Paralinguistic event detection from speech using probabilistic time-series smoothing and masking

Rahul Gupta, Kartik Audhkhasi, Sungbok Lee, Shrikanth Narayanan

Signal Analysis and Interpretation Lab (SAIL), Department of Electrical Engineering
University of Southern California, Los Angeles, CA

guptarah@usc.edu, audhkhas@usc.edu, sungbokl@usc.edu, shri@sipi.usc.edu

Abstract

Non-verbal speech cues serve multiple functions in human interaction such as maintaining the conversational flow as well as expressing emotions, personality, and interpersonal attitude. In particular, non-verbal vocalizations such as laughters are associated with affective expressions while vocal fillers are used to hold the floor during a conversation. The Interspeech 2013 Social Signals Sub-Challenge involves detection of these two types of non-verbal signals in telephonic speech dialogs. We extend the challenge baseline system by using filtering and masking techniques on probabilistic time series representing the occurrence of a vocal event. We obtain improved area under receiver operating characteristic (ROC) curve of 93.3% (10.4% absolute improvement) for laughters and 89.7% (6.1% absolute improvement) for fillers on the test set. This improvement suggests the importance of using temporal context for detecting these paralinguistic events.

Index Terms: Non-verbal vocalizations, receiver operating characteristic, time series smoothing, time series masking

1. Introduction

Non-verbal cues are widely used in human communication and serve many important functions such as supporting and maintaining interpersonal interactions, expressing emotions and conveying attitude [1]. Studies have also linked non-verbal communication to higher level behavioral predicates such as marital satisfaction [2] and language acquisition in children [3]. This paper concerns vocal non-verbal cues. Non-verbal vocalizations (NVVs) such as laughters are widely associated with affective expressions [4] and also act as a social lubricant [5]. On the other hand speech fillers are used to hold the floor during a conversation [6, 7]. The Interspeech 2013 Social Signals Sub-Challenge involves framewise detection of these two NVVs with area under the curve (AUC) for receiver operating characteristics (ROC) as the performance metric. [8] gives a detailed study of acoustic properties associated with laughter and observes laughter as composed of several bouts conceptualized as vowel like bursts. Likewise, fillers such as “uhm”, “eh”, “ah” also have specific acoustic properties. The baseline system for the challenge captures these acoustic patterns using a support vector machine (SVM) classifier trained on various frame level acoustic features. However the framewise classification fails to utilize contextual information from the neighboring frames. In this paper, we treat the framewise probability outputs as a time series and apply smoothing and masking techniques on it to improve the baseline system. Our approach is motivated by the success of time series modeling techniques such as hidden Markov models and conditional random fields which perform better as compared to static classification techniques such as Gaussian mixture models and maximum entropy models [9, 10, 11]. Our overall system utilizes framewise probabilities for each clip obtained from the baseline system and processes them to use temporal context for that frame and, notably aims to reduce the false alarm rate. We incorporate temporal context by low-pass filtering the time series followed by a stacked generalization [12] framework on the frame probabilities. We then mask the time series based on the properties of the obtained smoothed time series as well as the state occupation probabilities as output from an automatic speech recognition (ASR) system to identify potential regions of laughter and fillers. We obtain an AUC of 93.3% (baseline: 82.9%) for laughter detection and 89.7% (baseline: 83.6%) for filler detection on the test set.

We next describe the database and the baseline system in section 2 followed by our methodology for detection in section 3. We present the results in section 4 and our conclusions in section 5.

2. Database and baseline system

The Social Signals Sub-Challenge uses the “SSPNet Vocalization Corpus” (SVC) for the detection of laughter and fillers. It consists of 2763 audio clips. Each clip is 11 seconds long and contains at least one laughter or a filler between 1.5 seconds and 9.5 seconds. The data is manually segmented into laughter, fillers and garbage. 141 frame-wise features including MFCCs, F0, voicing probabilities etc. are extracted using openSMILE [13] for each clip at 100 frames per second.

The baseline system uses an SVM classifier on all the features. We replace this SVM classifier by a deep neural network (DNN) classifier [14, 15] as explained in the next section. Output probabilities for the $t\text{-}th$ frame $P_t(E)$ for an event $E \in \{\text{laughter, filler}\}$ are calculated using this DNN classifier. The garbage class consisting of all other speech and silence is downsampled by a factor of 20 to maintain a class balance during training. Whereas this form of training is likely to increase the detection rate of the event, the false alarm rate is also likely to increase. Further details of the database and the baseline system can be found in [6].

3. Detection system

Our detection system comprises two units that process the output from the DNN classifier. We train a four layer DNN with 141 input layer neurons, two hidden layers with 47 neurons each and 3 neurons in the output layer. The number of neurons in the hidden layers is tuned on the development set.

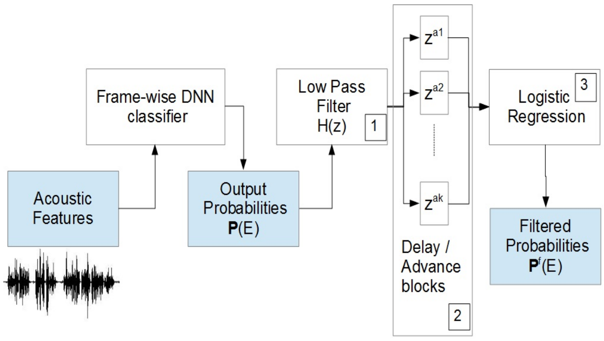

Figure 1: Time series smoothing setup as applied on baseline system

We define a time series of probability outputs by the DNN as $\mathbf{P}(E) = \{P_1(E), P_2(E), ..., P_{1100}(E)\}$ for an event E for each clip consisting of 1100 frames. We use time series filtering followed by masking to obtain our final probability predictions for a given clip. Time series smoothing is aimed at smoothing-out the probability signal and using temporal context to make a prediction. Time series masking is used to reduce the frequent false alarms introduced by the DNN classifier. We describe each individual unit in the following subsections.

3.1. Time series smoothing

In this scheme, we use the raw probability time series $\mathbf{P}(E)$ as output by the DNN system. This scheme initially removes the inherent noise in the probability signal due to frame-wise classification. We filter the probability signal using a low pass filter (block 1 in figure 1). We use a low pass filter $G(z)$ of order n cascaded with itself p times (equation 1). n determines the length of filter and p the sharpness of filter response around the poles. These parameters are tuned on the development set.

$$\text{low pass filter}: G(z) = \frac{1}{(\sum\limits_{i=0}^{n} z^i)^p} \tag{1}$$

We next capture the information from the surrounding frames to obtain smoothed probability estimates. These events happen over several frames and the decision on one frame should be dependent on its neighbors. The filtered output probabilities as obtained above are shifted as shown in block 2 of figure 1 to obtain probabilities for the surrounding frames. $\{a1, a2, ..., ak\} \in Z$ represent the shift introduced by each of the shift units. The output from these advance/delay blocks are then fed to a logistic regression model to obtain the final smooth probabilities. We chose 3 shift blocks with $\{a1$=-2, $a2$=0, $a3$=2$\}$ to make a decision for each frame. Using closer frames led to highly correlated features while training logistic regression, leading to numerical problems. Also, using a higher number of frames did not improve the performance. We obtain the smoothed probability time series $\mathbf{P^s} = \{P_1^s(E), P_2^s(E), ..., P_{1100}^s(E)\}$ after this operation. Figure 2 shows the plot of $\mathbf{P}$, intermediate filtered probability and $\mathbf{P^s}$ for one clip in the training set containing laughter. As can be seen, initial filtering removes the noise in the times series. Then the application of logistic regression further increases the probability values for regions corresponding to the event while at the same time attenuating the event probability for frames not corresponding to the event.

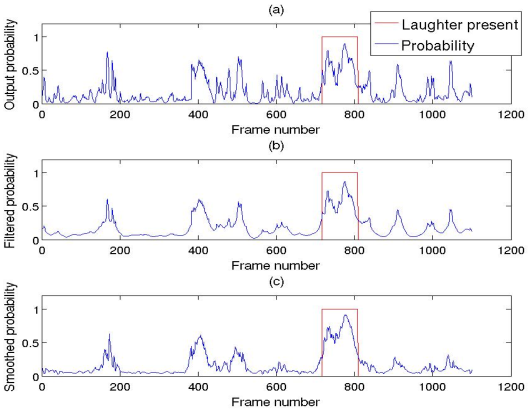

Figure 2: Time series of probabilities: (a) baseline (b) after filtering (c) smoothed

3.2. Time series masking

We use several techniques to reduce the false alarm rate during detection. The baseline model is more prone to false alarms as compared to missed detection as the majority class is down-sampled. We define masking of a probability as multiplying the frame probability $P_t(E)$ by a constant θ_t, $0 \leq \theta_t \leq 1$ based on certain criteria. We define these criteria based on smoothed probability time series as well as the state occupancy probabilities from an ASR system. We discuss these masking schemes below.

3.2.1. Masking based on smoothed probability time series

This masking scheme is based on our empirical observations from $\mathbf{P^s}$. We observe that in the regions of laughter and filler in a clip, at least one of the frames has a high probability value. This suggests that at least one frame during an event is confidently classified by the DNN system. We utilize this information by designing a mask as shown in equation 2. This scheme preserves the probability time series around the high probability frames and sets the ones with $P_t(E)$ less than a threshold T_1 to 0. We obtain the modified time series $P_t^{m1}(E)$ as shown in equation 3. N_1 determines the length of the window to be preserved and θ_t^1 is the multiplication factor for the t^{th} frame. S_H is the set of frames $\{t_{h1}, t_{h2}, ..t_{hi}, ..\}$ with $P_{t_{hi}}(E) > T_2$. We tune the parameters T_1, T_2 and N_1 on the development set.

$$\theta_t^1 = \begin{cases} 0, & \text{if } \min\limits_{t_h \in S_H} |t - t_h| > N_1 \text{ \& } P_t^s(E) < T_1 \\ 1, & otherwise \end{cases} \tag{2}$$

$$P_t^{m1}(E) = \theta_t^1 \times P_t^s(E) \tag{3}$$

We further mask frames that correspond to an event happening for extremely short time. If two non-zero frames are very close to each other after the previous operation, we set all the event probabilities between those frames to zero . It is unlikely that these events last only for a few frames and hence all such frames can be disregarded as not belonging to the event. The mask is shown in equation 4 and the modified time series in 5. S_0 is the set of frames $\{t_{01}, t_{02}, .., t_{0i}, ..\}$with $P_{t_{0i}}^{m1}(E) = 0$. N_2 determines the maximum length between two frames in set

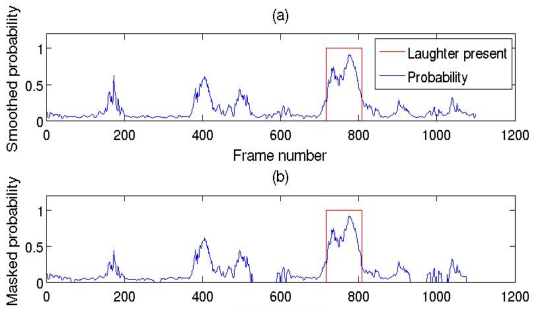

Figure 3: Time series of probabilities: (a) smoothed (b) masked

S_0 and T_3 ($> T_1$) gives an upper bound on $P^{m1}_{t_{0i}}(E)$ to use the mask.

$$\theta^2_t = \begin{cases} 0, & \text{if } \min\limits_{t_{0l} \in S_0} |t - t_{0(l-1)}| + |t_{0l} - t| < N_2 \\ & \& \quad P^{m1}_t(E) < T_3 \\ 1, & otherwise \end{cases} \tag{4}$$

$$P^{m2}_t(E) = \theta^2_t \times P^{m1}_t(E) \tag{5}$$

Finally, in order to improve the AUC, we pull up all the probabilities $P^{m2}_t(E)$ above a threshold T_4 to 1 (equation 6). We tune this threshold on the development set.

$$P^{m3}_t(E) = \begin{cases} 1, & \text{if } P^{m2}_t(E) > T_4 \\ P^{m2}_t(E), & \text{otherwise} \end{cases} \tag{6}$$

3.2.2. Masking based on state occupancy probabilities of an automatic speech recognition (ASR) system

We train an automatic speech recognition (ASR) system with the Kaldi toolkit [16] using a three item vocabulary - {laughter, filler, garbage}. We parameterize the audio signal using Mel frequency cepstral coefficients (MFCCs) over 25 msec frames with 10 msec shift. We then train monophone (context-independent) models using the Viterbi-EM algorithm. These models then align the input audio and the resulting alignments are used to train triphone (context-dependent) models. We use these models with a trigram language model to decode the development set audio files and generate word lattices. We compute the state occupancy probabilities P^{state}_t for the t^{th} frame using the forward-backward algorithm [9] on the lattices. We thus obtain a probability mass function over states for every frame and compute its entropy H_t (equation 7). We intuitively expect this entropy to be high in frames where the ASR is confused between many competing states and low when the ASR system is more confident.

$$H_t = - \sum_{\text{all states}} P^{state}_t \times log(P^{state}_t) \tag{7}$$

We observe that the entropy for frames corresponding to laughter is higher where as for the fillers, the entropy is low. Laughter has been hypothesized as composed of vowel like bursts [8, 17] with silences in between these bursts. A higher entropy for laughter hints towards a higher confusion during decoding possibly due to this heterogeneous nature of laughter. On the other hand, fillers like “em”, “um”, “ah” while training can be represented by fewer number of states as they are more or less acoustically homogeneous. Thus while decoding, only a few states are likely to be occupied during decoding and finding n-best paths. This leads to a lower entropy for fillers frames. We define different masks for laughter and filler time series as defined in equation 8 and 10 respectively, utilizing the above properties. A plot for the same clip as in figure 2 after the masking schemes is shown in figure 3.

$$\theta^l_t = \begin{cases} 1, & \text{if } \min\limits_{t_h \in S_H} |t - t_h| < N \;\&\; H_t > T_4 \\ k_l < 1, & otherwise \end{cases} \tag{8}$$

$$P^m_t(E) = \theta^l_t \times P^{m3}_t(E) \text{ for } E = \text{laughter} \tag{9}$$

$$\theta^f_t = \begin{cases} k_f < 1, & \text{if } H_t > T_5 \\ 1, & otherwise \end{cases} \tag{10}$$

$$P^m_t(E) = \theta^f_t \times P^{m3}_t(E) \text{ for } E = \text{filler} \tag{11}$$

4. Results and Discussion

The results are given in table 1 for the various suggested algorithmic improvements on the development set followed by the final results on the test set. We observe an increase in the AUC for both laughter and fillers after implementing each proposed modeling step. We achieve an overall increment of 8.2% in unweighted AUC (UWAUC) for the two events of interest on the test set. The majority of the improvement appears to come from the DNN system and probability time series smoothing. ASR system entropy based masking improves AUCs for both laughter and filler, however this is not significant in case of the fillers.

System		AUC		UWAUC
		Laughter	Filler	
Baseline system		86.2	89.0	87.6
DNN system		90.1	90.1	90.1
Probability smoothing		94.6	94.4	94.5
Probability Masking	Prob.time series based	94.8	94.7	94.8
	ASR system based	95.1	94.7	94.9
Test Set				
Baseline system		82.9	83.6	83.3
Our system		93.3	89.7	91.5

Table 1: AUC for laughter and fillers on the development and test set

We plot the histograms for normalized probability counts NC_i for the development set as defined in equation 13. In Equation 12, $\mathbf{P}^*(E|T)$ represents a time series downsampled from $\mathbf{P}^*(E)$ such that we only retain the frames probabilities $P^*_t(E|T)$ with $T \in$ {laughter, filler, garbage} being the true class for all the downsampled frames. $C_i(E|T)$ gives the count (#) of frames in $\mathbf{P}^*(E|T)$ with $P^*_t(E|T) \in [b_i, b_{i+1}]$ $0 \leq b_i \leq b_{i+1} \leq 1$. This is just a histogram count. We sum up $C_i(E|T)$ from all the clips in the development set and normalize by the total number of frames corresponding to the true class T, thereby making it a probability distribution. This probability distribution gives an empirical estimate for the probability density functions (PDF) for the false alarm rate and true positive rate of an event. An AUC of 100% corresponds to PDF of false alarm rate to be a delta function at zero and true positive rate a

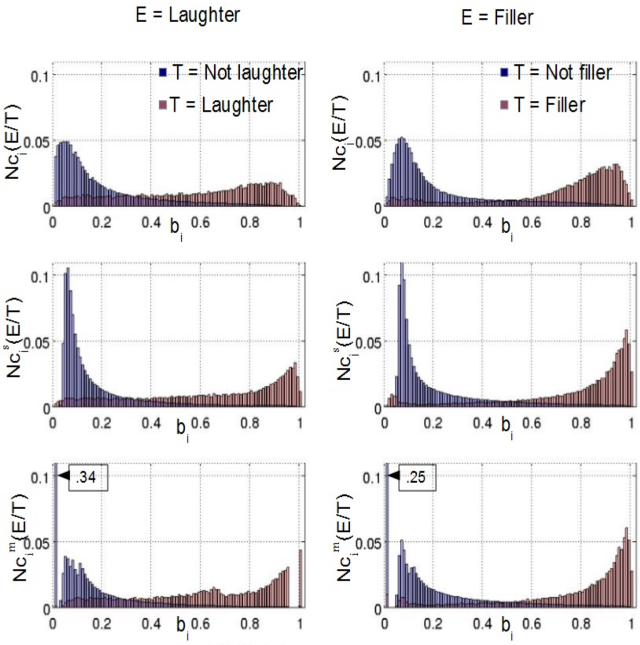

Figure 4: $NC_i^*(E)$ plots for laughter and filler events

delta function at one. Figure 4 gives the normalized counts for $\mathbf{P}^*(E) = \mathbf{P}(E), \mathbf{P}^s(E)$ and $\mathbf{P}^m(E)$. Figure 5 plots the receiver operating characteristic for the baseline system, the DNN system and after masking+smoothing systems.

$$C_i^*(E|T) = \#(b_i < P_t^*(E|T) < b_{i+1}) \quad \forall\, P_t^*(E|T) \in \mathbf{P}^*(E|T) \tag{12}$$

$$NC_i^*(E|T) = \sum_{\text{dev set}} \frac{C_i^*(E|T)}{\sum_{i=1}^{K} C_i^*(E|T)} \tag{13}$$

4.1. Discussion

The histograms in figure 4 show that each of our proposed ideas brings the raw time series of probabilities obtained from the baseline system closer to the ideal distribution (delta function at 0 for false alarm rate and delta function at 1 for true positive rate). The time series filtering shifts the PDFs such that there is a better demarcation between the false alarm and true positive case where as the masking helps us in assigning a good portion of PDF to 0 for false alarms. However, it is not as effective in assigning PDF for true positives to 1. The AUC curves (figure 5) reflect that whereas each subsequent AUC curve is better than the previous one, the biggest jump is achieved using the smoothing and masking system. A substantial gain is also seen using the DNN system for the laughter class. The masking schemes give us advantage particularly at the top right corner of the AUC curves, due to the fact that a large portion of false positive PDF is assigned to 0. This also suggests that the masking system is successful in reducing the false alarm rates particularly when the detection threshold is chosen close to 0.

Through our experiments we expect that improvement due to temporal context advocates for more investigation into using information from neighboring frames both while extracting features as well as fusing after obtaining a time series of the probabilities. In the future we plan on incorporating more features during masking. Our ASR is initially trained based on force alignment on the training data which can be improved by using the true time labels. Additionally, other ASR based confidence scores [18] can be used to further improve the masking scheme.

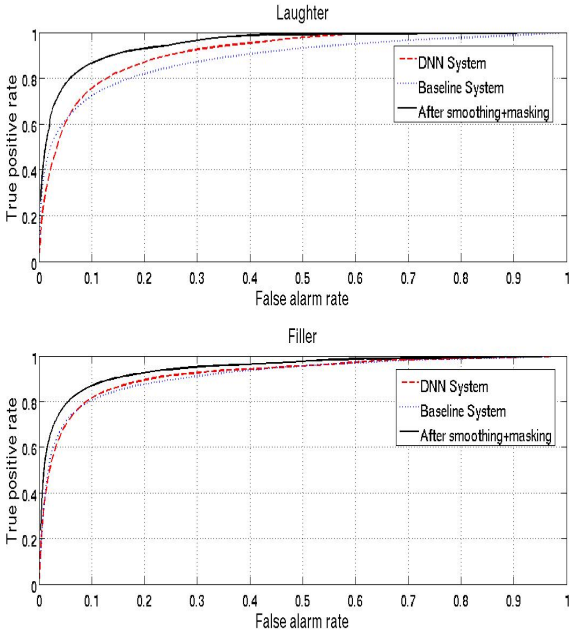

Figure 5: ROC curves for laughter and filler detection

5. Conclusion

In this paper, we improve upon the AUC baselines set in the Interspeech 2013 Social Signals Sub-Challenge for detection of laughter and fillers. We use filtering and masking techniques on probabilistic time series representing the occurrence of a laughter or a filler. We observe that, while most of the improvement comes from using temporal context, masking also helps us reduce the false alarm rates.

As an extension to this work, we would like to study the association of these vocal events with higher behavioral predicates such as emotions, distress, engagement etc.. Studies have suggested a correlation between non-verbal speech cues and these behavioral predicates. A quantification of those based on nonverbal cues can give us a deeper insight into human communication.

6. Acknowledgment

This research is supported by NSF, NIH and DARPA.

7. References

[1] Michael Argyle, Veronica Salter, Hilary Nicholson, Marylin Williams, and Philip Burgess, “The communication of inferior and superior attitudes by verbal and nonverbal signals*,” *British journal of social and clinical psychology*, vol. 9, no. 3, pp. 222–231, 2011.

[2] Malcolm Kahn, “Non-verbal communication and marital satisfaction,” *Family Process*, vol. 9, no. 4, pp. 449–456, 2004.

[3] Margaret Procyk Creedon, “Language development in

nonverbal autistic children using a simultaneous communication system.," 1973.

[4] John Morreall, *Taking laughter seriously*, State University of New York Press, 1983.

[5] Mark Van Vugt, Charlie Hardy, Julie Stow, and Robin Dunbar, "Laughter as social lubricant: A biosocial hypothesis about the functions of laughter and humour," *Unpublished manuscript*, 2007.

[6] Björn Schuller, Stefan Steidl, Anton Batliner, Alessandro Vinciarelli, Klaus Scherer, Fabien Ringeval, Mohamed Chetouani, Felix Weninger, Florian Eyben, Erik Marchi, et al., "The interspeech 2013 computational paralinguistics challenge: Social signals, conflict, emotion, autism," *Proc. Interspeech*, 2013.

[7] Herbert H Clark and Jean E Fox Tree, "Using uh and um in spontaneous speaking," *Cognition*, vol. 84, no. 1, pp. 73–111, 2002.

[8] Jo-Anne Bachorowski, Moria J Smoski, and Michael J Owren, "The acoustic features of human laughter," *The Journal of the Acoustical Society of America*, vol. 110, pp. 1581, 2001.

[9] Lawrence Rabiner and Bing-Hwang Juang, "An introduction to hidden Markov models," *ASSP Magazine, IEEE*, vol. 3, no. 1, pp. 4–16, 1986.

[10] Lawrence R Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[11] John Lafferty, Andrew McCallum, and Fernando CN Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," 2001.

[12] David H Wolpert, "Stacked generalization," *Neural networks*, vol. 5, no. 2, pp. 241–259, 1992.

[13] Florian Eyben, Martin Wöllmer, and Björn Schuller, "Opensmile: the munich versatile and fast open-source audio feature extractor," in *Proceedings of the international conference on Multimedia*. ACM, 2010, pp. 1459–1462.

[14] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[15] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[16] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukáš Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlıcek, Yanmin Qian, Petr Schwarz, et al., "The Kaldi speech recognition toolkit," in *Proc. ASRU*, 2011.

[17] Donald E Mowrer, Leonard L LaPointe, and James Case, "Analysis of five acoustic correlates of laughter," *Journal of Nonverbal Behavior*, vol. 11, no. 3, pp. 191–199, 1987.

[18] Hui Jiang, "Confidence measures for speech recognition: A survey," *Speech communication*, vol. 45, no. 4, pp. 455–470, 2005.