論文 / 著書情報
Article / Book Information

Title	Stream-weight optimization by LDA and adaboost for multi-stream speaker verification
Authors	Taichi Asami, Koji Iwano, Sadaoki Furui
Citation	Interspeech2005, Vol. , No. , pp. 2185-2188,
Pub. date	2005, 9
Copyright	(c) 2005 International Speech Communication Association, ISCA
DOI	http://dx.doi.org/

Stream-Weight Optimization by LDA and Adaboost for Multi-Stream Speaker Verification

Taichi Asami, Koji Iwano and Sadaoki Furui

Department of Computer Science, Tokyo Institute of Technology
2-12-1 Ookayama, Meguro-ku, Tokyo, 152-8552 Japan
{taichi, iwano, furui}@furui.cs.titech.ac.jp

Abstract

This paper proposes an automatic stream-weight optimization method for noise-robust speaker verification using multi-stream HMMs integrating spectral and prosodic information. The paper first shows the effectiveness of the multi-stream technique in our speaker verification framework. Next, a stream-weight adaptation method combining the linear discriminant analysis (LDA) and Adaboost techniques is proposed. Experiments were conducted using four-connected-digit utterances of Japanese contaminated by white noise with various SNRs. Experimental results show that 1) the verification performance was improved in all SNR conditions by using stream weights estimated by the LDA and 2) the performance is further improved by using the Adaboost in 10 - 30dB SNR conditions.

1. Introduction

Increasing noise-robustness is a major challenge in constructing practical speaker verification systems. Here, we investigate one of the methods increasing noise-robustness of speaker verification, the method using multi-stream HMMs for integrating some noise-robust features and spectral features. We have shown that verification performance can be increased in noisy environments by combining fundamental frequency (F_0) related features as prosodic information, which is extracted by the noise-robust method based on the Hough transform [1] with spectral (segmental) features. In our previous experiments, stream-weights of multi-stream HMMs were manually optimized. To construct practical systems, the stream-weights need to be automatically optimized according to the environments where the systems are used.

In this paper, we propose an automatic stream-weight optimization method using a development set, based on the combination of the LDA and Adaboost [2] techniques. The Adaboost method has a capability of constructing a high performance classifier by combining multiple simple classifiers, and it has been reported that the Adaboost is effective in improving the performance of speech recognition [3-6]. Specifically, [4] showed that performance was improved by applying the Adaboost to multimodal speech recognition using multiple features.

This paper is organized as follows. Section 2 explains a speaker verification method using multi-stream HMMs integrating segmental and prosodic information. In Section 3, our stream-weight optimization method based on the LDA and the Adaboost is explained. Experimental results are presented in Section 4, and Section 5 concludes this paper.

2. Speaker verification using multi-stream HMMs

Our speaker verification method [1] integrates segmental and prosodic information using multi-stream HMMs. The strategy for integration will be explained and its effectiveness shown in this section.

2.1. Integration of segmental and prosodic features

Each segmental feature vector has 25 elements consisting of 12 MFCC, their deltas and the delta log energy. The window length is 25ms and the frame interval is 10ms. Cepstral mean subtraction (CMS) is applied to each utterance.

Prosodic feature vectors consist of $\log F_0$ and $\Delta \log F_0$. F_0 is extracted by the noise-robust method based on the Hough transform [7]. The segmental and prosodic feature vectors are combined at each frame to build a segmental-prosodic feature vector.

2.2. Integration of segmental and prosodic models

The proposed method was evaluated using four-connected-digit speech in Japanese. Since timing of the change of F_0 transitions, such as "rising" and "falling", is highly related to that of CV syllable transitions in Japanese connected digit speech, segmental and prosodic features are integrated in our method using syllabic unit HMMs. The integrated syllable HMM is denoted by "SP-HMM (Segmental-Prosodic HMM)".

In order to make SP-HMMs, S-HMMs (Segmental HMMs) and P-HMMs (Prosodic HMMs) are first trained separately by segmental and prosodic features. Then, the S-HMMs and the P-HMMs are combined to construct SP-HMMs. Gaussian mixtures in the segmental stream of SP-HMMs are tied with corresponding S-HMM mixtures, while the mixtures in the prosodic stream are tied with corresponding P-HMM mixtures.

2.3. Multi-steam modeling

SP-HMMs are modeled as multi-stream HMMs. In recognition, the probability $b_j(\boldsymbol{O}_{sp})$ of generating segmental-prosodic observation $\boldsymbol{O}_{sp}$ at state j is calculated by:

$$b_j(\boldsymbol{O}_{sp}) = b_j(\boldsymbol{O}_s)^{\lambda_s} \cdot b_j(\boldsymbol{O}_p)^{\lambda_p} \quad (1)$$

where $b_j(\boldsymbol{O}_s)$ is the probability of generating segmental feature vectors $\boldsymbol{O}_s$, and $b_j(\boldsymbol{O}_p)$ is the probability of generating prosodic feature vectors $\boldsymbol{O}_p$. λ_s and λ_p are weighting factors for the segmental and prosodic streams, respectively. They are constrained by $\lambda_s + \lambda_p = 1$ $(0 \leq \lambda_s, \lambda_p \leq 1)$.

Table 1: Comparison of the EERs (%) of the case where the S-HMMs and P-HMMs are separately used with the case where the SP-HMMs are used.

SNR (dB)	S-HMM & P-HMM	SP-HMM
30	0.67	0.65
20	3.46	3.36
15	9.62	9.02
10	17.91	16.28
5	25.62	24.32

2.4. Verification score

The verification score after observing a feature set O is denoted by $q(O)$, which is calculated as

$$q(O) = \log p(O|S^c) - \log p(O|S^g) \quad (2)$$

where $p(O|S^c)$ is a likelihood value with claimed speaker's SP-HMM and $p(O|S^g)$ is a likelihood value with general speaker's SP-HMM.

If the score is larger than a threshold value θ, the speaker is accepted as the claimed speaker. Therefore, the discriminant function is $z = q(O) - \theta$. If z is positive, the speaker is accepted, and if less than or equal to 0, the speaker is rejected as being an imposter.

2.5. Effectiveness of multi-stream HMMs

We first investigated the effectiveness of using multi-stream HMMs to integrate segmental and prosodic information. The EERs (equal error rates) of the following two cases were compared:

(a) The case where the $q_{sp}(O_{sp})$ obtained from claimed and general speaker's SP-HMMs is used as the verification score.

(b) The case where $\omega_s q_s(O_s) + \omega_p q_p(O_p)$ which is the weighted sum of $q_s(O_s)$ and $q_p(O_p)$ obtained separately from S-HMMs and P-HMMs is used as the verification score.

The subscript m of q_m represents the model from which the score is calculated, S-HMMs, P-HMMs or SP-HMMs. In this experiment, m was either s, p or sp. (a) and (b) in Figure 1 show the flow charts of the speaker verification process in each case. The integration weights, ω_s and ω_p, and the stream-weights, λ_s and λ_p, were manually optimized from 0.0 to 1.0 at 0.1 intervals. The data used for this experiment is the same as that for the verification experiment described in section 4.

The results are shown in Table 1. In all SNR conditions, the EERs of the case where feature level integration was implemented by SP-HMMs were lower than that of the case where the scores obtained separately from S-HMMs and P-HMMs were integrated afterward. These results show that the speaker verification method using multi-stream HMMs for integrating segmental and prosodic information is effective. Our previous research on speech recognition [7] showed that using multi-stream HMMs which integrated segmental and prosodic features yielded better time alignment of digits in noisy conditions. Probably this is also the reason why EERs were decreased by using SP-HMMs in the speaker verification experiment described above.

3. Automatic stream-weight optimization methods

3.1. Stream-weight estimation by the LDA

As described in section 2.4, speaker verification by SP-HMM uses the discriminant function

$$z = q_{sp}(O_{sp}) - \theta \quad (3)$$
$$= \lambda_s q_{sp}(O_s) + \lambda_p q_{sp}(O_p) - \theta. \quad (4)$$

For estimating stream weights, we approximate z by

$$z \approx \lambda_s q_s(O_s) + \lambda_p q_p(O_p) - \theta \quad (5)$$

to reduce computation time. This means that a score q_{sp} calculated from SP-HMM is approximated by using q_s from S-HMM and q_p from P-HMM.

The stream-weights, λ_s and λ_p, are estimated by obtaining z by the LDA as follows. First, segmental and prosodic scores, $q_s(O_s)$ and $q_p(O_p)$, calculated from claimed speaker's and imposter's data included in the development set are plotted in two-dimensional space composed by $q_s(O_s)$ and $q_p(O_p)$. Then, the LDA is applied to the space so as to obtain the discriminant function z which distinguishes score distribution of claimed speakers from that of imposters. Since the obtained function $z = \lambda_s q_s(O_s) + \lambda_p q_p(O_p) - \theta$ does not satisfy $\lambda_s + \lambda_p = 1$, it is transformed so that the sum of the coefficients of $q_s(O_s)$ and $q_p(O_p)$ becomes 1. The estimated values of the stream-weights and the threshold of verification are

$$\frac{\lambda_s}{\lambda_s + \lambda_p}, \frac{\lambda_p}{\lambda_s + \lambda_p}, \frac{\theta}{\lambda_s + \lambda_p}. \quad (6)$$

In this paper, we estimate only the stream-weights, since the optimal value of the threshold of verification changes according to the applications of the verification system.

3.2. A stream-weight optimization method using the Adaboost

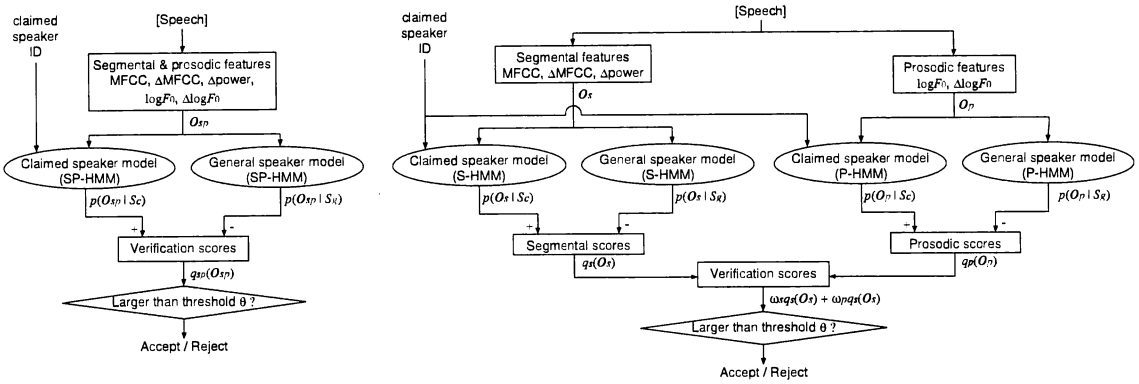
The Adaboost [2], a class of boosting algorithms, constructs a composite classifier by combining sequentially trained simple classifiers. A stream-weight optimization algorithm using linear discriminant functions obtained by the LDA as the simple classifiers for the Adaboost are given in the following subsections.

3.2.1. Adaboost algorithm

In the Adaboost algorithm, a training set is resampled according to the weights of data for every iteration. A classifier is trained by the resampled training set and given a weight according to its accuracy. Then, the weights of data are changed and a resampling of the training set is iterated. The final decision is made by a weighted majority voting of the iteratively trained classifiers.

Details of the Adaboost algorithm is as follows, where n represents the number of data in the training set and T represents the number of iterations. Let $\{x_i\}(i = 1, \cdots, n)$ be the labelled training set and $\{w_i\}(i = 1, \cdots, n)$ be the weights of each data.

1. Initialize the weights of data $w_i := 1/n$.
2. Iterate following processes for $t = 1, \cdots, T$.
   i) Choose n samples with duplicate from $\{x_i\}$ using $\{w_i\}$ as a probability distribution.
   ii) Obtain the linear discriminant function
   $$z_t = \lambda_s^{(t)} q_s(O_s) + \lambda_p^{(t)} q_p(O_p) - \theta^{(t)}$$
   by the LDA.



(a) SP-HMM (b) S-HMM & P-HMM

Figure 1: Flow of speaker verification process. (a) : The verification score $q_{sp}(O_{sp})$ is calculated from a segmental-prosodic feature vector O_{sp}. (b) : The scores $q_s(O_s)$ and $q_o(O_p)$ are separately calculated from segmental and prosodic feature vectors, O_s and O_p, and integrated afterward.

iii) Verify all data in the training set $\{x_i\}$ using z_t, and calculate the weighted discriminant error ϵ_t:

$$\epsilon_t := \sum_{i\text{: misclassify } x_i} w_i$$

where $0 < \epsilon_t \leq 1/2$. When $\epsilon_t > 1/2$, reverse the decisions of z_t and $\epsilon_t := 1 - \epsilon_t$. When $\epsilon_t = 0$, reset the weights as $w_i := 1/n$ and return to step i).

iv) Let the weight of z_t be $c_t := \frac{1}{2} \log \frac{1-\epsilon_t}{\epsilon_t}$.

v) Update w_i by following formula:

$$w_i := \begin{array}{ll} w_i \times e^{-c_t} & (i\text{: classify } x_i \text{ accurately}) \\ w_i \times e^{c_t} & (i\text{: misclassify } x_i) \end{array}$$

vi) Normalize $\{w_i\}$ to meet $\sum_{i=1}^{n} w_i = 1$.

3. Let the conclusive classifier z be the weighted majority vote of z_t:

$$z = \sum_{t=1}^{T} \{c_t \times sign(z_t)\}.$$

Classify by sign of z.

3.2.2. A stream-weight optimization method using the Adaboost

z, the result of the Adaboost algorithm, cannot be directly used for stream-weight estimation, since its form is not a linear discriminant function. Thus, we approximate z by $z = \sum_{t=1}^{T} (c_t z_t)$. Then, z can be represented by a linear discriminant function form:

$$z = \lambda_s^{(boost)} q_s(O_s) + \lambda_p^{(boost)} q_p(O_p) - \theta^{(boost)} \quad (7)$$

where $\lambda_s^{(boost)} = \sum_{t=1}^{T} (c_t \lambda_s^{(t)})$, $\lambda_p^{(boost)} = \sum_{t=1}^{T} (c_t \lambda_p^{(t)})$ and $\theta^{(boost)} = \sum_{t=1}^{T} (c_t \theta^{(t)})$.

As in section 3.1, the estimated values of segmental and prosodic stream-weights are normalized to

$$\frac{\lambda_s^{(boost)}}{\lambda_s^{(boost)} + \lambda_p^{(boost)}}, \frac{\lambda_p^{(boost)}}{\lambda_s^{(boost)} + \lambda_p^{(boost)}}. \quad (8)$$

Thus, the sum of the coefficients of $q_s(O_s)$ and $q_p(O_p)$ becomes 1.

4. Experiments

4.1. Database

Speech data were recorded at five sessions separated by intervals of approximately one month. The data were collected from 36 male speakers and sampled at 16kHz with a 16bit resolution. Each speaker uttered 50 strings of four connected digits in Japanese at each session.

The set of data recorded at sessions $1 \sim 3$ were used for training and data recorded at sessions 4 and 5 were used for weight optimization and testing. The database was separated into three groups in terms of speakers as shown in Figure 2. This figure shows the case where speaker M01 was used as the claimed speaker. The general speaker's model was trained using utterances by all the speakers in the speaker group 2, which did not include the claimed speaker nor the set of data for weight optimization. In the case where speaker #01 was used as the claimed speaker, an additional experiment was conducted, in which the general speaker's model was trained by the data of speaker group 3 and the stream-weights were optimized by the data of speaker group 2. There are 6 combinations of the training set, the development set and the testing set. The result averaged over the 6 experiments was used for evaluation.

White noise was added to the training set at a 30dB SNR level to increase robustness against noisy speech, and the development and testing sets were contaminated with white noise at 5, 10, 15, 20 and 30dB SNR conditions.

4.2. Experimental results

Table 2 shows the EERs using the stream-weights optimized by the proposed method with various numbers of Adaboost iterations in each SNR condition. The bold type represents the lowest EER in each SNR condition. "Baseline" in the table shows the results using only segmental information without using the multi-stream HMMs, and the results in "Adaboost t=1" correspond to those using the stream-weights estimated by only the LDA. The right most column shows the results of the experiment in section 2.5, in which the stream-weights were manually optimized. These results are worse than that of the proposed method in several SNR conditions. This is because the stream-weights were optimized at 0.01 intervals in the proposed method whereas they were optimized at 0.1 intervals in the manual optimization, as described in section 2.5.

Table 2: Comparison of the EERs when changing the number of Adaboost iterations in each SNR condition.

SNR (dB)	Baseline (S-HMM only)	Adaboost t=1 (LDA)	Adaboost t=2	Adaboost t=3	Adaboost t=4	Adaboost t=5	Manually optimized
30	0.88	0.74	**0.67**	0.75	0.77	0.77	0.65
20	4.91	3.54	**3.38**	3.41	3.49	3.48	3.36
15	14.67	9.04	**8.89**	8.90	9.06	8.98	9.02
10	27.10	16.29	**16.27**	16.31	16.73	16.63	16.28
5	37.48	**23.79**	23.89	23.94	24.04	24.01	24.32

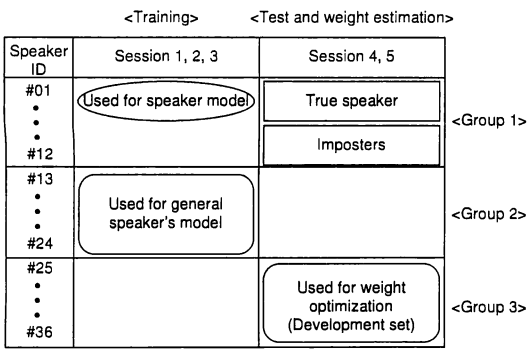


Figure 2: Training, testing and development sets for the verification experiment when the speaker M01 is used as the claimed speaker.

It has been confirmed that EERs can be reduced by the weight optimization method using the LDA and the Adaboost as compared to the baseline. Although the EER increases when the number of iterations increases, the loss of performance is modest and the performance is always increased compared to the baseline. This means that appropriate stream-weights are easily obtained by the proposed weight optimization method. The fact that the highest performance is obtained when t=2 in a wide range of noise conditions ($10 \sim 30$dB) indicates effectiveness of combining the weight optimization method by the Adaboost with the weight estimation by the LDA. Figure 3 shows the detection error tradeoff (DET) curves in 15dB SNR condition in terms of the number of the Adaboost iterations. The curve moves toward the origin as the number of iteration increases, and the performance becomes highest when t=2.

5. Conclusions

This paper first showed the effectiveness of using multi-stream HMMs integrating spectral and prosodic information for speaker verification. Next, an automatic stream-weight optimization method was proposed, in which the stream-weights are first estimated by the LDA and then optimized by applying the Adaboost. Experimental results using Japanese connected digit speech show that: 1) optimum stream-weights are obtained by the proposed method in all SNR conditions; 2) in $10 \sim 30$dB SNR conditions, the weight optimization method using the Adaboost is effective.

Our future works include: 1) investigating a method for estimating verification thresholds; 2) investigating a stream-weight estimation method using the score $q_{sp}(O_{sp})$ obtained from SP-HMMs without the approximation described in section 3.1, and comparing results with that reported in this paper; 3) investigating the weight optimization method without approximating the result of the Adaboost algorithm described in section

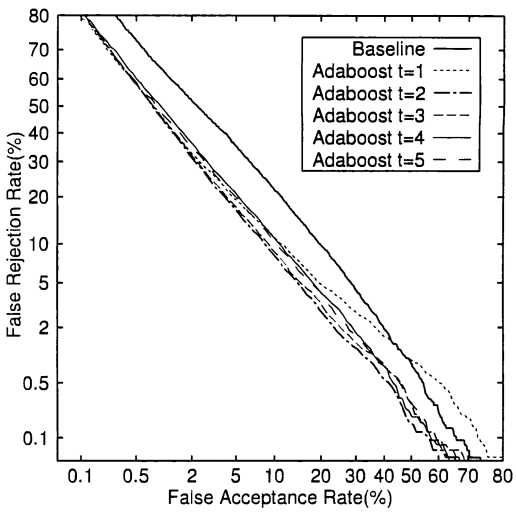


Figure 3: DET curves with various numbers of Adaboost iterations in 15dB SNR condition.

3.2.2; and 4) confirming the effectiveness of proposed method in noisy environments other than white noise.

6. References

[1] K. Iwano, T. Asami and S. Furui, "Noise-robust speaker verification using F_0 features," *Proc. ICSLP 2004*, vol.2, pp.1417-1420, Jeju Island, Korea, 2004.

[2] Y. Freund and R.E. Schapire, "A decision theoretic generalization of on-line learning and an application to boosting," *Journal of Computer and System Science*, vol.55, no.1, pp.119-139, 1997.

[3] S.W. Foo and L. Dong, "A boosted multi-HMM classifier for recognition of visual speech elements," *Proc. ICASSP 2003*, vol.2, pp.285-288, Hong Kong, 2003.

[4] P. Yin, I. Essa and J.M. Rehg, "Boosted audio-visual HMM for speech reading," *Proc. AMFG 2003*, pp.68-73, 2003.

[5] C. Dimitrakakis and S. Bengio, "Boosting HMMs with an application to speech recognition," *Proc. ICASSP 2004*, vol.5, pp.621-624, Montreal, Quebec, Canada, 2004.

[6] C. Meyer, "Utterance-level boosting of HMM speech recognition," *Proc. ICASSP 2002*, vol.1, pp.109-112, Orlando, Florida, 2002.

[7] K. Iwano, T. Seki and S. Furui, "Noise robust speech recognition using F_0 contour information," *IEICE Trans. on Information and Systems*, vol.E87-D, no.5, pp.1102-1109, 2004.