KEYWORD-CONDITIONED PHONE N-GRAM MODELING WITH CONTEXTUAL INFORMATION FOR SPEAKER VERIFICATION

Kyu J. Han, Jason Pelecanos, Mohamed K. Omar

IBM T. J. Watson Research Center
Yorktown Heights, NY 10598, USA
{kjhan,jwpeleca,mkomar}@us.ibm.com

ABSTRACT

In this paper we present our current work on automatic speaker recognition using keyword-conditioned phone N-gram modeling. We propose the use of contextual information around keywords in modeling a speaker's pronunciation characteristics at a phonetic level. Our approach is to add time margins around keywords when aligning keyword regions with keyword-specific phone events for feature vector generation. Including such additional information by incorporating time margins can capture idiosyncratic pronunciation information and is shown to help our keyword-conditioned phonetic speaker verification system achieve more than 50% (relative) performance improvement. This leads our high-level speaker verification system (i.e., fusion of non-conditioned and keyword-conditioned phonetic speaker verification systems) to currently achieve the best published result for the English 8-conversation enrollment telephony task of the 2008 NIST Speaker Recognition Evaluation for systems utilizing features not based directly on low-level acoustic information.

Index Terms— Speaker verification, keyword-conditioned phone N-gram modeling, contextual information, time margin

1. INTRODUCTION

For the past decade, there is increased research effort in the speaker recognition community on modeling speaker characteristics at a higher level (e.g., phoneme or word) to complement state-of-the-art speaker verification systems based on low-level acoustic features like Mel-Frequency Cepstral Coefficients (MFCCs). Doddington's work [1] is an example that showed the potential of exploiting high-level idiolectal information using word sequences for speaker recognition. It was shown (in [1]) that idiosyncratic speech patterns like 'you bet' or 'how shall' could be used to recognize familiar speakers. Inspired by this work, Andrews [2] developed a phonetic speaker verification framework to make use of idiolectal phone sequences by including relative frequencies for pruned phone N-grams as input features to a speaker detector based on calculating the log-likelihood ratio. This framework was further extended by Campbell [3], where Support Vector Machines (SVMs) were applied as classifiers to the phonetic speaker verification problem and the Term-Frequency Log-Likelihood Ratio (TFLLR) kernel was designed accordingly. Concurrently, a number of novel ideas were proposed to improve the accuracy of phonetic speaker verification, e.g., in terms of modeling [4], phone statistics [5], and parameter estimation [6].

More recently, Lei and Mirghafori [7] initiated the work on keyword conditioning for phonetic speaker verification. They empirically showed that it could further enhance phonetic speaker verification performance by processing the phone N-grams corresponding to selected words. It is noted that this work is somewhat related to word-conditioned phonetic modeling performed manually by some forensic phoneticians [8]. Motivated by this, we investigate keyword-conditioned phonetic speaker verification more in depth.

In this paper we propose the use of contextual information around keywords. The contextual information can provide extra information relating to idiolectal phone usage. To obtain such additional information, we apply time margins before and after keywords when conditioning phone events. This time margin framework is shown to not only improve the overall system performance significantly but to also yield a simplified system design compared to other approaches examined that utilize context near keywords.

The paper is structured as follows: in Section 2, we give a brief overview of our phonetic speaker verification system with keyword conditioning; Section 3 investigates how we can utilize contextual information and how it can improve system performance. In Section 4, we wrap up the paper by summarizing our findings.

2. KEYWORD-CONDITIONED PHONETIC SPEAKER VERIFICATION SYSTEM

Our keyword-conditioned phonetic speaker verification system processes the phone N-grams ($N = 1, 2,$ and 3) corresponding only to selected keywords rather than handling all phone N-grams for a given conversation side. From the ASR transcripts (obtained using IBM's English automatic speech recognizer) of a background conversation side set B, we select the 50 most frequent words as our keywords. We also choose the 2,000 most frequent phone events for each keyword as keyword-conditioned phone N-grams. For the background data set B, we use 7,736 English telephone conversation sides from the Switchboard-II corpus and the 2004/06

NIST Speaker Recognition Evaluation (SRE) data[1]. Given a conversation side, we concatenate the relative frequencies of keyword-conditioned phone N-grams across the keywords to generate a phone feature vector of 100,000 dimensions. The keyword-conditioned TFLLR kernel for SVM training and classification, which is adapted from [3], can thus be written as follows:

$$K(X,Y) = \sum_{w=1}^{W} \sum_{i=1}^{M} \frac{P(d_i^w|X)}{\sqrt{P(d_i^w|B)}} \frac{P(d_i^w|Y)}{\sqrt{P(d_i^w|B)}}, \quad (1)$$

where M is the number of the most frequent phone events per keyword, empirically chosen to be 2,000, W is the number of keywords (i.e., 50 in our case), d_i^w is the i^{th} phone event of the keyword-conditioned phone N-grams conditioned by the w^{th} keyword, and $P(d_i^w|\cdot)$ is the relative frequency of d_i^w in a given conversation side, i.e., $P(d_i^w|\cdot) = \frac{\#(d_i^w|\cdot)}{\sum_{u=1}^{W}\sum_{j=1}^{M}\#(d_j^u|\cdot)}$, where # is the soft count of d_i^w in the conversation side. X and Y are enrollment and test conversation sides. To obtain the soft count of each phone N-gram of interest, as calculated in [5], we utilize phone confusion networks, which are generated by using IBM's English phone recognizer and the SRILM toolkit [10]. An SVM classifier is implemented using the LIBSVM package [11] with a margin and classification error tradeoff of $c = 1$.

To obtain more reliable phone N-gram statistics for $P(d_i^w|X)$ and $P(d_i^w|Y)$, we apply the Maximum A Posteriori (MAP) adaptation technique [6, 12] and use these statistics in Eq. (1). We can view $P(d_i^w|\cdot)$ as a Maximum Likelihood (ML) estimate for d_i^w (whose statistics follow a multinomial distribution [6]), i.e., $P(d_i^w|\cdot) = P_{\text{ML}}(d_i^w|\cdot)$. Assuming a Dirichlet distribution as a conjugate prior, $P(d_i^w|\cdot)$ can be estimated in a MAP sense:

$$P_{\text{MAP}}(d_i^w|\cdot) = \frac{\#(d_i^w|\cdot) + \nu_i^w - 1}{\sum_{u=1}^{W}\sum_{j=1}^{M}\left[\#(d_j^u|\cdot) + \nu_j^u - 1\right]}, \quad (2)$$

where ν_i^w is the hyperparameter of the Dirichlet distribution corresponding to d_i^w. The τ-initialization scheme [13] is used to estimate ν_i^w,

$$\nu_i^w - 1 = \tau \cdot D \cdot W \cdot P(d_i^w|B), \quad (3)$$

where D is the dimension of a keyword-conditioned phone feature vector (i.e., 100,000) and τ is empirically chosen to be 0.0005.

3. CAPTURING CONTEXTUAL INFORMATION AROUND KEYWORDS

The basic idea of conditioning phone events by keyword is to compare pronunciation differences for a fixed set of words because such differences can be related to speaker identity. In spontaneous conversations, it is also idiosyncratic which words are spoken with certain keywords. For example, 'YOU' is followed by 'KNOW' to make a popular filler in American English conversations. Another example is 'I', which can be used with 'MEAN' very often for some speakers and may be used with 'KNOW' or 'JUST' for other speakers. Thus, it would be appropriate to consider contextual information around keywords as well when conditioning phone events for phonetic speaker verification. Fig. 1 illustrates how we can obtain contextual phonetic information around keywords. Instead of selecting the phone N-grams within the time boundaries of keywords, we apply *time margins* before and after keywords so that statistics for additional phone N-grams surrounding keywords can also be included in keyword-conditioned phone feature vectors.

Fig. 1. Illustration of how to obtain contextual phonetic information around keywords by applying time margins to keyword boundaries.

To indicate the utility of incorporating additional phonetic information from time margins, we ran a task from the NIST SRE '08 consisting of 8 English telephone conversation sides for enrollment and one English telephone conversation side as a testing example. We call this task *8conv-short3-Eng-Tel*. The Equal Error Rate (EER) and the minimum Detection Cost Function (minDCF)[2] are used to measure performance. The claim that including the time margin can help the system capture more speaker-relevant phonetic information per keyword and result in better speaker verification is empirically supported by Fig. 2. This figure shows how the performance of the keyword-conditioned phonetic speaker verification system changes as we change the time margin. From the figure we observe that the optimal performance is when a 0.3 second time margin is applied. The time margin of 0.3 seconds can be interpreted as being comparable to the average length of one English word based on the word duration statistics from the ASR transcripts of our background data set B. This means that by including a 0.3 second time margin we could capture the phone N-gram statistics approximately corresponding to one more word on either side of the keywords. The relative improvements of 57% in EER (from 11.2% to 4.8%) and 51.1% in minDCF $\times 10^3$ (from 54.6 down to 26.7) coming from the time margin of 0.3 seconds compared to the performance without any time margin shows the importance of applying time margins when generating phone feature vec-

[1]This data set is also used for both Nuisance Attribute Projection (NAP) [9] and SVM training (as negative examples).

[2]For further information regarding the evaluation metrics, please refer to [14].

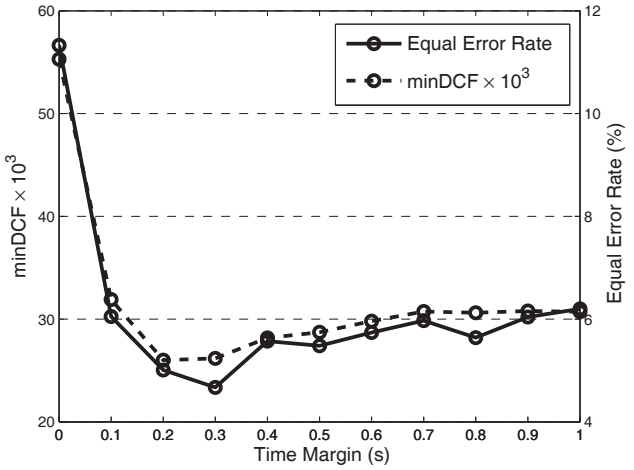

Fig. 2. Performance of the keyword-conditioned phonetic speaker verification system (with 50 keywords used) along with the time margins applied before and after the keywords. The task is 8conv-short3-Eng-Tel in the NIST SRE '08.

tors in the keyword-conditioned phonetic speaker verification system.

Applying a time margin around keywords can provide the system with further opportunity to capture idiolectal word and phone usage. Table 1 lists the most frequent phone trigrams for 10 chosen keywords (among 50) in both of the cases with and without the time margin of 0.3 seconds being applied to keyword-conditioned phone feature vector generation. Without the time margin, the most frequent phone trigrams obtained by keyword conditioning are nothing but the phone events corresponding to the given keywords. (In the middle column, we can observe that most of the trigrams listed represent the phone sequence events matching with the corresponding keywords in the pronunciation.) Utilizing the time margin, however, changes the most frequent keyword-conditioned phone trigrams for some words, e.g., 'I', 'YOU', 'OF' and 'IS'. For these words, after the time margin is applied, the most frequent phone trigrams are changed to 'M-IY-N', 'Y-UW-N', 'L-AA-T', and 'T-IH-Z', which is not surprising because it is well known that those words are mostly used in a characteristic form in spontaneous English conversations such as 'I mean...', 'you know', 'a lot of', and 'it is...'. By capturing additional content from a given conversation side, our keyword-conditioned phonetic speaker verification system with time margins can expand unigram keywords to capture information from bigram or even trigram keywords conceptually. This results in a huge boost to system performance, as was shown in Fig. 2.

Our time margin framework is compared in Table 2 with systems using bigram or trigram keywords without time margins. The main issue in using multi-gram keywords in a direct way is sparsity. Since they do not occur very often like unigram keywords in conversation sides, keyword-conditioned phone feature vectors in the systems with bigram or trigram keywords become sparse. Although smoothing helps compensate phone relative frequency statistics to some degree in the experiment, the two systems do not have the performance comparable with the proposed framework with the time margin of 0.3 seconds. While we can try to mix unigram, bigram, and trigram words manually to optimize the system performance, a related issue is how to optimally select such a mix. In contrast, the time margin framework utilizes information from multiple words without the severe sparsity issues.

Table 3 compares the time margin framework with systems explicitly assigning neighboring words around keywords. It is shown from the table that the time margin framework seems a better approach than its counterparts with regard to selection of contextual information. Although its performance in minDCF is not dramatically better than those for 1 or 2 more words used around keywords as contextual information, it can provide a noticeable benefit in terms of fusion[3] with the acoustic baseline. Its relative improvement is 10% while the improvement of the others is limited to less than 7%. One of reasons for this discrepancy is that the system utilizing the fixed number of words to capture contextual information around keywords would be susceptible to the case where there is a long pause or silence between a given keyword and its most adjacent word on either side. This could cause the system to obtain statistics that may be

Table 1. List of the most frequent phone trigrams for 10 chosen keywords in the cases with and without the time margin of 0.3 seconds. X: silence.

Keyword (Rank)	Most Frequent Phone Trigram	
	w/o Time Margin	w/ Time Margin
I (1)	X-X-AY	M-IY-N
YOU (2)	X-Y-UW	Y-UW-N
KNOW (7)	N-OW-OW	Y-UW-N
LIKE (8)	L-AY-K	L-AY-K
IT (9)	IH-T-X	IH-T-IH
UM (13)	AH-M-X	AH-M-X
OF (15)	AH-V-X	L-AA-T
IS (20)	IH-Z-S	T-IH-Z
ALL (46)	X-AO-L	AO-L-DH
GO (49)	G-OW-OW	G-OW-T

Table 2. Comparison between the time margin framework (0.3 second time margin) and systems directly using bigram or trigram keywords without time margins.

	Proposed	Bigram	Trigram
minDCF ($\times 10^3$)	26.7	83.9	99.2

[3]For optimized fusion weight selection, we utilize a held out data set of 1,600 conversation sides from the Switchboard-II corpus, which is separate from the data used for the background data set B.

Table 3. Comparison (minDCF $\times 10^3$) between our proposed framework with the time margin of 0.3 seconds and systems using more words around keywords. For fusion (with optimized weighting), an acoustic baseline using MFCCs is used. The acoustic system has NAP and ZT score normalization applied. The task is 8conv-short3-Eng-Tel in the NIST SRE '08.

System	Acoustic	Time Margin	Word Margin		
			1	2	3
Individual	5.7	26.7	27.7	27.9	30.4
Fused	-	5.1	5.3	5.4	5.6

Table 4. Comparison of individual and fused systems. ASV: Acoustic speaker verification using MFCCs, PSV: Non-conditioned phonetic speaker verification, and KW-PSV: Keyword-conditioned PSV.

System	minDCF ($\times 10^3$)
PSV	23.6
KW-PSV	26.7
PSV + KW-PSV	20.4
ASV (ZT-Norm Applied)	5.7
ASV + PSV	5.4
ASV + KW-PSV	5.1
ASV + PSV + KW-PSV	5.1

less relevant. From Tables 2 and 3 we can claim that our approach of adding time margins to keyword boundaries in phone feature vector extraction for phonetic speaker verification provides a simplified system design as well as captures more speaker-relevant information.

Table 4 presents a comparison of the time-margin framework and the non-conditioned phonetic system in terms of score fusion with the acoustic baseline. Note that the combined performance (20.4) of the two phonetic systems is the best published result on 8conv-short3-Eng-Tel in the NIST SRE '08 among systems using high-level information other than acoustic features. We observe that the keyword-conditioned phonetic system (with the time margin of 0.3 seconds) provides a 10% (relative) improvement to the acoustic baseline through fusion, which is almost doubled compared to the 5.3% (relative) improvement from the non-conditioned system.

4. CONCLUSIONS

In this paper, we proposed the use of contextual information around keywords for phone feature vector generation in the framework of phonetic speaker verification. By applying a time margin of 0.3 seconds before and after 50 unigram keywords when computing the relative frequencies of phone events of interest, we achieved an improvement of 51.1% (relative) in minDCF. We also obtained a fusion benefit from the proposed approach when we combined it with our MFCC acoustic baseline. This study not only highlighted the importance of proper constraints in phone feature vector generation but also underscored the importance of incorporating context around keywords.

5. ACKNOWLEDGEMENT

This research was funded by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), through the Army Research Laboratory (ARL). All statements of fact, opinion or conclusions contained herein are those of the authors and should not be construed as representing the official views or policies of IARPA, the ODNI, or the U.S. Government.

6. REFERENCES

[1] G. Doddington, "Speaker recognition based on idiolectal differences between speakers," in *Interspeech*, 2001, pp. 2521–2524.

[2] W. Andrews, M. Kohler, J. Campbell, and J. Godfrey, "Phonetic, idiolectal, and acoustic speaker recognition," in *Speaker Odyssey*, 2001, pp. 55–63.

[3] W. Campbell, J. Campbell, D. Reynolds, and T. Leek, "Phonetic speaker recognition with support vector machines," in *NIPS*, 2003, pp. 1377–1384.

[4] J. Navratil, Q. Jin, W. Andrews, and J. Campbell, "Phonetic speaker recognition using maximum-likelihood binary-decision tree modeling," in *ICASSP*, pp. 796–799.

[5] A. Hatch, B. Peskin, and A. Stolcke, "Improved phonetic speaker recognition using lattice decoding," in *ICASSP*, 2005, pp. 169–172.

[6] B. Baker, R. Vogt, M. Mason, and S. Sridharan, "Improved phonetic and lexical speaker recognition through MAP adaptation," in *Speaker Odyssey*, 2004, pp. 91–96.

[7] H. Lei and N. Mirghafori, "Word-conditioned phone N-grams for speaker recognition," in *ICASSP*, 2007, pp. 253–256.

[8] R. Schwartz, W. Shen, J. Campbell, S. Paget, J. Vonwiller, D. Estival, and C. Cieri, "Construction of a phonotactic dialect corpus using semiautomatic annotation," in *Interspeech*, 2007.

[9] W. Campbell, "Compensating for mismatch in high-level speaker recognition," in *Speaker Odyssey*, 2006, pp. 1–6.

[10] A. Stolcke, "SRILM - An extensible language modeling toolkit," in *Interspeech*, 2002.

[11] C. Chang and C. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011.

[12] J. Gauvain and C. Lee, "Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains," *IEEE Transactions on Speech and Audio Processing*, vol. 2, pp. 291–298, 1994.

[13] C. Lee and Q. Huo, "On adaptive decision rules and decision parameter adaptation for automatic speech recognition," *Proceedings of the IEEE*, vol. 88, no. 8, pp. 1241–1269, 2000.

[14] The National Institute of Standards and Technology (NIST), "The NIST year 2008 speaker recognition evaluation plan," http://www.itl.nist.gov/iad/mig/tests/sre/2008/sre08_evalplan_release4.pdf, 2008.