**The 23rd IEEE International Symposium on Robot and Human Interactive Communication
August 25-29, 2014. Edinburgh, Scotland, UK,**

How to Train Your Robot – Teaching service robots to reproduce human social behavior

Phoebe Liu, Dylan F. Glas, Takayuki Kanda, *Member, IEEE*, Hiroshi Ishiguro, *Member, IEEE*, and Norihiro Hagita, *Senior Member, IEEE*

Abstract— **Developing interactive behaviors for social robots presents a number of challenges. It is difficult to interpret the meaning of the details of people's behavior, particularly non-verbal behavior like body positioning, but yet a social robot needs to be contingent to such subtle behaviors. It needs to generate utterances and non-verbal behavior with good timing and coordination. The rules for such behavior are often based on implicit knowledge and thus difficult for a designer to describe or program explicitly. We propose to teach such behaviors to a robot with a learning-by-demonstration approach, using recorded human-human interaction data to identify both the behaviors the robot should perform and the social cues it should respond to. In this study, we present a fully unsupervised approach that uses abstraction and clustering to identify behavior elements and joint interaction states, which are used in a variable-length Markov model predictor to generate socially-appropriate behavior commands for a robot. The proposed technique provides encouraging results despite high amounts of sensor noise, especially in speech recognition. We demonstrate our system with a robot in a shopping scenario.**

I. Introduction

As a variety of service applications are being explored for social robots, different approaches have been taken to developing autonomous behaviors, including rule-based approaches, flowchart-style programming, and a variety of learning-based methods ranging from teleoperation-based input to teaching through face-to-face social interaction.

While each technique has its merits and drawbacks, we believe that it is worthwhile to consider a data-driven learning-by-demonstration approach based on passive observation of natural human-human interactions. As environmental sensor systems and wearable and mobile devices become more widely available, such an approach could take advantage of very large sets of training data.

For example, if a chain of retail stores installed sensors in several shops and recorded data for just a few weeks, it could provide tens of thousands of interactions which could be used for training a robot to sell products, in a scenario like that shown in Fig. 1. Similar scenarios can be imagined for cafes, restaurants, museums, or many other industries.

A. *Programming social behaviors*

It is necessary for social robots to interact in humanlike ways, using speech, gesture, and proxemics, but it can be difficult to use traditional methods of programming such as scripting or flowchart-based design [1] to develop such social interactions for two reasons. First, the programmer or interaction designer may not be consciously aware of their own actions – we often control our body positioning, gaze, and reactive conversational utterances without thinking, and it is difficult to explicate the tacit knowledge of these actions or the cues that trigger them [2]. Second, social interactions depend greatly on the partner's behavior, and it can be difficult to anticipate the details of the possible situations to which a robot will need to react.

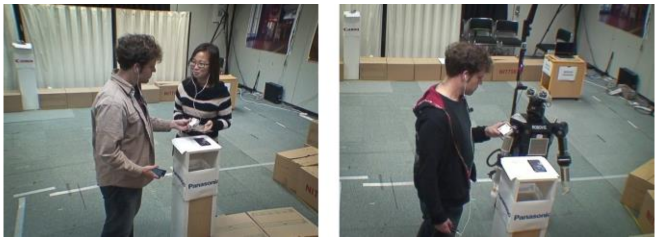


Figure 1. Teaching a robot to sell cameras. Abstracted motion and speech data from 178 human-human interactions were recorded (left) and used to train a robot to act as a shopkeeper (right).

We propose that, given a rich enough set of training situations, these rules and actions could be inferred from natural human interaction data using machine-learning techniques. Future systems for developing interactive social robot applications may even combine the two approaches: explicit programming and data-driven learning.

B. *Learning by Demonstration*

In robotic tasks like manipulation, machine learning is sometimes used as a behavior generation tool because it is easier and more intuitive to input poses by moving an arm manually, than to explicitly specify them numerically. Some of these learning techniques include trajectory following [3, 4] or joint motion replication [5] to reproduce tasks or gestures. Typically it is seen as a way to input sensory-motor patterns, but not cognitive and decision-making skills.

In social robotics, machine learning has been used to teach low-level behaviors, for example to mimic gestures and movements [6] and to learn how to direct gaze in response to gestural cues [7]. In one example, pointing and gaze behaviors were recognized in an imitative game using a hidden Markov model [8]. Lee et al. demonstrated a probabilistic approach for reproducing more structured tasks, such as a dance sequence

*This work was supported by JSPS KAKENHI Grant Number 25240042.

P. Liu, D. Glas, T. Kanda, and N. Hagita are with the ATR Intelligent Robotics and Communication Laboratories, Keihanna Science City, Kyoto, Japan, 619-0288 (corresponding author: P. Liu, phone: +81-774-95-1405; fax: +81-774-95-1405; e-mail: phoebe@ atr.jp).

H. Ishiguro is with the Intelligent Robotics Laboratory, Graduate School of Engineering Science, Osaka University, Toyonaka, Osaka, Japan, 560-8531 (e-mail: ishiguro@sys.es.osaka-u.ac.jp).

978-1-4799-6764-3/14/$31.00 ©2014 IEEE

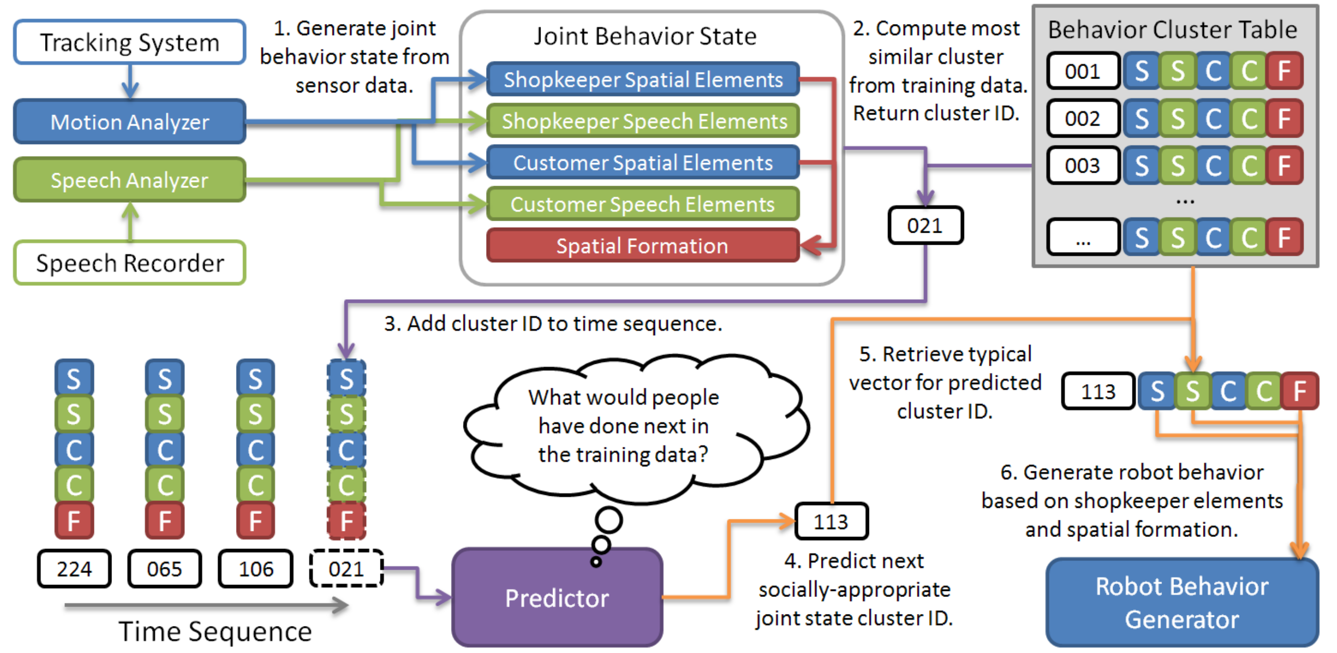

Tracking System
Motion Analyzer
Speech Analyzer
Speech Recorder
1. Generate joint behavior state from sensor data.
Joint Behavior State
Shopkeeper Spatial Elements
Shopkeeper Speech Elements
Customer Spatial Elements
Customer Speech Elements
Spatial Formation
2. Compute most similar cluster from training data. Return cluster ID.
Behavior Cluster Table
001 S S C C F
002 S S C C F
003 S S C C F
...
... S S C C F
021
3. Add cluster ID to time sequence.
S S C C F
224
S S C C F
065
S S C C F
106
S S C C F
021
Time Sequence
What would people have done next in the training data?
Predictor
113
4. Predict next socially-appropriate joint state cluster ID.
5. Retrieve typical vector for predicted cluster ID.
113 S S C C F
6. Generate robot behavior based on shopkeeper elements and spatial formation.
Robot Behavior Generator

Figure 2. Overall behavior generation procedure. 1: Sensor inputs are used to generate a joint behavior state for the shopkeeper (robot) and customer. 2: The current state vector is compared with clusters from the training data to find the best match. 3: The ID of the best match is added to the interaction history. 4: Based on the history, the predictor predicts what a human would have done in a similar situation in the training data. 5:The typical vector for that cluster ID is retrieved, and 6: Robot behaviors are generated based on the shopkeeper's behavior elements and spatial formation contained in that typical vector.

[9]. However, these techniques would not typically be expected to provide a level of decision-making complexity sufficient for entire social interactions

Yet, some social interactions exhibit repeatable patterns on a macro-scale. For example, sales staff in a shop might develop routine techniques for presenting products to customers, and they might respond to typical questions with similar answers each time a new customer asked them. In domains where sufficient repeatability is observed, we believe it may be possible to train robots to reproduce such high-level behavior patterns without modeling the underlying cognitive decision processes.

C. *Passive Sensing vs. Active Teaching*

Many studies have investigated methods of actively teaching tasks directly to robots or agents, where a human acts as a trainer or teacher and provides feedback to the robot, for example, [10] and [11].

It has been demonstrated that learning can be used to infer meanings from spoken utterances [12, 13]. These scenarios are usually restricted to a constrained set of vocabulary or using manual annotation to improve the learning algorithm.

Learning for social interactions has also been done based on passively-collected training data. In their "Crowdsourcing HRI" study, Breazeal et al. used a data-driven approach to develop HRI behaviors for a collaborative game, using an online simulation to provide large amounts of training data [14].

We propose to collect training data directly from real-world social interactions, by using a sensor network of environmental and mobile sensors. In this study, we create a simulated shop environment in which we record the motions and utterances of participants role-playing a customer-shopkeeper interaction. Based on this data, we train a robot to reproduce the shopkeeper's behavior through learning-by-demonstration.

II. Proposed Technique

In this section we will present our strategy for using learning by demonstration to reproduce human behavior in a robot.

A. *Strategy Overview*

Our overall strategy, summarized in Fig. 2, is to observe natural interactions between two people, and then to abstract their behavior into sequences of semantically-meaningful elements which are combined to form joint two-person states. We then build a model which can predict the next likely two-person state in a sequence. When a robot acts the role of one of the people, this model predicts what the human in the data set would have done in a similar situation. We then resolve the predicted states into action commands for the robot.

Of course, this high-level conceptual description is quite straightforward, and the challenge lies in the details. What kind of sensing is necessary? What abstraction models are meaningful, useful, and generalizable? What kind of classifier would be effective at generating socially-appropriate human behaviors?

Natural variation in social behaviors provides a major challenge - although humans understand that the utterances, "Hi, can I help you?" and "Do you need any assistance?" represent the same semantic meaning, a machine learning system does not know this relationship.

Sensor noise is also a major concern. In particular, speech recognition is a notoriously difficult problem in uncontrolled environments.

B. *Behavior elements*

Human behavior occupies a very high-dimensional feature space, considering the number of potential motions, utterances, gestures, and interactions we could perform at any given moment. In practice, however, the true variation of human behavior occupies only a small manifold within this high-dimensional space – people usually perform actions in predictable ways and follow common patterns.

To reduce this dimensionality for learning, we identify abstractions that we call **behavior elements**, semantically meaningful action units which constitute the building blocks of interactions. Our camera shop scenario includes **speech elements** like asking the price of a camera, and **spatial elements** like walking to the door or standing in front of a camera.

We define the requirements for behavior elements as follows:

1. They should represent atomic, semantically meaningful action units.
2. Semantically equivalent actions performed by different people should usually be mapped to the same abstract behavior element model.
3. They should be quickly detectable from a feed of live sensor data, so that the robot can react to people's behavior in real-time interactions.
4. They should be reproducible in a robot in a deterministic way, so that interaction data recorded from people can be used to generate robot behaviors.

It is necessary to create techniques for recognizing each behavior element, and for reproducing it in a robot.

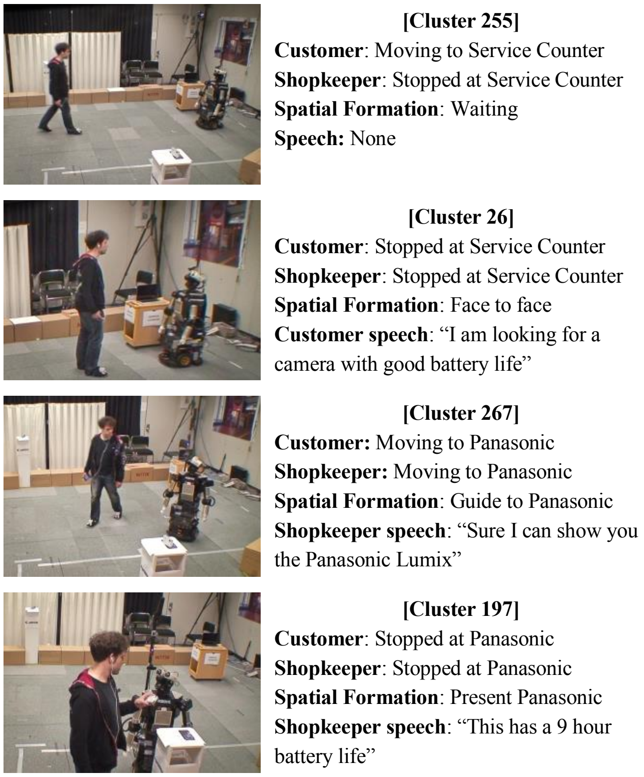
C. *Joint behavior representation*

Once the sensor data has been abstracted into behavior elements, we can examine their sequence over time. In a social interaction, actions taken by people are dependent upon both their own state and their interaction partner's state.

For example, if a customer asks for assistance while the shopkeeper is far away, the shopkeeper should approach the customer before responding; however, if the shopkeeper is already nearby, it is only necessary to respond verbally to the customer's question. The shopkeeper's next behavior is thus based on both the customer's state (location and utterance) and the shopkeeper's own state (location). For this reason, we combine the speech and motion behavior elements from both participants into a feature vector representing the **joint behavior state**, as shown in Fig. 2.

D. *Spatial formations and HRI models*

A number of studies in human-robot interaction have developed models for proxemics behavior in specific social situations. For example, the model proposed by Shi et al. [15] describes relative positioning for initiating conversation, and the model proposed by Yamaoka et al. describes the positions of two people talking about an object [16].



[Cluster 255]
Customer: Moving to Service Counter
Shopkeeper: Stopped at Service Counter
Spatial Formation: Waiting
Speech: None

[Cluster 26]
Customer: Stopped at Service Counter
Shopkeeper: Stopped at Service Counter
Spatial Formation: Face to face
Customer speech: "I am looking for a camera with good battery life"

[Cluster 267]
Customer: Moving to Panasonic
Shopkeeper: Moving to Panasonic
Spatial Formation: Guide to Panasonic
Shopkeeper speech: "Sure I can show you the Panasonic Lumix"

[Cluster 197]
Customer: Stopped at Panasonic
Shopkeeper: Stopped at Panasonic
Spatial Formation: Present Panasonic
Shopkeeper speech: "This has a 9 hour battery life"

Figure 3. Example sequence of joint behavior states from an interaction. Each joint state is comprised of several behavior elements and mapped to a cluster. This interaction sequence could be represented as 255-26-267-197. (Some intermediate steps have been omitted for brevity).

Such top-down models provide useful abstractions, because they can be used to specify proxemic constraints and other behavior at a detailed level for a robot. We detect several of these **spatial formations** based on HRI models, such as "presenting an object," and include them in the joint behavior state.

E. *Clustering joint behavior states*

Even after abstraction of the data into behavior elements, the number of joint states is quite large, so we cluster the observed vectors based on similarity to reduce the overall set of joint states to a tractable number. We define the state closest to a cluster's center as the typical joint behavior state for that cluster.

Each **joint behavior state cluster** is given a number and represents some common situation. Figure 3 shows an example sequence of joint behavior states from an interaction with a robot. After automatic clustering of the behavior vectors from the training set (see Section V) each joint behavior state cluster is assigned a number. In this example, as the customer approaches the shopkeeper at the service counter, the joint state corresponds to cluster 255. When he reaches the counter and speaks, it is recognized as cluster 26. After this, the robot and customer both move to the Panasonic camera, a joint state represented by cluster 267. When they stop at the camera and the robot talks about its battery life, the state is updated to cluster 197. As this example shows, joint state transitions can be initiated by either the customer or the shopkeeper.

F. Prediction and behavior generation

The time sequences of joint state clusters taken from all of the training interactions are used to train a variable-length Markov model **predictor** to predict the most likely joint state which will follow a given sequence. For example, in Fig. 3, clusters 255 and 26 have been observed in sequence and input to the predictor. The predictor outputs cluster 267 as the most likely joint state to come next.

We then extract the speech and motion elements for one person (in our case, the shopkeeper) from the typical vector of this predicted joint state to generate speech and motion commands for the robot, using the spatial formation to determine proxemic constraints for the robot's positioning. For cluster 267, the behavior generator would command the robot to move to stand near the customer at the Panasonic camera and speak the phrase, "Sure, I can show you the Panasonic Lumix."

III. Environment and Scenario

A. Sensor Environment

To capture people's motion and speech behaviors, we prepared a data collection environment with a sensor network including a human position tracking system and a set of handheld mobile phones to use for speech recognition.

The position tracking system consists of 16 ceiling-mounted Microsoft Kinect RGBD sensors, arranged in rows. Particle filters are used to estimate the position and body orientation of each person in the room based on point cloud data [17].

For speech recognition, we developed a smartphone application which uses the Android speech recognition API to recognize utterances, sending the text to a server via Wi-Fi. The user wears a hands-free headset and touches anywhere on the mobile screen to indicate the beginning and end of their speech, so no visual attention is required, making it possible to conduct natural face-to-face interactions without breaking eye contact.

Although the study was conducted in Japan, we found a greater variety of software tools available for analysis of English text, so the interactions in this study were carried out in English.

B. Training Interactions

To create a set of training interactions, we set up three product displays in a 8m x 11m experiment space, shown in Fig. 4. We chose a shopping scenario in a camera shop setting, so the product displays represented three different digital camera models. We also set up a service counter, where we instructed the shopkeeper to stand at the start of each interaction.

Participants were members of our laboratory. Four participants, including two native English speakers, played the role of shopkeeper. 10 participants, including one native English speaker, played the role of customer. Each customer took part in 10-20 interactions, for a total of 178 trials.

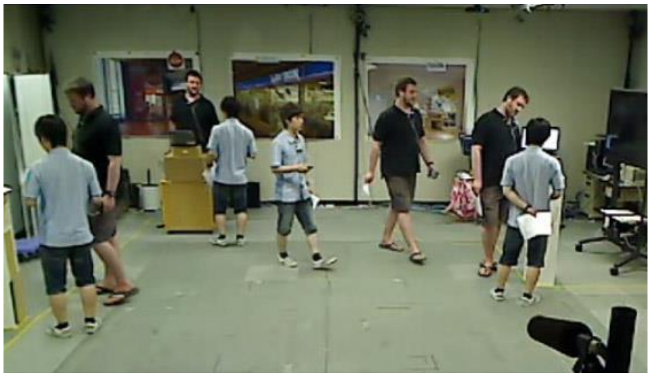


Figure 4. Environment for our data collection.

At the beginning of each interaction, the participants were trained to use the android phone and given a list of camera features to ask about. The shopkeeper was given a reference sheet containing a set of feature specifications for each camera.

In each trial, the customer was instructed to follow one of three scenarios: looking for a specific feature, comparison shopping between two cameras, or just browsing with no interest in the shopkeeper's help. The shopkeeper was not informed of the chosen scenario, and was instructed to allow the customer to browse, to answer any questions the customer had, and to gently introduce products when appropriate.

IV. Data Abstraction

A. Abstracting Motion Elements

In the abstraction of motion elements, our primary objective is to identify the person's *current location* if stopped, or, if moving, their *motion target*.

Using the approach described by Guéguen [18], we segmented all observed trajectories in the training data into "stopped" and "moving" segments. The probability density model from this analysis was saved for use in segmenting live data for the online system.

1) Stopping points

The geometric centers of each the stopped segments were computed and clustered spatially with k-means clustering, to identify typical stopping locations. Six locations were identified for the customer and five for the shopkeeper. The centroid of each cluster was defined as a "stopping point".

As Fig. 5 shows, many stopping points correspond to objects in the room. Each stopping point within 1 m of an object was given its label. The point in the middle of the room was designated "middle," although in more complex environments several unnamed stopping points could exist.

2) Motion Targets

The moving states were then clustered into 30 clusters for each role (shopkeeper, customer) using k-medioid clustering based on spatiotemporal matching using dynamic time warping (DTW). The medioid trajectory for each cluster was used as a reference trajectory, shown in Fig. 6, and each cluster was marked with a motion target, defined as the stopping point closest to the destination point of the reference trajectory, as shown in Fig. 5.

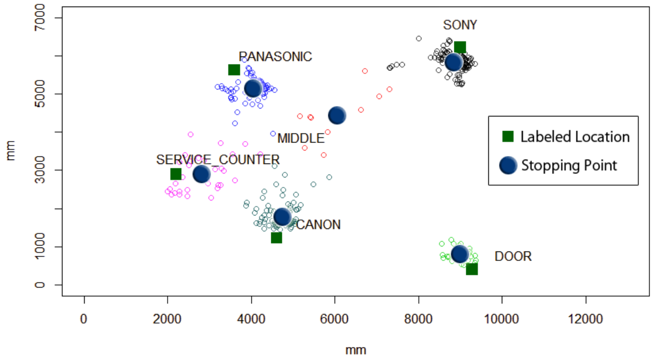


Figure 5. Small circles show the centroids of clusters of stopped trajectories the customer, which are marked as "stopping points" Five of the six correspo to labeled locations in the room.

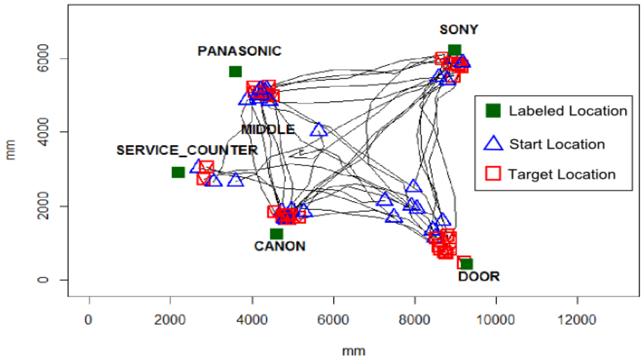


Figure 6. Reference trajectories for the customer, defined as the medioid trajectories from 30 trajectory clusters.

3) *Processing Online Data*

To generate the abstracted features in the online system, live position data is recorded once per second and collected into a buffer of size T_{max}. The trajectory is segmented using the thresholds determined in the offline analysis. When a new segment begins, the buffer is cleared. For stopped segments, the nearest stopping point to the centroid of the buffered points is returned as the *stopping location*. For moving trajectories, we perform a spatiotemporal comparison with each reference trajectory using DTW and return the stopping point nearest to its endpoint of the best match as the *motion target*.

4) *Back-Propagation of Motion Targets*

Estimation of a person's future motion target from sensor data is often unstable, introducing noise to the training data. However, for the shopkeeper, we found a way to remove this uncertainty.

Since training of the behavior predictor is performed offline, we can determine the human shopkeeper's motion target at any time by looking ahead in the training data to observe their actual future destination, rather than relying on our estimation technique from the sensor data.

This technique cannot be used for the customer, since the real-time system can only see the estimated motion target from the sensor data, but a behavior predictor trained with this knowledge can be used online for the robot shopkeeper, because we always know the robot's destination with certainty.

B. *Speech Elements*

1) *Speech Recognition*

Although the Android speech recognition API that we used can perform quite well in some situations, we found recognition accuracy to be poor in our data collection, because people were speaking to each other in a very natural way. An analysis of 400 utterances showed a 53% correct recognition rate. 30% of utterances included minor errors, e.g., "can it should video" rather than "can it shoot video," and 17% were complete nonsense, e.g. "is the lens include North Florida."

2) *Abstracted Representation*

We processed the textual content of the utterances to facilitate behavior clustering. Each utterance was represented by a large feature vector generated through Latent Semantic Analysis (LSA), a technique commonly used for classifying document similarity in text mining applications [19].

The high incidence of speech recognition errors made it difficult to accurately represent semantic similarity between utterances. To handle this noise, we employed a number of common preprocessing techniques. We removed stop words, applied a Porter stemmer [20] to remove conjugations, and enumerated N-grams (up to N=3), in order to capture sequences of words in addition to individual words. We then computed a term-document frequency matrix and pruned terms which occurred fewer than 3 times. We then calculated a term frequency – inverse document frequency (TF-IDF) matrix, and finally performed LSA, computing the singular-value decomposition of the TF-IDF matrix and truncating it to reduce the dimensionality of the space. The optimal dimensionality for the truncated LSA matrix was chosen to achieve a 50% "share" (percentage of cumulated singular values) as described in [21].

3) *Keyword extraction*

All of the procedures described so far are standard techniques used in text mining and merit little discussion here. However, we found that even after this process, we still had a low quality of clustering for speech behaviors. We believe that the nature of the short utterances and fragments in our speech data combined with the extremely high speech recognition noise made it harder to identify important key words.

To help reinforce the important topic words in phrases, we used AlchemyAPI[1], a cloud-based service for text analysis based on deep learning, to perform keyword extraction on the utterances. The collection of keywords returned for each utterance was processed using LSA, and those columns were added to the feature vector. Because many utterances included no keywords, we did not remove the LSA vectors from the original utterances.

C. *Abstracting Social Spatial formations*

The third type of abstraction used was the classification of **spatial formations**. Spatial behavior elements were used to identify two-person spatial formations corresponding to social behavior models which have been studied in HRI, which are useful in planning motion and proxemics. The abstractions used in this study are presented in Table I.

The *present object* spatial formation is based on the work of Yamaoka et al. [16], in which the body positions of two interaction participants discussing an object are modeled

[1] http://www.alchemyapi.com

TABLE I. SPATIAL FORMATIONS

Spatial formation	Condition
Present object	Both people near object Both people facing object or each other
Face to Face	Both people not near object Facing each other
Waiting	One person is at a designated waiting area Both people are not near each other
Approach Person	People are not near each other One person is moving towards the other
Guide Person to Object	Both people are close to each other Both people have the same motion target

based on considerations of field of view and interpersonal distances.

When two people are engaged in a face-to-face conversation and no object is involved, we label it as *face-to-face*, and we chose the robot's motion target position to be 1.5 m from the human, based on Hall's definition of close social distance [22].

Another model we used was *waiting*. Another study [23] has developed a robot waiting behavior based on modeling socially-appropriate waiting locations. In our abstraction, we associate this concept with the behavior of the shopkeeper returning to the service counter to let the customer browse.

We also included two moving formations: *approach person*, for which a model has been proposed by Satake et al. [24], and *guide person to object*, which we associate with the side-by-side walking model proposed by Saiki et al. [25]. However, we have not yet incorporated these two formations into our model.

We created simple rules mapping combinations of motion behavior elements to given formations. For example, when both participants were at a product, we classified the state as *present product*, and when participants were stopped near each other but not at an object, we classified the state as *face-to-face*.

V. BEHAVIOR GENERATION

To generate robot behaviors, we use the most recently observed sequence of joint states as an input to a variable-length Markov model (VLMM) which was trained offline using the training data. The model outputs an estimate of the next appropriate joint state, which is used to generate robot behaviors.

A. Clustering Joint Behavior States

To identify semantically-meaningful behaviors despite the great amount of noise in the sensor data, we used unsupervised clustering to group the observed state vectors (including utterances and spatial behaviors) into clusters representing unique behavior states.

1) Generating Clusters

Clustering of the state vectors was performed using dynamic hierarchical clustering [26]. To handle the high dimensionality of the speech vectors, we separated the state vectors into three groups for clustering: customer speech, shopkeeper speech, and non-speech. The numbers of dimensions, instances, and clusters for each group are presented in Table II.

TABLE II. BEHAVIOR STATE VECTOR CLUSTERING RESULTS

	Dimensions	Instances	Clusters
Customer Speech	328	1328	168
Shopkeeper Speech	437	1439	233
Non-Speech	35	17979	90

2) Extracting Typical Vectors

From each joint state cluster, a representative vector was selected for use in behavior generation. For speech vectors, we found that simply choosing the vector closest to the centroid of the cluster was often problematic. Sometimes this vector was not actually lexically similar to other utterances in the cluster. Instead we attempted to find the utterance with the highest level of lexical similarity to the most other utterances in the cluster.

To compute this, we use the original term-document matrix containing a term frequency vector for each utterance, rather than the reduced matrix computed by LSA. This is because LSA gives a good estimate of *semantic* similarity but not *lexical* similarity. For each utterance, we compute the cosine similarity of its term frequency vector with every other utterance in the same cluster, and we sum these similarity values. The utterance with the highest similarity sum is chosen as the typical utterance.

B. Prediction

To decide the robot's next action, we predict the most socially-appropriate behavior cluster to follow the current state in the social interaction. To capture the timing of behaviors, we separated the prediction process into two parts: predicting the next state, and predicting how long to maintain the current state.

1) Variable-Length Markov Chains

To take into consideration not only the current state but also recent interaction history, the prediction of socially-appropriate clusters is achieved by building a variable-length Markov model (VLMM) which predicts the next cluster c_i in a sequence based on up to a maximum history length n. VLMM's have often been used for prediction and recognition tasks [27] [28].

Our model is built by building N-grams of increasing length, $(\mathrm{n} \in 0 \dots \mathrm{N})$ encapsulating the most recent sequence of unique joint state clusters $\left(\mathrm{c}_{\mathrm{i-1}}, \mathrm{c}_{\mathrm{i-2}}, \dots, \mathrm{c}_{\mathrm{i-n}}\right)$ and computing the probability of the transition $P\left(\mathrm{c}_{\mathrm{i}} | \mathrm{c}_{\mathrm{i-1}}, \mathrm{c}_{\mathrm{i-2}}, \dots, \mathrm{c}_{\mathrm{i-n}}\right)$ as proportional to its relative frequency in the training data.

2) Duration Modeling

We also created a predictor to model the expected duration of each cluster, based on observed durations from the training data.

From the training data, each time a given N-gram $(c_{i-1}, c_{i-2}, \dots, c_{i-n})$ and following behavior c_i was observed, the duration of c_{i-1} (that is, the number of seconds before c_i was executed) was tabulated, and the mean and variance were computed, defining a normal distribution. In the real-time prediction system, once a cluster transition c_i is predicted, the delay before that cluster should be executed is selected randomly from its associated normal distribution.

C. *Generating Robot Actions from Behavior Clusters*

Once the predictor has produced a guess about what a human would have done in the training interactions, the predicted joint state vector is used to generate robot action commands. If the predicted cluster contains a speaking event, a command is sent to speak the utterance contained in the typical vector of the cluster.

If the typical vector of the predicted cluster represents a "moving" state, then the robot is commanded to drive to that location. If this motion is projected to result in a "*present object"* or "*face-to-face*" spatial formation, a target position is computed according to that formation's proxemics model. While it is moving, the robot projects the future position of the customer and computes its ideal motion target according to the proxemics model every second until it arrives.

VI. Example of a Robot Interaction

We tested the proposed behavior generation approach using a Robovie II humanoid robot. An example of a typical interaction with the robot as a shopkeeper is shown in the video attachment, and Table III shows a transcript of that interaction.

This interaction illustrates two points. First, it shows that the system we have developed can generate appropriate social behaviors – the utterances and movements of the robot were natural and appropriate in response to the customer's actions. Second, it shows that this technique can enable a robot to imitate a person's interaction style. In training, the shopkeeper was instructed to seem busy, and to only approach the customer if asked a question or if the customer had been there for a long time. In this interaction the robot was able to reproduce that behavior, waiting for the customer to initiate the interaction.

In our testing, the robot's behavior was not always correct – it sometimes spoke meaningless or incorrect utterances or performed socially-inappropriate behaviors like walking away in the middle of a conversation. However, in our informal testing we observed that the robot's behavior was appropriate in the majority of cases – it usually answered questions correctly and moved to appropriate locations. In future work, we plan to conduct a formal evaluation of its performance.

VII. Discussion and Conclusions

A. *Effectiveness of this approach*

In comparison with more traditional methods, this data-driven approach to learning social interactions by demonstration may appear inefficient for the scenario we have chosen – even after 178 training interactions, our robot still made mistakes in a relatively simple interaction, and social interactions of higher complexity may pose a much greater challenge.

However, we believe the power of this approach lies in the scalability of "big data"– when sufficiently large data sets are collected, performance should improve significantly. By using passive sensing in real environments, we should easily be able to collect many thousands of interactions. With training data on this scale, a fully-automated, data-driven approach to learning social interactions seems much more promising.

TABLE III. Interaction Transcript

WAITING
Customer moves to Canon **Customer** stops at Canon **Customer** moves to Panasonic. **Customer**: Says "Excuse me" at Panasonic
APPROACH PERSON
Robot: Can I help you with anything today?
PRESENT PANASONIC
Customer: Yeah, how much optical zoom does it have? **Robot**: It has 5 times optical zoom **Customer**: Oh, nice. And how is the battery life? **Robot**: This actually a very long battery life of 9 hours **Customer**: Wow that's great, how much does it cost? **Robot**: $300 **Customer**: Hmm, and does this come in any other colors? **Robot**: and we have many colors available: pink, yellow, grey, and black. **Customer:** Thank you very much
WAITING
Robot moves back to Service Counter

B. *Speech recognition and generation*

One merit of our approach is that it performs well despite poor speech recognition. Many of our participants were non-native English speakers, and their accents and speech styles varied a great deal. Since the robot was trained based on these noisy detections, it was able to respond appropriately despite a wide range of accents and grammatical errors. This could be an advantage of our approach over grammar-based speech systems.

Another merit of our approach is the lifelike variation of behaviors learnt by the robot. Explicitly programming multiple phrasings of utterances requires time and effort, but our system implicitly learns to use a variety of synonymous phrases, which can help keep interactions interesting and lifelike.

C. *Future Work*

Many aspects of this system remain to be explored and improved. It would be interesting to observe if the robot can adapt to different interaction styles, such as aggressive vs. soft sales behavior. We would also like to extend this work to include behavior primitives such as gesture and gaze. By doing so, we could incorporate additional HRI models which have been developed to address gaze and pointing gestures [29, 30]. Finally, we would like to investigate techniques such as modeling hidden states or applying other kinds of generalizable models to improve its robustness and flexibility.

D. Conclusion

We have presented a prototype system enabling a social robot to be trained in an interactive task through a fully autonomous procedure based on observation of human-human interactions. We collected 178 trials of humans interacting in a retail camera store scenario. By applying abstraction and clustering techniques to the captured speech and motion data, and using a variable length Markov model predictor, we successfully trained the robot to react with appropriate timing and behaviors in live social interactions.

We believe that with today's trends towards big-data systems and cloud robotics, techniques like this will become important methods for generating robot behaviors in the future.

References

[1] D. F. Glas, S. Satake, T. Kanda, and N. Hagita, "An Interaction Design Framework for Social Robots," in Proceedings of Robotics: Science and Systems, Los Angeles, CA, USA, 2011.

[2] D. F. Glas, K. Wada, M. Shiomi, T. Kanda, H. Ishiguro, and N. Hagita, "Never too old for teleoperation: Helping elderly people control a conversational service robot," in RO-MAN, 2013 IEEE, Gyeongju, Korea, 2013, pp. 703-710.

[3] M. Nicolescu and M. J. Mataric, "Task learning through imitation and human-robot interaction," Models and mechanisms of imitation and social learning in robots, humans and animals: behavioural, social and communicative dimensions, 2005.

[4] S. P. Chatzis and Y. Demiris, "Nonparametric mixtures of Gaussian processes with power-law behavior," Neural Networks and Learning Systems, IEEE Transactions on, vol. 23, pp. 1862-1871, 2012.

[5] M. Ogino, H. Toichi, Y. Yoshikawa, and M. Asada, "Interaction rule learning with a human partner based on an imitation faculty with a simple visuo-motor mapping," Robotics and Autonomous Systems, vol. 54, pp. 414-418, 2006.

[6] B. M. Scassellati, "Foundations for a Theory of Mind for a Humanoid Robot," Massachusetts Institute of Technology, 2001.

[7] Y. Nagai, "Learning to comprehend deictic gestures in robots and human infants," in Robot and Human Interactive Communication, 2005. ROMAN 2005. IEEE International Workshop on, 2005, pp. 217-222.

[8] S. Calinon and A. Billard, "Teaching a humanoid robot to recognize and reproduce social cues," in Robot and Human Interactive Communication, 2006. ROMAN 2006. The 15th IEEE International Symposium on, 2006, pp. 346-351.

[9] K. Lee, Y. Su, T.-K. Kim, and Y. Demiris, "A syntactic approach to robot imitation learning using probabilistic activity grammars," Robotics and Autonomous Systems, vol. 61, pp. 1323-1334, 2013.

[10] W. B. Knox and P. Stone, "Interactively shaping agents via human reinforcement: the TAMER framework," presented at the Proceedings of the fifth international conference on Knowledge capture, Redondo Beach, California, USA, 2009.

[11] S. Calinon and A. Billard, "A framework integrating statistical and social cues to teach a humanoid robot new skills," in Proc. IEEE intl conf. on robotics and automation (ICRA), workshop on social interaction with intelligent indoor robots, 2008.

[12] P. E. Rybski, J. Stolarz, K. Yoon, and M. Veloso, "Using dialog and human observations to dictate tasks to a learning robot assistant," Intelligent Service Robotics, vol. 1, pp. 159-167, 2008.

[13] J. Orkin and D. K. Roy, "Understanding Speech in Interactive Narratives with Crowdsourced Data," in AIIDE, 2012.

[14] C. Breazeal, N. DePalma, J. Orkin, S. Chernova, and M. Jung, "Crowdsourcing Human-Robot Interaction: New Methods and System Evaluation in a Public Environment," Journal of Human-Robot Interaction, vol. 2, pp. 82-111, 2013.

[15] C. Shi, T. Kanda, M. Shimada, F. Yamaoka, H. Ishiguro, and N. Hagita, "Easy development of communicative behaviors in social robots," in Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on, 2010, pp. 5302-5309.

[16] F. Yamaoka, T. Kanda, H. Ishiguro, and N. Hagita, "How close?: model of proximity control for information-presenting robots," in Proceedings of the 3rd ACM/IEEE international conference on Human robot interaction, Amsterdam, The Netherlands, 2008, pp. 137-144.

[17] D. Brscic, T. Kanda, T. Ikeda, and T. Miyashita, "Person Tracking in Large Public Spaces Using 3-D Range Sensors," Human-Machine Systems, IEEE Transactions on, vol. 43, pp. 522-534, 2013.

[18] L. Guéguen, "Segmentation by Maximal Predictive Partitioning According to Composition Biases," in Computational Biology. vol. 2066, O. Gascuel and M.-F. Sagot, Eds., ed: Springer Berlin Heidelberg, 2001, pp. 32-44.

[19] T. K. Landauer, P. W. Foltz, and D. Laham, "An introduction to latent semantic analysis," Discourse processes, vol. 25, pp. 259-284, 1998.

[20] M. F. Porter, "An algorithm for suffix stripping," Program: electronic library and information systems, vol. 14, pp. 130-137, 1980.

[21] F. Wild, C. Stahl, G. Stermsek, and G. Neumann, "Parameters driving effectiveness of automated essay scoring with LSA," presented at the Proceedings of the 9th CAA Conference, Loughborough, UK, 2005.

[22] E. T. Hall, The Hidden Dimension. London, UK: The Bodley Head Ltd, 1966.

[23] T. Kitade, S. Satake, T. Kanda, and M. Imai, "Understanding suitable locations for waiting," in Proceedings of the 8th ACM/IEEE international conference on Human-robot interaction, 2013, pp. 57-64.

[24] S. Satake, T. Kanda, D. F. Glas, M. Imai, H. Ishiguro, and N. Hagita, "A Robot that Approaches Pedestrians," IEEE Trans. Robotics, 2012.

[25] L. Y. M. Saiki, S. Satake, R. Huq, D. Glas, T. Kanda, and N. Hagita, "How do people walk side-by-side?: using a computational model of human behavior for a social robot," in Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction, Boston, Massachusetts, USA, 2012, pp. 301-308.

[26] P. Langfelder, B. Zhang, and S. Horvath, "Defining clusters from a hierarchical cluster tree: the Dynamic Tree Cut package for R," Bioinformatics, vol. 24, pp. 719-720, 2008.

[27] Y.-M. Liang, S.-W. Shih, A. Chun-Chieh Shih, H.-Y. Liao, and C.-C. Lin, "Learning atomic human actions using variable-length Markov models," Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, vol. 39, pp. 268-280, 2009.

[28] A. Sastry, "N-Gram modeling of tabla sequences using variable-length hidden Markov models for improvisation and composition," Ph.D., Center for Music Technology, Georgia Institute of Technology, 2011.

[29] Y. Hato, S. Satake, T. Kanda, M. Imai, and N. Hagita, "Pointing to space: modeling of deictic interaction referring to regions," in Human-Robot Interaction (HRI), 2010 5th ACM/IEEE International Conference on, 2010, pp. 301-308.

[30] P. Liu, D. F. Glas, T. Kanda, H. Ishiguro, and N. Hagita, "It's not polite to point: generating socially-appropriate deictic behaviors towards people," in 8th ACM/IEEE International Conference on Human-Robot Interaction, 2013, pp. 267-274.