Time-Contrastive Learning Based DNN Bottleneck Features for Text-Dependent Speaker Verification

Achintya kr. Sarkar
Department of Electronic Systems
Aalborg University, Denmark
akc@es.aau.dk

Zheng-Hua Tan
Department of Electronic Systems
Aalborg University, Denmark
zt@es.aau.dk

Abstract

In this paper, we present a time-contrastive learning (TCL) based bottleneck (BN) feature extraction method for speech signals with an application to text-dependent (TD) speaker verification (SV). It is well-known that speech signals exhibit quasi-stationary behavior in and only in a short interval, and the TCL method aims to exploit this temporal structure. More specifically, it trains deep neural networks (DNNs) to discriminate temporal events obtained by uniformly segmenting speech signals, in contrast to existing DNN based BN feature extraction methods that train DNNs using labeled data to discriminate speakers or pass-phrases or phones or a combination of them. In the context of speaker verification, speech data of fixed pass-phrases are used for TCL-BN training, while the pass-phrases used for TCL-BN training are excluded from being used for SV, so that the learned features can be considered generic. The method is evaluated on the RedDots Challenge 2016 database. Experimental results show that TCL-BN is superior to the existing speaker and pass-phrase discriminant BN features and the Mel-frequency cepstral coefficient feature for text-dependent speaker verification.

1 Introduction

Speaker verification (SV) aims to either accept or reject a person by his/her voice. It is broadly categorized into text-dependent (TD) and text-independent (TI) SV. In TD-SV, speakers are obliged to speak the same pass-phrases/sentences during both enrollment and test phases. In TI-SV, speakers are free to speak any pass-phrase/sentence during enrollment and test phases. As TD-SV maintains a matched phonetic content during both the training and test phrases, it outperforms TI-SV.

Due to the quasi-stationary behavior of speech within a short duration, short-time ceptral feature extraction is commonly deployed for speaker [1] and speech recognition [2]. Deep neural networks (DNNs) [3] have recently gained great attention from the speech and speaker recognition community. In SV, DNNs are generally adopted either for extracting the posteriori statistics [4–6] with respect to the pre-defined phonetic classes (called senones) for incorporating phonetic knowledge into i-vector extraction [7] or for discriminative feature extraction, where a DNN is trained to discriminate speakers, pass-phrases, phonetic classes, phones or a combination of them. The outputs of the DNN hidden layers are either directly used for speaker characterization *called d-vector* [8] or projected onto a low dimensional space *called bottleneck (BN)* feature [9–12]. Many studies in literature have demonstrated [10–12] that BN feature based systems either perform better than or provide complementary information to conventional short-time cepstral feature based speaker recognition.

A recent study [10] shows that the performance of TD-SV using BN features extracted based on the discrimination of both speaker and pass-phrases is similar to that of features based on discrimination of either speakers or speaker+phone. Among them, augmentation of a cepstral feature with the speaker+pass-phrase discriminant BN feature yields the lowest error rates. However, all of these

DNN BN feature extraction methods exploit label/supervision information of data. The success of these systems highly depends on the availability of well-labeled data.

Inspired by the time contrastive learning (TCL) concept, a type of unsupervised feature learning used for classification of magnetoencephalography (MEG) data in [13], we study the potential of TCL for speech signals. In [13], a TCL method is presented to classify few different states of brain that generally evolve over the time and can be measured by MEG signals. Speech and MEG signals, however, are quite different in nature, namely speech signals contain much richer information for which the tasks in hand often involve classification of much more classes.

In this paper, we explore the TCL concept for speech feature extraction. Two different strategies are considered. The first strategy randomly concatenates training utterances to form a single speech stream and then the stream is evenly partitioned into segments, each of which is assumed to contain a single content belonging to a class. Let N denotes the number of class in TCL. We take N segments each time, and the data points within the nth, $n \in \{1, 2, ..., N\}$, segment are assigned to class n. Then we take the next N segments and assign data points the same way and so on. Afterwards, a DNN is trained to discriminate the data points. Finally, the output of a selected hidden layer is projected onto a low dimensional space to get BN features. The second strategy is similar to the previous one with a key difference that *each speech utterance* is uniformly divided into a number of segments (i.e. the number of classes in TCL) regardless of speakers and contents.

The TCL-BN feature is experimentally compared with cepstral and existing BN features in TD-SV on the RedDots Challenge 2016 database consisting of short utterances. Experimental results show the superior performance of the TCL-BN feature. The Gaussian mixture model - universal background model (GMM-UBM) [14] technique is used for SV since it is well-established that a GMM based classifier [15, 16] outperforms the i-vector [7] method for SV with short utterances.

2 Cepstral and DNN bottleneck features

Cepstral feature: The Mel-frequency cepstral coefficients (MFCC) feature vectors (with RASTA [17] filtering) are of 57 dimensions consisting of static C_1-C_{19}, Δ and $\Delta\Delta$ coefficients. The frame shift is 10 ms and a 20 ms Hamming window is used. An energy based voice activity detection is applied to remove the less energized frames. High energy frames are normalized to zero mean and unit variance at utterance level.

DNN bottleneck features: Two DNN training approaches [10] are considered for BN feature extraction. In the first case, DNNs are trained to optimize a cross-entropy based objective function for discriminating speakers. In the second case, DNNs are trained to optimize two cross-entropy based objective functions simultaneously: one for discriminating speakers (*spkr*) and the other for discriminating pass-phrases. There are two types of output nodes: one predicting speakers and the other predicting pass-phrases. Average of the two criteria is used as a final criterion in the DNN multi-task learning procedure [18]. The output of a DNN hidden layer at frame-level is then projected onto a lower dimensional space *called bottleneck features*, as illustrated in Fig.1(a).

3 Time-contrastive learning concept and TCL based speech BN feature

In the TCL concept [13], multivariate time series data X are first divided into a number of uniform segments (say N), and then *data-points* within a particular segment are assigned to one class label,

$$\underbrace{(x_1, ..., x_M)}_{\text{class 1}}, ..., \underbrace{(x_{iM+1}, ..., x_{iM+M})}_{...}, ..., \underbrace{(x_{(N-1)M+1}, ..., x_{NM})}_{\text{class } N} \tag{1}$$

where i and M indicate the segment index and the number of data points within a segment, respectively. Finally, a DNN is trained to classify the data points. In [13], the output of the last hidden layer is used as a feature to classify different states of brain using magnetoencephalography (MEG) data. We adopt this concept and devise two strategies to extract speech features as follows.

Stream-wise TCL (sTCL): Speech utterances of training data are randomly concatenated to form a single speech stream which is then partitioned into segments of $d = 6$ frames each in order to capture short-time speech events. For a N number of classes in TCL, N segments are taken at a time, and the data points within the nth, $n \in \{1, 2, ..., N\}$, segment are assigned to class n. Then, the next consecutive N segments are taken and assign data points the same way. The process continues

until we finish assigning all data points in the stream. Afterwards, a DNN is trained to discriminate the data points assigned to N different classes. Finally, the output of one of the DNN hidden layers is projected onto a low dimensional space to get BN features for TD speaker verification.

Utterance-wise TCL (uTCL): This method considers *each training utterance separately* and then uniformly divides it into N segments. Class assignment for data points and DNN training are done the same way as in the previous method. The motivation here is to segment each utterance the same way, namely the first segment is the beginning of an utterance and the last the end of it. This consistency in segmentation is expected to be beneficial in particular when there are utterances of the same textual content, e.g. training data of fixed pass-phrases. As there are textually-repeating utterances, this can be regarded as weak supervision. Note that the pass-phrases appeared in the training data are excluded from evaluation, so the learned feature is not phrase-specific. The TCL-BN feature extraction methods are illustrated in Fig.1(b).
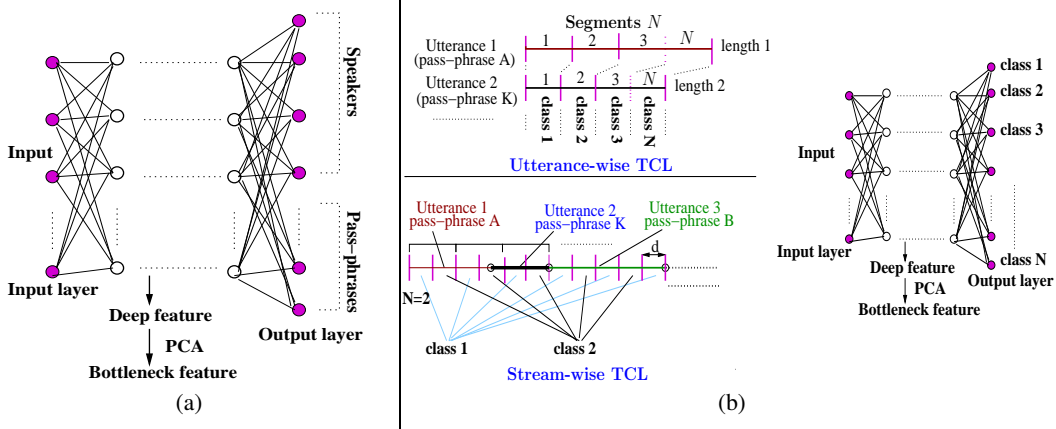

Figure 1: *Bottleneck feature extraction in (a) speakers + pass-phrases (b) TCL*

4 Experiments

For both existing and TCL based BN feature extraction, DNNs are trained using the RSR2015 [15] database. Data used are ≈ 72764 utterances recorded on 9 sessions for 27 pass-phrases from 300 non-target speakers (157 male, 143 female). All DNNs are 7 layer feed-forward networks and are trained using the same learning rate and the same number of epochs. Each hidden layer consists of 1024 sigmoid units. The input layer is of 627 dimensions, based on 57 dimensional MFCC features with a context window of 11 frames (i.e. 5 frames left, current frame, 5 frames right). In existing BN, the speaker-discriminant DNN consists of an output layer equal to the number of speakers, i.e. 300 nodes, while the speaker+pass-phrase discriminant DNN consists of 327 output nodes (300 speakers and 27 pass-phrases). To obtain the final BN feature, the output from a chosen hidden layer, a 1024 dimensional deep feature, is projected onto a 57 dimensional space to align with the dimension of MFCC feature for a fair comparison. Deep features are normalized to zero mean and unit variance at utterance level before using principle component analysis (PCA) for dimension reduction.

Experiments for text-dependent speaker verification are conducted on a different database, which is Part I (male speakers) of the RedDots database [19] as per protocol. Three types of non-target trials are available for evaluation. For details about the database and number of trials refer to [19].

Gender-independent GMM-UBM (512 mixtures, having diagonal covariance matrix) is trained using non-target speakers (438 male, 192 female) data (6300 utterances) from the TIMIT database [20]. The GMM-UBM training data are reused for the PCA. Speaker models are derived from the GMM-UBM (updating mean vector of mixtures) with maximum a posteriori (MAP) adaptation using their respective training data. In the test phase, a test utterance $Y = \{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_T\}$ is scored against the target specific model (obtained in training) λ_r and GMM-UBM λ_{ubm}. Finally, a log likelihood ratio (LLR) value is calculated using the scores between the two models $LLR(Y) = \frac{1}{T}\sum_{t=1}^{T}\{\log\ p(\mathbf{y}_t|\lambda_r) - \log\ p(\mathbf{y}_t|\lambda_{ubm})\}$, $\mathbf{y}_t$ denotes the tth frame/feature vector. Three iterations (with value of relevance factor 10.0) are used in MAP. System performance is evaluated in terms of equal error rate (EER) and minimum detection cost function (minDCF) [21].

5 Results and discussion

Table 1 shows the effect of number of classes (N) in TCL-BNs (for the $L2$ layer) on TD-SV. It is observed that sTCL and uTCL give lower TD-SV error rates when $N = 15$ and $N = 10$, respectively, and hence those values are experimentally chosen to represent their optimal performance in this setting. In this work, TCL-BN features are extracted from the $L2$ output layer of DNN, which gives slightly better performance than from $L4$. The uTCL approach outperforms sTCL, which could be explained by the factor that uTCL takes advantage of consistently segmenting utterances of the same textual content in contrast to sTCL that concatenates all utterances together before segmentation. Table 2 compares the performance of TD-SV on different feature sets (BN feature for the best DNN layer output in the respective systems: L4 in existing and L2 for TCL) for different non-targets.

Table 1: *Effect of number of classes (N) in TCL-BNs (L2 layer output of DNN) on TD-SV. The average EER is calculated over the system performance for different types of non-target (see Table2)*

Feature set	Number of classes (N) in TCL [% average EER]			
	5	10	15	20
sTCL-BN	2.91	2.88	**2.83**	2.86
uTCL-BN	1.98	**1.89**	2.96	2.95

Table 2: *Comparing performance of TD-SV for different features on RedDots m_part_01.*

Feature set	Non-target type [%EER/(minDCF$\times$ 100)]			Avg. (EER
	Target-wrong	Impostor-correct	Impostor-wrong	/minDCF)
MFCC	5.12/2.176	3.33/1.401	1.14/0.474	3.19/1.350
BN-spkr	4.59/1.654	3.05/1.355	1.11/0.380	2.91/1.130
BN-spkr+phrases	4.53/1.644	3.07/**1.348**	1.17/0.385	2.92/1.125
sTCL-BN [N=15]	4.33/1.662	**3.02**/1.384	1.14/0.391	2.83/1.145
uTCL-BN [N=10]	**1.88/0.654**	3.14/1.444	**0.64/0.195**	**1.89/0.764**

Table 2 shows TD-SV results for different features. Similarly to [10], existing BN features (for the $L4$ layer) show better TD-SV performance than the cepstral feature, and the two existing BN features show performance close to each other. TCL-BN shows lower average error rates for TD-SV compared with the cepstral and existing BN features. Specially, uTCL-BN shows significant reduction of error rates for the target- and impostors-wrong cases, which could be due to its ability to capture phonetic discriminate information. Overall, the TCL based features are effective. Moreover, it is observed in [10] that the TD-SV performance of BN features extracted from DNNs trained to discriminate speaker and triphone state labels (from speech recognition with a supervised mode) is very similar to those of *BN-spkr* and *BN-spkr+phrases*. However, speech recognition systems need annotation/labeled data for training. On the other hand, the TCL method does not use any speaker/pass-phrase/phonetic specific label information.

6 Conclusion

In this paper, we explored the time-contrastive learning (TCL) concept for training DNN based BN features for text-dependent speaker verification (TD-SV), in which DNNs are trained to discriminate the temporal events across a speech signal. This is realized by uniformly segmenting the speech signal into a number of segments and assigning the same label to all speech frames in one segment but different labels to different segments. DNNs are then trained to discriminate data across the different time segments without any speaker or phonetic label in contrast to the existing DNN BN feature extraction approaches. Experimental results confirmed the effectiveness of the proposed methods.

References

[1] T. Kinnunen and H. Li, "An Overview of Text-independent Speaker Recognition: from Features to Supervectors," *Speech Communication*, vol. 52, pp. 12–40, 2010.

[2] Z.-H. Tan and B. Lindber, "Low-Complexity Variable Frame Rate Analysis for Speech Recognition and Voice Activity Detection," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, pp. 798–807, 2010.

[3] G. Hinton et al., "Deep Neural Networks for Acoustic Modeling in Speech Recognition," in *IEEE Signal Process. Mag.*, 2012, pp. 82–97.

[4] M. McLaren, Y. Lei, and L. Ferrer, "Advances in Deep Neural Network Approaches to Speaker Recognition," in *Proc. of IEEE Int. Conf. Acoust. Speech Signal Processing (ICASSP)*, 2015.

[5] P. Kenny, V. Gupta, T. Stafylakis, P. Ouellet, and J. Alam, "Deep Neural Networks for Extracting Baum-welch Statistics for Speaker Recognition," in *Proc. of Odyssey Speaker and Language Recognition Workshop*, 2014, pp. 293–298.

[6] E. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent Pre-trained Deep Neural Networks for Large Vocabulary Speech Recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, pp. 30–42, 2012.

[7] N. Dehak, P. Kenny, R. Dehak, P. Ouellet, and P. Dumouchel, "Front-End Factor Analysis for Speaker Verification," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 19, pp. 788–798, 2011.

[8] E. Variani, X. Lei, E. McDermott, I. Lopez-Moreno, and J. Gonzalez-Dominguez, "Deep Neural Networks for Small Footprint Text-dependent Speaker Verification," in *Proc. of IEEE Int. Conf. Acoust. Speech Signal Processing (ICASSP)*, 2014, pp. 4080–4084.

[9] T. Fu, Y. Qian, Y. Liu, and Kai Yu, "Tandem Deep Features for Text-dependent Speaker Verification," in *Proc. of Interspeech*, 2014, pp. 1327–1331.

[10] Y. Liu, Y. Qian, N. Chen, T. Fu, Y. Zhang, and K. Yu, "Deep Feature for Text-dependent Speaker Verification," *Speech Communication*, vol. 73, pp. 1–13, 2015.

[11] C.-T. Do, C. Barras, V.-B. Le, and A. K. Sarkar, "Augmenting Short-term Cepstral Features with Long-term Discriminative Features for Speaker Verification of Telephone Data," in *Proc. of Interspeech*, 2013, pp. 2484–2488.

[12] S. Ghalehjegh and R. Rose, "Deep Bottleneck Features for i-vector based Text-independent Speaker Verification," in *Proc. of IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2015, pp. 555–560.

[13] A. Hyvarinen and H. Morioka, "Unsupervised Feature Extraction by Time-Contrastive Learning and Nonlinear ICA," in *Proc. of Neural Information Processing systems (NIPS)*, 2016.

[14] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker Verification using Adapted Gaussian Mixture Models," *Digital Signal Processing*, vol. 10, pp. 19–41, 2000.

[15] A. Larcher, K. A. Lee, B. Ma, and H. Li, "Text-dependent Speaker Verification: Classifiers, Databases and RSR2015," *Speech Communication*, vol. 60, pp. 56–77, 2014.

[16] H. Delgado, M. Todisco, M. Sahidullah, A. K. Sarkar, N. Evans, T. Kinnunen, and Z.-H. Tan, "Further Optimisations of Constant Q Cepstral Processing for Integrated Utterance and Text-dependent Speaker Verification," in *Proc. of Spoken Language Technology Workshop (SLT)*, 2016.

[17] H. Hermanksy and N. Morgan, "Rasta Processing of Speech," *IEEE Trans. on Speech and Audio Processing*, vol. 2, pp. 578–589, 1994.

[18] A. Agarwal et al., "An Introduction to Computational Networks and the Computational Network Toolkit," 2016.

[19] "The reddots challenge: Towards characterizing speakers from short utterances," https://sites.google.com/site/thereddotsproject/reddots-challenge.

[20] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, N. L. Dahlgren, and V. Zue, "TIMIT Acoustic-Phonetic Continuous Speech Corpus LDC93S1," 1993, Web Download. Philadelphia: Linguistic Data Consortium.

[21] A. Martin, G. Doddington, T. Kamm, M. Ordowskiand, and M. Przybocki, "The Det Curve in Assessment of Detection Task Performance," in *Proc. of Eur. Conf. Speech Commun. and Tech. (Eurospeech)*, 1997, pp. 1895–1898.