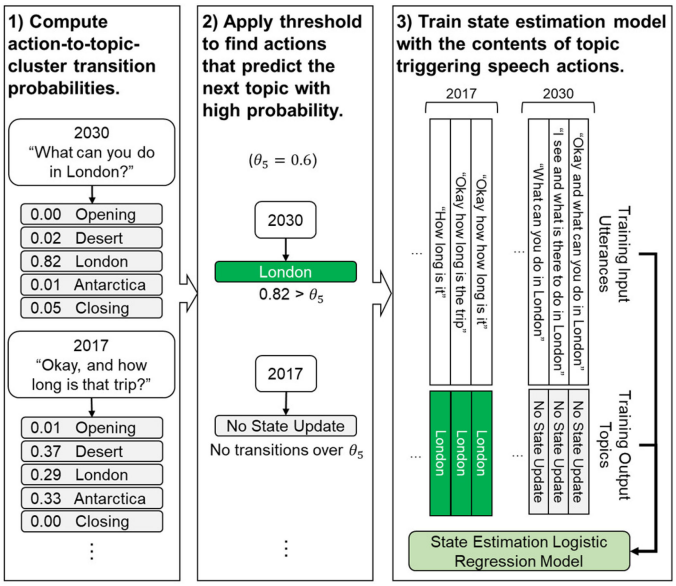

Fig. 4. Procedure for training the state estimator.

not directly observable from the actions alone, since there are many ambiguous actions and the topic clustering algorithm only identifies a subset of the topic-specific actions. To estimate the topic state for robot action prediction a topic state estimator was designed (Fig. 2).

The set of possible topic states S was defined to represent the topics of conversation discovered by the topic clustering algorithm. That is, each topic cluster τ, has a corresponding topic state s_τ, $s_\tau \in S$. Each timestep t of an interaction has some topic state s_t. At each step the topic state estimator determines whether the topic state s_t remains the same or changes based on the most recent action a_t.

The topic state estimator is a logistic regression classifier that takes an utterance vectorization as input and outputs the new topic state. The topic state is estimated after each utterance, but since some utterances, such as ambiguous questions, do not change the topic of conversation, an additional class, "no state update", was used to indicate no update or insufficient evidence to update the topic state. Training the topic state estimator consists of three steps (Fig. 4).

1) Compute action-to-topic-cluster transition probabilities. For each speech action a the probability of transitioning to topic τ was computed from the training data by finding each instance of a, a_t, and looking ahead at the topic cluster of the next travel agent action, $a_{TA,t+1}$, using (4). Since customer actions are often ambiguous, the transition probabilities were based on the topic of the next travel agent action, which provided more information about the topic of conversation.

$$\mathrm{P}_{\mathrm{trans}}(a,\tau) = \frac{|\{a_t;\ a_t = a \ \wedge\ a_{TA,t+1} \in \tau\}|}{|\{a_t;\ a_t = a\}|}. \quad (4)$$

2) Apply a threshold, $\theta_5 (=0.60)$, to the action-to-topic-cluster transition probabilities to select actions that predict the next topic with high probability. Such actions indicate that the topic state should be updated based on the most highly predicted topic cluster. For example, action 2030, with the typical utterance "What can you do in London?," was followed by actions in topic cluster three (London) 82% of the time. This is sufficient evidence to change the topic state to s_3 when the system observes utterances similar to those in action 2030.

Actions for which transition probabilities to all topics were below θ_5 were selected as candidate examples of the "no state update" class. Since this set of actions was large, candidates for training the classifier were randomly selected until the number of "no state update" example actions was on par with the number of example actions for the other topic classes. Unselected candidate actions were not used to train the classifier.

The advantage of selecting all actions that predict the next topic with high probability, instead of the only actions that predict topic transitions (between two topics) is that it enables more robust topic state estimation that can recover from errors. For example, when the estimator misses the topic state update at the first utterance in a topic run, there is a chance of updating the topic state correctly during the next utterance in the topic run.

3) Train the logistic regression classifier on the utterances from the customer actions (speech clusters) and topics found in Step 2. The training inputs to the logistic regression classifier consist of the utterance vectorizations of all the utterances in the speech clusters that predict the next topic with high probability. Their corresponding outputs are the most highly predicted topic states s_τ or the "no state update" class.

Training a logistic regression model on the contents (utterances) of the customer speech actions takes advantage of statistical learning to make the topic state estimation more robust to variations in speech not represented in the training data, while still providing some degree of interpretability since the actions and topics that were used for training are easy to see.

The topic state also needs to be updated based on the robot's actions, which is a simpler procedure than updating based on human actions since the robot's actions are explicitly represented in the system during run-time. Whenever the robot action predictor outputs a speech action that predicts the next topic with high probability, the topic state estimator updates the topic state directly. An example of a robot action that updates the state is, 3016 "Okay, that's a great choice" → Closing.

After training, the topic state estimator achieved a prediction accuracy of 70.3% on a manually annotated subset of 15 interactions containing 424 utterances.

D. Robot Action Prediction With Topic State

After the topic states are estimated for each turn in the training interactions, the interaction rules are trained. The topic state was incorporated into the previously proposed robot action predictor (Fig. 2) by including topic state in the conditional probability that is used to learn the interaction rules, as shown in (5), where s_t is the topic state at time t.

$$a_{TA,t} = \underset{a \in A_{TA}}{\operatorname{argmax}}\ P\left(a | a_{C,t}, s_t\right). \quad (5)$$

TABLE V
EXAMPLES OF RULES LEARNED BY THE SYSTEM WITH TOPIC STATE

Input Customer Action	Customer Action Typical Utterance	State	Predicted Travel Agent Action
2001	Hello.	Opening	Hi there, how can I help you today?
2017	Okay, and how long is that trip?	Desert	That is 14 nights.
		London	Five nights.
		Antarctica	That is 10 days.
2022	Okay, I think I'm going to go with the London trip.	Closing	Okay, that's a great choice.

Thus, a customer speech action/topic state pair is input to the action predictor and the most frequent travel agent action observed following that pair in the training data is returned. The rest of the system works the same as described in Section IV.

E. Interaction Rules With Topic State

Readability of the interaction rules learned by the system with topic state (Table V) allows a human designer to understand how the robot will respond to input customer actions, and the rules that include topic state reveal how the topic of conversation affects the robot's decision.

The rules learned with topic state allow the system to respond correctly to ambiguous customer actions. For example, Table V shows three rules learned for responding to the ambiguous customer action 2017 ("Okay and how long is the trip?"). Each of the three rules provides the correct information about the travel package indicated by the topic state. Incorporation of topic state solves the problem of ambiguity illustrated in Section IV-C, where the without-state system could learn only a single response to customer action 2017.

In a second preliminary offline evaluation, the system with topic state responded to ambiguous customer utterances with a higher rate of correctness than the without-state system (62% versus 39%, $\chi 2(1, \mathrm{N} = 190) = 17.70, p < .001$), and performed better overall on all customer utterances (67% versus 55%, $\chi 2(1, \mathrm{N} = 470) = 12.06, p < .001$). Thus, incorporating knowledge of the interaction structure aided in resolving ambiguous speech.

VI. OFFLINE SYSTEM EVALUATION

A more thorough offline evaluation was conducted on the human–human data to compare the performance of the proposed system to several systems using state-of-the-art techniques.

A. Experimental Design

The focus of this study was to develop a technique for generating appropriate robot behaviors by modeling the structure of the interaction, so the experimental design compared the behaviors of six action prediction conditions.

Nearest neighbor worked by matching the customer's utterance to the closest customer utterance in the training data using the cosine distance between the utterance vectors (Section III-E). The robot then performed the same travel agent utterance that followed in the training data. This is a commonly used baseline for data-driven dialog systems [34], [37].

Without-state was the same as the proposed system but without the topic state estimation module (presented in Section V). This condition was chosen to examine the effects of incorporating the state estimation module.

Non-negative matrix factorization (NMF) topic was the same as the proposed system but used NMF instead of the proposed topic state estimator. NMF, an unsupervised topic modeling algorithm similar to LDA, is a state-of-the-art method for topic modeling of small datasets [48], [49]. The topic model was trained by setting the NMF parameters to discover five topics from the human–human dataset, using the term frequency-inverse document frequency representations of utterances as input. During runtime, the topic state was updated whenever the maximum NMF topic of an utterance was greater than an empirically-set threshold (0.05), otherwise the previous topic was retained. This condition was chosen to compare to an existing topic modeling method.

Recurrent neural network (RNN) consisted of a many-to-one recurrent neural network with three layers: An input layer corresponding to the customer utterance vector, a 100-unit RNN layer, and a softmax output layer with each unit corresponding to a robot speech cluster ID. The RNN was trained for 1000 epochs using the categorical cross entropy loss function and the Adagrad update function [50]. The learning rate was 0.001 and the batch size was 128. For prediction, the customer's utterance vectorizations were input and the typical utterance of the predicted speech cluster was the final output. RNNs are a widely used machine learning approach for end-to-end dialog systems [35], [51] and other natural language processing tasks [52], [53]; therefore, they are a suitable for comparison.

Long short-term memory (LSTM) was identical to the RNN system, except the RNN layer was replaced by an LSTM layer. LSTMs were designed to overcome the shortcomings of RNNs in modeling long term dependencies [54] and have recently been used for dialog systems [51]. The LSTM system can potentially retain topic information in memory longer, so it was hypothesized to perform better than the RNN system.

The proposed consisted of the complete system introduced in the previous sections.

The prediction accuracies for the subsets of ambiguous and unambiguous customer actions were also separately analyzed. It was expected that all the systems would perform equally well on the unambiguous actions, but the proposed system would perform better on ambiguous actions than the other systems.

B. Evaluation Procedure

Hold-one-out cross validation was used to train and test the system. That is, to evaluate on each of the 192 interactions, the system was trained on the other 191 interactions (192 folds). 36 interactions (about a third of the dataset), containing 500 utterance predictions, were randomly selected for evaluation by a human evaluator. An expert annotator labeled the 500 customer utterances as either ambiguous or unambiguous. The

TABLE VI
OFFLINE ACTION PREDICTION RESULTS ON ALL CUSTOMER UTTERANCES, AND THE AMBIGUOUS AND UNAMBIGUOUS SUBSETS OF CUSTOMER UTTERANCES (ASTERISKS REPRESENT COMPARISON WITH PROPOSED * $p < .05$, ** $p < .01$, *** $p < .001$)

	Customer Utterances					
	Ambiguous		Unambiguous		All	
Nearest Neighbor	36%	***	59%	***	49%	***
Without-state	38%	***	71%	n.s.	58%	***
NMF Topic	47%	**	73%	n.s.	62%	*
RNN	45%	**	73%	n.s.	62%	*
LSTM	48%	**	**78%**	n.s.	66%	n.s.
Proposed	**62%**		75%		**69%**	

TABLE VII
OFFLINE EVALUATION YATES' CHI-SQUARED TEST RESULTS

		Customer Utterances		
		Ambiguous N=195	Unambiguous N=274	All N=469
Proposed vs. Nearest Neighbor	χ^2	24.63	14.47	37.41
	p	**<.001**	**<.001**	**<.001**
Proposed vs. Without-state	χ^2	20.77	0.59	13.40
	p	**<.001**	.442	**<.001**
Proposed vs. NMF Topic	χ^2	8.11	0.15	5.15
	p	**.004**	.698	**.023**
Proposed vs. RNN	χ^2	10.56	0.04	5.78
	p	**.001**	.846	**.016**
Proposed vs. LSTM	χ^2	7.55	1.01	1.24
	p	**.006**	.314	.265

ambiguous subset of the data contained 201 utterance pairs and the unambiguous subset contained 299 utterance pairs.

One human evaluator (F, age 38), blind to the experimental conditions, evaluated each predicted robot action with a binary label of either correct or incorrect. To receive a correct label the robot's action must have provided the correct information. If the customer did not request any specific information, then the criterion was that the robot's action must be socially appropriate. The evaluations were made based on transcripts of the human–human interactions automatically transcribed using ASR. In the case that the customer's utterance was clipped or contained too many ASR errors for the evaluator to determine if the predicted action was correct or not, the instance was not included in the evaluation (6% of all instances). Evaluations by a second evaluator on ten percent of the 500 instances showed a high degree of agreement, with a kappa coefficient of 0.80, so the evaluations were judged to be reliable.

C. Offline Evaluation Results

Table VI shows the results of the offline evaluation. A Yates' chi-squared test was used to test for statistical significance between the proposed system and the other systems (Table VII).

On the ambiguous customer utterances the proposed system performed significantly better than all other conditions. On the unambiguous customer utterances there were no significant differences between the proposed system and the other systems, except the nearest neighbor system (proposed 74.5% versus nearest neighbor 58.5%, $\chi 2(1, N = 469) = 14.47$, $p < .001$). On all utterances the proposed system performed significantly better than the other systems, except the LSTM (proposed 69.3% versus LSTM 65.5%, $\chi 2(1, N = 469) = 1.24$, $p = .265$).

The proposed system outperformed all other systems in handling ambiguous customer questions, the focus of this study. Additionally, the proposed system provided a human readable representation of the interaction logic. Although there was no significant difference between the proposed system and LSTM on all utterances, the proposed system performed significantly better on the ambiguous customer utterances (proposed 62.1% versus LSTM 47.7%, $\chi 2(1, N = 195) = 7.55$, $p = .006$).

Although LSTMs were designed for modeling long term dependencies, the LSTM system failed to learn the topic dependence of ambiguous utterances in this evaluation. LSTMs (and RNNs) typically require very large amounts of data to effectively capture such dependencies, so the small dataset size may have caused their poor performance. The ability to learn from a small dataset is one of the strengths of the proposed system.

Thus, the proposed system outperformed the state-of-the-art unsupervised learning techniques at dealing with ambiguity in human speech in the travel agent interaction scenario.

VII. USER EVALUATION

A user study was conducted to evaluate the proposed system with a real robot interacting with real people. A trial example and some special cases are shown in the supplementary video.

A. Experimental Design

The experimental design consisted of a within-participants design with three experimental conditions: 1) the proposed system, 2) the without-state system, and 3) the nearest neighbor based system, as described in Section VI-A.

B. Participants

Fifteen external participants (nine male, six female, mean age 34.3, s.d. 8.1) were recruited, all fluent English speakers with little to no experience interacting with robots, and no knowledge of the details of the experiment beyond the instruction's contents.

C. Experimental Setup

Participants interacted with ERICA, an android in the form of a young woman (Fig. 1) [55]. The android has 19 DOF, including 13 actuators in the face for eye gaze, facial expressions, and speech movements. She uses Hoya's VoiceText software (http://voicetext.jp/) to synthesize speech and ASR available in the Google Speech API to transcribe human speech collected through a handheld microphone. An array of ceiling-mounted sensors allows the robot to track the positions of people in the room and direct her gaze at them [56].

During the interactions, the participant's speech was collected in real time and fed into the run-time system (Fig. 2). The robot's actions were selected by the robot action predictor and synthesized using the android's speech synthesis. Facial expressions and gestures (smile, head nod, etc.) were randomized during each robot action to make her more animated. The robot's lip and

trunk movements were automatically generated in synchrony with the speaking rhythm based on the audio output from the speech synthesizer [57]. Interactions were recorded using three 1080p high resolution webcams and a microphone for the participant's speech, and the android's speech stream was recorded directly to file.

D. Procedure

The participant played the role of a customer and the robot acted as a travel agent. Participants were instructed to role-play three interactions in each of the experimental conditions (nine interactions per participant). In order to test the robot's behaviors in response to a variety of different customer behavior patterns, each interaction was role-played as one of the three customer types used in the human–human data collection.

Each interaction started with the participant standing near the door, and then they were instructed to walk up to and greet the android before stating what they were looking for. Furthermore, they were instructed to finish each interaction by stating their decision—which travel package they wished to purchase or if they were still undecided—and then walk back to the starting location. Last, the participants were instructed to treat each interaction as if it was the first and to forget whatever information was collected in previous interactions.

The order of the experimental conditions was varied for each participant and the order of the customer types was kept the same for each condition to reduce ordering effects. After each round of three interactions in one condition, the participant was instructed to fill out a questionnaire. Since it was not clear to the participants whether some of the robot's behaviors were correct or not (e.g., when providing a price), the experimenter kept a tally of the number of correct and incorrect robot actions during each condition and provided the participant with that information. The participant then completed a questionnaire.

E. Measurement

The effects of the experimental conditions on the robot's behavior were measured in two ways. A questionnaire was used to collect the participant's subjective, qualitative judgments of the robot's behaviors, and transcripts of the experiment interactions were annotated to obtain an objective count of the number of correct and incorrect actions.

1) Questionnaire: A questionnaire containing five questions was designed to compare the participants' subjective ratings.

- Q1. How understandable was the wording and flow of the robot's speech? (1 = "very difficult to understand", 7 = "very easy to understand"). The proposed and with-state systems should be the easiest to understand since the robot speaks the typical utterances of speech clusters.
- Q2. Were you able to get the information you asked for? (1 = "not at all", 7 = "yes, completely"). The proposed system should provide the most information since it is designed to respond to ambiguous questions.
- Q3. How much effort was required to get the information you asked for? (1 = "completely effortless", 7 = "maximum effort"). The proposed system should require the least effort since it should make fewer mistakes, minimizing the need to rephrase questions.
- Q4. How well did the robot respond to ambiguous questions? (1 = "failed completely", 7 = "greatly succeeded"). The proposed system should respond to ambiguous questions the most accurately since it can track the topic of conversation.
- Q5. How do you think the robot performed overall (including the above points)? (1 = "failed completely", 7 = "greatly succeeded"). The proposed system should perform the best overall for the reasons stated above.

TABLE VIII
USER STUDY RESULTS (p VALUES REPRESENT POST-HOC TUKEY TEST COMPARISON WITH PROPOSED)

	Nearest Neighbor		Without-state		Proposed
Q1. Understandability	4.2	$p<.001$	**6.0**	$p=.845$	5.8
Q2. Information	3.6	$p<.01$	4.3	$p<.05$	**5.1**
Q3. Effort	4.2	$p<.001$	3.2	$p<.05$	**2.4**
Q4. Ambiguity	2.9	$p<.001$	3.2	$p<.001$	**4.1**
Q5. Overall	3.3	$p<.001$	4.1	$p<.001$	**5.1**
Mean Rate of Correct Robot Actions	49%	$p<.001$	54%	$p<.01$	**65%**

2) Quantitative Measurement: In addition to asking the participants their subjective judgements, a quantitative evaluation was conducted in which the robot's actions were rated by a human evaluator.

One evaluator (male, age 22), blind to the experimental conditions, evaluated the robot responses in all 135 interactions (15 participants × three conditions × three interactions per condition) using the same procedure as in Section VI-B. 3% of all instances were excluded from the evaluation because of ASR errors in the experiment transcript that made them impossible to judge. Evaluations by a second evaluator on ten percent of the data showed a high degree of agreement, with a kappa coefficient of 0.80. So, the evaluations were judged to be reliable.

It was hypothesized that the proposed system would have the highest rate of correct actions, followed by the without-state system, and the nearest neighbor system performing the worst.

F. Results

1) Questionnaire Results: Table VIII shows the results for the questionnaire. One-way repeated measures ANOVA with Greenhouse–Geisser correction determined that there was a statistically significant effect of the experimental conditions on each of the questionnaire responses. Q1. $F(1.283, 17.963) = 14.884$, $p < .001$; Q2. $F(1.642, 22.988) = 14.512$, $p < .001$; Q3. $F(1.554, 21.753) = 19.409$, $p < .001$; Q4. $F(1.746, 24.450) = 18.984$, $p < .001$; and Q5. $F(1.523, 21.322) = 41.323$, $p < .001$.

Posthoc Tukey tests (Table VIII) found significant differences between the proposed system and the other two systems on all questions, except for the proposed and without-state systems on Q1.

In conclusion, the questionnaire responses supported the hypotheses, with proposed rated the best on all questions, followed

by without-state (with the exception of Q1), and the nearest neighbor system coming in with the worst ratings.

2) Quantitative Results: The last row of Table VIII shows the quantitative evaluation results, each percentage representing the mean robot action prediction correctness computed over all experiment trials ($N = 15$) in the indicated experimental condition.

The proposed system achieved a mean robot action prediction correctness rate of 65.2%, without-state 54.2%, and nearest neighbor 49.2%. A one-way repeated measures ANOVA determined that the experimental conditions had a statistically significant effect on the mean rates of robot action correctness: $F(2, 28) = 12.821$, $p < .001$. A posthoc Tukey test (Table VIII) found statistically significant differences between the proposed condition and the two other conditions. Thus, the proposed system performed better than the without-state and nearest neighbor systems. Therefore, this evaluation validates the effectiveness of the proposed system.

G. Topic State Estimation Results

The proposed topic state estimator estimated the state correctly 69% of the time and recovered on the next turn 18% of the time in the case of errors. Thus, the estimator was able to generalize from the training data and was robust to some errors.

H. Analysis of Errors

The system made incorrect responses for five reasons: Action matching errors (33% of all errors), customer utterances not present in the training data (23%), topic state estimation errors (18%), incorrectly learned rules (18%), and ASR errors (8%). This suggests that the system's performance could be increased by focusing improvements on the utterance-to-action matching mechanism and dealing with out-of-scope customer utterances.

VIII. Discussion and Conclusion

The main objectives of this work fit under the general goal of learning interaction behaviors for a robot using data-driven methods, without input from a human designer. First, the system was to learn how to respond to ambiguous human actions that depend on hidden state. Second, the learned behaviors were to be represented in a human-readable way.

A. Human-Readable Interaction Rules

Human-readability enables debugging of faulty robot behaviors by a designer. In future work a hybrid system could learn social robot interaction logic from data but also allow for manual debugging and adjustment. Additionally, rule interpretability allows a designer to examine the list of robot interaction rules in order to validate them for safe operation.

The proposed system achieves human-readability in two ways: First, by finding an explicit set of interaction rules, which map a clearly-defined set of human actions to a clearly-defined set of robot responses, and second, defining a discrete set of topics makes the dependence on hidden state explicit. Both are important for interpretability of the interaction logic.

In contrast, neural network architectures such as RNNs, LSTMs, and deep neural networks with attention, as presented in [7] may achieve higher prediction accuracy with larger datasets, but they are black boxes. The downside of such machine learning models in general is that the reasoning behind their decisions is relatively unclear to human understanding.

B. Learning Topic From Action Co-Occurrences

This work presents a novel approach to discover topic by clustering actions based on action co-occurrences. The proposed topic clustering algorithm is unique since it is based on abstract, discrete symbols representing actions, and not any lower level information, like words or language. There are at least two advantages to discover topics through action co-occurrences rather than textual analysis. First, by forgoing textual analysis, and representing actions with discrete symbols instead, the proposed technique can be applied to other languages and even other modalities. Second, techniques that rely on textual analysis (e.g., [18], [19]) are reliant on having error-free, grammatically correct sentences, so it is difficult to apply them to automatically transcribed, natural, human conversation data with many disfluencies and ASR errors.

An additional result of operating at the level of actions is that the topic clustering algorithm learns structure in the pragmatic, social level, rather than the semantic level. This enables it to discover that two actions are about the same topic even when it is not clear from the textual content. For example, the question "How much is the Antarctica vacation?" and the answer "It is $3000" have no distinguishing words that hint they belong to the same topic, but the high frequency of co-occurrence of these two actions suggests that they are about the same thing.

C. Applicability to Multiple Modalities

Many techniques focus on text, but the proposed system can potentially be applied to any modality whose data can be clustered to discover a set of abstract, discrete actions, such as individuals' walking trajectories and stopping points [5]–[7]. In the future this work could be extended to learn facial expressions, hand gestures, and vocal inflections for heightened robot expressivity. Also, visual attention [24] could be integrated to enable communication about the physical world.

D. Limitations of the Human–Human Interaction Dataset

The main limitation of the travel agent dataset (Section III) is the lesser number (six) of participants. In the future, our proposed system may be applied to data collected in the real world. Therefore, the lab-collected dataset should have the same characteristics (but at a smaller scale).

It is reasonable to have only a few participants playing the travel agent in the training data, as this reflects the reality of the target domain, where the number of human employees available for training the system is also limited. Moreover, sometimes it may even be desirable to train the system on a single travel agent's data to learn their specific character and interaction style. Therefore, the training dataset is sufficient in this regard.

In contrast, it is important to have variation in customer behavior in the training dataset to demonstrate that the proposed system can learn correct customer speech clusters and interaction rules despite variation.

Greater variation of customer behavior may make customer speech clustering more difficult. However, the core customer actions (asking about travel package features, etc.) would remain the same regardless of the number of customers, so the increase in variation could be offset by collecting more data, containing more examples of the core actions. Thus, while the number of customer participants in the training dataset might have mixed impact on system performance, it is not a critical weakness, and it does not invalidate the study's results.

Furthermore, the number of customers did not affect the proposed system's generalizability: In the offline experiment, in which the system was evaluated on interactions from the training dataset with six participants, the rate of correct robot actions was 69%. In the user study, which was conducted with 15 new participants, the rate of correct robot actions was 65%. The small difference in performance between these two experiments (4%) suggests that the proposed system was able to generalize despite the small training dataset size.

E. Generalizability to Broader Domains and Larger Datasets

In a broader domain (e.g., a real, nonlimited travel agent scenario with dozens of travel packages), the number of actions and topics would increase. However, with a large enough dataset, with sufficient examples of each action and topic for speech and topic clustering, the proposed system will still be capable of learning the repeatable aspects of the interaction.

Concerning dataset size (quantity), the set of common abstract actions will remain the same regardless. The number of phrasing variations of actions may increase with the size of the dataset, but so also will the number of examples of each phrasing. The purpose of the proposed system is to learn these common actions, i.e., the repeatable, core aspects of the interaction. Therefore, the proposed system's performance is expected to improve if the size of the dataset is increased and the set of abstract actions remained constant.

F. Conclusion

This study presented a technique for automatically learning social robot interaction behaviors, in the form of human-readable rules, from unannotated human–human interaction examples that are full of ASR errors, natural speech variation, and disfluencies. Additionally, a novel topic clustering algorithm was introduced that discovered topics and phases of interaction based on action co-occurrences, to resolve ambiguous human speech. In evaluation, the proposed system performed significantly better than the five other systems. The proposed technique was demonstrated in a travel agency scenario in which human participants interacted with a real robot. In the future, this behavior learning approach could be extended to multiple modalities, such as gestures and facial expressions. It would also be beneficial to incorporate online learning to learn during real-time human-robot interaction.

REFERENCES

[1] K. Kuwamura, S. Nishio, and H. Ishiguro, "Designing robots for positive communication with senior citizens," in *Intelligent Autonomous Systems 13. Advances in Intelligent Systems and Computing*, E. Menegatti, N. Michael, K. Berns, and H. Yamaguchi Eds. Cham, Switzerland: Springer-Verlag, 2016, pp. 955–964.

[2] H. A. Samani, A. D. Cheok, F. W. Ngiap, A. Nagpal, and M. Qiu, "Towards a formulation of love in human-robot interaction," in *Proc. Int. Symp. Robot Human Interact. Commun.*, 2010, pp. 94–99.

[3] S. Guo, J. Lenchner, J. H. Connell, M. Dholakia, and H. Muta, "Conversational bootstrapping and other tricks of a concierge robot," in *Proc. Int. Conf. Human–Robot Interact.*, 2017, pp. 73–81.

[4] S. Rosenthal, J. Biswas, and M. Veloso, "An effective personal mobile robot agent through symbiotic human-robot interaction," in *Proc. Int. Conf. Auton. Agents Multiagent Syst.*, 2010, pp. 915–922.

[5] P. Liu, D. F. Glas, T. Kanda, and H. Ishiguro, "Data-driven HRI: Learning social behaviors by example from human–human interaction," *IEEE Trans. Robot.*, vol. 32, no. 4, pp. 988–1008, Aug. 2016.

[6] P. Liu, D. F. Glas, T. Kanda, H. Ishiguro, and N. Hagita, "How to train your robot-teaching service robots to reproduce human social behavior," in *Proc. Int. Symp. Robot Human Interact. Commun.*, 2014, pp. 961–968.

[7] P. Liu, D. F. Glas, T. Kanda, and H. Ishiguro, "Learning proactive behavior for interactive social robots," *Auton. Robots* vol. 42, no. 5, pp. 1067–1085, 2017.

[8] M. G. Core, H. C. Lane, M. Van Lent, D. Gomboc, S. Solomon, and M. Rosenberg, "Building explainable artificial intelligence systems," in *Proc. 18th Conf. Innov. Appl. Artif. Intell.*, 2006, pp. 1766–1773.

[9] D. Gunning, "Explainable artificial intelligence (XAI)," Defense Advanced Research Projects Agency, Arlington, VA, USA, Tech. Rep. DARPA-BAA-16-53, 2017.

[10] D. Martens, B. Baesens, T. Van Gestel, and J. Vanthienen, "Comprehensible credit scoring models using rule extraction from support vector machines," *Eur. J. Oper. Res.*, vol. 183, no. 3, pp. 1466–1476, 2007.

[11] J. Orkin and D. Roy, "The restaurant game: Learning social behavior and language from thousands of players online," *J. Game Develop.*, vol. 3, no. 1, pp. 39–60, 2007.

[12] J. Orkin and D. Roy, "Automatic learning and generation of social behavior from collective human gameplay," in *Proc. Int. Conf. Auton. Agents Multiagent Syst.*, 2009, pp. 385–392.

[13] C. Breazeal, N. DePalma, J. Orkin, S. Chernova, and M. Jung, "Crowdsourcing human-robot interaction: New methods and system evaluation in a public environment," *J. Human Robot Interact.*, vol. 2, no. 1, pp. 82–111, 2013.

[14] H. Admoni and B. Scassellati, "Data-driven model of nonverbal behavior for socially assistive human-robot interactions," in *Proc. 16th Int. Conf. Multimodal Interact.*, 2014, pp. 196–199.

[15] D. M. Blei, "Probabilistic topic models," *Commun. ACM*, vol. 55, no. 4, pp. 77–84, 2012.

[16] L. Hong and B. D. Davison, "Empirical study of topic modeling in twitter," in *Proc. 1st Workshop on Social Media Anal.*, 2010, pp. 80–88.

[17] H. M. Wallach, "Topic modeling: Beyond bag-of-words," in *Proc. 23rd Int. Conf. Mach. Learn.*, 2006, pp. 977–984.

[18] B. J. Grosz, S. Weinstein, and A. K. Joshi, "Centering: A framework for modeling the local coherence of discourse," *Comput. Linguistics*, vol. 21, no. 2, pp. 203–225, 1995.

[19] B. J. Grosz and C. L. Sidner, "Attention, intentions, and the structure of discourse," *Comput. Linguistics*, vol. 12, no. 3, pp. 175–204, 1986.

[20] K. P. Murphy, *Dynamic Bayesian Networks: Representation, Inference and Learning*. Berkeley, CA, USA: Univ. California, 2002.

[21] M. Purver, T. L. Griffiths, K. P. Körding, and J. B. Tenenbaum, "Unsupervised topic modelling for multi-party spoken discourse," in *Proc. 21st Int. Conf. Comput. Linguistics and 44th Annu. Meeting Assoc. Comput. Linguistics*, 2006, pp. 17–24.

[22] K. E. Boyer *et al.*, "Investigating the relationship between dialogue structure and tutoring effectiveness: A hidden Markov modeling approach," *Int. J. Artif. Intell. Educ.*, vol. 21, no. 1/2, pp. 65–81, 2011.

[23] J. F. Maas, T. Spexard, J. Fritsch, B. Wrede, and G. Sagerer, "Biron, what's the topic? A multi-modal topic tracker for improved human-robot interaction," in *Proc. Int. Symp. Robot Human Interact. Commun.*, 2006, pp. 26–32.

[24] P. Lanillos, J. F. Ferreira, and J. Dias, "Designing an artificial attention system for social robots," in *Proc. Int. Conf. Intell. Robots Syst.*, 2015, pp. 4171–4178.

[25] A. Stolcke *et al.*, "Dialogue act modeling for automatic tagging and recognition of conversational speech," *Comput. Linguistics*, vol. 26, no. 3, pp. 339–373, 2000.
[26] R. Higashinaka *et al.*, "Unsupervised clustering of utterances using nonparametric Bayesian methods," in *Proc. INTERSPEECH*, pp. 2081–2084, 2011.
[27] A. Ezen-Can and K. E. Boyer, "Understanding student language: An unsupervised dialogue act classification approach," *J. Educ. Data Mining*, vol. 7, no. 1, pp. 51–78, 2015.
[28] H. Sacks, E. A. Schegloff, and G. Jefferson, "A simplest systematics for the organization of turn-taking for conversation," *Language*, vol. 50, pp. 696–735, 1974.
[29] T. D. Midgley, S. Harrison, and C. MacNish, "Empirical verification of adjacency pairs using dialogue segmentation," in *Proc. 7th SIGdial Workshop Discourse Dialogue*, 2009, pp. 104–108.
[30] K. E. Boyer, R. Phillips, E. Y. Ha, M. D. Wallis, M. A. Vouk, and J. C. Lester, "Modeling dialogue structure with adjacency pair analysis and hidden Markov models," in *Proc. Conf.: Human Lang. Technol. North Amer. Chapter Assoc. Comput. Linguistics*, 2009, pp. 49–52.
[31] D. Jurafsky and J. H. Martin, *Speech and Language Processing*. London, U.K.: Pearson, 2014.
[32] K. Clark and C. D. Manning, "Deep reinforcement learning for mention-ranking coreference models," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2016, pp. 2256–2262.
[33] T. Wolf, State-of-the-art neural coreference resolution for chatbots, Accessed: Nov. 28, 2017. [Online]. Available: https://medium.com/huggingface/state-of-the-art-neural-coreference-resolution-for-chatbots-3302365dcf30
[34] A. Ritter, C. Cherry, and W. B. Dolan, "Data-driven response generation in social media," in *Proc. Conf. Empirical Methods Natural Lang. Process*, 2011, pp. 583–593.
[35] O. Vinyals and Q. Le, "A neural conversational model," 2015, arXiv:1506.05869.
[36] J. D. Williams, A. Raux, D. Ramachandran, and A. W. Black, "The dialog state tracking challenge," in *Proc. SIGdial Workshop Discourse Dialogue*, 2013, pp. 404–413.
[37] A. Bordes, Y.-L. Boureau, and J. Weston, "Learning end-to-end goal-oriented dialog," 2016, arXiv:1605.07683.
[38] M. Henderson, "Machine learning for dialog state tracking: A review," in *Proc. 1st Int. Workshop on Mach. Learn. Spoken Lang. Process.*, 2015.
[39] M. Henderson, B. Thomson, and S. Young, "Robust dialog state tracking using delexicalised recurrent neural networks and unsupervised adaptation," in *Proc. IEEE Spoken Lang. Technol.Workshop*, 2014, pp. 360–365.
[40] K. Jokinen, H. Tanaka, and A. Yokoo, "Context management with topics for spoken dialogue systems," in *Proc. 36th Annu. Meeting Assoc. Comput. Linguistics and 17th Int. Conf. Comput. Linguistics*, 1998, pp. 631–637.
[41] A. Breuing and I. Wachsmuth, "Let's talk topically with artificial agents! Providing agents with humanlike topic awareness in everyday dialog situations," in *Proc. 4th Int. Conf. Agents Artif. Intell.*, 2012, pp. 62–71.
[42] P. Liu, D. F. Glas, T. Kanda, and H. Ishiguro, "Two demonstrators are better than one-a social robot that learns to imitate people with different interaction styles," *IEEE Trans. Cogn. Develop. Syst.*, 2017, doi: 10.1109/TCDS.2017.2787062.
[43] I. Leite, A. Pereira, A. Funkhouser, B. Li, and J. F. Lehman, "Semi-situated learning of verbal and nonverbal content for repeated human–robot interaction," in *Proc. 18th ACM Int. Conf. on Multimodal Interact.*, 2016, pp. 13–20.
[44] M. Shiomi, D. Sakamoto, T. Kanda, C. T. Ishi, H. Ishiguro, and N. Hagita, "A semi-autonomous communication robot–A field trial at a train station," in *Proc. IEEE Int. Conf. Human-Robot Interact.*, 2008, pp. 303–310.
[45] P. Langfelder, B. Zhang, and S. Horvath, "Defining clusters from a hierarchical cluster tree: The Dynamic tree cut package for R," *Bioinformatics*, vol. 24, no. 5, pp. 719–720, 2008.
[46] R. Agrawal, T. Imieliński, and A. Swami, "Mining association rules between sets of items in large databases," *ACM SIGMOD Rec.*, vol. 22, no. 2, pp. 207–216, 1993.
[47] S. E. Schaeffer, "Graph clustering," *Comput. Sci. Rev.*, vol. 1, no. 1, pp. 27–64, 2007.
[48] X. Yan, J. Guo, S. Liu, X. Cheng, and Y. Wang, "Learning topics in short texts by non-negative matrix factorization on term correlation matrix," in *Proc. SIAM Int. Conf. Data Mining*, 2013, pp. 749–757.
[49] S. Arora *et al.*, "A practical algorithm for topic modeling with provable guarantees," in *Proc. 30th Int. Conf. Mach. Learn.,* 2013, pp. 280–288.
[50] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *J. Mach. Learn. Res.*, vol. 12, pp. 2121–2159, 2011.
[51] R. T. Lowe, N. Pow, I. V. Serban, L. Charlin, C.-W. Liu, and J. Pineau, "Training end-to-end dialogue systems with the Ubuntu dialogue corpus," *Dialogue Discourse*, vol. 8, no. 1, pp. 31–65, 2017.
[52] M. Henderson, B. Thomson, and S. Young, "Word-based dialog state tracking with recurrent neural networks," in *Proc. 15th Annu. Meeting Special Interest Group Discourse Dialogue*, 2014, pp. 292–299.
[53] J. Williams, A. Raux, and M. Henderson, "The dialog state tracking challenge series: A review," *Dialogue Discourse*, vol. 7, no. 3, pp. 4–33, 2016.
[54] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[55] D. F. Glas, T. Minato, C. T. Ishi, T. Kawahara, and H. Ishiguro, "Erica: The erato intelligent conversational android," in *Proc. 25th IEEE Int. Symp. Robot Human Interact. Commun.*, 2016, pp. 22–29.
[56] D. Brscic, T. Kanda, T. Ikeda, and T. Miyashita, "Person tracking in large public spaces using 3-D range sensors," *IEEE Trans. Human–Mach. Syst.*, vol. 43, no. 6, pp. 522–534, Nov. 2013.
[57] K. Sakai, T. Minato, C. T. Ishi, and H. Ishiguro, "Speech driven trunk motion generating system based on physical constraint," in *Proc. 25th IEEE Int. Symp. Robot Human Interact. Commun.*, 2016, pp. 232–239.

Malcolm Doering received the B.S. degree in computer science and the B.A. degree in linguistics in 2013, the M.S. degree in computer science from Michigan State University, East Lansing, MI, USA, in 2015. He is currently working toward the Ph.D. degree at Osaka University, Osaka, Japan, and is a Research Intern with the Advanced Telecommunications Research Institute International, Kyoto, Japan.

His research interests include human-robot interaction, natural language processing, machine learning, linguistics, and cognitive science.

Dylan F. Glas (M'18) received the M.Eng. degree in aerospace engineering from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 2000 and the Ph.D. degree in Robotics from Osaka University, Osaka, Japan, in 2013.

He was a Group Leader and a Senior Researcher with the Intelligent Robotics and Communication Laboratories and Hiroshi Ishiguro Laboratories at Advanced Telecommunications Research Institute International, Kyoto, Japan, and a Guest Associate Professor with Osaka University, Osaka, Japan. He is currently working as a Senior Robotics Software Architect at Futurewei Technologies, San Francisco, CA, USA.

Hiroshi Ishiguro (M'14) received the D.Eng. degree in systems engineering from Osaka University, Osaka, Japan, in 1991.

He is currently a Professor of the Department of Systems Innovation with the Graduate School of Engineering Science at Osaka University, Osaka, Japan, and a Distinguished Professor of Osaka University. He is also the Visiting Director of Hiroshi Ishiguro Laboratories at Advanced Telecommunications Research Institute International (ATR) and an ATR fellow. His research interests include sensor networks, interactive robotics, and android science.