



Insights into Deep Neural Networks for Speaker Recognition

Daniel Garcia-Romero and Alan McCree

Human Language Technology Center of Excellence
The Johns Hopkins University, Baltimore, MD 21218, USA

dgromero@jhu.edu, alan.mccree@jhu.edu

Abstract

Traditional i-vector speaker recognition systems use a Gaussian mixture model (GMM) to collect sufficient statistics. Recently, replacing this GMM with a deep neural network (DNN) has shown promising results. In this paper, we study a number of open issues that relate to performance, computational complexity, and applicability of DNNs as part of the full speaker recognition pipeline. The experimental validation is performed on the female part of the SRE12 telephone condition 2, where our DNN-based system produces the best published results. The insights gained by our study indicate that, for the purpose of speaker recognition, not using fMLLR speaker adaptation and early stopping of the DNN training allow significant computational reduction without sacrificing performance. Also, using a full covariance universal background model (UBM) and a large set of senones produces important performance gains. Finally, the DNN-based approach does not exhibit a strong language dependence as a DNN trained on Spanish data outperforms the conventional GMM-based system on our English task.

Index Terms: speaker recognition, i-vectors, deep neural networks

1. Introduction

Current speaker recognition systems model i-vectors [1] with variants of Probabilistic Linear Discriminant Analysis (PLDA) [2, 3, 4, 5, 6, 7]. Given a large collection of labeled data (speaker labels), PLDA provides a powerful data-driven mechanism to separate speaker information from other sources of undesired variability.

The traditional i-vector framework [1] uses a GMM to collect sufficient statistics (stats). The work in [8] has shown that replacing this GMM with a DNN to compute stats produces significant improvements in an in-domain setup. More recently, the work in [9] has shown that these gains can also be obtained in an unsupervised domain adaptation setup. The role of the DNN is to effectively leverage transcribed data and to produce soft classifications (in terms of posterior probabilities) of speech frames into sub-phonetic categories (senones). The alignment of speech frames to sub-phonetic categories facilitates the comparison of speakers when they are producing the same content. In the same spirit, the earlier work in [10] proposed the use of a phonetically-aware UBM obtained from an automatic speech recognition (ASR) system. However, the performance improvements obtained by the recent DNN approach are much larger. Also, the same concept of stats computation with DNNs was explored by [11] and found less promising. The results in this paper are more in line with the optimistic findings in [8].

Unlike in the ASR community, where for the past five years DNNs have disrupted the field and received much attention, it

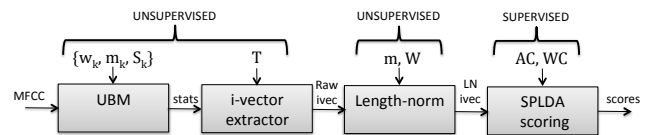


Figure 1: Block diagram of speaker recognition system indicating which parameters are trained in supervised and unsupervised mode.

was not until recently (2014) that DNNs have produced performance improvements for speaker recognition. Due to the different role that the DNNs play in the speaker recognition pipeline, it is necessary to gain more insights into their effective use as well as their applicability.

In this paper, we study a number of open issues that relate to performance, computational complexity, and applicability of DNNs as part of the full speaker recognition pipeline. In particular, we first evaluate the influence of having the best possible DNN (in terms of ASR accuracy) in our downstream speaker recognition performance. Specifically, we explore the importance of speaker adaptation, with feature-space maximum likelihood linear regression (fMLLR) [12], and amount of training iterations of the DNN. Also, we study how the granularity of the partition of the input space (determined by the number of senones) affects speaker recognition. Moreover, we evaluate the effects of using diagonal or full covariance Gaussians to model the acoustic features in each region of the partition. Finally, we study the language-dependence of the approach by comparing a matched-language DNN with a mismatched one.

By improving our understanding of these issues, the final goal of the paper is to provide guidelines that result in efficient performance maximization. All our experimental validation is performed on the female part of the SRE12 telephone condition 2 where our DNN-based system produces the best published results on this task.

The remainder of the paper is organized as follows. Section 2 describes the system architecture and summarizes the role of the DNN. Section 3 describes our experimental setup, the procedure to train the DNNs, and the experimental analysis. Finally, section 4 provides the conclusions.

2. Speaker Recognition System

Figure 1 shows a block diagram of a state-of-the-art i-vector speaker recognition system. The first two blocks serve as a data-driven front-end that maps sequences of MFCCs into a low-dimensional vector denoted as i-vector [1]. The third block is a pre-processing stage that conditions the i-vectors so that they conform to the Gaussian modeling assumptions of the last

10.21437/Interspeech.2015-298

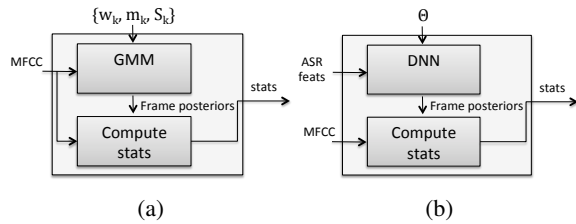


Figure 2: Diagram of the (a) GMM-based and (b) DNN-based sufficient statistics computation.

block [6]. The goal of the final block is to produce a similarity score, based on the PLDA model [6], that is higher as the likelihood of an i -vector \mathbf{x}_i belonging to speaker i increases. An efficient computation of this score is presented in [13]. On top of each block, Figure 1 shows the set of parameters that need to be trained.

2.1. Role of the DNN

Traditional i -vector systems rely on a GMM-UBM to provide soft alignments of acoustic frames (i.e. MFCCs) to compute sufficient statistics [1]. Each mixture of the GMM represents a region/class and provides a context in which to characterize how speakers differ from each other. Ideally one would like these regions to correspond to phonetic content (i.e. to allow comparisons of how speakers differ in pronouncing the same content). However, the unsupervised nature of the GMM training does not guarantee this behavior. To enforce this property, the authors in [8, 11] have proposed to replace the GMM with a DNN that has been explicitly trained to discriminate between tied triphone states (senones). In this way, the DNN is in charge of providing the class/region alignments for the stats computation.

Figure 2 highlights the differences between the GMM and the DNN approaches. Notice that while the traditional GMM-based approach uses the same acoustic features (i.e. MFCCs designed for good speaker recognition performance) to compute the stats and to obtain the alignments (frame posteriors), the DNN-based approach uses ASR specific features to compute the alignments and then speaker features for the stats. Moreover, the DNN parameters Θ are trained using a transcribed training set. This extra piece of supervision is what allows the DNN to provide alignments that are phonetically-aware.

3. Experiments

3.1. Datasets

3.1.1. Speaker recognition data

For our experiments we use the female part of the SRE12 telephone data (extended condition 2). This evaluation subset includes 1155 speaker models trained from 11,549 speech cuts (uneven number of cuts per model). There are 4524 target and 8,798,181 non-target trials. To train all the parameters indicated in Figure 1 we use 43,218 telephone call sides from 7692 speakers taken from Switchboard-II, Switchboard cellular, Fisher English part 1, and previous SRE collections.

3.1.2. DNN training data

Most of our analysis is based on DNNs trained on the Fisher English (FE) corpus (parts 1 and 2) which comprises 1200h

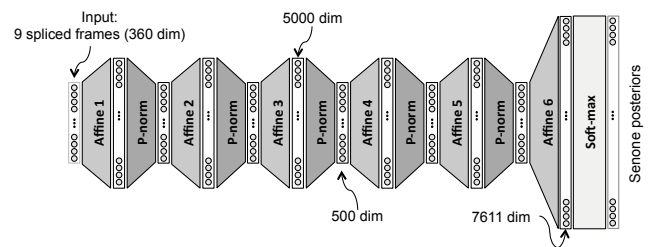


Figure 3: Block diagram of the DNN trained on 1200h of Fisher English data. There are 5 hidden layers of p -norm nonlinearities with $p = 2$ and approximately 15.5M free parameters.

of speech from approximately 20K conversation sides including around 10K speakers using cellphone and landline phones. For some contrasting experiments we also trained DNNs using the Fisher Spanish (FS) and Switchboard-I (SWB) corpora. FS comprises 130h of speech from around 1500 conversation sides including 136 speakers using cellphone and landline phones. SWB comprises 300h of speech from around 4600 conversation sides including 543 speakers using only landline phones.

3.2. Speaker recognition system setup

3.2.1. GMM-based baseline

The baseline system in Figure 1 uses 40-dimensional MFCCs (20 base + deltas) with short-time mean and variance normalization. It is configured in a completely gender-independent way. It uses a 2048 mixture diagonal UBM with a 600 dimensional i -vector extractor, and a speaker subspace of 400 dimensions for PLDA. We report recognition performance in terms of equal error rate (EER) and/or normalized minimum detection cost function (DCF) [14] with probability of target trial set to either 10^{-2} or 10^{-3} , and the cost of misses and false alarms set to 1.

3.2.2. DNN-based systems

The only differences between the DNN and GMM-based configurations are due to the alternative ways to compute the frame posteriors. The posteriors of the DNN-based system are used to compute the stats and to define an ancillary UBM needed for the i -vector computation [8, 11]. The number of mixtures of this UBM is given by the number of senones (after removing senones from non-speech states). We use full covariance mixtures for our best performing system, but also show results in the case of diagonal covariances.

3.3. DNN training

All the DNNs in this work are trained using the Kaldi speech recognition toolkit. The labels (i.e. frame alignments to senones) for the DNNs are obtained from a standard tied-state triphone GMM-HMM system trained with maximum likelihood. The senone set is obtained by clustering the states using a decision tree and the number of total Gaussians is set to 300K. For the experiments where we vary the number of senones, we keep the number of total Gaussians fixed. The input features for the GMM-HMM system are 40 dimensional vectors obtained from an LDA+MLLT projection of 7 spliced frames of 13 MFCCs. These features are further processed by an fMLLR transform to perform speaker adaptation.

Our DNN architectures comprise either 4 or 5 hidden layers of p -norm non-linearities with $p = 2$ and an input/output

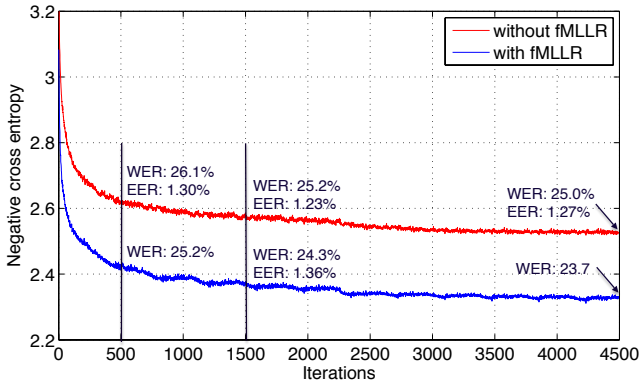


Figure 4: Plot of the negative cross entropy on a held-out validation set for the DNN trained on Fisher English data as a function of the number of training iterations. The overlaid annotations correspond to the word error rate (WER) on the validation set, and the EER on our speaker recognition task. An epoch comprises almost 500 iterations. The annotations loosely correspond to 1, 3, and 9 epochs.

dimension ratio of 10 [15]. The input to the DNNs are 9 spliced vectors of the 40 dimensional LDA+MLLT projected features (total of 360 dim). fMLLR transforms are not used since (as shown in our analysis) they add significant computational complexity without improving speaker recognition performance. Figure 3 shows an example of the specific configuration of our best performing DNN trained on Fisher English.

The DNN training algorithm performs back propagation using mini-batch natural gradient descent and parameter averaging [16]. Parallelization is accomplished by training n replicas of the DNN ($n = 4$ for our system) independently on disjoint subsets of data and combining them periodically by averaging their parameters. Once a new updated DNN is obtained, it gets replicated and asynchronous training is performed again. We refer to each one of these stages of “replicate-train-merge” as an iteration. An epoch (i.e. a run over the entire training dataset) consists of a fixed number of these iterations. During an iteration, each replica does back propagation over 400,000 training examples (using mini-batches of size 512). We use an exponential decay schedule for the learning rate and minimize negative cross entropy for a fixed number of epochs. Convergence is monitored on a validation set in terms of cross entropy and frame accuracy.

3.4. Analysis

3.4.1. Impact of fMLLR

Feature-space maximum likelihood linear regression, also known as constrained MLLR, is an affine feature transform that is commonly used for speaker adaptation [12] in state-of-the-art ASR systems. The estimation of the fMLLR transforms requires a first pass of the data through the HMM-GMM and language model to find an alignment [12]. This is computationally expensive at runtime and therefore its use needs to be justified by a significant boost in performance. While this is true for ASR, it is not clear what the effects of fMLLR are in our speaker recognition pipeline. To facilitate a better understanding of this issue, Figure 4 shows the impact of fMLLR in the cross entropy as well as WER and EER. Cross entropy and WER are evaluated on a held-out validation set. For WER eval-

uation we used a trigram language model (LM) trained on 3M words of Switchboard training transcripts and then interpolated it with another trigram LM trained on 11M words of the Fisher English Part 1 transcripts. The speaker recognition performance is shown in terms of EER but the DCFs are available in Table 1.

Looking at Figure 4, it is quite clear that fMLLR has a significant effect in cross-entropy and WER. As expected, fMLLR improves WER, and better cross entropy translates into better WER. However, the speaker recognition performance does not follow the same trend. In fact, the results are slightly worse in terms of EER and DCF. Therefore, removing fMLLR is a good way to reduce runtime computational cost without performance degradation.

3.4.2. Convergence of the DNN

Another interesting observation from Figure 4 and Table 1 is that speaker recognition performance does not benefit from a lot of training iterations of the DNN. As an example, for the 1200h of Fisher English DNN, each iteration took an average of 110s to complete (using 4 GPUs in parallel). Therefore, a DNN trained for 500 iterations can be obtained overnight (15h), whereas waiting for the 4500 iterations would take around 6 days, and would not produce any performance gains. This knowledge can be used by the researcher to expedite experimental exploration as well as by the practitioner to do fast deployment of DNN-based systems.

3.4.3. Diagonal vs full covariance UBM

The role of the DNN is to provide a context in which to characterize how speakers differ from each other. Unlike in the GMM-UBM approach, the partition of the space accomplished by the DNN is optimized to discriminate between senones. Due to the complex nature of this phonetic space, these regions are not well modeled by simple Gaussians with diagonal covariance. To quantify this statement, we took the Fisher English DNN with 1500 iterations (row 3 of Table 1) and instead of a full covariance UBM we used a diagonal one. The performance of the diagonal UBM system at the three operating points is: $DCF10^{-3} = 0.246$, $DCF10^{-2} = 0.145$, and $EER = 1.75\%$. Compared to the full covariance system, we can observe a noticeable reduction in performance (still much better than the baseline). Although the full covariance UBM has many more parameters than the diagonal one, the final complexity of the i-vector extraction process is not significantly increased (i.e. a similarity transformation of the first order stats that standardizes the covariances of the UBM facilitates efficient computation). Therefore, full covariance UBMs are recommended for

Table 1: Performance comparison of the baseline GMM system and the DNN-based systems. The influence of the number of DNN training iterations and the effect of fMLLR on the input features are presented.

System	Iter.	$DCF10^{-3}$	$DCF10^{-2}$	EER(%)
GMM (2048)	-	0.362	0.195	1.82
Fisher English DNN (7611)	500	0.221	0.119	1.30
	1500	0.218	0.117	1.23
	4500	0.220	0.117	1.27
+ fMLLR	1500	0.220	0.121	1.36

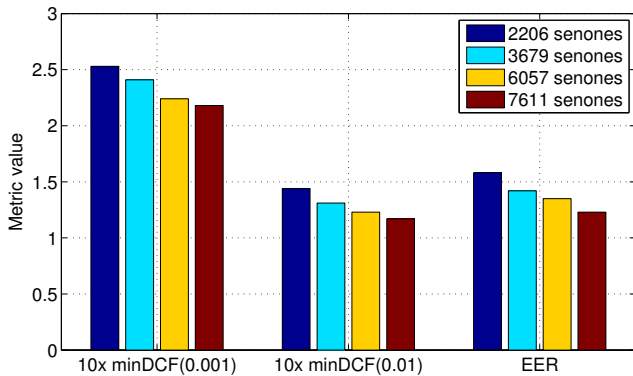


Figure 5: Speaker recognition performance for our three metrics as a function of the number of target senones of the DNN trained on Fisher English. Note that the number of senones defines the number of mixtures of the UBM.

DNN-based systems.

3.4.4. Size of senone set

The number of regions in which the acoustic space is partitioned is given by the senone set (discarding non-speech senones). This set is obtained by clustering triphone states using a decision tree and its size has a strong repercussion in the total memory and computational cost of the final system. To characterize the relationship between number of senones and speaker recognition performance, we took the Fisher English DNN with 1500 iterations (row 3 of Table 1) and changed the number of target senones of the DNN. The bar plot in Figure 5 shows the results for the three operating points. We can see that the more senones the better the performance for the three metrics. It is quite remarkable that the performance does not saturate, for the large range considered, as we increase the partitioning of the acoustic space. This is a new behavior that differs from that of the unsupervised GMM-UBM approach (where performance saturates quite fast [8]). Therefore, it is the supervised nature of the DNN partitioning process (i.e. leveraging transcribed data) that enables a fine-grained partitioning that results in important performance gains. We plan to explore bigger sizes in the future to find out where the saturation point starts. However, these bigger systems might be too computationally expensive for practical purposes.

3.4.5. Language of DNN training data

Using transcribed data to train the DNN brings into question the amount of language dependence that might be introduced into the speaker recognition system. If performance gains are only attained by using a matched-language paradigm, the applicability of the technique is more limited. A first look into this issue was presented in [17] where Farsi and Arabic were studied using a convolutional neural network (CNN). Their results suggested little language dependence. Here we expand that work using a more mainstream dataset (conversational telephone calls) with a DNN and a different language pair (Spanish and English). Table 2 shows the performance of the system using a DNN trained on Fisher Spanish, along with the baseline and other DNN-based systems trained on English data (two from SWB and our best system from FE). The second column shows the amount of transcribed data used by the DNNs. Note

Table 2: Performance comparison of the baseline GMM system and DNN-based systems trained on various datasets. The influence of the language, type, and amount of data is presented. See section 3.1.2 for more detail about the composition of the datasets.

System	DNN Data (hours)	DCF10 ⁻³	DCF10 ⁻²	EER(%)
GMM (2048)	-	0.362	0.195	1.82
SWB DNN-1	100	0.304	0.175	1.92
Fish. Spanish	130	0.309	0.186	1.97
Fish. English	1200	0.218	0.117	1.23

that the types of channels and number of speakers also vary (see section 3.1.2), so the performance may also be affected by these factors. We can see that the Fisher Spanish system outperforms the baseline at both DCFs and gets a small degradation at EER. This behavior is similar to that of the SWB DNN-1 system, but slightly worse. Given the many factors of variation, we need to be cautious about making definitive statements. However, it seems safe to suggest that although there might be some language dependence, it is not too strong. Also, for the low false alarm region the Fisher Spanish system is as good as the SWB DNN-1. Nevertheless, a multilingual DNN systems could be a good approach to further reduce it, and it is part of our future work.

4. Conclusions

In this paper, we have conducted a large number of experiment to gain insights into the effective use of DNNs for speaker recognition. The results were presented on the female part of the SRE12 telephone data, where our top performing DNN provides the best published results on this task. The insights gained by our study indicate that not using fMLLR speaker adaptation and early stopping of the DNN training allow significant computational reduction without sacrificing performance. Also, using a full covariance universal background model (UBM) and a large set of senones is important for maximum performance. Finally, the DNN-based approach does not exhibit a strong language dependence as a DNN trained on Spanish data outperforms the conventional GMM-based system on our English task.

5. References

- [1] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, May 2011.
- [2] S. Prince and J. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *IEEE International Conference on Computer Vision (ICCV)*, Rio de Janeiro, 2007.
- [3] N. Brümmer and E. De Villiers, "The speaker partitioning problem," in *Odyssey: The Speaker and Language Recognition Workshop*, Brno, Czech Republic, 2010.
- [4] P. Kenny, "Bayesian speaker verification with heavy-tailed priors," in *Odyssey: The Speaker and Language Recognition Workshop*, Brno, Czech Republic, 2010.
- [5] J. Villalba and N. Brümmer, "Towards fully Bayesian

- speaker recognition: Integrating out the between-speaker covariance,” in *Interspeech*, Florence, Italy, August 2011.
- [6] D. Garcia-Romero and C. Espy-Wilson, “Analysis of i-vector length normalization in speaker recognition systems,” in *Interspeech*, Florence, Italy, August 2011.
- [7] D. Garcia-Romero, X. Zhou, and C. Espy-Wilson, “Multicondition training of Gaussian PLDA models in i-vector space for noise and reverberation robust speaker recognition,” in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Kyoto, Japan, 2012.
- [8] Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, “A novel scheme for speaker recognition using a phonetically-aware deep neural network,” in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2014.
- [9] D. Garcia-Romero, X. Zhang, A. McCree, and D. Povey, “Improving speaker recognition performance in the domain adaptation challenge using deep neural networks,” in *IEEE Spoken Language Technology Workshop (SLT)*, 2014.
- [10] M. Omar and J. Pelecanos, “Training universal background models for speaker recognition,” in *Odyssey: The Speaker and Language Recognition Workshop*, 2010.
- [11] P. Kenny, V. Gupta, T. Stafylakis, P. Ouellet, and J. Alam, “Deep neural networks for extracting Baum-Welch statistics for speaker recognition,” in *Odyssey: The Speaker and Language Recognition Workshop*, 2014.
- [12] M. Gales, “Maximum likelihood linear transformations for HMM-based speech recognition,” *Computer Speech and Language*, vol. 12, pp. 75–98, 1998.
- [13] D. Garcia-Romero and A. McCree, “Subspace-constrained supervector PLDA for speaker verification,” in *Interspeech*, 2013.
- [14] “The NIST year 2010 Speaker Recognition Evaluation plan.” (Available at http://www.itl.nist.gov/iad/mig/tests/sre/2010/NIST_SRE10_evalplan.r6.pdf), 2010.
- [15] X. Zhang, J. Trmal, D. Povey, and S. Khudanpur, “Improving deep neural network acoustic models using generalized maxout networks,” in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2014.
- [16] D. Povey, X. Zhang, and S. Khudanpur, “Parallel training of deep neural networks with natural gradient and parameter averaging,” in *International Conference on Learning Representations (ICLR)*, submitted, 2015.
- [17] M. McLaren, Y. Lei, N. Scheffer, and L. Ferrer, “Application of convolutional neural networks to speaker recognition in noisy conditions,” in *Interspeech*, Singapore, 2014.