INTERSPEECH 2016
September 8–12, 2016, San Francisco, USA

Mahalanobis Metric Scoring Learned from Weighted Pairwise Constraints in I-vector Speaker Recognition System

Zhenchun Lei[1], *Yanhong Wan*[1], *Jian Luo*[1], *Yingen Yang*[1]

[1] School of Computer Information Engineering, Jiangxi Normal University, Nanchang, China

zhenchun.lei@hotmail.com, wyanhhappy@126.com,
luo.jian@hotmail.com, ygyang@jxnu.edu.cn

Abstract

The i-vector model is widely used by the state-of-the-art speaker recognition system. We proposed a new Mahalanobis metric scoring learned from weighted pairwise constraints (WPCML), which use the different weights for the empirical error of the similar and dissimilar pairs. In the new i-vector space described by the metric, the distance between the same speaker's i-vectors is small, while that of the different speakers' is large. In forming the training set, we use the traditional way in random fashion and develop a new nearest distance based way. The results on the NIST 2008 telephone data shown that our model can get better performance than the classical cosine similarity scoring. When using the nearest distance based way to form the training set, our model is better than the state-of-the-art PLDA. And the results on the NIST 2014 i-vector challenge show that our model is also better than the PLDA.

Index Terms: speaker recognition, Mahalanobis metric scoring, i-vector model

1. Introduction

The i-vector [1] based technique represent the state-of-the-art in speaker recognition [2]. A low dimensional i–vector is a compact representation of a supervector of Gaussian Mixture Model (GMM) [3], which captures most information of the high dimensional supervector variability. As i-vectors contain both speaker and channel variability, there is a requirement that intersession compensation approaches should be implemented to reduce the effects of channel variability in the i-vector speaker representations. Consequently, raw i-vectors are not suited for speaker discrimination directly and should be handled by intersession variability compensation methods, such as Linear Discriminative Analysis (LDA) [4] and Within-Class Covariance Normalization (WCCN) [5], which are performed to reduce the variability and enhance discrimination. Moreover, length normalization [6] is to reduce mismatch and allow for effective scoring. The cosine similarity scoring [7] for a trial between a set of i-vectors is used for its effectiveness and simplicity, and the state-of-the-art probabilistic linear discriminant analysis (PLDA) [8] based scoring generally shows relatively good properties.

Cumani [9] rewrote the log-likelihood ratio score in the PLDA as a dot-product in an i–vector pairs expanded space, and proposed the use of a 2nd order SVM kernel for the binary classification of basic trials. Like that, the Mahalanobis distance can also be rewrote as a quadratic function of the i-vector pair in a trial, and has been used for speaker recognition scoring [10]. In this paper, we also use the Mahalanobis-based scoring to measure the similarity between i-vectors. Most metric learning approaches [11, 12] learn a Mahalanobis-like distance: $\mathrm{d}_M(x_i, x_j) = (x_i - x_j)^T M (x_i - x_j)$, where M is a positive semi-definite (PSD) matrix satisfying the training constraints. Whether Mahalanobis distance can measure the similarity of the samples correctly or not depends significantly on the metric. A good metric relies on the simple geometric intuition that if all points in the same class could be mapped into a single location while those in other classes mapped to other location.

Cao et al. proposed the subspace similarity metric learning algorithm (SUB-SML) [13], which formulate the objective function by incorporating the robustness to the large intra-personal variations and the discriminative power of similarity metrics. In this paper, we will propose a new weighted pairwise constraints metric learning algorithm (WPCML), which adds a weight to the dissimilar pairs. The optimization problem is convex and our algorithm is completely specified by our objective function.

In the NIST SRE development dataset, the speaker ids were provided by the organizers, but the metric learning algorithm need the matched pairs and unmatched pairs information. If all training segments form the matched pairs and unmatched pairs with each other by speaker ids, the size of sets is very big and is very hard to handle. So we will also propose a new way to select the unmatched pairs to train the metric, while it is selected by random in traditional fashion.

This paper is organized as follows: Section 2 reviews the i-vector speaker recognition system. Section 3 details the proposed WPCML algorithm. Section 4 studies the methods of forming the training pair sets. Section 5 describes the experiments and results. Section 6 concludes the paper.

2. I-vector for Speaker Recognition

2.1. I-vector extractor

The most popular speaker model is the i-vector model [1], which is based on the Gaussian mixture model-universal background model (GMM-UBM) [14]. The speaker- and channel-dependent supervector can be expressed as:

$$\mu = m + Tw \tag{1}$$

where m is the UBM mean supervector, and T is a total variability space matrix containing speaker and channel variability. Speaker supervector μ fits the normal distribution of $N(m, TT^T)$. The post distribution of hidden total factor w is Gaussian distribution, and the mean of it is the i-vector of the speaker utterance, which is a more low dimensional compacted vector meeting the standard normal distribution. Suppose we have a sequence of L frames $\{\mathrm{x}_1, \mathrm{x}_2, \ldots, \mathrm{x}_L\}$and an

http://dx.doi.org/10.21437/Interspeech.2016-1071

UBM composed of K mixture components defined in some feature space of dimension *D*. The Baum-Welch statistics needed to estimate the i-vector for a given speech utterance *u* are obtained by:

$$N_k = \sum_{t=1}^{L} P(k \mid x_t, \lambda_{UBM}) \tag{2}$$

$$F_k = \sum_{t=1}^{L} P(k \mid x_t, \lambda_{UBM})(x_t - \mu_k) \tag{3}$$

Where μ_k is the kth mean of Gaussian distribution, and $P(k|x_t, \lambda_{UBM})$ is the post possibility of kth Gaussian distribution for speaker feature vector x_t. The i-vector of u is:

$$w = (I + T^T \Sigma^{-1} NT)^{-1} T^T \Sigma^{-1} F \tag{4}$$

where N is defined as a diagonal matrix of dimension $KD \times KD$ whose diagonal blocks are $N_k I(k = 1, \cdots, K)$. F is a supervector of dimension KD obtained by concatenating all statistics F_k for a given utterance u. Σ is a diagonal covariance matrix of dimension $KD \times KD$ estimated during factor analysis training and it models the residual variability not captured by the total variability matrix T.

2.2. Session variability compensation

After the i-vectors have been extracted, some session variability compensation techniques are used, such as Linear Discriminate Analysis (LDA), Within-Class Covariance Normalization (WCCN) and Length Normalization.

It is supposed that S is the total number of speakers, and n_s is number of i-vectors of speaker s, $\bar{w}_s$ is the mean of all the i-vectors of speaker s, and $\bar{w}$ represents the mean of all the i-vectors of all speakers. The target of LDA is to minimize intra-class distance and maximize between-class distance by projecting the data onto a subspace. The between-class variance S_b and within-class variance S_w are calculated as:

$$S_b = \sum_{s=1}^{S} (\bar{w}_s - \bar{w})(\bar{w}_s - \bar{w})^T \tag{5}$$

$$S_w = \sum_{s=1}^{S} \frac{1}{n_s} \sum_{h=1}^{n_s} (w_{s,h} - \bar{w}_s)(w_{s,h} - \bar{w}_s)^T \tag{6}$$

The optimal subspace is comprised by the eigenvectors of $S_b v = \lambda S_w v$.

WCCN maximizes the orthogonal direction of speaker-dependent to make the speaker space as orthogonal as possible. The within-class covariance matrix is computed as follows:

$$W = \frac{1}{S} \sum_{s=1}^{S} \frac{1}{n_s} \sum_{h=1}^{n_s} (w_{s,h} - \bar{w}_s)(w_{s,h} - \bar{w}_s)^T \tag{7}$$

After we get the matrix W, the i-vectors can be projected as: $\tilde{w} = Bw$. B is obtained by Cholesky decomposition: $W^{-1} = BB^T$.

The length normalization of i-vector is follows:

$$\tilde{w} = w / \|w\| \tag{8}$$

2.3. Scoring

In a trial, the i-vector speaker recognition system usually uses the cosine similarity scoring, PLDA scoring or Mahalanobis metric scoring to measure the similarity between the target i-vector w_{target} and the test i-vector w_{test}.

Cosine distance score can be computed as follows:

$$score_{\cos} = w_{target}{}^T w_{test} / (w_{target} w_{test}) \tag{9}$$

And the zt-normalization [15] was used for improving performance.

Probabilistic linear discriminant analysis (PLDA) model defines a speaker- and channel-dependent i-vector as:

$$w_r = \bar{w} + U_1 x_1 + U_2 x_{2r} + \varepsilon_r \tag{10}$$

where for given speakers $r = 1, \cdots, R$, U_1 is the eigenvoice matrix and U_2 is the eigenchannel matrix, x_1 and x_{2r} are the speaker and channel factors and ε_r is the residuals. For PLDA classification, scoring is conducted using the log-likelihood ratio as follows:

$$score_{PLDA} = \log\left(\frac{P(w_{target}, w_{test} \mid H_1)}{P(w_{target} \mid H_0) P(w_{test} \mid H_0)}\right) \tag{11}$$

where H_1 denotes that i-vectors (w_{target}, w_{test}) represent the same speaker and H_0 denotes that they do not.

The Mahalanobis metric scoring function is [10]:

$$score_M = -(w_{target} - w_{test})^T M (w_{target} - w_{test}) \tag{12}$$

3. Weighted Pairwise Constraints Metric Learning

Cao et al. [13] developed a regularization framework to learn similarity metrics for unconstrained face verification, which combine the cosine similarity and the Mahalanobis distance and proposed the SUB-SML algorithm. We proposed a new Mahalanobis metric scoring learned from weighted pairwise constraints(WPCML), which adds a weight to the dissimilar pairs.

In the following sections, the notations *S* and *D* denote the set of similar pairs (from the same speaker) and that of dissimilar pairs (from different speakers), and $\mathcal{P} = S \cup D$. Metric learning usually focuses on the Mahalanobis distance defined, for two i-vectors (w_i, w_j):

$$d_M(w_i, w_j) = (w_i - w_j)^T M (w_i - w_j) \tag{13}$$

where M is a positive semi-definite matrix.

3.1. Intra-speaker subspace projection

Like the SUB-ML, to reduce the effect of large intra-personal variations, we map all i-vectors to the intra-speaker subspace. The intra-speaker covariance matrix is defined by:

$$C_S = \sum_{(i,j) \in S} (w_i - w_j)(w_i - w_j)^\top \tag{14}$$

where $\Lambda = \{\lambda_1, \cdots, \lambda_k\}$, and $V = (v_1, \cdots, v_k)$ be the eigenvalues and eigenvectors of C_S .The mapping of the i-vector to the intra-speaker subspace is defined by the whitening process:

$$\tilde{w} = diag(\lambda_1^{-1/2}, \cdots, \lambda_k^{-1/2}) V^T w \tag{15}$$

3.2. Algorithm

If i-vector w_i is similar to w_j (from the same speaker), define its associated binary output $y_{ij} = 1$ and -1 otherwise. To better discriminate similar pairs from dissimilar pairs, we should learn M from the available data such that reports a large score for $y_{ij} = 1$ and a small score otherwise. We derive the formulation of the empirical discrimination using the hinge loss:

$$\varepsilon_{emp}(M) = \sum_{(i,j)\in P} (1 - y_{ij} d_M(w_i, w_j))_+ \tag{16}$$

For differentiating the importance of the similar pairs and dissimilar pairs, we applied a weight β to the dissimilar pairs:

$$\begin{aligned} \varepsilon'_{emp}(M) = &\sum_{(i,j)\in S} (1 - y_{ij} d_M(w_i, w_j))_+ \\ &+ \beta \sum_{(i,j)\in D} (1 - y_{ij} d_M(w_i, w_j))_+ \end{aligned} \tag{17}$$

Minimizing the above empirical error with respect to M will encourage the discrimination of similar pairs from dissimilar ones. Then the regularization term $\|M - I\|_F^2$ is introduced to improve the generalization ability of loss function $\varepsilon'_{emp}(M)$. When the loss function is over fitted in the training process, the regularization term can modify the value of it. $\|\cdot\|_F$ denotes the Frobenius normalization. Minimization of loss function $\varepsilon'_{emp}(M)$ can express as:

$$\min_M \varepsilon'_{emp}(M) + \frac{\gamma}{2}\|M - I\|_F^2 \tag{18}$$

The factor γ can be used to balance the effect of regularization term. Formulation (18) is identical to a standard convex optimization problem by introducing the slacking variables:

$$\begin{aligned} &\min_M \sum_{(i,j)\in P} \xi_{ij} + \frac{\gamma}{2}\|M - I\|_F^2 \\ &s.t.\ \ y_{ij}[d_M(w_i, w_j)] \geq 1 - \xi'_{ij}, \forall (i,j) \in S. \\ &\qquad y_{ij}[d_M(w_i, w_j)] \geq \beta(1 - \xi'_{ij}), \forall (i,j) \in D. \\ &\qquad \xi_{ij} \geq 0 \end{aligned} \tag{19}$$

The dual formulation can be written as:

$$\begin{aligned} &\max_{0\leq\alpha\leq 1} \sum_{(i,j)\in S} \alpha_{ij} + \sum_{(i,j)\in D} \beta\alpha_{ij} \\ &+ \sum_{(i,j)\in S} \alpha_{ij} y_{ij} (w_i - w_j)^2 + \sum_{(i,j)\in D} \beta\alpha_{ij} y_{ij} (w_i - w_j)^2 \\ &- \frac{1}{2\gamma} (\sum_{(i,j)\in S} \alpha_{ij} y_{ij} (w_i - w_j)(w_i - w_j)^T)^2 \\ &- \frac{1}{2\gamma} (\sum_{(i,j)\in D} \beta\alpha_{ij} y_{ij} (w_i - w_j)(w_i - w_j)^T)^2 \end{aligned} \tag{20}$$

If the optimal solution is denoted by α^* then the optimal solution M^* is given by:

$$\begin{aligned} M^* = I &- \frac{1}{\gamma} \sum_{(i,j)\in D} y_{ij} \alpha_{ij} (w_i - w_j)^\top (w_i - w_j) \\ &- \frac{1}{\gamma} \sum_{(i,j)\in S} \beta y_{ij} \alpha_{ij} (w_i - w_j)^\top (w_i - w_j) \end{aligned} \tag{21}$$

Formulation (20) is a standard quadratic programming problem, but we use the accelerated first-order algorithm proposed in [16] which is suitable for large-size datasets.

4. Forming Training Sets

The forming training sets used to learn a metric is a key issue in metric learning. An appropriate training pair containing proper information can instruct the training process correctly, while that with a great error will affect the training of the right metric.

4.1. Traditional way

As we can see in [17], similar pairs set is formed as follows. First, from the dataset, a speaker is chosen at random. Next, two i-vectors are drawn uniformly at random from among the i-vectors of the given speaker. If the two i-vectors are identical or if the i-vector pair of the specific speaker is already chosen previously as a similar pair, then the whole process is repeated. Otherwise the pair is added to the set of similar pairs.

Dissimilar pairs are formed as follows. First, from the set of speaker in the set, two speakers are chosen uniformly at random. One i-vector is then chosen uniformly at random from the set of i-vectors for each speaker. If this particular i-vector pair is already chosen previously as a dissimilar pair, then the whole process is repeated. Otherwise the pair is added to the set of dissimilar pairs.

4.2. Nearest distance pair sets for dissimilar pairs

Traditional way to form training pair sets is simple and effective, but the i-vector pair sets are selected at random. We propose a new way to form the training set, which is based on the Euclidean distance between two i-vectors.

For similar pairs, one speaker's all i-vectors form the training set, and they form the similar pairs by each other. The dissimilar pairs can also be formed between the i-vectors belonging to the different speaker, but the set size is very big and infeasible for calculating in the model. So we select the nearest Euclidean distance pairs as the training set from all dissimilar pairs. This make the training model is feasible and can also improve performance.

5. Experiments

5.1. Results on NIST 2008

Our experiments were run on the short2-short3 telephone core-task in NIST SRE 2008 dataset firstly. NIST SRE 2004, SRE 2005, and SRE 2006 telephone datasets were used for training two gender-dependent UBMs with 1024 Gaussian components, and estimating the total variability space. The dimension of the i-vectors in the total factor space is 400.

The features were derived from the waveforms using 13 mel-frequency cepstral coefficients on a 20 millisecond frame every 10 milliseconds. Delta and delta-delta coefficients were computed making up a 39 dimensional feature vector. And the band limiting was performed by retaining only the filter bank outputs form the frequency range 300-3400 Hz. Mean removal, preemphasis and a hamming window were applied, and energy-based end pointing eliminated nonspeech frames.

There are 6609 utterances from 491 male speakers and 9136 utterances from 703 female speakers are used to form the training pair set in NIST SRE 04, 05, 06. Like the cosine similarity scoring, the LDA, WCCN and Length normalization are used for session compensation. There are five models tested:

1. **Cosine+ZTNORM:** This is the classical model in speaker recognition field, and the cosine similarity scoring with zt-norm is used. LDA, WCCN and Length normalization are used for session variability compensation after the i-vectors are extracted from utterances.
2. **PLDA:** This is the most used model in the modern speaker recognition system. The PLDA classifier is used in the trial with the whitened i-vectors.
3. **SUB-ML:** The SUB-ML algorithm is used to learn a Mahalanobis metric with the traditional way to form the training sets.
4. **SUB-ML-N:** The SUB-ML algorithm is also used to learn a Mahalanobis metric, but the proposed nearest distance based way to form the training sets is used.
5. **WPCML-N:** Finally, we use the proposed WPCML algorithm to learn a Mahalanobis metric, and the forming training sets way is based on the nearest distance.

The results of the gender-dependent experiments are shown in the table 1 and table 2. For measuring the performance, we used equal error rate (EER) and the minimum decision cost function (minDCF), calculated using C_{miss}=10, C_{FA}=1 and P_{target}=0.01.

Table 1. *The results of cosine similarity scoring, PLDA and Mahalanobis metric scoring on the NIST 2008 (male)*

Model	EER(%)	minDCF(08)
Cosine+ZTNORM	4.74	0.027
PLDA	4.28	0.024
SUB-ML	4.68	0.023
SUB-ML-N	4.47	0.023
WPCML-N	**4.16**	**0.020**

Table 2. *The results of cosine similarity scoring, PLDA and Mahalanobis metric scoring on the NIST 2008 (female)*

Model	EER(%)	minDCF(08)
Cosine+ZTNORM	6.32	0.033
PLDA	5.10	0.024
SUB-ML	5.11	0.023
SUB-ML-N	5.00	0.024
WPCML-N	**4.75**	**0.022**

We can see that the discriminative performance of Mahalanobis metric scoring with metric learned from SUB-ML algorithm is better than the cosine similarity scoring. SUB-ML algorithm uses the constraint information of the pairwise i-vectors, so the learned metric can be more discriminative in the i-vector space, and make the Mahalanobis-based scoring be more predictive to the unknown utterances. When we select the dissimilar pairs according to the nearest distance, the performance of SUB-ML-N is better than SUB-ML's. The results of WPCML-N model show that our method can improve the performance compared to the SUB-ML algorithm, and can get the better performance than the state-of-the-art PLDA model.

5.2. Results on NIST 2014 i-vector challenge

In the NIST 2014 i-vector challenge [18], the 600 dimensional ivectors were trained by previous years NIST SRE data and provided by the organizers. The development data containing 36,572 speech files and 4,958 speakers were used to train the PLDA model. In the testing phase, there are 6,530 target i-vectors from 1,306 speakers and 9,634 test i-vectors. The trials were randomly divided into two separate subsets, 40% in the progress set and 60% in the evaluation set.

The performances of the proposed Mahalanobis metric scoring on the NIST 2014 i-vector machine learning challenge are shown in Table 3. The performance was evaluated the EER and normalized minDCF in [18]. The proposed WPCML-N model achieves 13.4% and 13.3% relative EER reduction against the PLDA baseline on the progress and evaluation subset respectively.

Table 3. *The results of PLDA and Mahalanobis metric scoring on the NIST 2014 challenge*

Model	EER(%)		norm minDCF(14)	
	prog	eval	prog	eval
PLDA	3.13	2.78	0.301	0.287
WPCML-N	2.71	2.41	0.270	0.262

6. Conclusions

This paper proposed the Mahalanobis metric scoring learned from the weighted pairwise constraints metric learning algorithm (WPCML) for speaker recognition, and the system performance can be improved. This paper also studied the way of forming training pair sets, and proposed a new way which is based on the Euclidean distance between the i-vectors. The performance of the speaker recognition system can be improved when using the selected similar pairs and dissimilar pairs. Compared to Cumani's method, our model is more simple and can also get better performance than the PLDA. Metric learning algorithms usually have more complexity of time and space, so how to make the metric learning algorithm takes less time and space is a key issue in future.

7. Acknowledgements

This work is supported by National Natural Science Foundation of P.R.China (61365004).

8. References

[1] Dehak N, Kenny P J, Dehak R, et al. "Front End Factor Analysis for Speaker Verification," *IEEE Transactions on Audio, Speech and Language Processing*, 2010.

[2] Kinnunen T, Li H. “An overview of text-independent speaker recognition: From features to supervectors,” *Speech Communication*, 52(1):12–40, 2010.

[3] Reynolds, Douglas A. “A gaussian mixture modeling approach to text independent speaker identification,” Georgia Institute of Technology, 1992.

[4] ESTOUP ARNAUD, et al. “Estimation of demo-genetic model probabilities with Approximate Bayesian Computation using linear discriminant analysis on summary statistics,” *Molecular Ecology Resources*, pp.846–855. 2012.

[5] A. Hatch, S. Kajarekar, and A. Stolcke, “Within-class covariance normalization for SVM-based speaker recognition,” *in Proc. Int. Conf. Spoken Lang. Process.*, Pittsburgh, PA, Sep. 2006.

[6] Garcia-Romero D, Espy-Wilson C Y. “Analysis of i-vector Length Normalization in Speaker Recognition Systems,” *Proceedings of Interspeech*, pp.249-252, 2011.

[7] Dehak, Najim, et al. “Cosine Similarity Scoring without Score Normalization Techniques,” *Proceedings of Odyssey 2010 - The Speaker and Language Recognition Workshop*, 2010.

[8] Kanagasundaram A, Vogt R, Dean D, et al. “PLDA based Speaker Recognition on Short Utterances,” *Speaker & Language Recognition Workshop*, 2012.

[9] S.Cumani, N.Brummer, L.Burget, P.Laface, O.Plchot, and V.Vasilakakis, “Pairwise discriminative speaker verification in the i-vector space,” *IEEE Transactions on Audio, Speech, and Lanaguage Processing*, vol.21, no.6, pp1217-1227, 2013

[10] Bousquet, P. M. “Intersession Compensation and Scoring Methods in the i-vectors Space for Speaker Recognition,” *Proceedings of Interspeech*, 2011.

[11] Kulis B. “Metric learning: a survey”. *Foundations & Trends in Machine Learning*, 2012.

[12] Bellet A, Habrard A, Sebban M, “A survey on metric learning for feature vectors and structured data,” *Technical report*, 2013

[13] Cao, Qiong, Y. Ying, and P. Li. “Similarity Metric Learning for Face Recognition,” *2013 IEEE International Conference on Computer Vision (ICCV)IEEE Computer Society*, pp. 2408-2415 2013.

[14] Reynolds, Douglas A., T. F. Quatieri, and R. B. Dunn. “Speaker Verification Using Adapted Gaussian Mixture Models," *Digital Signal Processing*, pp.19-41, 2000.

[15] Auckenthaler, Roland, M. Carey, and H. Lloyd-Thomas. "Score Normalization for Text-Independent Speaker Verification Systems," *Digital Signal Processing*, 2000.

[16] Beck, Amir, and M. Teboulle. “A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems,” *Siam Journal on Imaging Sciences*, 2009.

[17] Huang, Gary B., et al. “Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments,” *Workshop on Faces in 'Real-Life' Images: Detection*, Alignment, and Recognition, 2007.

[18] NIST, "NIST i-vector Challenge Homepage," *[Online]. Available: http://www.nist.gov/itl/iad/mig/ivec.cfm*.