INTERSPEECH 2017
August 20–24, 2017, Stockholm, Sweden

Online End-of-Turn Detection from Speech based on Stacked Time-Asynchronous Sequential Networks

Ryo Masumura, Taichi Asami, Hirokazu Masataki, Ryo Ishii, Ryuichiro Higashinaka

NTT Media Intelligence Laboratories, NTT Corporation, Japan

{masumura.ryo, asami.taichi, masataki.hirokazu, ishii.ryo, higashinaka.ryuichiro}@lab.ntt.co.jp

Abstract

This paper presents a novel modeling called stacked time-asynchronous sequential networks (STASNs) for online end-of-turn detection. An online end-of-turn detection that determines turn-taking points in a real-time manner is an essential component for human-computer interaction systems. In this study, we use long-range sequential information of multiple time-asynchronous sequential features, such as prosodic, phonetic, and lexical sequential features, to enhance online end-of-turn detection performance. Our key idea is to embed individual sequential features in a fixed-length continuous representation by using sequential networks. This enables us to simultaneously handle multiple time-asynchronous sequential features for end-of-turn detection. STASNs can embed all of the sequential information between a start-of-conversation and the current end-of-utterance in a fixed-length continuous representation that can be directly used for classification by stacking multiple sequential networks. Experiments show that STASNs outperforms conventional modeling with limited sequential information. Furthermore, STASNs with senone bottleneck features extracted using senone-based deep neural networks have superior performance without requiring lexical features decoded by an automatic speech recognition process.

Index Terms: online end-of-turn detection, sequential networks, time-asynchronous sequential features, senone bottleneck features.

1. Introduction

In human-computer interaction, systems that can perform human-like behavior have been needed [1]. For the human-like behavior, end-of-turn detection that decides whether a user's utterance is ended or not is an important technology because turn-taking behavior strongly affects to user's impressions [2,3]. This paper focuses on speech-based end-of-turn detection that can be used for interactive voice response (IVR) systems.

A simple end-of-turn detection is based on non-speech duration that is determined by speech activity detection (SAD) [4]. However, rhythm gets worse if the duration is increased; incorrect turn-takings occur frequently if the duration is decreased. In fact, SAD-based end-of-turn detection is reported to significantly stress users [5]. Therefore, more sophisticated end-of-turn model that can accurately determine a turn-taking point in a real time manner is desired.

A lot of work has been examined in order to model end-of-turn detection [6–16]. In speech-based end-of-turn detection, past speech context, such as prosodic, phonetic, and lexical features had been often utilized. More specifically, fixed-range sequential features just before the target end-of-utterance [6–14] or simplified utterance-level features such as maximum, minimum, or average values [12–16] were employed. On the other hand, previous work could not handle both long-range sequential features of the target utterance and features extracted from past utterances. This is because conventional discriminative modeling such as decision trees or support vector machines could not support variable-length features.

This paper establishes a modeling that can manage such rich speech context information. Our key idea is to introduce multiple sequential networks with a recurrent neural network (RNN) structure. The sequential networks can embed a variable-length sequential feature into a fixed length continuous vector that can be directly used for classification. We can expect to simultaneously handle multiple time-asynchronous sequential features and features in past utterances by using multiple sequential networks thoughtfully.

In this paper, stacked time-asynchronous sequential networks (STASNs) are proposed. The STASNs can manage the whole of multiple sequential features behind individual end-of-utterance points in two stages. In the first stage, each sequential feature within an utterance is individually embedded to a fixed length continuous vector using multiple feature-level sequential networks. This enables us to convert utterance-level sequential information into a fixed length vector at each end-of-utterance point. Additionally, in the second stage, the utterance-level vectors individually extracted at each end-of-utterance are also embedded to a fixed length continuous vector that is directly used for classification using an utterance-level sequential network. In the STASN, individual sequential networks can perform in an asynchronous manner, making it suitable for online end-of-utterance detection.

In addition, this paper attempts to build an accurate online end-of-turn detection system without introducing lexical features. Although the lexical features were the most informative features for end-of-turn detection [14], time latency and misrecognition problems must be caused by an automatic speech recognition (ASR) process. Instead of them, this paper utilizes senone bottleneck features as phonetic information, which can be extracted from a bottleneck layer of senone-based deep neural networks (DNNs) [17]. The senone bottleneck features had been applied for speaker recognition and language identification [17–21]. It can be expected that the STASN with the senone bottleneck features is a great solution for building non-lexical online end-of-turn detection because the bottleneck features involve similar information to the lexical features.

Main contributions are summarized as follows.

- This paper proposes the STASN. We introduce words, mel-frequency cepstrum coefficients (MFCCs), fundamental frequencies (F0s), and senone bottleneck features as time-asynchronous sequential features.
- This paper reveals that long-range sequential features of the target utterance and features extracted from past ut-

http://dx.doi.org/10.21437/Interspeech.2017-651

terances can improve end-of-turn detection performance compared with only using limited context information behind the end-of-utterance point.

- This paper presents the results of both non-lexical systems and lexical systems. We show that non-lexical systems with senone bottleneck features can yield comparable performance to lexical systems with an ASR process.

2. Related Work

In end-of-turn detection, multimodal features have been examined, whereas this paper only deals with features extracted from speech. The multimodal features include gaze behavior, head gesture, and respiration [22–25]. These features can be used as one of the sequential features for STASNs.

In addition to time-asynchronous sequential features, fixed-length features including utterance duration or speaking rate have been utilized for end-of-turn detection from speech [15, 16]. Although these features are available to STASNs, this paper only focuses on sequential features. Moreover, this paper does not focus on complicated lexical features such as dialogue acts because non-lexical systems are our ideal form [11, 26].

A related problem of end-of-turn detection is backchannels [6, 27]. It is known that similar modeling and features were utilized for backchannel point estimation. We expect that this work can be applied to backchannel point estimation.

Human-computer conversational data sets have also been used for evaluation, while this paper uses human-human conversational data sets [9, 10]. Human-computer interactions are reported to show different attributes from human-human interactions [28]. Our goal is that computers can perform human-like behavior by acquiring skills from human-human interactions.

3. Online End-of-Turn Detection

End-of-turn detection is a problem that detects whether each end-of-utterance point is a turn-taking point or not. Figure 1 details the end-of-turn detection problem. The utterance is defined as an internal pause unit (IPU) that is a unit surrounded by non-speech unit in which duration q is more than σ [6]. The speech/non-speech unit is estimated by speech activity detection (SAD).

In online end-of-turn detection, all past information behind the target end-of-utterance can be used for context information. Thus, a non-speech duration immediately after the target end-of-utterance cannot be used. An estimated label is one of two behaviors: end-of-turn or not. The label of the t-th end-of-utterance in a conversation can be decided by:

$$\hat{l}^t = \underset{l}{\operatorname{argmax}}\, P(l|\boldsymbol{X}^1, \cdots, \boldsymbol{X}^t, \boldsymbol{\Theta}), \tag{1}$$

where $\boldsymbol{\Theta}$ denotes a model parameter. $\hat{l}^t$ is an estimated label of the t-th end-of-utterance. $\boldsymbol{X}^t$ represents multiple asynchronous sequential features within t-th utterance and involves N kinds of sequential features:

$$\boldsymbol{X}^t = \{\boldsymbol{x}_1^t, \cdots, \boldsymbol{x}_N^t\}, \tag{2}$$

where $\boldsymbol{x}_n^t$ represents the n-th sequential feature in the t-th utterance. $\boldsymbol{x}_n^t$ involves frame-level features:

$$\boldsymbol{x}_n^t = \{\boldsymbol{v}_{n,1}^t, \cdots, \boldsymbol{v}_{n,I_n}^t\}, \tag{3}$$

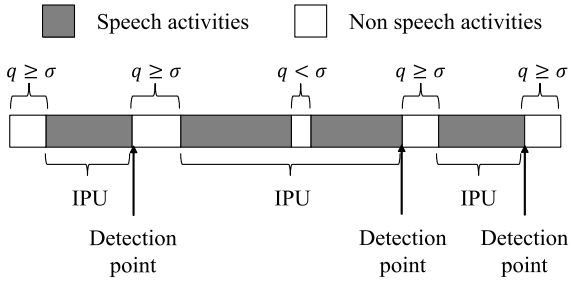

Figure 1: *Details of an end-of-turn detection problem.*

where $\boldsymbol{v}_{n,i}^t$ is the i-th frame's feature of the n-th sequence in the t-th utterance. In this paper, a word embedding, MFCC, F0, and a senone bottleneck feature correspond to the frame-level feature. Note that the length of each sequence I_n has an index n; i.e., each sequence has a different length.

4. Proposed Method

4.1. Modeling

A novel modeling of $P(l^t|\boldsymbol{X}^1, \cdots, \boldsymbol{X}^t, \boldsymbol{\Theta})$ is proposed. Figure 2 shows the model structure of the proposed method with three time-asynchronous sequential features. The proposed method introduces multiple sequential networks with an RNN structure for embedding entire sequential information into a continuous representation. This paper uses LSTM-RNNs for the sequential networks. The proposed method manages multiple time asynchronous-sequential features in two stages. In the first stage, feature-level sequential networks are applied for individual sequential features within an utterance. We call this modeling time-asynchronous sequential network (TASN). In the second stage, an utterance-level sequential network is also applied to utterance-level continuous representations. We call this modeling stacked TASN (STASN).

4.1.1. Time-Asynchronous Sequential Networks

In TASN, each feature within an utterance is individually embedded into a continuous representation in an asynchronous manner. Sequential networks are prepared for individual sequential features. Each sequential network embeds sequential information as:

$$\boldsymbol{h}_{n,i}^t = \texttt{LSTM}(\boldsymbol{v}_{n,1}^t, \cdots, \boldsymbol{v}_{n,i}^t; \boldsymbol{\theta}_n^{\texttt{F}}), \tag{4}$$

$$= \texttt{LSTM}(\boldsymbol{v}_{n,i}^t, \boldsymbol{h}_{n,i-1}^t; \boldsymbol{\theta}_n^{\texttt{F}}), \tag{5}$$

where $\boldsymbol{h}_{n,i}^t$ denotes a continuous representation that embeds the n-th sequential feature within the t-th utterance from a start-of-utterance to the i-th frame. $\texttt{LSTM}()$ represents a function of the unidirectional LSTM-RNN layer. $\boldsymbol{\theta}_n^{\texttt{F}}$ is a model parameter for the n-th sequence. For each sequential feature, this procedure is repeated until a end-of-utterance point.

At the end-of-utterance point, continuous representations individually composed from each sequential feature are merged as an utterance-level continuous representation:

$$\boldsymbol{H}^t = [\boldsymbol{h}_{1,I_1}^{t\top}, \cdots, \boldsymbol{h}_{N,I_N}^{t\top}]^\top, \tag{6}$$

where $\boldsymbol{H}^t$ means the utterance-level continuous representation for the t-th utterance.

In an output layer of TASN, the utterance-level continuous representation $\boldsymbol{H}^t$ is directly used for end-of-turn detection of

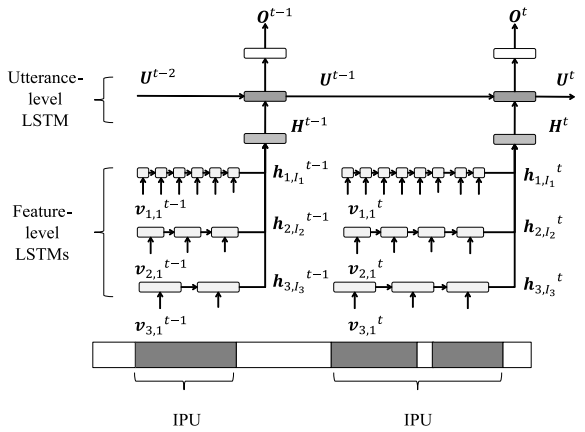

Figure 2: *Detailed structure of STASN.*

the t-th utterance:

$$\boldsymbol{O}^t = \texttt{SOFTMAX}(\boldsymbol{H}^t; \boldsymbol{\theta}^{\texttt{O}}), \tag{7}$$

where $\texttt{SOFTMAX}()$ is a softmax function, and $\boldsymbol{\theta}^{\texttt{O}}$ is a model parameter for the softmax function. The k-th dimension in $\boldsymbol{O}^t$ corresponds to $P(l^t|\boldsymbol{X}^1, \cdots, \boldsymbol{X}^t, \boldsymbol{\Theta})$.

4.1.2. Stacked Time-Asynchronous Sequential Networks

In STASN, an utterance-level sequential network is additionally introduced for end-of-turn detection. The network can embed sequences of utterance-level continuous representations that are composed by the TASN into a continuous representation:

$$\begin{aligned} \boldsymbol{U}^t &= \texttt{LSTM}(\boldsymbol{H}^1, \cdots, \boldsymbol{H}^t; \boldsymbol{\theta}^{\texttt{I}}), & (8)\\ &= \texttt{LSTM}(\boldsymbol{H}^t, \boldsymbol{U}^{t-1}; \boldsymbol{\theta}^{\texttt{I}}), & (9) \end{aligned}$$

where $\boldsymbol{U}^t$ denotes a continuous representation that embeds sequential information of all information behind the t-th end-of-utterance. $\boldsymbol{\theta}^{\texttt{I}}$ is a model parameter for the utterance-level sequential network.

In an output layer of STASN, end-of-turn detection in the t-th end-of-utterance is defined as:

$$\boldsymbol{O}^t = \texttt{SOFTMAX}(\boldsymbol{U}^t; \boldsymbol{\theta}^{\texttt{O}}). \tag{10}$$

This modeling can consider not only a target utterance but also past utterances for end-of-turn detection.

4.1.3. Optimization

Trainable parameters of both TASN and STASN are represented as:

$$\begin{aligned} \boldsymbol{\Theta}_{\texttt{TASN}} &= \{\boldsymbol{\theta}^{\texttt{F}}_1, \cdots, \boldsymbol{\theta}^{\texttt{F}}_N, \boldsymbol{\theta}^{\texttt{O}}\}, & (11)\\ \boldsymbol{\Theta}_{\texttt{STASN}} &= \{\boldsymbol{\theta}^{\texttt{F}}_1, \cdots, \boldsymbol{\theta}^{\texttt{F}}_N, \boldsymbol{\theta}^{\texttt{I}}, \boldsymbol{\theta}^{\texttt{O}}\}. & (12) \end{aligned}$$

In training, the parameter can be optimized by minimizing cross entropy between a reference probability and an estimated probability:

$$\hat{\boldsymbol{\Theta}} = \underset{\boldsymbol{\Theta}}{\text{argmin}} - \sum_{d \in \mathcal{D}} \sum_t \sum_l \hat{\boldsymbol{O}}_l^{t,d} \log \boldsymbol{O}_l^{t,d}, \tag{13}$$

where $\hat{\boldsymbol{O}}_l^{t,d}$ and $\boldsymbol{O}_l^{t,d}$ are a reference probability and an estimated probability of label l for the t-th end-of-utterance in d-th conversation, respectively. $\mathcal{D}$ represents a training data set.

Table 1: *Experimental data sets.*

Topics	#calls	#utterances	#turns
Finance	50	3,459	2,214
Internet provider	64	3,411	1,843
Local government unit	58	3,232	1,639
Mail-order	52	3,277	1,878
PC repair	45	2,838	1,934
Mobile phone	61	3,701	2,062
Total	330	19,918	11,570

4.2. Features

In TASN and STASN, various sequential features can be used. This paper uses MFCCs, F0s, words, and senone bottleneck features as the sequential features. Each feature corresponds to $\boldsymbol{v}_i^t$ shown in Eq. (3). Here, we add some explanations to features that require some transformational functions with a trainable parameter.

4.2.1. Symbolic features

The sequential features include not only continuous vector sequences such as MFCCs but also symbolic sequences such as words. In TASN and STASN, the symbolic features are used by converting them into continuous vectors. The i-th symbol in the t-th utterance w_i^t is converted as:

$$\boldsymbol{v}_i^t = \texttt{EMBEDDING}(w_i^t; \boldsymbol{\theta}^{\texttt{W}}) \tag{14}$$

where $\texttt{EMBEDDING}()$ is a linear transformational function to convert a symbol to a continuous vector. $\boldsymbol{\theta}^{\texttt{W}}$ is a model parameter for the function. $\boldsymbol{\theta}^{\texttt{W}}$ can be jointly optimized with other trainable parameters in STASN.

4.2.2. Senone bottleneck features

Senone bottleneck features can be extracted from a senone-based DNN with a bottleneck layer. The bottleneck feature means the output of the bottleneck layer. For the senone-based DNN, an input is composed by stacking a currently-being-processed source feature and its left-right contexts. This paper uses MFCC as the source feature. The i-th frame's senone bottleneck feature in the t-th utterance is calculated by:

$$\bar{\boldsymbol{r}}_i^t = [{\boldsymbol{r}_{i-M}^t}^\top, \cdots, {\boldsymbol{r}_i^t}^\top, \cdots, {\boldsymbol{r}_{i+M}^t}^\top]^\top, \tag{15}$$

$$\boldsymbol{v}_i^t = \texttt{BOTTLE}(\bar{\boldsymbol{r}}_i^t; \boldsymbol{\phi}), \tag{16}$$

where $\boldsymbol{r}_i^t$ is a i-th frame's source feature in the t-th utterance. $\texttt{BOTTLE}()$ is a function for extracting a bottleneck feature using a senone-based DNN. $\boldsymbol{\phi}$ is a model parameter for the function, which is preliminarily optimized before STASN training. Although M frames of anticipative processes are required, causing delay time can be ignored compared with the ASR process.

5. Experiments

5.1. Setups

We used the Japanese simulated contact center dialogue data sets for experiments, which include 330 dialogues and 6 topics. One dialogue means one telephone call between one operator and one customer, in which each speaker's speech was separately recorded. Each data set was divided into speech units and non-speech units using DNN-based speech activity detector [29] trained from various Japanese speech. In order to define utterances, σ was set to 200 ms. We manually annotated

Table 2: *Experimental results (%).*

			Features	Recall	Precision	F-value	Accuracy
(1).	Non-Lexical	Baseline	F0	79.6	60.0	68.4	57.3
(2).		Baseline	MFCC	**86.0**	65.1	74.1	65.1
(3).		Baseline	MFCC+F0	75.6	69.3	72.3	66.4
(4).		TASN	F0	81.8	68.5	74.6	67.6
(5).		TASN	MFCC	80.6	75.4	77.9	73.5
(6).		TASN	MFCC+F0	81.0	76.9	78.4	74.4
(7).		STASN	F0	78.9	71.4	75.4	69.8
(8).		STASN	MFCC	81.0	76.0	78.4	74.1
(9).		STASN	MFCC+F0	80.8	77.6	79.1	75.3
(10).		STASN	BOTTLE	85.1	78.6	81.7	77.9
(11).		STASN	BOTTLE+F0	84.6	**79.4**	**82.0**	**78.3**
(12).	Lexical	Baseline	WORD	81.9	77.2	79.5	75.5
(13).		Baseline	WORD+MFCC+F0	78.9	78.8	78.8	75.4
(14).		TASN	WORD	84.2	77.6	80.8	76.7
(15).		TASN	WORD+MFCC+F0	84.3	79.3	81.7	78.1
(16).		STASN	WORD	84.9	78.0	81.3	77.3
(17).		STASN	WORD+MFCC+F0	85.2	**79.7**	82.4	78.8
(18).		STASN	WORD+BOTTLE+F0	**86.4**	79.5	**82.8**	**79.1**

turn-taking points and backchannel points to all dialogues. We excluded utterances with backchannel labels and only evaluated customer's data sets in order to simulate IVR applications. The evaluation was a 6-fold cross validation in which training and validation data were 5 topics and test data were 1 topic. Detailed setups are shown in Table 1 where #calls, #utterances, and #turns represent number of calls, utterances and end-of-turn points, respectively.

In our evaluation, four sequential features were introduced. **F0** is 2 dimensional sequential features of F0 and ΔF0. The frame shift was set to 5 ms. **MFCC** is 36 dimensional sequential features of 12 MFCCs, 12 ΔMFCCs, and 12 $\Delta\Delta$MFCCs. The frame shift was set to 10 ms. **WORD** is 1 dimensional sequential feature of words, which was used by converting into 64 dimensional continuous vectors. In training, manual transcriptions were used. In testing, hypotheses generated by ASR were used. The average word error rate of ASR was 29.4% . **BOTTLE** is 64 dimensional sequential features extracted from the Japanese senone-based DNN. For the source features, M in Eq. (15) was set to 5, the frame shift was 10 ms. The senone-based DNN had five hidden layers. The fourth hidden layer was a bottleneck layer whose unit size was set to 64, and the other hidden layers had 512 units. The DNN was trained from the corpus of spontaneous Japanese [30].

We evaluated 3 modeling methods.

- **Baseline:** Utterance-level neural network with limited-context information. The neural network had one hidden layer with 256 units. As the limited-context information, 50 frames of **F0**, 10 frames of **MFCC**, and 2 **WORD** behind the end-of-utterance were used. For training, the mini-batch size was set to 20 utterances, and Adam optimization was used.
- **TASN:** TASN using LSTM-RNNs. Each LSTM-RNN has 256 units. For training, the mini-batch size was set to 10 calls, and RMSprop optimization was used.
- **STASN:** STASN using LSTM-RNNs. Each LSTM-RNN has 256 units. For training, the mini-batch size was set to 10 calls, and RMSprop optimization was used.

In training, a part of training sets were used for data sets for early stopping. We constructed five models by varying an initial parameter for individual conditions and evaluated averaged performance.

5.2. Results

Our evaluation was examined in non-lexical conditions without WORD features and lexical conditions. The evaluation metrics are recall, precision, macro F-value, and accuracy. Table 2 shows the experimental results.

Lines (1)-(11) show the results for the non-lexical conditions. TASN and STASN outperformed Baseline when MFCC, F0, and MFCC+F0 were used. The performance could be considered an improvement because TASN and STASN can deal with long-range sequential information. In addition, STASN was superior to TASN. This means that past information behind the target utterance can improve end-of-turn detection performance. STASN with BOTTLE achieved remarkable higher performance that that with MFCC. The result confirms that explicitly extracted phonetic information is better than raw MFCC. In non-lexical conditions, the best results of F-value and accuracy were attained by STASN with BOTTLE+F0.

Lines (12)-(18) show the results for the lexical conditions. As with non-lexical conditions, TASN and STASN outperformed Baseline, and STASN was superior to TASN. In terms of features, WORD was an effective feature compared with F0 and MFCC for all modeling. A remarkable point is that STASN with BOTTLE was comparable to that with WORD. This confirms that STASN can exploit similar information to WORD from BOTTLE. The best results of F-value and accuracy were attained by STASN with WORD+BOTTLE+F0.

6. Conclusions

This paper proposed STASNs for online end-of-turn detection. STASNs can utilize multiple asynchronous sequential features between the start-of-conversation and the current end-of-utterance. Our experiments revealed that the long-range sequential information of both the target utterance and past utterances improves end-of-turn detection performance compared with only using limited context information behind the end-of-utterance point. Moreover, we verified that non-lexical systems based on STASN with senone bottleneck features can yield comparable performance to lexical systems with an ASR process. In future work, we will enhance STASN by utilizing not only target speaker's utterance information but also collocutor's utterance information.

7. References

[1] N. G. Ward and D. D. Vault, "Ten challenges in highly-interactive dialog systems," *AAAI Spring Symposium, Turn-Taking and Coordination in Human-Machine Interaction*, pp. 104–107, 2015.

[2] H. Sacks, H. A. Schegloff, and G. Jefferson, "A simplet systematics for the organization of turn-taking for conversation," *Language*, pp. 696–735, 1974.

[3] R. Meena, G. Skantze, and J. Gustafson, "Data-driven models for timing feedback responses in a map task dialogue system," *Computer Speech and Language*, vol. 28, pp. 903–922, 2014.

[4] R. Hariharan, J. Hakkinen, and K. Laurila, "Robust end-of-utterance detection for real-time speech recognition applications," *In Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 249–252, 2001.

[5] N. G. Ward, A. G. Rivera, K. Ward, and D. G. Novick, "Root causes of lost time and user stress in a simple dialog system," *In Proc. Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 1565–1568, 2005.

[6] H. Koiso, Y. Horiuchi, S. Tutiya, and A. Ichikawa, "An analysis of turn-taking and backchannels based on prosodic and syntactic features in japanese map task dialogs," *Language and Speech*, vol. 41, pp. 295–321, 1998.

[7] E. Shriberg, A. Stolcke, D. Hakkani-Tur, and G. Tur, "Prosody-based automatic segmentation of speech into sentences and topics," *Speech Communication*, vol. 32, pp. 127–154, 2000.

[8] D. Schlangen, "From reaction to prediction: Experiments with computational models of turn taking," *In Proc. Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 17–21, 2006.

[9] A. Gravano and J. Hirschberg, "Turn-taking cues in task-oriented dialogue," *Computer Speech and Language*, vol. 25, pp. 601–634, 2011.

[10] R. Sato, R. Higashinaka, M. Tamoto, M. Nakano, and K. Aikawa, "Learning decison trees to determine turn-taking by spoken dialogue systems," *In Proc. International Conference on Spoken Language Processing (ICSLP)*, pp. 861–864, 2002.

[11] N. Guntakandla and R. D. Nielsen, "Modelling turn-taking in human conversations," *AAAI Spring Symposium, Turn-Taking and Coordination in Human-Machine Interaction*, pp. 17–22, 2015.

[12] L. Ferrer, E. Shriberg, and A. Stolcke, "In the speaker done yet? faster and more accurate end-of-utterance detection using prosody in human-computer dialog," *In Proc. International Conference on Spoken Language Processing (ICSLP)*, pp. 2061–2064, 2002.

[13] L. Ferrer, E. Shriberg, and A. Stolcke, "A prosody-based approach to end-of-utterance detection that does not require speech recognition," *In Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 608–611, 2003.

[14] M. A. T. Baumann and D. Schlangen, "Towards incremental end-of-utterance detection in dialogue systems," *In Proc. International Conference on Computational Linguistics (COLING)*, pp. 11–14, 2008.

[15] H. Arsikere, E. Shriberg, and U. Ozertem, "Computationally-efficient endpointing features for natural spoken interaction with personal-assistant systems," *In Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 3241–3245, 2014.

[16] H. Arsikere, E. Shriberg, and U. Ozertem, "Enhanced end-of-turn detection for speech to a personal assistant," *In Proc. AAAI Spring Symposium, Turn-Taking and Coordination in Human-Machine Interaction*, pp. 75–78, 2015.

[17] L. Ferrer, Y. Lei, M. McLaren, and N. Scheffer, "Study of senone-based deep neural network approaches for spoken language recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, pp. 105–116, 2016.

[18] Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, "A novel scheme for speaker recognition using a phonetically-aware deep neural network," *In Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 1695–1699, 2014.

[19] L. Ferrer, Y. Lei, M. McLaren, and N. Scheffer, "Spoken language recognition based on senone posteriors," *In Proc. Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 2150–2154, 2014.

[20] M. McLaren, L. Ferrer, and A. Lawson, "Exploring the role of phonetic bottleneck features for speaker and language recognition," *In Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 5575–5579, 2016.

[21] R. Masumura, T. Asami, H. Masataki, and Y. Aono, "Parallel phonetically aware DNNs and LSTM-RNNs for frame-by-frame discriminative modeling of spoken language identification," *In Proc. International Conference on Acoustics, Speech, and Signal Processing ICASSP*, pp. 5260–5264, 2017.

[22] I. de Kok and D. Heylen, "Multimodal end-of-turn prediction in multi-party meetings," *In Proc. International Conference on Multimodal Interaction (ICMI)*, pp. 91–98, 2009.

[23] G. Skantze, A. Hjalmarsson, and C. Oertel, "Turn-taking, feedback and joint attention in situated humanrobot interaction," *Speech Communication*, vol. 65, pp. 50–66, 2014.

[24] R. Ishii, K. Otsuka, S. Kumano, and J. Yamato, "Analysis of respiration for prediction of who will be next speaker and when in multi-party meetings," *In Proc. International Conference on Multimodal Interaction (ICMI)*, pp. 18–25, 2015.

[25] M. Johansson and G. Skantze, "Opportunities and obligations to take turns in collaborative multi-party human-robot interaction," *In Proc. Annual SIGdial Meeting on Discourse and Dialogue (SIGDIAL)*, pp. 305–314, 2015.

[26] T. Meshorer and P. A. Heeman, "Using past speaker behavior to better predict turn transitions," *In Proc. Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 2900–2904, 2016.

[27] N. Ward and W. Tsukahara, "Prosodic features which cue back-channel responses in english and japanese," *journal of Pragmatics*, vol. 32, pp. 1177–1207, 2000.

[28] C. Doran, J. Aberdeen, L. Damianos, and L. Hirschman, "Comparing several aspects of human-computer and human-human dialogues," *In Proc. Annual SIGdial Meeting on Discourse and Dialogue (SIGDIAL)*, pp. 48–57, 2001.

[29] X.-L. Zhang and J. Wu, "Deep belief networks based voice activity detection," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, pp. 697–710, 2013.

[30] K. Maekawa, H. Koiso, S. Furui, and H. Isahara, "Spontaneous speech corpus of Japanese," *In proc. International Conference on Language Resources and Evaluation (LREC)*, pp. 947–952, 2000.