Language Identification Using Time Delay Neural Network D-Vector on Short Utterances

Conference Paper · August 2016

DOI: 10.1007/978-3-319-43958-7_53

5 authors, including:

Maxim Tkachenko

5 PUBLICATIONS **12** CITATIONS

SEE PROFILE

Language Identification using Time Delay Neural Network D-Vector on Short Utterances

Maxim Tkachenko[1], Alexander Yamshinin[1], Nikolay Lyubimov[3], Mikhail Kotov[2], and Marina Nastasenko[1]

[1] Vector I LLC, Moscow, Russia
[2] ASM Solutions LLC, Moscow, Russia
[3] Lomonosov Moscow State University
{makseq,lex.sapfir,lubimov.nicolas,kotov.mike,marina.nastasenko}@gmail.com

Abstract. This paper describes d-vector language identification (LID) system on short utterances using time delay neural network (TDNN) acoustic model for the speech recognition task. The acoustic TDNN model is chosen for ASR system of ICQ messenger and it's applied for the LID task. We compared LID TDNN d-vector results to i-vector baseline. It was found that the TDNN system performance is close at any durations while i-vector shows good results only at long time. Open-set test is conducted. Relative improvement of 5.5% over the i-vector system is shown.

Keywords: language identification, i-vector, d-vector, speech recognition acoustic model, neural networks

1 Introduction

I-vector is a gold-standard approach for speaker and language identification [1, 2]. Whereas neural networks have a rising power. Deep neural networks (DNN) and Long Short-Term Memory (LSTM) were introduced. LSTM has demonstrated a high performance for ASR and LID [3–5] tasks. Auto-encoders and bottleneck features provided by NNs have also improved the performance in all speech processing tasks. D-vectors become popular within DNNs. The goal of our team is to explore the acoustic model of our production ASR based on TDNN for d-vector [6].

We have built Russian ASR in ICQ messenger. More than 15% of data queries is not in Russian. Under a high load there is a need to truncate unwanted traffic with no Russian speech.

The paper is organized as follows. Section 2 refers to I-Vector Baseline system. Section 3 describes D-vector and TDNN in detail. Section 4 describes dataset and data preparation. Section 5 presents and analyzes the results on 3 seconds durations. Section 6 presents the conclusions and interesting findings.

2 I-Vector Baseline System

2.1 About

I-vector is the state of the art technique that effectively represents speech utterance as low dimensional vector. The underlying idea behind i-vectors is based on supervectors over concatenated Gaussian Mixture Models (GMM) means M, factorized as

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

where m is concatenated Universal Background Model (UBM) means, T forms the subspace covering the important variability (both language- and session-specific) in the supervector space, and w is a random vector distributed as N(0,1). For each observation sequence representing an utterance the corresponding i-vector can be estimated using the maximum a posteriori (MAP) method. For more detail on i-vector extraction see [2], [8].

2.2 Configuration

First, the UBM GMM is trained. The next step is to calculate Total Variability and Sigma matrices on the special dataset from train. The i-vector extractor uses Baum-Welch statistics calculated from voice frames, followed by Support Vector Machine scoring procedure. We have used RBF kernels to model non-linear relationship in total variability space. (Fig. 1).

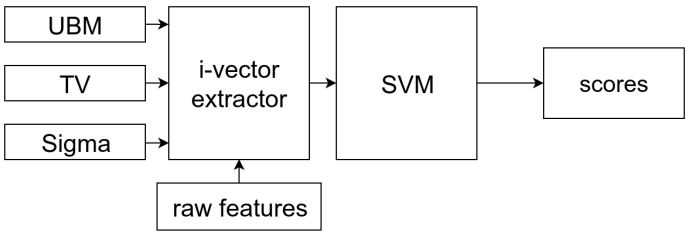

Fig. 1. I-vector system configuration

3 TDNN D-Vector System

3.1 About

In proposed LID system we use d-vectors instead of i-vectors as the input language features for the SVM classifier. D-vectors are obtained using our best for ASR acoustic model. We hypothesize that amount of uncertainty in the neural

network output, produced by non-target language, leads to the shift in hidden layer activations behavior relatively target language (Russian in our case). Therefore their averaged representations (d-vectors) must give good discriminative feature for binary identification task, but not for multiclass identification.

3.2 Extracting improved D-Vector

Assume we have a set of raw features of the whole utterance $X_{utt} = \{x_1, \ldots, x_T\}$ $X_{utt} \in \mathbb{R}^{FxT}$ and the last hidden layer activations corresponding to raw features $H_{utt} = \{h_1, \ldots, h_T\}$ $H_{utt} \in \mathbb{R}^{LxT}$ from TDNN where F is a raw feature dimension, L is a number of neurons in the last hidden layer and T is a number of frames in the utterance. Next we compute mean and standard deviation of H_{utt} and concatenate them into one single vector. Now we have improved version of d-vectors per utterance as compared with [7].

TDNN is chosen as d-vector extractor because it can model long term temporal dependencies with training times comparable to standard feed-forward DNNs and shows better Word Error Rate (WER) in speech recognition tasks [6]. Training TDNN is done using Kaldi toolkit [9].

Scheme below depicts an example of TDNN architecture with sub-sampling $\{-3, 3\}$, $\{-1, +1\}$ and $\{-2, +1\}$ applying to its hidden layers correspondingly (Fig. 2).

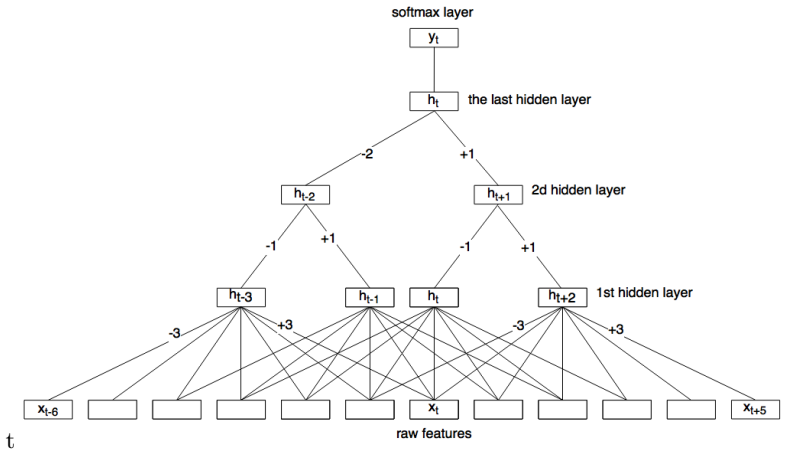

Fig. 2. An example of TDNN architecture with sub-sampling. D-vector is extracted from the last hidden layer activations

3.3 Configuration

To make the comparison clear d-vector is configured as in subsection 2.2 of I-Vector Baseline System. However, a slight difference is still present — Prin-

cipal Component Analysis (PCA) was used to whiten d-vectors and reduce the dimensionality (Fig. 3).

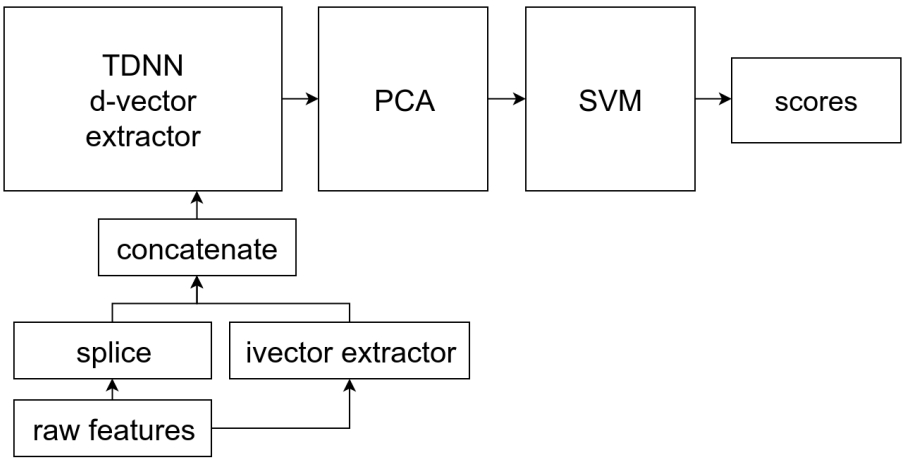

Fig. 3. TDNN d-vector system configuration

4 Experiment setup

4.1 Dataset

ASR TDNN model, i-vector UBM, TV & Sigma parameters were obtained on the proprietary Russian corpus collected from the microphone and telephone speech.

NIST Language Recognition Evaluation 2007 (LRE07) is a popular corpora and it contains Russian language in train and test sets. We have used the next 11 languages to train the classifier: Russian, Arabic, Bengali, Chinese (Min), Spanish (Mexican), Tamil, Thai, Chinese (Taiwan, Wu, Cantonese), Hindustani(Urdu). It was prepared 26 items for open-set test: Arabic, Bengali, Chinese (Cantonese, Mainland, Taiwan, Min, Wu), English (American, Indian) Farsi, French, German, Hindustani (Hindi, Urdu), Indonesian, Italian, Japanese, Korean, Punjabi, Russian, Spanish (Caribbean, non-Caribbean), Tagalog, Tamil, Thai, Vietnamese.

We run Voice Activity Detector over train set to split long durations of source audio files into small parts (Fig. 4). I-vector and d-vector extractors results were considered at this training corpora.

4.2 I-Vector

13-dimensional MFCCs with double deltas and no normalization were used as input to the i-vector extractor. UBM has 512 GMMs. Output is 400-dimensional

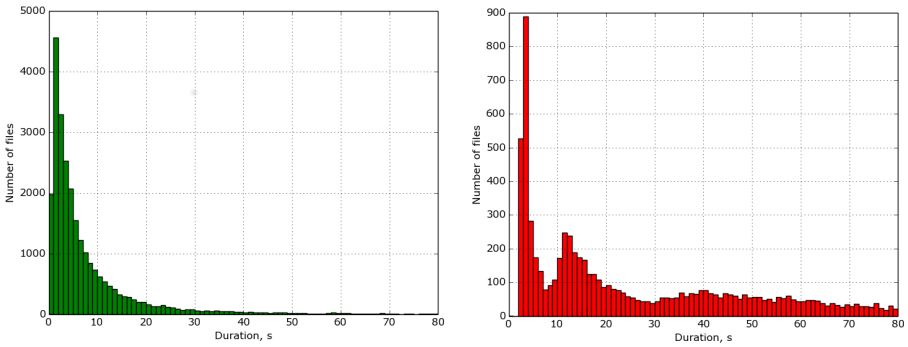

Fig. 4. Train (left) and test (right) durations histogram

vector. It's about 20 hours of audio in UBM training and 50 hours for TV & Sigma.

4.3 D-Vector

Perceptual Linear Prediction (PLP) with 13 cepstral coefficients, without cepstral mean and variance normalization, were used as input to the TDNN at the each time step. Also we concatenated a 100-dimensional i-vector with the PLP input.

TDNN consists of 6 nonlinear hidden layers and we use the following subsampling scheme (Tab. 1) on the first three layers only.

Table 1. Sub-sampling scheme

Layer	Input context
1	{-4, 4}
2	{-2, 2}
3	{-4, 4}

Neural network was trained using stochastic gradient descent after that sequence training based on a state-level variant of the Minimum Phone Error (MPE) criterion was applied to get the final acoustic model.

The output dimension of d-vector after PCA is 400. It's about 450 hours of audio for the TDNN train.

5 Results

The research purpose was the investigation of the short utterance LID. Here are presented DET plots of i-vector and d-vector open-set General LR 3 sec test on

Russian language (Fig. 5). We used a lot of short audio fragments (after VAD) in train and didn't expect a great work of i-vectors & d-vectors at 10s and 30s (Tab. 2). But as you can see d-vectors outperforms at 10s and 30s i-vectors too.

Table 2. I-vector and D-vector EER on Russian open-set test, %

System	3s	10s	30s
i-vector	28.08	24.09	22.51
d-vector	22.42	21.23	20.34

We found that d-vectors need less data to reach the best result while i-vectors take more data to train SVM model. It makes SVM very slow because the high number of support vectors increases the computing time.

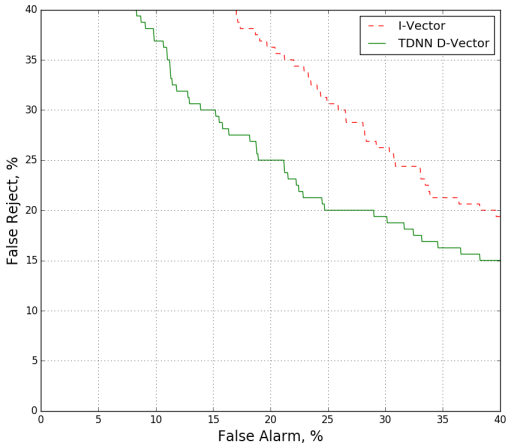

Fig. 5. I-vector (red, dashed) and D-vector (green) DET curves on Russian open-set test

All the results, scores and other meta information about the experiments are stored in the Testarium — research tool and experiment repository [10]. Using this tool we made a grid search of SVM gamma, C, data limits and other parameters of our setup.

6 Conclusions

In this paper we have successfully applied TDNN framework to model acoustic features within language verification scenario. The feed-forward architecture of

TDNN allows training and adapting parameters faster than more sophisticated recurrent nets, whereas gathering sufficiently wide context is important to make the system robust against outliers. The accurate modeling of target language phonemes seems to be crucial step for gaining performance when the small portion of acoustic information is given. Furthermore an interesting observation was concluded: TDNN d-vectors using in training are not so sensitive to audio durations in contrast to i-vectors.

Though only Russian language was the case study, we believe in the same effect for other languages. Also we want to build multiple TDNN acoustic models for other languages and concatenate their d-vectors to reach the best performance. The validation addressed to the future work.

References

1. Dehak, N., Kenny, P., Dehak, R., Dumouchel, P., Ouellet, P.: Front-end factor analysis for speaker verification. In: IEEE Transactions on Audio, Speech, and Language Processing, vol. 19, pp. 788-798. IEEE Press (2011)
2. Martinez, D., Plchot, O., Burget, L., Glembek, O., Matejka, P.: Language Recognition in iVectors Space. In: 12th Annual Conference of the International Speech Communication Association (INTERSPEECH), vol. 2011, no. 8, pp. 861864. ISCA, Florence (2011)
3. Graves, A., Mohamed, A., Hinton, G.: Speech Recognition with Deep Recurrent Neural Networks. In: International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6645-6649. IEEE Press, Vancouver (2013)
4. Gonzalez-Dominguez, J., Lopez-Moreno, I., Sak, H., Gonzalez-Rodriguez, J., Moreno, P.: Automatic Language Identification using Long Short-Term Memory Recurrent Neural Networks. In: 16th Annual Conference of the International Speech Communication Association (INTERSPEECH). ISCA, Dresden (2015)
5. Zazo, R., Lozano-Diez, A., Gonzalez-Dominguez, J., Toledano, D., Gonzalez-Rodriguez, J.: Language Identification in Short Utterances Using Long Short-Term Memory (LSTM) Recurrent Neural Networks. PLoS ONE 11(1): e0146917 (2016)
6. Peddinti, V., Povey, D., Khudanpur, S.: A time delay neural network architecture for efficient modeling of long temporal contexts. In: 16th Annual Conference of the International Speech Communication Association (INTERSPEECH). ISCA, Dresden (2015)
7. Variani, E., Lei, X., McDermott, E., Moreno, I.L., Gonzalez-Dominguez, J.: Deep Neural Networks for small footprint Text-Dependent Speaker Verification. In: International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE Press, Singapore (2014)
8. Kenny, P., Oullet, P., Dehak, V.N., Gupta, Dumouchel, P.: A Study of Interspeaker Variability in Speaker Verification. In: IEEE Transactions on Audio, Speech, and Language Processing, vol. 16, pp. 980-988. IEEE Press (2008)
9. Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlicek, P., Qian, Y., Schwarz, P., Silovsky, J., Stemmer, G., Vesely, K.: The kaldi speech recognition toolkit. In: IEEE 2011 Workshop on Automatic Speech Recognition and Understanding. Hawaii (2011)
10. Testarium. Research tool and experiment repository, `http://testarium.makseq.com`