Iterative Perceptual Learning for Social Behavior Synthesis

Iwan de Kok, Ronald Poppe, Dirk Heylen

***Abstract*—We introduce Iterative Perceptual Learning (IPL), a novel approach for learning computational models for social behavior synthesis from corpora of human-human interactions. The IPL approach combines perceptual evaluation with iterative model refinement. Human observers rate the appropriateness of synthesized individual behaviors in the context of a conversation. These ratings are in turn used to refine the machine learning models. As the ratings correspond to those moments in the conversation where the production of a specific social behavior is inappropriate, we can regard features extracted at these moments as negative samples for the training of a machine learning classifier. This is an advantage over traditional corpus-based approaches, in which negative samples at extracted at random from moments in the conversation where the specific social behavior does not occur.**

We perform a comparison between the IPL approach and the traditional corpus-based approach on the timing of backchannels for a listener in speaker-listener dialogs. While both models perform similarly in terms of precision and recall scores, the results of the IPL model are rated as more appropriate in the perceptual evaluation. We additionally investigate the effect of the amount of available training data and the variation of training data on the outcome of the models.

***Index Terms*—Social behavior synthesis, Machine learning, Perceptual evaluation, Backchannel**

I. Introduction

In this paper, we address the learning of computation models for the synthesis of human behavior. We target the setting where a human interacts verbally and nonverbally with an intelligent virtual agent (IVA, or virtual human). The aim is to make this human-machine interaction as close as possible to natural human-human interaction. From a machine perspective, this requires that appropriate responsive behavior is displayed to the human (see Figure 1(top)). One common approach to endow IVAs with this ability is to learn conditional responsive behaviors from a corpus of human-human dialogs. The verbal and nonverbal behavior of a dialog partner is continuously encoded as features, e.g. speech activity, gaze direction or body movement. In addition, discrete social behaviors are identified in time. Examples are smiles as a reaction to observed facial movements or backchannels as a reaction to a speaker's speech and gaze. The task of the classifier is to associate (probability) scores for the synthesis of specific behaviors to feature instances of the dialog partner's behavior.

The application of this corpus-based learning approach for human behavior synthesis is widespread, but suffers from two main drawbacks. First, the evaluation of the synthesized behavior is typically measured by comparing it to the actually performed behavior in the corpus. While this is an objective measure, it does not take into account the *optionality* (or individuality) of social behavior. We argue that social behavior performed differently from the dialog partner's in the corpus can also be appropriate. However, objective measures will discredit such altenative behavior which hinders generalization of behavior synthesis models.

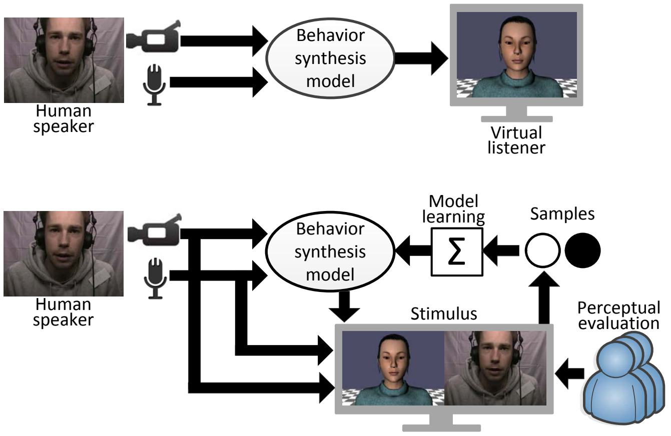

Fig. 1. Schematic overview of social behavior synthesis for an artificial listing agent (top) and the setting of our IPL framework (bottom).

Second, a classifier is trained with feature instances extracted slightly before the occurrence of a social behavior. These instances are considered *positive* samples. Typically, random feature instances that do not overlap with positive samples are used as *negative* samples. However, while a social behavior was not performed in the actual dialog, there is no guarantee that if it had been performed, it would be inappropriate. Consequently, some of the negative samples can also be regarded as positive samples. Again, this hinders the learning of behavior synthesis models.

In this paper, we describe a novel approach that addresses these drawbacks. Instead of relying on objective measures, we obtain subjective ratings regarding the appropriateness of the synthesized behavior. Subsequently, we use these ratings not only to evaluate the quality of the behavior but also to iteratively improve the classifier. We have termed our proposed approach *Iterative Perceptual Learning* (IPL). Figure 1 shows a coarse schematic overview of the IPL approach. IPL is general in the sense that it can be applied to the learning and synthesis of a broad range of social behaviors in dialogs. In addition, the approach is independent of the choice of machine learning classifier and features.

I. de Kok, R. Poppe and D. Heylen are with the Human Media Interaction Group, University of Twente, P.O. Box 217, 7500AE, Enschede, The Netherlands. E-mail: {i.a.dekok,r.w.poppe,d.k.j.heylen}@utwente.nl

The contributions of this paper are:

1) **Perceptual evaluation**: we use subjective, perceptual ratings to measure the appropriateness of individual instances of social behavior. We thus avoid comparing the synthesized behavior with the specific behavior that was performed in the corpus. This allows us to evaluate the quality of the synthesized behavior in a more general sense. In addition, we obtain samples (moments in the dialog) where the production of a specific social behavior is regarded as inappropriate.
2) **Iterative learning**: given the availability of positive and negative samples, we learn a classifier for the synthesis of the timings of social behaviors. By iteratively training and evaluating the resulting synthesized behavior, we refine the performance of the classifier. This approach allows us to focus the negative samples on those feature instances that are relevant.
3) **Experiment on the synthesis of backchannel timings**: we evaluate the merits of the IPL approach for the synthesis of backchannel timings in speaker-listener dialogs. We compare IPL to the common corpus-based approach where negative samples are obtained from the pool of non-positive samples. Our experiment involves several hours of dialog. We analyze the influence of the type of negative samples and the amount of available data on both the objectively and subjectively measured quality of the synthesized listening behavior.

The remainder of this paper is organized as follows. In the next section, we discuss related work on learning social behavior synthesis models. We introduce the IPL approach in Section III. In Sections IV and V, we describe, respectively, the setup and the results of an experiment on the synthesis of backchannel timings. We conclude with Section VI.

II. Related Work

The field of social signal processing [1], [2] addresses computational approaches towards the automatic understanding, modeling and generation of human social behavior in artificial agents and robots. In this work, we focus on the synthesis of nonverbal behavioral cues. Previous work on this topic has addressed, among others, the synthesis of backchannels [3], [4], [5], eye gaze [6], [7], smiles [8], [9] or head gestures during speech [10], [11].

These synthesis models are typically based either on handcrafted rules [7], [12] or on machine learning algorithms [5], [9]. Both give a (probability) score for the production of a social signal, given a feature instance at a selected moment. Due to the real-time nature of interactions, the methods use shallow features in the sense that they are non-semantic and are derived directly from the audio or video signal. While handcrafted rules are usually intuitive and can be based on known patterns in human social behavior, specifying these rules based on shallow features is not trivial. Therefore, recent work has increasingly addressed employing machine learning algorithms to learn behavior synthesis models.

Machine learning models are trained by providing samples to a learning algorithm. For social behavior modeling, positive samples correspond to feature instances extracted at moments in a dialog where the production of a specific behavior is appropriate. The dominant approach to obtain these samples is to record a corpus of human-human interactions in a similar conversational setting and to identify the moments in time where a specific behavior is displayed. In general, the number of such moments is relatively small and there are probably many appropriate moments where no social behavior has been produced. This is due to individual differences in behavior between subjects (e.g. in the amount and timing), which is a consequence of the optional nature of social signals.

Negative samples are usually extracted at random moments within the conversation with the constraint that they should not overlap with positive samples. As a consequence, these negative samples could be extracted at moments in time where the production of a social signal is appropriate, but was not produced in the corpus. The classifier will therefore try to label these positive samples as negative, which is likely to reduce the quality of the classifications. To prevent this form of overfitting and deal with this type of noise, much more data is needed.

Currently, this optional nature of social signals is also not reflected in the evaluation practice of machine learning models that generate their timings. In general, the quality of a behavior synthesis model is evaluated in terms of precision and recall of the generated social behaviors compared to those performed by the actual subject in the corpus. Any deviation from the actually performed behavior results in lower scores. This is an undesired effect as there is no guarantee that the generated listening behavior is also *perceived* as less appropriate.

In sum, one of the key challenges in social behavior synthesis is to obtain appropriate positive and negative samples. This will help in learning behavior synthesis models that are better able to generalize. In addition, it allows for perceptual evaluation of the synthesized social behavior.

Several studies have addressed this challenge. To obtain more samples, De Kok and Heylen [4] recorded three listeners that interacted in parallel with the same speaker. The result of their Paralel Listener Consensus approach is a larger pool of positive samples compared to the setting where only a single listener interacted with the speaker. In addition, by analyzing when multiple listeners produced a social signal, moments in time can be identified where this production is more likely to occur. The method also allows for the investigation of the variation in timing and differences between human observers.

To overcome the complex recording setting of [4], Huang et al. [13] introduced Parasocial Consensus Sampling (PCS). With this method, human observers watch a video of a conversational partner and act as if they were in the conversation. Every time they would produce a social signal, they are to press a button. The approach has been used to collect positive samples of backchannels [13] and speaker turn endings [14]. Despite the fact that the observers are not part of the conversation and pressing a button is artificial, the results of PCS in terms of quantity and timing of social signals was comparable to those produced by the actual subjects in the corpus. For social behavior synthesis, increased generalization was observed when considering as positive samples only the moments in time where the majority of the human observers indicated they would produce a social signal.

Both of the above methods address obtaining more positive samples, which reduces the moments in time where negative samples can be extracted. Still, there is no guarantee that a negative sample corresponds to a moment in time where the production of a social signal is inappropriate. To this end, Poppe et al. [15] had human observers watch a video of a speaker and an animation of a listener side-by-side. The listener was a virtual human that produced specific social signals at predetermined moments in time. Motivated by the observation that humans are sensitive to flaws in animated social behavior, the human observers were instructed to press a button when they judged the produced social behavior as inappropriate. This approach was used as a subjective, perceptual evaluation measure for synthesized social behavior. However, it can also be used to obtain negative samples as we do in this research.

III. Iterative Perceptual Learning

We target a dyadic conversational setting where we aim at generating appropriate social signals for a virtual human in real-time, based on the observed social behavior of a human conversational participant. We consider social signals that (1) are performed as a reaction to the observable behavior of the conversational partner and (2) have an optional nature. We further assume that the observations can be described as feature vectors. This allows us to use machine learning techniques that output a probability or score for the production of a social signal based on a feature vector instance. These assumptions are common for learning social signal models. Examples of this application setting are the animation of head movement as a reaction to the speech of the conversational partner, or backchannels as a reaction to a speaker's speech and gaze (see Section IV).

In this research, we learn social behavior synthesis models in an iterative manner. The basis is a machine learning model which we will treat as a black box. At each iteration, we learn the model given the available positive and negative samples. As we cannot obtain negative samples from the corpus directly, we resort to a generate-and-test approach. We use a virtual, computer-animated, copy of the conversant and animate social signals according to a trained classifier. We then have human subjects rate the (in)appropriateness of the displayed social signals in the context of the conversation. Based on these ratings, we obtain negative samples which are used to train the models in the next iteration. In addition to an increased number of available samples, both positive and negative, we expect that the models are progressively more accurate. The subjective ratings double as perceptual evaluation measures. This allows us to determine, at each iteration, the subjective quality of the generated listening behavior.

A schematic representation of the IPL framework appears in Figure 2. In the following, we discuss the generation, evaluation and learning stages of the framework, respectively. We also address the bootstrapping of the approach. For the sake of simplicity, we consider a dialog with a sender and a receiver. Social behaviors will be synthesized for the receiver, as a response to features extracted from the behavior of the sender.

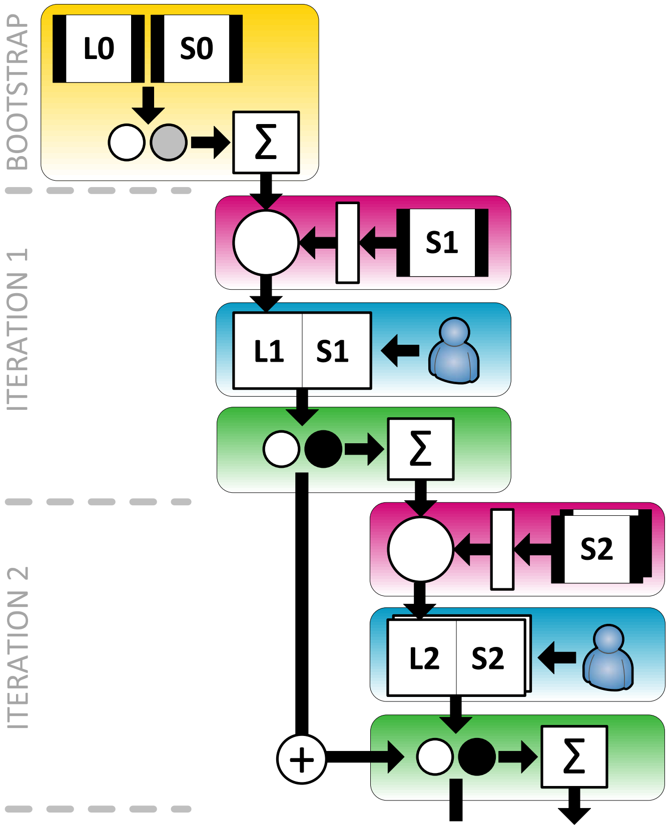

Fig. 2. Schematic representation of the Iterative Perceptual Learning framework. The generation, evaluation and learning stage are shown in pink, blue and green, respectively. Please refer to the text for details. Best viewed in color.

A. Generation

An iteration starts with the generation of the stimuli. Each stimulus is a video of the sender, combined with an animation of the receiver, placed side-by-side. There are three steps involved in the generation stage (see also Figure 2): feature extraction, feature classification and stimulus generation.

The sender is observed, for example using microphone or camera. From these recordings, we calculate feature vectors at each time step. These can be audio features such as pitch and intensity, video features such as amount of movement or head orientation, or any combination of features.

We then classify each feature vector with the model that was trained in the previous iteration (see Section III-C). This results in a numerical output, for example a probability or a score. Given an entire video, we thus obtain a score for each time instant.

The next step is to convert the sequence of scores into a set of social signal timings. To this end, one can apply a threshold, or select the moments corresponding to the top n scores. Additional constraints such as minimum time between two social signal timings, or a minimum or maximum number of social signals per minute can be enforced at this stage as well. Computer animation software is used to generate a virtual copy of the receiver, where social signals are synthesized at the determined timings. Finally, we place this animation of the receiver side-by-side with the video of the sender and make sure both are synchronized in time.

B. Evaluation

In the evaluation stage (blue areas in Figure 2), human subjects rate the (in)appropriateness of the animated social signals. Similar to [16], [15], human raters watch the stimuli and press a button (the *yuck* button) whenever they think an animated social signal of the receiver is inappropriate.

After watching and rating a stimulus, the raters' yucks are matched to the animated social signals, and a typical response delay is taken into account. When several raters watched the same stimuli, their yucks can be aggregated. This results in a percentage of raters that judged a certain social signal instance as (in)appropriate. These numbers can be thresholded to filter out accidental mis-presses, or used directly to determine which social signal samples are to be considered negative ones. The social signal instances that received no or only a few yucks can be regarded as positive samples, in addition to the social signals performed by the human listener in the recorded conversation.

C. Learning

A trained machine learning model is the result after the learning stage (green areas in Figure 2). In this stage, all positive and negative samples are used to train the classifier. As mentioned before, the specifics of the classifier are not important at this point.

In each iteration (except for the first, as we discuss below), the positive and negative samples are added to those of the previous iteration. There is thus an increasing amount of data available for training at each subsequent iteration. As more positive and negative training samples are available, we expect that our classifier will improve. As a result, we will generate social signals at more appropriate moments. Still, some of these instances will be perceived as inappropriate and these end up as negative samples for the next iteration. This approach can therefore be seen as a form of reinforcement learning. It allows us to fine-tune the model by focusing on those feature instances that are relevant.

D. Bootstrap

As we do not have access to negative samples in the first iteration, we bootstrap the process by learning a model with negative samples extracted at random moments where no positive samples occur. This is the exact same approach as is typical for corpus-based learning. After the generation and evaluation phases, we then obtain positive and negative samples, which are then used at each following iteration. The initial samples are discarded.

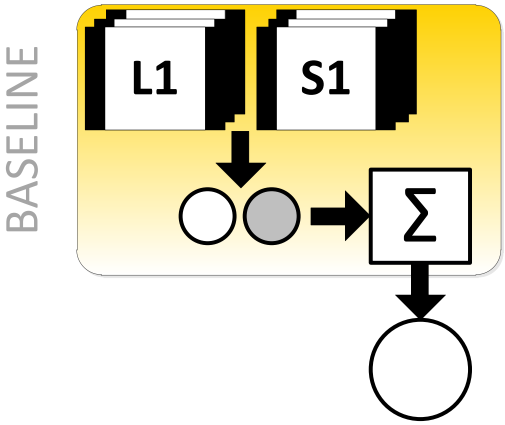

Fig. 3. Schematic representation of the Baseline approach. Please refer to the text for details.

IV. Iterative Perceptual Learning for the Timing of Backchannels

To illustrate the use of the Iterative Perceptual Learning framework for social behavior synthesis we target the scenario of a face-to-face conversation with a speaker and a listener. In this setting, the listener is to signal continued attention, interest and understanding to the speaker, for example with a nod, a short vocalization ("uh-huh") or a smile. These social signals are commonly referred to as backchannels [17], listener responses [18] or continuers [19]. Our aim is to learn computational models to synthesize listening behavior, conditioned on the observed behavior of a human speaker [20]. Specifically, we predict here the timing of backchannels in these speaker-listener dialogs.

We present an experiment in which we learned a backchannel prediction model for the listener using the IPL approach (*IPL*) and compare this to a model learned using the standard corpus-based approach (*baseline*). We evaluate the influence of several factors on both the objectively and perceptual quality of the models.

In the following, we will explain the data on which the models are learned and evaluated. Subsequently, the two models are explained in more detail. Finally, the experimental setup is presented. The results and discussion of the experiment are discussed in Section V.

A. Corpus

We used the Dutch-spoken MultiLis corpus [21] for learning and the evaluation of our models. The corpus consists of mediated human-human interactions between pairs of subjects. In the first interaction, one subject assumed the role of speaker and one subject was assigned the role of listener. In a second interaction, the roles were switched. Figure 4(left) shows a still of a speaker in the conversation. In total, 32 subjects (29 male, 3 female, mean age 25) participated in 32 recordings, with a total duration of 131 minutes.

The speakers were instructed to either summarize a short video they just saw before the interaction or to provide the instructions of a recipe they had just studied for 10 minutes.

Fig. 4. Example stimulus presented to the participants during the evaluation.

Listeners had to remember as many details as possible, because questions were asked afterwards about the video or the recipe. A pair of subjects was assigned either the video task or the recipe task.

Subjects were seated in cubicles and interacted through a remote videoconferencing system. The camera was placed behind an interrogation mirror on which the other subject was projected. This allowed subjects to look directly at the camera and this created the feeling of eye contact. In addition, this setting allowed us to robustly analyze mutual gaze.

B. Feature Preprocessing

From the audio and the video of the MultiLis corpus we extracted three types of features: *prosody* (112 features), *speaking* (1 feature) and *looking* (1 feature). We subsequently explain the procedure for the extraction and processing of each feature type.

From each speaker's audio channel, we extracted prosody features pitch, intensity and the first 12 mel-frequency ceptrum coefficients (MFCC) at a frequency of 100Hz using OpenEAR [22]. Pitch detection is typically noisy and can fail for a few frames during speech. To solve this issue, we linearly interpolated the pitch values for gaps smaller than 8 frames, which is in line with [23]. Between subjects, prosodic signals can vary significantly. For instance, pitch is higher in females than in males, people speak with different volume and/or had the microphone closer to their mouth. We normalized these signals to account for these differences between speakers by converting each signal into the z-score equivalent. The means and standard deviations needed for calculating the z-score were calculated on the first 10 seconds of each session. Therefore, the first 10 seconds of each interaction are excluded from the training data.

As we assume that a classifier is applied to each frame of data independently (see Section IV-D3), we need to capture the temporal aspect to some extent. To this end, we calculated the mean and the slope of each signal over a period of 50ms, 100ms, 200ms and 500ms. The slope was calculated by fitting a first order polynomial (linear regression) to the signal.

The *speaking* feature indicates if and for how long the speaker is talking and is extracted using SHoUT automatic speech recognizer [24]. The *looking* feature indicates if and for how long the speaker is looking at the listener and is based on the manual annotations provided with the MultiLis corpus.

Both signals are initially binary, but we wanted our features to represent sequentiality. To achieve this, for both the speaking and looking features, we calculated the relative offset to the moment where the speaker starts talking or starts looking at the listener, respectively. Specifically, the first frame the speaker is talking will be denoted 1 and this increases by 1 each frame he continues to speak. The first frame the speaker stops talking is -1 and this decreased by 1 each frame until the speaker starts speaking. For looking, a similar processing was applied.

In summary, we extracted 14 prosodic signals, calculated their z-scores and obtained their means and slopes for four different window lengths. This resulted in a total of 112 prosodic features. In addition, we used one speaking feature and one looking feature, which gave a total of 114 features. We concatenated all these features into one 114-dimensional vector per time instance.

C. Baseline Model

In the experiment, the baseline model represents the common corpus-based approach for social behavior synthesis. We will use this model as a means to illustrate the shortcomings of this traditional approach and the benefits of the IPL approach for social behavior synthesis. A schematic representation of the baseline model is shown in Figure 3.

In the traditional approach, feature vectors together with their corresponding ground truth labels are presented to a classifier. Based on the vectors, the classifier learns a model that approximates the ground truth labels. For our experiment, we use a Support Vector Machine (SVM) as a classifier. SVMs are commonly used in (social) signal processing, are well-known to the general public and output a (confidence) score for each input feature vector. We are interested in the relative performance of both approaches and do not focus on obtaining an optimally performing model. Therefore, we used the default settings of the libSVM library [25] without optimization of the parameters involved. These settings are a SVM with the RBF kernel with $c = 1$ and $\gamma = 1/|x|$, where $|x|$ is the dimensionality of the input vector.

The ground truth labels are divided into two classes: positive and negative samples of backchannel opportunities. The positive samples correspond to the first frame of each annotated backchannel in the corpus. The negative samples are randomly selected frames from moments where no backchannel is annotated in the corpus. Note that these negative samples possibly included false negatives, due to the optionality of the social behavior. Typically, there is only a small number of positive samples available in a corpus. To increase the amount of training data and to make the models less dependent on these single frames, we selected four additional frames around the positive frame. We sample these frames from a normalized Gaussian distribution with a σ such that 95% of the samples falls within 250ms of the positive sample. Finally, we made sure that we selected an equal number of negative samples.

In order to obtain backchannel timing predictions, we apply the trained SVM to each input vector, sampled at 100Hz. Instead of the per-frame binary classifications of the SVM, we used the numerical decision values, which can be regarded as confidence scores for the synthesis of a backchannel. By sequencing these decision values over time, we obtained curves representing the appropriateness to provide a backchannel. To remove artifacts due to the potentially highly non-linear output of the SVM, we smoothed these curves with a 10 frame moving average. After this filtering, we consider the highest peaks in this curve to correspond to the most likely moments to predict a backchannel. A threshold can be used to determine at which peaks a backchannel should be synthesized in the listener, similar to [5].

D. Iterative Perceptual Learning Model

The IPL model is learned according to the framework presented in Section III. In the following we will explain the details of the design decisions for each of the steps generation, evaluation and learning.

1) Generation: For each stimulus video, we synthesized for the listener shallow head nods as backchannels at the timings predicted by our trained SVM. For this, we used the Elckerlyc virtual human platform [26]. To control the number of backchannels, we determined the mean backchannel rate over all interactions in the MultiLis corpus, which was approximately 7.7 backchannels per minute. We decided to generate 25% extra backchannels (corresponding to a rate of 9.6 backchannels per minute) with the aim of potentially collecting more negative samples to be used in subsequent iterations. Based on these numbers, we determined the value of the treshold for the peak selection. The only restriction that we applied was that two backchannels could not be within 2 seconds from each other. Stimuli were obtained by putting side-by-side the video of the actual speaker and the animation of the virtual listener.

2) Evaluation: Each stimulus was evaluated perceptually by a number of participants in the experiment. Participants had to press the yuck button whenever they perceived an individual backchannel from the virtual listener as inappropriate. We matched these presses to the last preceeding backchannels if they occurred within 5000 ms of the onset. Finally, we determined for each synthesized backchannel the number of yucks, which we used as a measure of inappropriateness of the backchannel.

3) Learning: We used the exact same machine learning classifier as for the baseline model. The only difference between the two is the way the negative samples were selected. Instead of randomly selecting negative samples from moments where no backchannel was annotated, we used the timings of the generated backchannels which were yucked during the evaluation of the previous iteration as negative samples.

Again, we balanced the number of positive samples and number of negative samples. The number of positive samples was multiplied by five, in line with the baseline model. Next, we calculated the sampling factor for the negative samples. We determined this factor by dividing the increased number of positive samples by the number of individual yucks. The sampling of both the positive and negative samples was performed in the same way as in the baseline model, using a normalized Gaussian distribution. For individual backchannel moments, this meant that more yucks would result in a higher number of negative samples around this moment.

TABLE I
OVERVIEW OF THE SETS USED IN EACH ITERATION AND EACH PHASE OF THE IPL PROCESS.

Phase	Learned on	# Interact.	Evaluated on	# Interact.
Bootstrap	Boot set	1	Set 1	1
Iteration 1	Set 1	1	Set 2	2
Iteration 2	Sets 1, 2	3	Set 3	3
Iteration 3	Sets 1, 2, 3	6	Set 4	6
Iteration 4	Sets 1, 2, 3, 4	12	Test set	6

E. Experiment

We describe the experiment for the prediction of backchannel timings, where we compared a model learned using the Iterative Perceptual Learning framework proposed in this paper (*IPL*) to the common corpus-based model (*baseline*). IPL can be applied iteratively and here we used one bootstrap phase and four iterations. At each iteration of the IPL model, we learned a model using the baseline approach to allow for comparison between the two approaches. After each phase, we evaluated the results of the IPL and baseline models using both objective and subjective measures.

1) Stimuli: Participants of the experiment were shown a video of a speaker from the MultiLis corpus side-by-side with an animated listener, see Figure 4. The virtual listener nodded her head, while making an utterance ("uh-huh") each time the model controlling the virtual listener predicted a backchannel. Other behaviors such as head movement, posture shifts, facial expressions and eye blinks were not animated to prevent these factors to contribute to the perception. As a result, the synthesized listening behavior was rather minimal. For each interaction in a set we created an animation of the virtual listener based on the IPL model and a virtual listener based on the baseline model. The mean duration of a stimulus was approximately 4 minutes.

2) Procedure: The experiment consisted of five phases. We started with a bootstrap phase, followed by four iterations of IPL in which novel stimuli were presented to the participants. In the bootstrap phase, a baseline model was learned on a single interaction. This model was then evaluated perceptually on one other interaction.

For each subsequent iteration, all positive and negative samples obtained from all previous iterations were used to learn the IPL model. Only for the first iteration, the samples used to learn the bootstrap model were discarded. This was because the negative samples were selected at random and were not perceptually rated as inappropriate. In each subsequent iteration, there were more positive and negative samples available to learn the IPL model (see also Figure 2). In addition, we evaluated the model on a larger set of stimuli. An overview of the number of stimuli used to learn and evaluate the IPL models in each iteration is given in Table I.

In the fourth and final iteration, we learn the IPL model based on 12 interactions, and test it's performance on a test set of six interactions. This test set is never used for training.

To compare the performance of the IPL model with that of the baseline approach, we also perceptually evaluated the performance of the baseline approach after each iteration. We learned models on the same interactions according to Table I, but with negative samples selected randomly without overlap with positive samples, as explained before. As both models, after each iteration, were trained on the same interactions, we can make a fair comparison. To this end, we perceptually evaluate the resulting IPL and baseline models after each iteration on the test set. The evaluation results for the IPL model double as negative samples for model learning in the subsequent iteration.

To evaluate the quality of the models, participants of the experiment were shown stimuli through a webpage. It was explained to them that they would be participating in an experiment to determine the quality of synthesized listening behavior. After entering their name, gender and age, the participants were presented a set of (at most) 6 stimuli. They were asked to press the spacebar each time the virtual listener performed a backchannel they judged as inappropriate. Participants could replay the stimulus from the start, which would discard all previously issued yucks for that stimulus. Each participant was shown the same interaction twice in succession, once with the virtual listener based on IPL, once based on the baseline model. The order of the models was varied systematically. This design choice allowed us to evaluate the difference between the two models pair-wise. This is essential as there are typically differences in the amount of yucks between participants. In total, the experiment lasts around 30 minutes.

3) Participants: Each stimulus was rated by 5 participants. As set 4 and the test set contained six stimuli, we decided to split these sets into two. Including the evaluation on the test set for iterations 1 - 4, this gives us 13 groups of stimuli. Consequently, we required 65 participants to rate the stimuli, 25 for the evaluation of sets 1 - 4 and 40 for the evaluation of the test sets. Participants were requited among colleagues and students. Several persons participated more than once. As we perform a pair-wise comparison of the two models per iteration, this does not bias the results. Of the 65 trials, 8 and 57 were completed by females and males, respectively (mean age of 28, min. 18, max. 47).

4) Evaluation Measures: In the experiment, we used two performance measures: one objective measure and one novel subjective measure. For the objective measure, we compared the predicted timing of the backchannels with those performed by the actual listener in the MultiLis corpus, as is common for corpus-based learning. We calculated the precision and recall and combined these by taking the weighted harmonic mean of the two into the F_1 measure: $F_1 = \frac{2\ p\ r}{p+r}$, with p and r the precision and recall scores, respectively.

For the subjective measure, we used the yucks collected in the perceptual evaluation. We calculated the percentage of backchannels that did not receive any yucks. In addition, we calculated the average number of yucks per backchannel.

V. Results and Discussion

First, we analyzed the performance of both models on the test set after the fourth and final iteration. On the objective measure, both approaches performed the same with F$_1$ scores of 0.323, see Figure 5 (left). However, the subjective measures show a slightly different effect. In total, 239 backchannels are generated with each of the models. The number of yucks obtained from five participants per stimulus is lower for IPL than for the baseline (219 and 238, respectively). On average, a backchannel synthesized with the IPL model received 0.92 yucks from all participants, whereas this number was 1.0 for a backchannel generated from the baseline model. A breakdown of the number of yucks per backchannel is given in Figure 6.

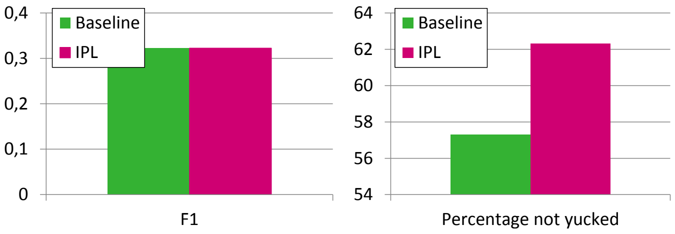

Fig. 5. Performances of the two models we trained on F$_1$ measure and percentage of generated head nods that are yucked by at least 1 subject.

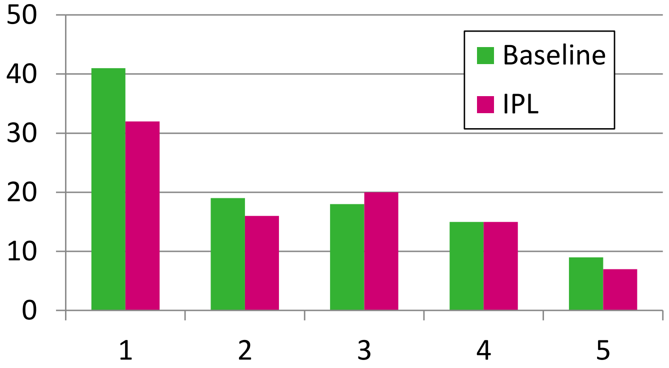

Fig. 6. Frequency histogram for number of yucks per synthesized backchannel on the test set, for baseline and IPL models after iteration 4.

When we looked at the number of backchannels that did not receive any yucks, these numbers are even more in favor of the IPL model. For IPL, 149 (62.3%) did not receive any yucks, compared to only 137 (57.3%) for the baseline model.

This means that both models generate behavior which replicates the original listener similarly in terms of co-occurring backchannels, but that the behavior generated based on the IPL model is perceived as more natural. In the following, we look at the amount of available data on the subjective and objective performance, and at the variation of the performance on different sets.

A. *Effect of Amount of Data*

Typically, as more data comes available for training, one would expect that the performance of the resulting learned model improves. This is due to the fact that models typically generalize better when they have been trained on a wider variety of positive and negative samples.

We have learned models for both IPL and the baseline approach after each iteration, which we tested on the test set. The results of the evaluation are shown in Figure 7 for the percentage of backchannels that did not receive any yucks. From this figure, a couple of observations can be

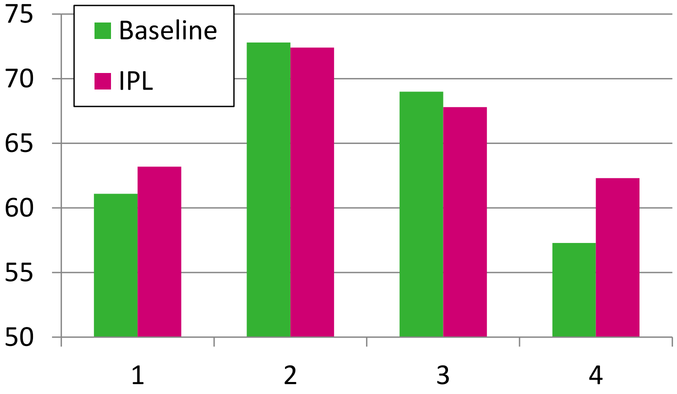

Fig. 7. Percentage of synthesized backchannels that did not receive any yucks on the test set, for the baseline and IPL models after each iteration.

made. First, the performance is not monotonically increasing for an increasing number of available training interactions. Even though these numbers are calculated on the same set of interactions, they are not completely comparable as they are obtained from different participants. More evaluations, or a within-subject design for iterations, are needed to make a more definitive conclusion whether there is a trend or not. Still, as model (IPL or baseline) was a within-subject factor, we can compare the results pair-wise. It becomes clear that the IPL model learned after the first and fourth iterations is better than the baseline approach.

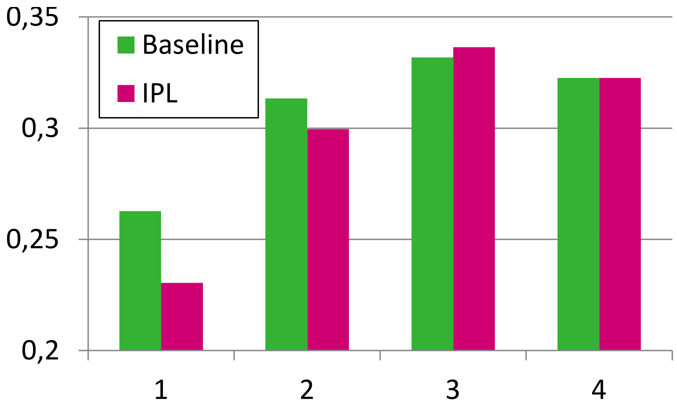

Fig. 8. F_1 measure on the test set, for the baseline and IPL models after each iteration.

To gain more insight into this observation, we turn to the objective F_1 measure. In Figure 8, a positive trend can be observed for the amount of training data on the F_1 measure. However, in the final iteration, the scores are lower for both models, which is again unexpected. Apart from the first iteration, differences between the two models are relatively small. Surprisingly, after the first iteration, the baseline approach outperforms the IPL approach on the objective measure, whereas this effect is reversed for the subjective measure.

In summary, there is no evidence from our data that an increasing amount of training data will lead to better models. We might attribute this finding to two causes. First, the features that we use do not contain sufficiently detailed information to clearly differentiate between appropriate and inappropriate moments to produce a backchannel. In our experiment, it might cause the learning of the models to saturate quickly. In this case, more meaningful features related to what has been said or the inclusion of features that have a known relation to turn-taking and backchanneling (e.g. mutual gaze, head orientation, smiling) might give better results.

TABLE II
THE F_1 MEASURES OBTAINED FOR IPL / BASELINE MODELS AND EVALUATED ON SUBSEQUENT (TEST) SETS.

	IPL / baseline evaluated on			
Model	**Set 2**	**Set 3**	**Set 4**	**Test**
Iter. 1	0.165 / 0.154	0.206 / 0.222	0.152 / 0.216	0.230 / 0.263
Iter. 2	-	0.222 / 0.222	0.158 / 0.191	0.300 / 0.313
Iter. 3	-	-	0.189 / 0.222	0.336 / 0.332
Iter. 4	-	-	-	0.323 / 0.323

A second explanation is the variation among the different training sets. There is typically a substantial variation in the amount and timing of backchannels between listeners [21]. It might be that certain patterns in backchannel placement are learned from one set that do not occur in another set, and vice versa. To investigate whether there are such differences between the sets, we conduct additional analyses in the next section.

B. *Effect of Variation in Training Set*

To gain more insight into the variation in backchannel placement between training sets, we evaluate models trained after a certain iteration on all training sets that are to be used in subsequent iterations. These tests are explicitly not part of the common IPL or baseline procedure. We calculated the F_1 measures for all combinations as we did not have subjective evaluations for the output of the models. Results are summarized in Table II.

It immediately becomes clear that all models perform rather poorly on set 4. Potentially, the backchannel behavior of the listener or the backchannel inviting behavior of the speaker in this set are different from that in other sets. We expect that this poor performance is the reason why the performance drops in the following iteration. This can be explained as follows. Both IPL and the baseline approach aim at learning a generalized model for predicting backchannel opportunities, applicable to every speaker and every listener. But individuals differ in their interaction styles (ways to deliver information, construct sentences and produce them), and the models are not capable of attuning to each individual. During training, they converge to the behavior of the average speaker paired with the average listener. Apparently, the models are better at generalizing to the behavior of the interactions in the test set using the interactions used so far than including the interactions from set 4. The interactions in set 4 might deviate more from the average behavior in the test set, and the models might be attuned to behavior not present in the test set. It is an interesting and needed avenue for future research to develop models that can adapt to different interaction styles.

VI. Conclusion and Future Work

We introduced Iterative Perceptual Learning (IPL), a novel approach for learning computational models for social behavior synthesis. The approach combines two innovative components to deal with the optionality of social behavior and individual differences in their production in interactions.

First, IPL uses subjective, perceptual evaluation measures instead of the common corpus-based metric such as precision and recall. Human observers rate the quality of stimuli of synthesized behavior, based on the output of trained models. These ratings are given at the level of individual synthesized behaviors. Specifically, observers press a button to indicate that the behavior is inappropriate in the context of the conversation. By analyzing the ratings of several observers, we can measure the appropriateness of individual behavior instances.

Second, the behavior synthesis model of IPL is refined iteratively using the perceptual ratings. Instead of a random selection of samples where no positive sample is recorded, we use these ratings as negative samples of social behavior in the model learning phase. This creates a more reliable ground truth. By iteratively applying this technique, we are able to tune our models to social behavior that is rated as appropriate.

We have demonstrated these innovations in a case study on the timing of backchannels in speaker-listener dialogs. We compared IPL to the traditional corpus-based approach. While both models performed similarly in terms of precision and recall, the results of the IPL model were rated as more appropriate in the perceptual evaluation. Differences between IPL and baseline model were small and varied between sets of stimuli. We expect the features did not contain sufficiently detailed information to clearly differentiate between appropriate and inappropriate moments to produce a backchannel. This might have caused the learning of the models to saturate quickly.

There are several ways in which we intend to improve our work. First, several design choices in the experiment were suboptimal from a performance point of view. The SVM model, not being a sequential model, might not have been the most suitable machine learning technique. Furthermore, we consider the use of semantic and lexical features for the prediction of backchannel opportunities [27]. Finally, future work should address the use of adaptive models to attune to different interaction styles.

References

[1] M. Pantic, R. Cowie, F. D'Errico, D. Heylen, M. Mehu, C. Pelachaud, I. Poggi, M. Schröder, and A. Vinciarelli, "Social signal processing: The research agenda," in *Visual Analysis of Humans - Looking at People*, T. B. Moeslund, A. Hilton, V. Krüger, and L. Sigal, Eds., 2011, ch. 26, pp. 511–538.

[2] A. Vinciarelli, M. Pantic, D. Heylen, C. Pelachaud, I. Poggi, F. D'Errico, and M. Schröder, "Bridging the gap between social animal and unsocial machine: A survey of social signal processing," *IEEE Transactions on Affective Computing*, to appear.

[3] R. M. Maatman, J. Gratch, and S. Marsella, "Natural behavior of a listening agent," in *Proceedings of Intelligent Virtual Agents (IVA)*, 2005, pp. 25–36.

[4] I. de Kok, D. Ozkan, D. Heylen, and L.-P. Morency, "Learning and evaluating response prediction models using parallel listener consensus," in *Proceeding of International Conference on Multimodal Interfaces and the Workshop on Machine Learning for Multimodal Interaction (ICMI-MLMI)*, 2010, p. 3.

[5] L.-P. Morency, I. de Kok, and J. Gratch, "A probabilistic multimodal approach for predicting listener backchannels," *Journal of Autonomous Agents and Multi-Agent Systems*, vol. 20, no. 1, pp. 70–84, 2011.

[6] A. Fukayama, T. Ohno, N. Mukawa, M. Sawaki, and N. Hagita, "Messages embedded in gaze of interface agents - impression management with agent's gaze," in *Proceedings of the conference on Human factors in computing systems (CHI)*, 2002, pp. 41–48.

[7] C. Peters, C. Pelachaud, E. Bevacqua, M. Mancini, and I. Poggi, "A model of attention and interest using gaze behavior," in *Proceedings of Intelligent Virtual Agents (IVA)*, 2005, pp. 229–240.

[8] E. Bevacqua, S. Hyniewska, and C. Pelachaud, "Positive influence of smile backchannels in ECAs," in *Proceedings of the Workshop on Interacting with ECAs as Virtual Characters at the International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, 2010.

[9] I. de Kok and D. Heylen, "When do we smile? Analysis and modeling of the nonverbal context of listener smiles in conversation," in *Affective Computing and Intelligent Interaction (ACII)*, 2011, pp. 477–486.

[10] J. Lee, A. Neviarouskaya, H. Prendinger, and S. Marsella, "Learning models of speaker head nods with affective information," in *Proceedings of Intelligent Virtual Agents (IVA)*, 2009, pp. 1–6.

[11] G. Hofer and H. Shimodaira, "Automatic head motion prediction from speech data," in *Proceedings of Interspeech*, 2007, pp. 722–725.

[12] R. Poppe, K. P. Truong, D. Reidsma, and D. Heylen, "Backchannel strategies for artificial listeners," in *Proceedings of Intelligent Virtual Agents (IVA)*, 2010, pp. 146–158.

[13] L. Huang, L.-P. Morency, and J. Gratch, "Parasocial consensus sampling: Combining multiple perspectives to learn virtual human behavior," in *Proceedings of Autonomous Agents and Multi-Agent Systems (AAMAS)*, 2010, pp. 1265–1272.

[14] ——, "A multimodal end-of-turn prediction model: learning from parasocial consensus sampling," in *Proceedings of Autonomous Agents and Multi-Agent Systems (AAMAS)*, 2011, pp. 1289–1290.

[15] R. Poppe, K. P. Truong, and D. Heylen, "Backchannels: Quantity, type and timing matters," in *Proceedings of Intelligent Virtual Agents (IVA)*, 2011, pp. 228–239.

[16] I. de Kok and D. Heylen, "Appropriate and inappropriate timing of listener responses from multiple perspectives," in *Proceedings of Intelligent Virtual Agents (IVA)*, 2011, pp. 248–254.

[17] V. H. Yngve, "On getting a word in edgewise," in *Sixth Regional Meeting of the Chicago Linguistic Society*, vol. 6, 1970, pp. 657–677.

[18] D. Xudong, *The Pragmatics of Interaction*. John Benjamins Publishing, 2009, ch. Listener response, pp. 104–124.

[19] E. A. Schegloff, "Discourse as an interactional achievement: Some uses of 'uh huh' and other things that come between sentences," in *Analyzing discourse: Text and Talk*, D. Tannen, Ed., 1982, pp. 71–93.

[20] D. Heylen, E. Bevacqua, C. Pelachaud, I. Poggi, J. Gratch, and M. Schröder, "Generating listening behaviour," in *Emotion-Oriented Systems*, R. Cowie, C. Pelachaud, and P. Petta, Eds., 2011, pp. 321–347.

[21] I. de Kok and D. Heylen, "The MultiLis corpus - dealing with individual differences of nonverbal listening behavior," in *Toward Autonomous, Adaptive, and Context-Aware Multimodal Interfaces: Theoretical and Practical Issues*, A. Esposito, A. Esposito, R. Martone, V. C. Müller, and G. Scarpetta, Eds., 2011, pp. 374–387.

[22] F. Eyben, M. Wöllmer, and B. Schuller, "openEAR - introducing the munich open-source emotion and affect recognition toolkit," in *Affective Computing and Intelligent Interaction (ACII)*, 2009, pp. 576–581.

[23] N. Ward and W. Tsukahara, "Prosodic features which cue back-channel responses in English and Japanese," *Journal of Pragmatics*, vol. 32, no. 8, pp. 1177–1207, 2000.

[24] M. Huijbregts, "Segmentation, diarization and speech transcription: Surprise data unraveled," PhD Thesis, University of Twente, 2008.

[25] C.-C. Chang and C.-J. Lin, "LIBSVM: a library for support vector machines," *ACM Transactions on Intelligent System and Technology*, vol. 2, no. 3, pp. 1–27, 2011.

[26] H. van Welbergen, D. Reidsma, Z. M. Ruttkay, and J. Zwiers, "Elckerlyc - a BML realizer for continuous, multimodal interaction with a virtual human," *Journal on Multimodal User Interfaces*, vol. 3, no. 4, pp. 271–284, 2010.

[27] I. de Kok and D. Heylen, "Observations on listener responses from multiple perspectives," in *Proceedings of the Nordic Symposium on Multimodal Communication*, 2011, pp. 48–55.