Towards a General, Continuous Model of Turn-taking in Spoken Dialogue using LSTM Recurrent Neural Networks

Gabriel Skantze
Department of Speech Music and Hearing, KTH
Stockholm, Sweden
skantze@kth.se

Abstract

Previous models of turn-taking have mostly been trained for specific turn-taking decisions, such as discriminating between turn shifts and turn retention in pauses. In this paper, we present a predictive, continuous model of turn-taking using Long Short-Term Memory (LSTM) Recurrent Neural Networks (RNN). The model is trained on human-human dialogue data to predict upcoming speech activity in a future time window. We show how this general model can be applied to two different tasks that it was not specifically trained for. First, to predict whether a turn-shift will occur or not in pauses, where the model achieves a better performance than human observers, and better than results achieved with more traditional models. Second, to make a prediction at speech onset whether the utterance will be a short backchannel or a longer utterance. Finally, we show how the hidden layer in the network can be used as a feature vector for turn-taking decisions in a human-robot interaction scenario.

1 Introduction

One of the most fundamental aspects of dialogue is the organization of speaking between the participants. Since it is difficult to speak and listen at the same time, the interlocutors need to take turns speaking, and this turn-taking has to be coordinated somehow. This poses a challenge for spoken dialogue systems, where the system needs to coordinate its speaking with the user to avoid interruptions and (inappropriate) gaps and overlaps.

For a full account of turn-taking, there are many different aspects that need to be modelled. For example, the system should be able to detect whether the user is likely to continue speaking after a brief silence, or whether the system should respond (Meena et al., 2014; Ferrer et al., 2002). Another related issue is to detect places where it is appropriate to give brief feedback (so-called backchannels) while the user is speaking (Morency et al., 2008). If the user starts speaking, it is also important to estimate whether the user is most likely initiating a longer utterance, or a shorter listener response (Neiberg and Truong, 2011; Selfridge et al., 2013). When the system is speaking, it is important to assess whether the user will interpret pauses in the system's speech as turn-yielding (an opportunity to take the turn) or not, depending on how the system's utterance is synthesized (Hjalmarsson, 2011). So far, these different problems have mostly been addressed as separate issues, using different models.

In this paper, we present a general, continuous model of turn-taking, trained on dialogue data. The model is *general*, in that we do not train it for specific turn-taking decisions, but instead train it to forecast the probability that the speakers will continue speaking over a future time window. The model is *continuous*, in that it does this at every time step, and not at certain events (such as when someone stopped speaking). We argue that this predictive model is potentially useful for a number of different types of predictions and decisions that are relevant for spoken dialogue systems.

A similar approach was taken by Ward et al. (2010). However, their experiments only yielded modest improvements over the baseline. An explanation for this might be that turn-taking is a highly context-dependent phenomenon, and that representation of dialogue context is a challenging task, typically involving a lot of heuristics and feature engineering. To address this problem, and make as few assumptions as possible, we train the model using Long Short-Term Memory (LSTM) Recurrent Neural Networks (RNN), where the context-modelling is left to the net-

work, and we feed it with fairly basic features representing cues known to be relevant for turn-taking.

The paper is organized as follows. We start with a review of related work on the problem of turn-taking in dialogue, and give a brief overview of RNNs. We then describe the proposed model in more detail, how it was applied in this study, and how features were extracted. Using the HCRC Map Task Corpus (Anderson et al., 1991), we then present two experiments on turn-taking predictions, both at pauses and at speech onset. Finally we investigate how the model can be applied to make predictions on human-computer dialogue data.

2 Background

2.1 Turn-taking in Spoken Dialogue

Traditionally, spoken dialogue systems have rested on a very simplistic model of turn-taking, where a certain amount of silence (e.g., 700ms) is used as an indicator that the user has stopped speaking, and that the turn is yielded to the system. One obvious problem with this model is that turn-shifts often are supposed to be much more rapid than this, with very short gaps, and that pauses within a turn often might be longer. Thus, the system will sometimes appear to give sluggish responses, and sometimes interrupt the user. Several studies have shown that humans coordinate their turn-taking using much more sophisticated cues. For example, an incomplete syntactic clause or a filled pause (such as "uhm") typically indicates that the speaker is not yielding the turn (Clark and Fox Tree, 2002), and turn-taking is related to information density in the words spoken (Dethlefs et al., 2016). Prosodically, a rising or falling pitch at the end of a segment tend to be turn-yielding, whereas a flat pitch is turn-holding (Edlund and Heldner, 2005). The intensity of the voice tends be lower when yielding the turn, and the duration of the last phoneme tends to be shorter. Gaze has also been found to be an important cue – speakers tend to not look at the addressee during an utterance, but then shift the gaze towards the addressee when yielding the turn (Kendon, 1967). Studies have also shown that the more turn-yielding cues are presented together, the more likely it is that the other speaker will take the turn (Gravano and Hirschberg, 2011; Koiso et al., 1998; Duncan and Niederehe, 1974).

Several models have been presented for taking these different cues into account and to predict turn-taking events. A common approach is to segment the speech into so-called Inter-Pausal Units (IPU), which is a stretch of audio from one speaker without any silence exceeding a certain amount (such as 200ms). Given the end of an IPU, the model has to predict whether the speaker is making a pause and "holding" the turn, or whether the speaker is yielding the turn. Various feature sets and machine learning algorithms have been proposed, and tested on both human-human and human-machine dialogue data (Meena et al., 2014; Schlangen, 2006; Neiberg and Gustafson, 2011; Johansson and Skantze, 2015; Ferrer et al., 2002; Kawahara et al., 2012).

These kinds of models assume that turn-taking only occurs when a speaker has stopped speaking. However, in studies of human-human dialogue it is clear that overlaps are fairly frequent (Heldner and Edlund, 2010). A common phenomenon, that often leads to overlapping speech, is *backchannels* – short utterances (such as "mhm" or "yeah"), which the listener provides to show continued attention (Yngve, 1970). Models have been proposed to continuously detect where in the speech these are suitable (Morency et al., 2008). Given that a listener starts to speak, the current speaker must also detect whether the listener is simply providing a backchannel (so that the speaker may continue), or is intending to claim the floor to produce a longer response (Neiberg and Truong, 2011).

Another limitation of IPU-based models of turn-taking is that they are purely reactive. Several studies have shown that humans are able to predict upcoming turn-taking events (Tice and Henetz, 2011), and that this prediction facilitates rapid and accurate turn-taking (Ruiter et al., 2006). To implement this behaviour in spoken dialogue systems, it is important that they can process speech incrementally (Skantze and Schlangen, 2009), and not wait until the user is done speaking. The model proposed in this paper is based on an incremental and predictive notion of turn-taking, where the model continuously monitor the speech from the two interlocutors and makes predictions about future turn-taking events.

2.2 Modelling Context with Recurrent Neural Networks

Most attempts at creating computational models of turn-taking have only considered a brief window before the turn-taking decision is being made. Also, any dynamic events (such as a raise in pitch) in this window need to be transformed

into a single feature vector using heuristics and careful feature engineering. This is an obvious drawback, since turn-taking is likely to be dependent on various contextual properties, such as previous speaking activity. To address this problem, we propose to use Recurrent Neural Networks (RNNs), which are especially designed to learn representations of context from low-level features. Whereas a typical feedforward neural network only transforms a single feature vector into an output vector (possibly through a number of hidden layers), RNNs are neural networks with loops that allow information to persist from one step in time to the next, as illustrated in Figure 1. During training and backpropagation, the updates are fed back in time, in order to adjust the weights at previous time steps, and thereby potentially learn long-term dependencies.

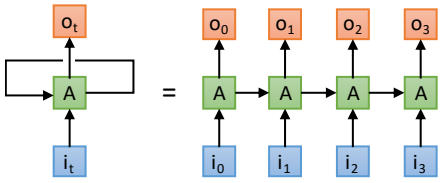


Figure 1. The principle behind RNNs with an unrolled view to the right. The neural network, *A*, looks at the input i_t at time *t* and outputs a value o_t. The loop allows the network to remember the state at time *t-1*.

A limitation of traditional RNNs is their inability to learn dependencies over longer time sequences. The reason for this is that the update gradients become too small over longer distances. This can be especially problematic for the continuous model proposed here, since important events may occur many frames before the turn-taking prediction is being made. To address this problem, it is common to use an extension called Long Short-Term Memory (LSTM), which have a cell state and a gating mechanism that allow information to pass longer paths in the network history, thereby avoiding the vanishing gradient problem (Hochreiter and Schmidhuber, 1997). LSTM has been successfully applied to a number of tasks related to speech and language processing, such as voice activity detection (Eyben et al., 2014), speech recognition (Graves et al., 2013), and spoken language understanding (Liu and Lane, 2016). To our knowledge, this is the first attempt at using LSTM RNNs for a continuous model of turn-taking.

3 Model and Data

3.1 The Model

The general principles for the model are illustrated in Figure 2. An RNN is trained to make continuous predictions about the speech activity for one of the speakers (speaker S_0) for an upcoming fixed time window, based on previous events in both speaker channels. The speech signals for the two speakers (S_0 and S_1) are segmented into equally sized frames (or time steps). For each frame, features from both speakers are extracted and fed into an RNN with one LSTM layer. For each frame, the RNN outputs an *N*-dimensional vector with predictions of the probability that S_0 will speak or not for the next *N* frames. For the experiments in this paper, we use a frame size of 50ms (20 frames per second), and a prediction window of 3 seconds (60 frames).

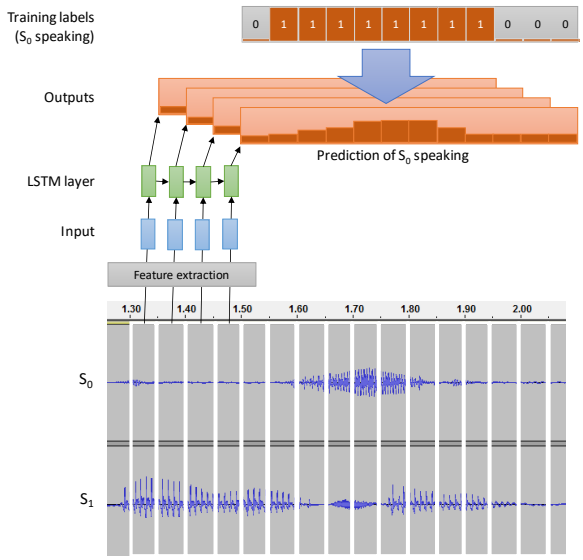


Figure 2. How the model makes predictions and is trained, with an unrolled view of the RNN. For each frame (50ms), the network predicts the probability of speaker S_0 speaking over the next *N* frames (with one output node per frame).

To train the model, we use human-human dialogue data, with the voice activity of speaker S_0 for the next *N* frames as target labels. Although these labels are binary, the output nodes will be trained to provide a probabilistic score (between 0 and 1). To allow the model to train to make predictions for both speakers, the same network is trained on each dialogue twice, with each speaker serving as both speaker S_0 and S_1.

When applying the model, two network instances are used, one in which speaker A serves as S_0 (to get predictions for speaker A), and one where speaker B serves as S_0 (to get predictions

for speaker B), with the speaker features switched between the two networks. Some examples of what the predictions can look like are shown in the Appendix[1]. Note that although we will here assume two speakers, the model is not limited to dyadic interaction. In principle, it could be applied to dialogues with any number of speakers, where each speaker is modelled with its own network at application time.

The model should also be applicable for making decisions in dialogue systems. By feeding the two networks (as described above) in real time with both the user's speech and the system's own speech, the user's network will make predictions of how likely it is that the user will speak in the near future. But the system's network will also predict how likely it is that the system *should* be speaking in the future time window, given the assumption that a human-like behaviour is desired. The output of the two models could then be combined to make decisions of whether the system should speak or not. In the simplest case, the two predictions can be compared, and if the system's network has a stronger prediction than the user's network, it would constitute a good place to take the turn. Since the model is probabilistic, a more sophisticated decision theoretic approach could take the probabilities of the predictions, together with a utility, into account. For example, it could still be desirable for the system to take the turn even if it is an unlikely place to do so, given that the system has something important to say. Since the probabilities are updated continuously, even during silences, the model could naturally generate variable gap lengths in the system's response.

Another potential application of the model would be for the generation of system responses. Given different prosodic and syntactic realisations of a response, the model could predict whether the user is likely to take the turn, for example in pauses. To select a response which signals the intended turn-taking cues, the system could feed different candidate responses into the networks and predict how the user would react to them. Yet another application would be to enhance Voice Activity Detection (VAD) with the probability that the user will be speaking, given the dialogue context.

In this paper, we will mainly evaluate the model on its predictive power when observing human-human interaction. However, we will also investigate whether it could be used for turn-taking decisions in a spoken dialogue system, according to the simple method outlined above.

3.2 Data

To train and evaluate the model, we have used the HCRC Map Task corpus (Anderson et al., 1991). This corpus consists of 128 dialogues, where one speaker (the information giver) is explaining a route on a map to another speaker (the follower), using landmarks on the map. The gender of the speakers is balanced, in some dialogue with mixed gender and in other dialogues with same gender. In half of the sessions, the speakers knew each other, in the other half they didn't. Another variable was whether they could see each other (face-to-face) or not.

For our experiments, the data set was split into one training set with 96 dialogues, and one test set with 32 dialogues. Care was taken to balance the variables described above across training set and test set. The average dialogue length was 6.7 minutes, giving 10.7 hours of training data and 3.6 hours of test data. Since the frame rate was 20 frames per second and the model was trained for both speakers, the RNN was trained on about 1 540 800 frames.

3.3 Feature extraction

Features were chosen based on the findings in related literature. For each frame (spanning 50ms), we produce a feature vector as input for the network. We only use momentary features (e.g., the current pitch level), and do not encode delta (such as a rising pitch) or context (e.g., for how long someone has been speaking), with the assumption that these derivations in the time-domain will be learned by the RNN.

Voice activity: A binary feature representing the current voice activity (speech/no speech) of the two speakers. The voice activity was extracted from the manual annotation of the corpus. These features are also used for the target labels during training (the projection of voice activity for the next 3 seconds), as can be seen in Figure 2.

Pitch: The pitch was automatically extracted using the Snack toolkit (Sjölander and Beskow, 2000), transformed into semitones, and then z-normalized for the individual speaker. Both the relative and absolute values were used as individual features. In addition, a binary feature indicating whether the current frame was voiced or not was included.

[1] A video of live predictions can be seen at https://www.youtube.com/watch?v=wE2pPZQGR6U

Power: The power (intensity) in dB, was automatically extracted using Snack, and then z-normalized for the individual speaker.

Spectral stability: Since final lengthening is known to be an indicator for turn-taking, a measure of spectral stability was derived. First, the Snack FFT analysis was used to get the power spectrum divided into *N* bands (up to 4 kHz), at each time step. Then the following equation was used to calculate the stability S_t at time *t*:

$$S_t = \sum_{n=0}^{N} p_{n,t} - \sum_{n=0}^{N} abs(p_{n,t} - p_{n,t-1})$$

where $p_{n,t}$ is the power in band *n* at time *t*. As is evident from the equation, S_t will be high when the total power in the spectrum is high, but when the power profile of the spectrum is stable, and should therefore be an indication of phonetic lengthening. Just like with the other prosodic features, this stability score was z-normalized for the individual speaker.

Part-of-Speech (POS): Previous studies have found the final POS tags to be indicative of turn-taking (Gravano and Hirschberg, 2011; Koiso et al., 1998). The corpus was already manually annotated with 59 different POS tags. A one-hot representation (with 59 features per speaker) was used. These features were all set to 0 as default, but 100ms after a word ended, the corresponding POS feature was set to 1 for one frame. This was done to simulate what could ideally be achieved in a real dialogue system, given that the spoken word would be available from an incremental speech recognizer immediately after it is spoken. Although this is a somewhat idealistic assumption, it serves an indication of the upper limit performance.

Since the POS features are the most challenging to extract in a live system, and the value of prosodic and syntactic features for turn-taking prediction has been debated (Ruiter et al., 2006; Edlund and Heldner, 2005), we are interested in evaluating two sets of features. The first set (**Full**) comprises all features listed above. The second set (**Prosody**), uses all features except POS, i.e., features that can be extracted directly from the speech signal without any speech recognition. In total, 12 features were used for the Prosody model (6 for each speaker), and 130 features for the Full model (65 for each speaker).

4 Experiments

4.1 Training the Model

To train and evaluate the model, we used the Deeplearning4j framework (Deeplearning4j, 2017). The training data was partitioned into mini-batches of 32 examples, with a sequence length of 60 seconds. Since these sequences are too computationally demanding to fully train, the Truncated Back-propagation Through Time (BPTT) procedure was applied, with a length of 10 seconds. The learning rate was set to 0.01. To avoid overfitting, an l2 regularization of 0.001 was used. The weights were updated using RMSProp, which is often used for LSTM. A sigmoid activation function was used for the output layer, and a *tanh* function for the hidden layer. The network was optimised using a mean-squared error loss function.

For the Full model, we used 40 hidden nodes in the LSTM layer, and for the Prosody model we used 10 hidden nodes, reflecting the different number of input nodes. Both models were trained for 100 epochs. This took about 2 days for the Full model on an Intel core i7 laptop.

Some examples of the predictions the model makes on the test set are shown in the Appendix. To evaluate the performance of the model, we measured the Mean Absolute Error across all 60 output nodes, at all time steps, when applying the model to the test set. The average performance of different sets of output nodes (covering different future windows) for the Full model, are shown in Figure 3.

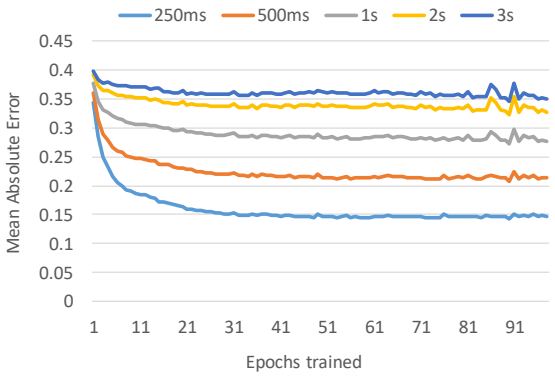


Figure 3. Prediction performance of the Full model on the test set, for different time windows (prefixes of output vectors) and depending on the number of epochs trained.

As can be seen, the performance varied a lot depending on the time window – predictions within the first second are much more accurate

than predictions further into the future. It also looks like the network seems to learn and stabilize the performance fairly early on. However, it is important to stress that this is a crude overall performance over all time steps. As we will see in the next section, it might hide improvements for more specific predictions.

4.2 Predictions at Pauses

One of the most common turn-taking decisions that has been modelled in related work is to predict whether a speaker will continue speaking when a brief pause is detected (HOLD), or whether the turn will shift to the other speaker (SHIFT). This is important to model in spoken dialogue systems, in order to know when the system should take the turn, but it could also be applied to predict whether the user is likely to take the turn or not after the system has made a pause.

To investigate whether the trained model could be used for such predictions in the test set (without being specifically trained for this decision), we identified all places where 10 frames (500ms) of silence had just passed since the last speaker was speaking (we will investigate different pause lengths further down). This amounted to 2876 instances in the test set. Of these, we selected instances where one (and only one) of the speakers continued within 1 second (2079 in total). We then averaged the predictions of the first second for the two networks associated with each speaker. The network with the highest average score was selected as the predicted next speaker. This binary classification task (SHIFT vs. HOLD) gives us an F-score with which we can compare the performance of different network configurations. Since the two classes are fairly well balanced (881 vs. 1198), a majority-class baseline (always HOLD) only yields an F-score of 0.421.

Figure 4 shows the performance for the Prosody and the Full models, depending on the total number of epochs trained. As can be seen, the performance of this specific decision is fairly unstable across epochs – probably because the model is not specifically trained towards this decision – and thus it might be hard to know which epoch model to choose. However, we found that the performance on the test set and the training set were highly correlated across epochs (r = 0.98). Thus, if the model that performs best on the training set is chosen, it will most likely be optimal for the test set. As the figure shows, the performance of the Prosody model quickly stabilizes and reaches an F-score of 0.724 at epoch 30

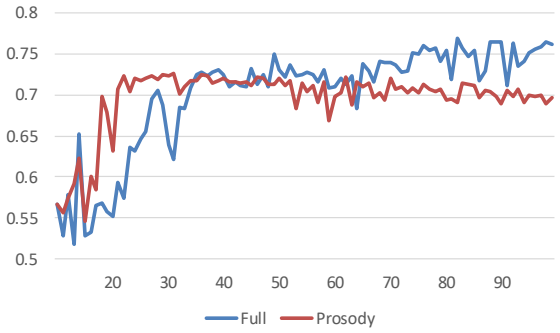


Figure 4. Prediction performance (F-score) of turn-shifts at pauses for the two models when applied to the test set, depending on the number of epochs trained.

(and then degrades somewhat), whereas the Full model continues to learn, reaching an F-score of 0.762 at epoch 100.

In the experiments above, we have studied the prediction performance after a pause of 500ms. However, turn-shifts might of course be much more rapid than this, and a dialogue system should be able to assess whether it should take the turn immediately when a pause is detected, or possibly wait a longer time if it is uncertain. Previous approaches have done this by training specific models at different pause lengths, which are then applied after each other as the pause progresses (Ferrer et al., 2002). Since our model is continuous, it can be directly applied at each time step during a pause. To assess the performance of the model after very brief pauses, we also evaluated the model after just 50ms (1 frame) or 250ms (5 frames) of silence. The results are shown in Table 1.

Table 1: Prediction performance of turn-shifts at pauses for the Full model, depending on pause length.

	50ms	**250ms**	**500ms**
Instances	4933	3405	2079
% HOLD	62.3%	59.8%	57.6%
Precision SHIFT	0.752	0.726	0.711
Recall SHIFT	0.583	0.703	0.738
Precision HOLD	0.778	0.805	0.802
Recall HOLD	0.884	0.822	0.780
F-score	**0.763**	**0.774**	**0.762**
Baseline F-score (always HOLD)	0.479	0.448	0.421

Interestingly, as the precision/recall numbers show, the model seems to be biased towards making HOLD predictions early on in the pause. This is arguably a good trade-off, since it means that the model would be inclined to wait a little

bit longer to make another decision, instead of interrupting the user. In any case, the F-score is very similar regardless of pause length, which shows that a relatively good prediction performance can be achieved already after very brief pauses, potentially allowing dialog systems to give responses with barely any gap.

It is not obvious what to compare the performance with. Since a lot of turn-taking behaviour is optional, and we are evaluating the model based on what the humans actually did, we could never expect these predictions to be 100% correct. One comparison is Neiberg and Gustafson (2011), who also used the HCRC Map Task data to predict turn SHIFT vs. HOLD, with a model specifically trained for this. Using Gaussian Mixture Modelling with prosodic features derived right before the pause, their best performance was an average recall of 0.578–0.614, depending on which part of the corpus they were targeting. However, since their data preparations and definitions were not exactly the same as ours, we also trained a set of more traditional models on our data set, using Naive Bayes, Support Vector Machines and Logistic Regression, to classify each 500ms pause as either HOLD or SHIFT. Since these are not sequential models, we cannot use the features directly in the same way as was used for the RNN. Instead, we used feature engineering similar to Meena et al. (2014), including syntactic features (last POS unigram and bigram), prosodic features (pitch slope, mean pitch, mean intensity, and mean spectral stability in the final 300ms voiced region), and context (length of last IPU and last turn). The models were trained on the training set and evaluated on the test set. The best result on the full feature set was obtained using Naive Bayes, which yielded an F-score of 0.677. When using only prosodic features, Logistic Regression yielded the best F-score of 0.590, which similar to Neiberg and Gustafson (2011). These performances are clearly below the performance of our model, even though we did not train it specifically for this decision.

Another possible comparison is how well a human would perform the task. To test this, we used the Crowdflower platform, where human subjects were paid to judge which speaker would continue after a brief silence, given 10 seconds of interaction ending just after a pause of 500ms (i.e., the same task ask the RNN was given). To simplify the task, we selected a random subset of the corpus where there was a man and a woman talking (207 instances), and asked the annotator "do you think the man or the woman will speak next?" As a quality control question, we also asked whether it was the man or the woman that was the last speaker, and excluded annotators who gave an incorrect answer. Three different annotators judged each instance. Using the majority vote, the humans reached an F-score of 0.709, which is below the performance of our best models. A summary of the different comparisons made here with our model is shown in Table 2.

Table 2: Summary of F-score comparisons for predicting turn-shifts at 500ms pauses.

Majority-class baseline	0.421
Human performance	0.709
Logistic Regression, Prosody only	0.590
RNN, Prosody only	0.724
Naive Bayes, All features	0.677
RNN, All features	0.762

4.3 Predictions at Speech Onset

Next, we wanted to see if the same model can be applied to a different task: to predict utterance length at the onset of speech. As discussed in 2.1, this prediction would be useful for a dialogue system, in order to determine whether it should stop speaking or not, given that the user has just started to speak. If the user is just giving a brief response (i.e., a backchannel), the system typically does not have to stop speaking. However, if the user is initiating a longer response, the system might decide to stop speaking and allow the user to "barge-in" (cf. Selfridge et al., 2013) .

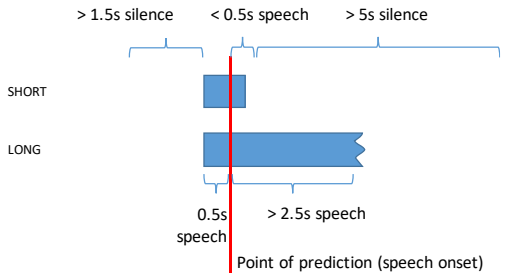


Figure 5. Definitions of SHORT and LONG utterances.

We therefore wanted to test if our model can, already at the speech onset, predict whether the utterance will be very brief or longer. To test this, we identified instances in the data where a speaker had just initiated a LONG or a SHORT utterance (i.e., something like a backchannel). The definitions of these categories are illustrated in Figure 5. To fall in any of these categories, at least 1.5s of silence by one participant has to be followed by an onset of 500ms of speech. If this

onset was followed by a maximum of 500ms of more speech, and then no speech (by the same speaker) for 5s, it was categorized as a SHORT utterance. If it was followed by at least 2.5s of speech, it was categorized as a LONG utterance. With these definitions, the test set contained 196 SHORT utterances and 179 LONG utterances. At each onset, the prediction score over the 60 output nodes in the model were averaged. Figure 6 shows the number of instances in the test set that received different prediction scores (rounded to deciles) by the Full model, depending on whether it was in fact a SHORT or LONG utterance. As is evident, the model manages to make a fairly good separation between short and long utterances. Using the best prediction score separation threshold derived from the training set (0.404), the F-score for classifying SHORT vs. LONG utterances in the test set was 0.786.

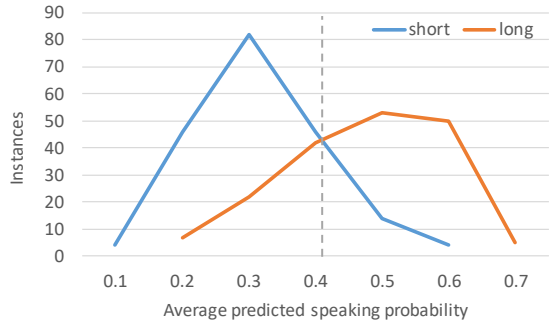


Figure 6. Number of instances with different prediction scores in the test set, using the Full model, at the onset of short and long utterances.

As a minimum comparison, a majority class baseline yields an F-score of 0.359. Another comparison is (Neiberg and Truong, 2011), who trained a model specifically for this decision and achieved a somewhat lower performance. However, they used a different dataset and it is therefore not directly comparable. Just like for the previous task above (4.2), we therefore also trained more traditional models for comparison. We used features that were deemed to be relevant for the task, including the preceding POS unigrams and bigrams for the two speakers, the mean power of the speech onset, whether it was voiced, whether it was overlapping with the other speaker, and time since last speech for both speakers. The best F-score of 0.684 was achieved using a Naive Bayes classifier. Again, our generic model achieves a better performance than traditional non-sequential models that were trained specifically for the task.

4.4 Application to Spoken Dialogue Systems

One important question is whether the models trained on human-human data could also be used to predict turn-taking in human-computer dialogue. Or, rather, could they be used to predict a *desired* behaviour for the system, given the dialogue history between the human and the computer up to some point in time, as discussed in 3.1 above? This is of course challenging, partly because human-human interaction and human-computer interaction typically look very different, but also because human-human turn-taking behaviour might not necessarily be a role model for how we want systems to behave. To test this, we used data from a previous study on human-robot interaction (Johansson et al., 2016). In that setting, the user was asked to tell the robot about a past visit to a foreign country, while the robot listened actively by giving backchannels and asking various follow-up questions to elicit more elaborate descriptions. The corpus consists of 30 dialogues with 15 different subjects. Each end of an IPU was manually annotated as either HOLD, OPTIONAL or TAKE. To make the task clearer, we excluded the OPTIONAL instances, and tested whether the model could distinguish between HOLD (213 instances) and TAKE (303 instances).

For this data, we used the Prosody model (at epoch 30), since we did not have any POS features. We first applied the model directly according to the simple approach outlined in 3.1 above, i.e., we fed the user's and the system's speech into two networks and then compared the predictions for the user and the system at the end of each IPU. If the system's prediction was stronger than the user's, a TAKE was selected, otherwise a HOLD. However, this only yielded an F-score of 0.582, which is a very modest improvement over the majority class baseline of 0.434.

As discussed above, there are a number of reasons why it is hard for the model to make direct predictions towards the labels in this dataset. A training set more similar to the testing set is most likely needed. However, it is still possible that the network might model phenomena relevant to turn-taking in the dialogue, and be useful for feature extraction. To test this, we partitioned the human-robot interaction data into a training and testing set, and applied a Logistic Regression model trained on the manual annotations (TAKE/HOLD). As input features, we used the hidden nodes in the RNN network, at the time of the prediction. In a 10-fold cross validation, this

yielded an F-score 0.751. Thus, it seems like the network had learned to transform the feature space, and the logistic regression only has to make a final linear separation in this new feature space. This would also mean that it should be possible to train with relatively few training examples. Indeed, when training on only 20% of the data (and evaluating on the other 80%), this approach still yields a relatively high average F-score of 0.72. This is promising, since it means that the model could at least be used for feature extraction to make turn-taking decisions in spoken dialogue systems, with only a small amount of manually labelled training data.

5 Conclusions and Discussion

In this paper, we have presented a first step towards a general model of turn-taking in spoken dialogue. Unlike most previous models, the proposed model is not trained towards specific turn-taking decisions, but instead makes continuous predictions of future speech activity. To evaluate the model, we have applied it to two different turn-taking decisions for which it was not specifically trained. First, to detect the next speaker at pauses, where the model achieves a better performance than more traditional attempts on the same dataset, and better than human performance. Second, to project the length of an utterance at speech onset, where the model also yields a better performance than traditional models. Finally, we have tested the model on human-robot dialogue data. Most likely due to the large differences in training and testing conditions, the model was not directly applicable for making turn-taking decisions in this setting. However, it could at least be used for feature extraction to train a separate model on a small set of manually labelled data.

So far, we have relied on manually labelled POS features (for the Full model). For future studies, we would like to see how well the model would cope with automatic online POS tagging of ASR results. Although we have worked with manually annotated speech segments, these could also be extracted with a VAD. All other features were automatically extracted.

As noted earlier, the model should be applicable to multi-party interaction. Another obvious extension is to use multi-modal features, such as gaze and gestures, which have shown to be important for turn-taking (Kawahara et al., 2012; Johansson and Skantze, 2015).

So far, we have only tested the model on binary decisions, in order to make the results as clear and comparable as possible. However, this clearly only hints at some of the potential applications of the model (which can be grasped by looking at the examples in the Appendix). For example, since the model is continuous and predictive, it should be possible to use it for preparing a dialogue system to make responses before the user's utterance is completed. Since the model is probabilistic, it should be possible to use it in a decision-theoretic framework, as discussed in 3.1 above. However, to make the model directly applicable to spoken dialogue systems, it should probably be trained on a more diverse set of interactions, more similar to the actual dialogue system application.

Acknowledgements

This work is supported by the SSF (Swedish Foundation for Strategic Research) project COIN, and the Swedish research council (VR) project *Online learning of turn-taking behaviour in spoken human-robot interaction*.

References

A Anderson, M Bader, E Bard, E Boyle, G Doherty, S Garrod, S Isard, J Kowtko, J McAllister, J Miller, C Sotillo, H Thompson, and R Weinert. 1991. The HCRC Map Task corpus. *Language and Speech*, 34(4):351–366.

H H Clark and J E Fox Tree. 2002. Using uh and um in spontaneous speaking. *Cognition*, 84(1):73–111.

Deeplearning4j. 2017. Deeplearning4j: Open-source distributed deep learning for the JVM, Apache Software Foundation License 2.0.

Nina Dethlefs, Helen Hastie, Heriberto Cuayahuitl, Yanchao Yu, Verena Rieser, and Oliver Lemon. 2016. Information density and overlap in spoken dialogue. *Computer Speech and Language*, 37:82–97.

S Duncan and G Niederehe. 1974. On signalling that it's your turn to speak. *Journal of Experimental Social Psychology*, 10(3):234–247.

Jens Edlund and Mattias Heldner. 2005. Exploring prosody in interaction control. *Phonetica*, 62(2–4):215–226.

Florian Eyben, Felix Weninger, Stefano Squartini, Bjorn Schuller, and Björn Florian Eyben ; Felix Weninger ; Stefano Squartini ; Schuller. 2014. Real-life voice activity detection with LSTM Recurrent Neural Networks and an application to Hollywood movies. In *Proceedings of ICASSP*, volume 1, pages 3709–3713. IEEE, May.

L Ferrer, E Shriberg, and A Stolcke. 2002. Is the speaker done yet? Faster and more accurate end-of utterance detection using prosody. In *Proceedings of ICSLP*, pages 2061–2064.

Agustín. Gravano and Julia. Hirschberg. 2011. Turn-taking cues in task-oriented dialogue. *Computer Speech & Language*, 25(3):601–634.

Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. 2013. Speech recognition with deep recurrent neural networks. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 6645–6649. IEEE, May.

Mattias Heldner and Jens Edlund. 2010. Pauses, gaps and overlaps in conversations. *Journal of Phonetics*, 38(4):555–568.

Anna Hjalmarsson. 2011. The additive effect of turn-taking cues in human and synthetic voice. *Speech Communication*, 53(1):23–35.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780, November.

Martin Johansson, Tatsuro Hori, Gabriel Skantze, Anja Höthker, and Joakim Gustafson. 2016. Making turn-taking decisions for an active listening robot for memory training. In *Proceedings of the International Conference on Social Robotics*, volume 9979 LNAI, pages 940–949.

Martin Johansson and Gabriel Skantze. 2015. Opportunities and Obligations to Take Turns in Collaborative Multi-Party Human-Robot Interaction. In *Proceedings of SigDial*, number September, page 402.

Tatsuya Kawahara, Takuma Iwatate, and Katsuya Takanashi. 2012. Prediction of Turn-Taking by Combining Prosodic and Eye-Gaze Information in Poster Conversations. In *Proceedings of Interspeech*.

A Kendon. 1967. Some functions of gaze direction in social interaction. *Acta Psychologica*, 26:22–63.

Hanae Koiso, Yasuo Horiuchi, Syun Tutiya, Akira Ichikawa, and Yasuharu Den. 1998. An analysis of turn-taking and backchannels based on prosodic and syntactic features in Japanese map task dialogs. *Language and Speech*, 41 (Pt 3-4:295–321.

Bing Liu and Ian Lane. 2016. Joint Online Spoken Language Understanding and Language Modeling with Recurrent Neural Networks. In *Proceedings of Sigdial 2016*, number September, pages 22–30.

Raveesh Meena, Gabriel Skantze, and Joakim Gustafson. 2014. Data-driven models for timing feedback responses in a Map Task dialogue system. *Computer Speech and Language*, 28(4):903–922.

L P Morency, I de Kok, and J Gratch. 2008. Predicting listener backchannels: A probabilistic multimodal approach. In *Proceedings of IVA*, pages 176–190, Tokyo, Japan.

Daniel Neiberg and Joakim Gustafson. 2011. Predicting speaker changes and listener responses with and without eye-contact. In *Proceedings of Interspeech*, number August, pages 1565–1568.

Daniel Neiberg and Khiet P. Truong. 2011. Online detection of vocal Listener Responses with maximum latency constraints. In *Proceedings of ICASSP*, pages 5836–5839. IEEE, May.

Jan-Peter De Ruiter, Holger. Mitterer, and N. J. Enfield. 2006. Projecting the end of a speaker's turn: A cognitive cornerstone of conversation. *Language*, 82(3):515–535.

D Schlangen. 2006. From reaction to prediction: experiments with computational models of turn-taking. In *Proceedings of Interspeech 2006, Pittsburgh, PA, USA*, 2010-2013, September.

Ethan O Selfridge, Iker Arizmendi, Peter A Heeman, and Jason D Williams. 2013. Continuously Predicting and Processing Barge-in During a Live Spoken Dialogue Task. *SIGdial 2013*(August):384–393. NULL.

Kåre Sjölander and Jonas Beskow. 2000. WaveSurfer - an open source speech tool. In *Proceedings of ICSLP 2000*, volume 4, pages 464–467, Beijing.

Gabriel Skantze and David Schlangen. 2009. Incremental dialogue processing in a micro-domain. In *Proceedings of EACL*, number April, pages 745–753. Association for Computational Linguistics.

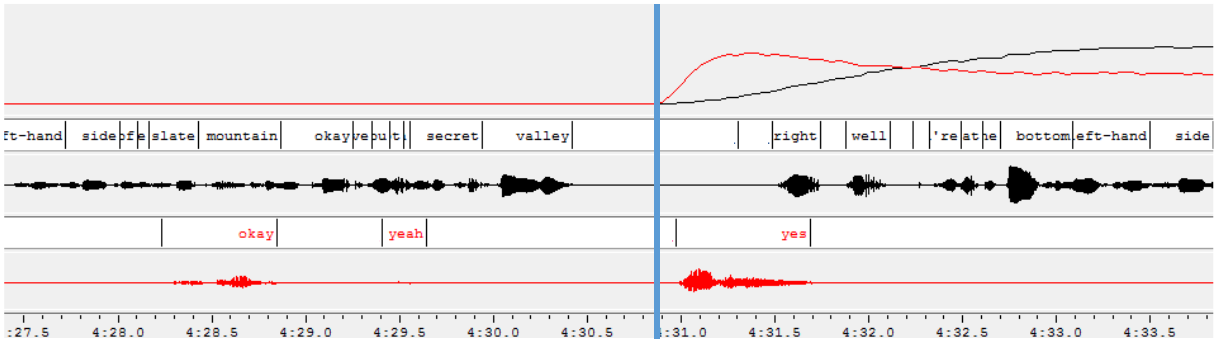
Marisa Tice and Tania Henetz. 2011. The eye gaze of 3rd party observers reflects turn-end boundary projection. In *Proceedings of SemDial*, pages 204–205, Los Angeles, CA, US, September.

Nigel G Ward, Olac Fuentes, and Alejandro Vega. 2010. Dialog Prediction for a General Model of Turn-Taking. In *Proceedings of Interspeech-2010*.

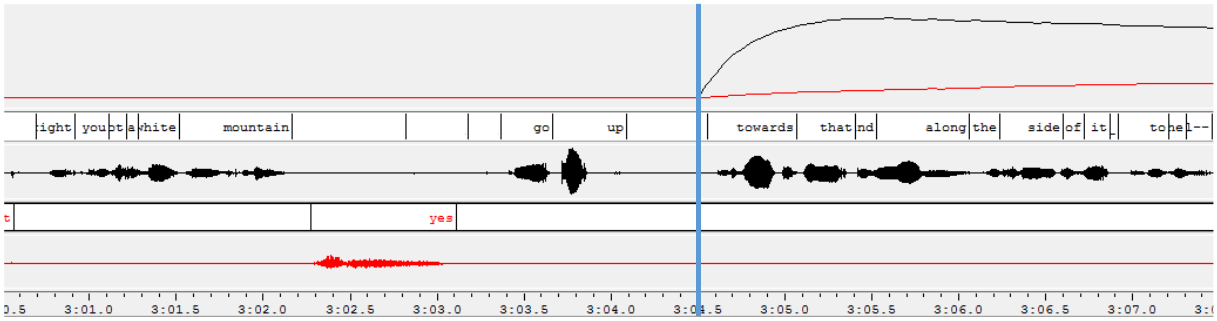
Victor H Yngve. 1970. On getting a word in edgewise. In *Papers from the sixth regional meeting of the Chicago Linguistic Society*, pages 567–578, Chicago, April.

Appendix – Examples of model predictions

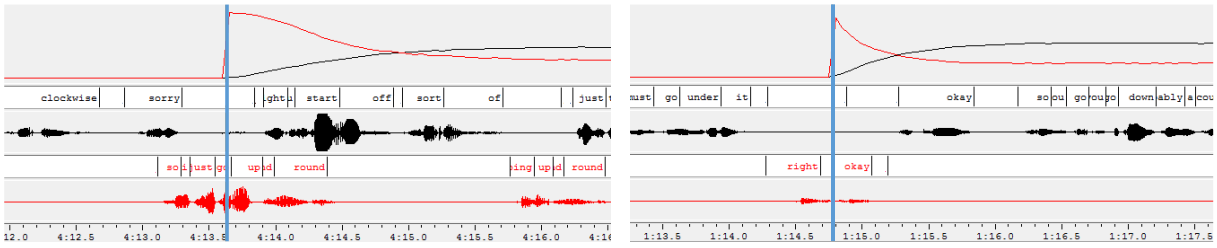
These are some examples of the output of the model, when applied to unseen test data. The blue vertical bar shows the point of prediction (i.e., no predictions are shown before this point), and the curves show the predictions for the future 3 seconds window. One speaker is represented with black (the information giver) and the other with red (the information follower).



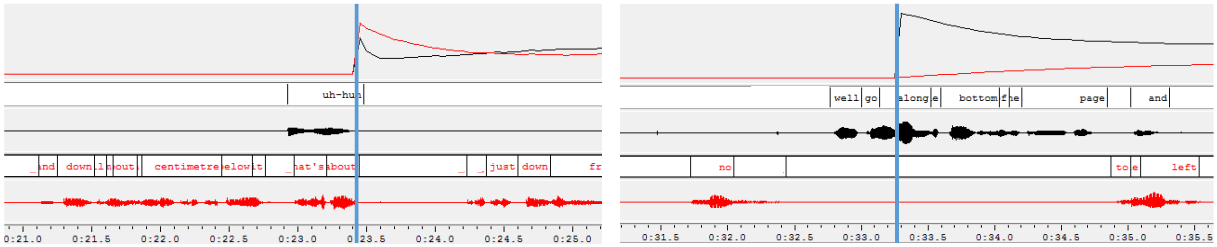
Example 1: Prediction in a pause. The model predicts that the red speaker will give a (short) response, but also that the black speaker will continue later on.



Example 2: Prediction in a pause. The model predicts that the black speaker will continue, and that the red speaker will not respond.



Example 3: Prediction at speech onset. On the left, the red speaker has just started a longer utterance (but is eventually interrupted by the black speaker). On the right, the speaker has only started a brief response (a backchannel). This is reflected by a stronger prediction for the red speaker in the left picture compared to the right picture.



Example 4: Prediction at speech onset, similar to Example 3. However, notice that it is the information giver that gives the backchannel here, and that it is still correctly distinguished from the longer utterance on the right.