JOINT I-VECTOR WITH END-TO-END SYSTEM FOR SHORT DURATION TEXT-INDEPENDENT SPEAKER VERIFICATION

Zili Huang, Shuai Wang, Yanmin Qian[†]

Key Laboratory of Shanghai Education Commission for Intelligent Interaction and Cognitive Engineering
SpeechLab, Department of Computer Science and Engineering
Brain Science and Technology Research Center
Shanghai Jiao Tong University, Shanghai, China
{huangziliandy@sjtu.edu.cn, wsstriving@gmail.com, yanminqian@tencent.com}

ABSTRACT

Factor analysis based *i*-vector has been the state-of-the-art method for speaker verification. Recently, researchers propose to build DNN based end-to-end speaker verification systems and achieve comparable performance with *i*-vector. Since these two methods possess their own property and differ from each other significantly, we explore a framework to integrate these two paradigms together to utilize their complementarity. More specifically, in this paper we develop and compare four methodologies to integrate traditional *i*-vector into end-to-end systems, including score fusion, embeddings concatenation, transformed concatenation and joint learning. All these approaches achieve significant gains. Moreover, the hard trial selection is performed on the end-to-end architecture which further improves the performance. Experimental results on a text-independent short-duration dataset generated from SRE 2010 reveal that the newly proposed method reduces the EER by relative 31.0% and 28.2% compared to the *i*-vector and end-to-end baselines respectively.

Index Terms— speaker verification, end-to-end, i-vector, triplet loss, hard trial selection

1. INTRODUCTION

Speaker verification(SV) is a binary classification task that aims to accept or reject a given speech sequence for a claimed identity. According to different test conditions, speaker verification can be categorized into text-dependent and text-independent [1, 2]. The former requires the phrases for enrollment and test to be the same, while the latter impose no constraints on the utterance content. This work focuses on the short duration text-independent speaker verification.

I-vector followed by Probabilistic Linear Discriminant Analysis (PLDA) represents the state-of-the-art approach in text-independent speaker verification. *I*-vector is a low-dimensional representation that models speaker and channel variability in a single total variability space. PLDA serves as a scoring back-end and compensates the channel distortion. Recent success of deep neural networks (DNN) in speech recognition [3, 4, 5, 6] has inspired its application in the field of speaker verification. In [7], DNN is utilized to replace the role of GMM in the *i*-vector framework. An alternative approach is to use DNN to extract bottleneck features [8, 9, 10, 11] or speaker representations directly [12, 13, 14], among which *d*-vector [12] is the most typical one. More recently, end-to-end frameworks have been explored in speaker recognition tasks and shown comparable or even better performance than the classic *i*-vector approach [15, 16, 17, 18]. The work in [15, 16] proposed the end-to-end framework with binary cross entropy loss for text-dependent speaker verification, and others adopted triplet loss [17, 18].

Since these two frameworks, i.e. *i*-vector and end-to-end, differ from each other hugely, it's natural to utilize their potential complementary property to achieve a better system performance. In this study, we develop and compare several methods to integrate the two state-of-the-art approaches into one complete framework, which takes advantage of both technologies. Experimental results reveal that the speaker information captured by *i*-vectors and end-to-end embeddings are highly complementary to each other. The direct scoring fusion or embedding concatenation works well but fails to fully explore the complementary property. Introducing *i*-vectors into the end-to-end model training process with transformed concatenation and joint learning will further improve the system performance significantly.

The rest of the paper is organized as follows. Section 2 briefly reviews the *i*-vector framework. Section 3 introduces the triplet-loss based end-to-end framework and the triplet sampling strategy used in this paper. The integrated framework to combine the above two technologies is described in Section 4. Experiments and analysis are presented in Section 5. Conclusions are drawn in Section 6.

2. I-VECTOR

Systems based on *i*-vector and Probabilistic Linear Discriminant Analysis (*i*-vector/PLDA framework) represent the current state-of-the-art in text independent speaker verification. In the *i*-vector framework [19], the speaker- and session-dependent super-vector $\mathbf{M}$ (derived from UBM) is modeled as

$$\mathbf{M} = \mathbf{m} + \mathbf{T}\mathbf{w} \tag{1}$$

where $\mathbf{m}$ is a speaker and session-independent super-vector, $\mathbf{T}$ is a low rank matrix that captures speaker and session variability, *i*-vector

[†] Yanmin Qian is the corresponding author and now he is with Tencent AI Lab, Tencent, Bellevue, WA, USA.

This work was partly supported by the China NSFC projects (No. U1736202 and No. 61603252), and the Shanghai Sailing Program No. 16YF1405300. Experiments have been carried out on the PI supercomputer at Shanghai Jiao Tong University.

978-1-5386-4658-8/18/$31.00 ©2018 IEEE

is the posterior mean of $\mathbf{w}$. After extracting *i*-vectors, PLDA is usually adopted as the scoring back-end. It compensates the impact of channel in the *i*-vector space and achieves better performance than simple cosine similarity scoring.

3. END-TO-END SPEAKER VERIFICATION

End-to-end speaker verification can follow different paradigms [15, 16, 17, 18, 20]. In this paper, the triplet-loss based end-to-end system is adopted, the architecture is illustrated in Figure 1. In the training stage, frame level features are extracted and fed into a deep model. Frame embeddings derived from deep models are averaged in the temporal pooling layer to form utterance embeddings which are then L2 normalized onto an unit hypersphere. Triplet loss is calculated using the utterance embeddings in the same triplet and back-propagation algorithm is performed to update parameters. In the evaluation stage, enrolled utterance embeddings from the same speakers are averaged to obtain speaker embeddings. Euclidean distance between speaker embeddings and test utterance embeddings are calculated, which can be utilized for the final speaker verification decision. In this work, a VGG-style convolution neural network (CNN) [21] is used as the deep model, which will be described in Section 5.

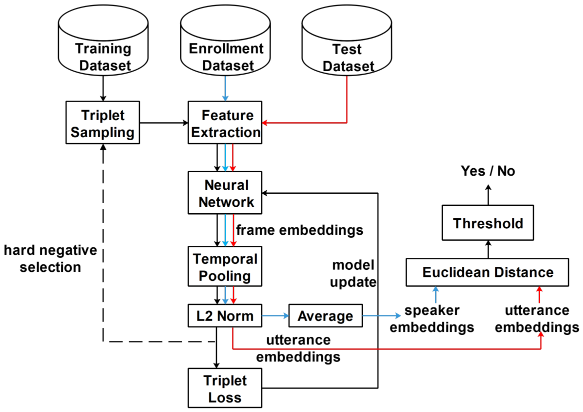


Fig. 1. End-to-end speaker verification system architecture with triplet loss

3.1. Triplet Loss

Aiming to minimize the within-class distance and simultaneously maximize the between-class distance, triplet loss is not as "greedy" as the pair-wise loss used in [15]. Triplet loss takes three inputs, including an anchor (an utterance from a specific speaker), a positive sample (an utterance from the same speaker) and a negative sample (an utterance from a different speaker). The loss L for an utterance triplet (u^a, u^p, u^n) is defined as

$$L(u^a, u^p, u^n) = [\|f(u^a) - f(u^p)\| - \|f(u^a) - f(u^n)\| + \alpha]_+ \quad (2)$$

where $f(u)$ denotes the embedding of the utterance u, α is an empirically defined margin enforced between positive and negative pairs and the operator $[x]_+ = max(x, 0)$. $\|f(u_1) - f(u_2)\|$ denotes the Euclidean distance between two embeddings $f(u_1)$ and $f(u_2)$. The total loss is the sum of loss computed on all triplets.

3.2. Triplet Sampling Strategy

Triplet sampling strategy plays a vital role in the training of the neural network. A good triplet sampling strategy leads to fast convergence and high verification accuracy. In our study, the similar triplet sampling strategy in [18] is followed. We divide the speakers into different groups and generate triplets in the same group. To be specific, given each group consists of n speakers and each person has k utterances, we create triplets for every positive pairs and the negative samples are randomly selected. In each epoch, $n \times k \times (k-1)/2$ triplets are created and we further reduce the number of triplets by only keeping triplets that violate the constraint $\|f(u^a) - f(u^p)\| + \alpha < \|f(u^a) - f(u^n)\|$.

3.2.1. *Hard trial selection*

In addition to the basic triplet sampling strategy [18], hard trial selection is applied to improve system performance. We select hard negative samples at utterance level or speaker level. Hard negative sampling at utterance level means that for each triplet we select the negative sample whose Euclidean distance is closest to the anchor. Hard negative sampling at speaker level gathers the speakers with similar embeddings into the same group. More specifically, we randomly select one speaker from the training set as the center and find his $(n-1)$-nearest neighbors in the speaker embedding space to form a group containing n speakers and create triplets among them. In our experiment, hard negative sampling at speaker level clearly outperforms that on utterance level and obtained considerable EER reduction.

4. JOINT I-VECTOR WITH END-TO-END SYSYEM

Factor analysis based *i*-vector approach follows a generative modeling paradigm, whereas neural networks based end-to-end model is discriminatively trained. We believe that the speaker information they obtained is highly complementary to each other. Accordingly we want to combine these two architectures into one integrated framework to take both advantages. Four combination strategies are explored and compared.

4.1. Score Fusion

Score fusion has been widely used due to its simplicity and effectiveness. The scores obtained via end-to-end system and *i*-vector system are normalized to comparable scales and averaged to obtain the final score for the decision.

4.2. Model Fusion

Rather than operating on the scores, we also explore the methods to combine two systems at model level. Three distinct modes for model fusion are proposed.

4.2.1. *Direct concatenation of embeddings*

The last layer of the end-to-end system can be regarded as an embedding extraction layer. The output of this layer is perceived as embedded speaker representation, which is similar to DNN embedding in [22]. The learned embedding can then be directly concatenated with the standard *i*-vector to form a new combined vector for speaker representation. It should be noted that the direct concatenation mode is also simple and requires no additional training stage, which is different from the next two methods.

4.2.2. *Transformed concatenation of embeddings*

Direct concatenation without any additional training stages is simple. However, it may not fully explore the complete complementary property from both speaker embeddings. Moreover, the direct concatenation increases the vector dimensions which consumes more computational cost in testing. Thus we proposed a newly trained transformation for the embedding concatenation as illustrated in Figure 2. The whole architecture can be divided into two parts, speaker embedding learning and embedding fusion learning. We wish to extract speaker discriminant features in the first part and learn how to effectively combine different speaker embeddings in the second part. For this transformed concatenation, we keep the parameters of the front-end CNN network fixed assuming that the quality of speaker discriminant features is high enough, and only train the linear transformed projection layer. The same triplet loss as described above is used to optimize the transformed projection layer.

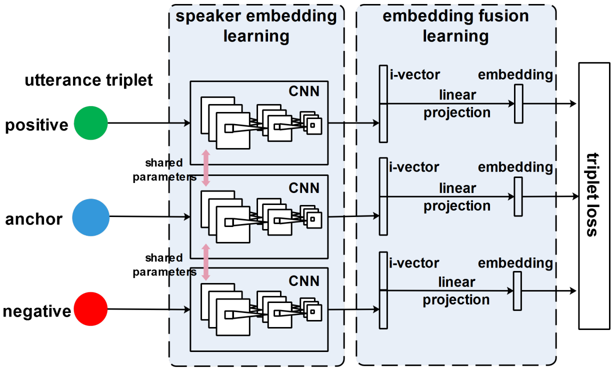


Fig. 2. The proposed architecture of integrating *i*-vector with end-to-end framework for speaker verification

4.2.3. *Joint learning*

The same architecture in the previous transformed embeddings concatenation is utilized in joint learning mode. The only difference is that instead of keeping the parameters of the speaker embedding learning part unchanged, the whole system is optimized and updated in an end-to-end training manner. By joint learning with *i*-vector, the new embeddings from the projection layer tend to be more effective. Experiments reveal that this system has the lowest equal error rate (EER) and the derived embeddings with joint learning show the highest discrimination ability among different speaker embeddings.

5. EXPERIMENTS

5.1. Data Preparation

We evaluate the performance of our proposed methods on a short-duration dataset generated from the NIST SRE corpus. This short duration text-independent task is more difficult and interesting for speaker verification. The training set consists of selected data from SRE04-08, Switchboard II phase 2, 3 and Switchboard Cellular Part1, Part2. After removing silence frames using an energy-based VAD, the utterances are chopped into short segments (ranging from 3-5s). The final training set contains 4000 speakers and each speaker has 40 short utterances. The enrollment set and test set are derived from NIST SRE 2010 following a similar procedure. The enrollment set contains 300 models (150 from male speakers and 150 from female speakers) and each model is enrolled by 5 utterances. The test set contains 4500 utterances from the 300 models in the enrollment set. The trial list we create contains 392660 trials. There are 15 positive samples and 1294 negative samples on average for each model. No cross-gender trial exists.

5.2. Implementation Details

The baseline is a standard *i*-vector / PLDA system based on Kaldi SRE10 V1 recipe [23]. The front-end features are 20-dimension MFCCs with a frame-length of 30ms. Delta and acceleration are appended to create 60-dimension feature vectors. 2048-mixture full covariance UBM and total variability matrix are trained using the generated training set. The dimension of extracted *i*-vectors is 400. PLDA serves as a scoring back-end.

In our end-to-end system, 36-dimension Fbank features are extracted as front-end features. The 17-frame context window is appended to form the 17×36 time-frequency feature maps for each frame. The VGG-style CNN [21], shown in Figure 3, is adopted in our system. It contains 4 convolution layers, 2 pooling layers and 1 fully-connected layer to produce the frame embeddings. The frame embeddings are then averaged to utterance embeddings with temporal pooling and L2 normalization. 2400 utterances from 60 speakers are selected in each epoch during the training process. For each positive pair, we randomly select another negative utterance to create a triplet. $60 \times 40 \times 39/2 = 46800$ triplets are generated in each epoch.

The performances of the *i*-vector and end-to-end baselines are shown on the top position of Table 1. In our experiment, 5 utterances are used for enrollment. Experimental results reveal that our end-to-end system performs slightly better than the *i*-vector system, which is consistent with the work in [16, 18].

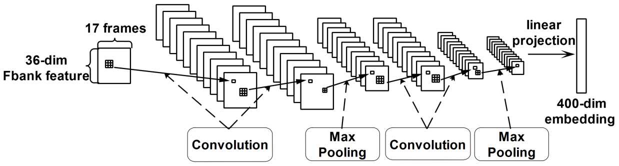
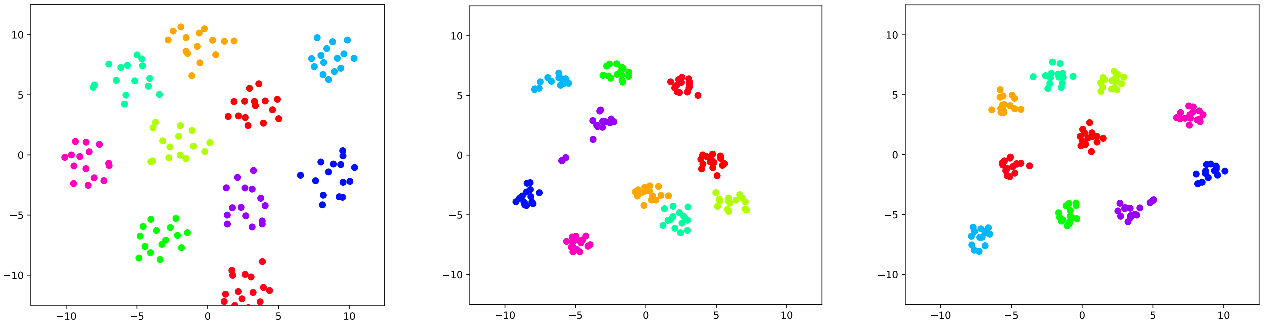


Fig. 3. VGG-style CNN architecture in our end-to-end system

5.3. Results and Analysis

5.3.1. *Evaluation on integrating i-vector with end-to-end speaker verification systems*

The proposed new approaches to integrate the *i*-vector with end-to-end framework are evaluated and the results are illustrated on the bottom part of Table 1 (with 5 utterances for enrollment). In Table 1, “basic” and “hard trial” refer to two triplet sampling strategies described in Section 3.2. It is observed that hard trial sampling strategy consistently outperforms basic sampling strategy for all end-to-



(a) *i*-vector (b) embedding: end-to-end (c) embedding: joint learning

Fig. 4. Visualization of different speaker embeddings: (a) *i*-vector. (b) embedding from the basic end-to-end system. (c) embedding from the proposed joint learning end-to-end system.

end involved speaker verification systems. Compared with two baselines, integrating *i*-vector with end-to-end systems improves system performance no matter which combination mode is adopted. Direct score fusion and naive embedding concatenation achieve obvious improvements. However, the improvements are not as large as the other two methods with parameter updating. Several points are revealed by these results: (1) Training end-to-end systems requires careful data preparation and trial selection, e.g. strategies such as hard trial selection help a lot. (2) *I*-vector system and end-to-end system contain a huge complementarity on the speaker knowledge representation, which can be utilized to improve system performance. (3) The complementary property can not be fully exploited by direct score fusion or naive embedding concatenation methods, in contrast embedding concatenations with parameter updating obtain a much larger improvement.

Among all the systems, the end-to-end speaker verification system with joint learning *i*-vector integration achieves the best system performance. The EER is reduced from 4.96% to 3.42%, relative 31.0% improvement compared to the *i*-vector system.

Table 1. Equal error rate (EER, %) comparison of different approaches to integrating *i*-vector with the end-to-end system

Method	**basic**	**hard trial**
i-vector/PLDA	4.96	
basic end-to-end	4.91	4.76
score fusion	4.67	4.51
direct concatenation	4.31	4.04
transformed concatenation	4.16	3.53
joint learning	3.96	**3.42**

Then, the influence of different enrollment utterance numbers is investigated on the proposed systems. The EER comparison is given in Table 2. Our newly proposed architecture integrating *i*-vector with end-to-end systems by joint learning significantly outperforms the traditional *i*-vector and basic end-to-end systems under all conditions with different enrollment utterance numbers. Another interesting finding is that the performance gap between the new proposed approach and previous method is enlarged significantly by the increased enrollment utterance numbers.

Table 2. EER (%) comparison of different enrollment utterance numbers

utt number	1	3	5	10
i-vector/PLDA	8.53	5.47	4.96	4.44
basic end-to-end	8.84	5.51	4.76	4.44
joint learning	7.64	4.13	3.42	**2.93**

5.3.2. Speaker embeddings visualization and analysis

Finally, different speaker embeddings, including the standard *i*-vectors, embeddings from the basic end-to-end system and embeddings from the joint learning end-to-end system, are visualized and compared, as shown in Figure 4. Each point represents a projected utterance embedding by t-SNE [24] and each color represents one speaker. It is observed that although *i*-vector has obvious discrimination between speakers, the within-speaker variability is large. The embeddings extracted from the basic end-to-end system shows reduced within-speaker variance, which is benefited from the triplet loss criterion on model optimization. However, the between-speaker distance among some of speakers is not large enough. The embeddings extracted from the newly proposed joint learning end-to-end system take both advantages from the previous two speaker embeddings and show superior property on both the within-speaker variance and between-speaker distance. This observation is also consistent with the results in Table 1 and Table 2.

6. CONCLUSION

This work shows that factor analysis based *i*-vector and deep model based end-to-end system contain highly complementary speaker knowledge. Accordingly we explore a framework to integrate both *i*-vector and end-to-end technologies into a paradigm to improve the system performance. Four combination approaches are developed and evaluated on a short duration text-independent speaker verification dataset based on SRE 2010. Compared to the *i*-vector baseline, the proposed joint learning framework reduces the EER by 31.0% relatively. This improvement can be further enlarged to 34.0% when with more enrollment utterances.

7. REFERENCES

[1] Joseph P Campbell, "Speaker recognition: A tutorial," *Proceedings of the IEEE*, vol. 85, no. 9, pp. 1437–1462, 1997.

[2] Douglas A Reynolds, Thomas F Quatieri, and Robert B Dunn, "Speaker verification using adapted gaussian mixture models," *Digital signal processing*, vol. 10, no. 1-3, pp. 19–41, 2000.

[3] Geoffrey Hinton, Li Deng, Dong Yu, George E Dahl, Abdelrahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N Sainath, et al., "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012.

[4] Frank Seide, Gang Li, and Dong Yu, "Conversational speech transcription using context-dependent deep neural networks," in *Twelfth Annual Conference of the International Speech Communication Association*, 2011.

[5] George E Dahl, Dong Yu, Li Deng, and Alex Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *IEEE Transactions on audio, speech, and language processing*, vol. 20, no. 1, pp. 30–42, 2012.

[6] Yanmin Qian, Mengxiao Bi, Tian Tan, and Kai Yu, "Very deep convolutional neural networks for noise robust speech recognition," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 24, no. 12, pp. 2263–2276, 2016.

[7] Yun Lei, Nicolas Scheffer, Luciana Ferrer, and Mitchell McLaren, "A novel scheme for speaker recognition using a phonetically-aware deep neural network," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2014*. IEEE, 2014, pp. 1695–1699.

[8] Tianfan Fu, Yanmin Qian, Yuan Liu, and Kai Yu, "Tandem deep features for text-dependent speaker verification.," in *Interspeech*, 2014, pp. 1327–1331.

[9] Yuan Liu, Yanmin Qian, Nanxin Chen, Tianfan Fu, Ya Zhang, and Kai Yu, "Deep feature for text-dependent speaker verification," *Speech Communication*, vol. 73, pp. 1–13, 2015.

[10] Yao Tian, Meng Cai, Liang He, and Jia Liu, "Investigation of bottleneck features and multilingual deep neural networks for speaker verification.," in *Interspeech*, 2015, pp. 1151–1155.

[11] Fred Richardson, Douglas Reynolds, and Najim Dehak, "A unified deep neural network for speaker and language recognition," *arXiv preprint arXiv:1504.00923*, 2015.

[12] Ehsan Variani, Xin Lei, Erik McDermott, Ignacio Lopez Moreno, and Javier Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2014*. IEEE, 2014, pp. 4052–4056.

[13] Lantian Li, Yiye Lin, Zhiyong Zhang, and Dong Wang, "Improved deep speaker feature learning for text-dependent speaker recognition," in *Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2015 Asia-Pacific*. IEEE, 2015, pp. 426–429.

[14] Nanxin Chen, Yanmin Qian, and Kai Yu, "Multi-task learning for text-dependent speaker verification," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[15] Georg Heigold, Ignacio Moreno, Samy Bengio, and Noam Shazeer, "End-to-end text-dependent speaker verification," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2016*. IEEE, 2016, pp. 5115–5119.

[16] Shi-Xiong Zhang, Zhuo Chen, Yong Zhao, Jinyu Li, and Yifan Gong, "End-to-end attention based text-dependent speaker verification," in *Spoken Language Technology Workshop (SLT),2016*. IEEE, 2016, pp. 171–178.

[17] Hervé Bredin, "Tristounet: triplet loss for speaker turn embedding," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2017*. IEEE, 2017, pp. 5430–5434.

[18] Chunlei Zhang and Kazuhito Koishida, "End-to-end text-independent speaker verification with triplet loss on short utterances," *Proc. Interspeech 2017*, pp. 1487–1491, 2017.

[19] Najim Dehak, Patrick J Kenny, Réda Dehak, Pierre Dumouchel, and Pierre Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[20] David Snyder, Pegah Ghahremani, Daniel Povey, Daniel Garcia-Romero, Yishay Carmiel, and Sanjeev Khudanpur, "Deep neural network-based speaker embeddings for end-to-end speaker verification," in *Spoken Language Technology Workshop (SLT), 2016*. IEEE, 2016, pp. 165–170.

[21] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[22] David Snyder, Daniel Garcia-Romero, Daniel Povey, and Sanjeev Khudanpur, "Deep neural network embeddings for text-independent speaker verification," *Proc. Interspeech 2017*, pp. 999–1003, 2017.

[23] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, et al., "The kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*. IEEE Signal Processing Society, 2011, number EPFL-CONF-192584.

[24] Laurens van der Maaten and Geoffrey Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, no. Nov, pp. 2579–2605, 2008.