BULLETIN OF NATIONAL ACADEMY OF SCIENCES
OF THE REPUBLIC OF KAZAKHSTAN
ISSN 1991-3494
Volume 3, Number 379 (2019), 36 – 43 **https://doi.org/10.32014/2019.2518-1467.66**

UDC 004.932.75

**O. Zh. Mamyrbayev[1], M. Othman[3], A. T. Akhmediyarova[1],
A. S. Kydyrbekova[2], N. O. Mekebayev[2]**

[1]Institute of Information and Computational Technology, Almaty, Kazakhstan,
[2]al-Farabi Kazakh National University, Almaty, Kazakhstan,
[3]University Putra, Malaysia.
E-mail: morkenj@mail.ru, mothmanupm@gmail.com,
aat.78@mail.ru, kas.aizat@mail.ru, nyrbapakas.aizat@mail.ru

VOICE VERIFICATION USING I-VECTORS AND NEURAL NETWORKS WITH LIMITED TRAINING DATA

Abstract. This study proposes an approach to voice identification based on neural networks (DNN) for i-Vector. Modern voice identification systems based on DNN use large amounts of labeled training data. Using the LRE i-Vector Machine Learning Challenge restricts access to ready-to-use i-Vector for learning and testing the voice identification system. This poses unique challenges in developing DNN-based voice identification systems, since optimized external interfaces and network architectures can no longer be used. We propose to use the training i-Vectors to train the initial DNN to identify the voice. Next, we present a novel strategy for using this initial DNN to strip the language labels of the inappropriate set from the development data. The final DNN for voice identification is trained using the original training data and the estimated out-of-set language data. We show that augmenting the training set with out-of- set labels leads to a significant improvement in voice identification performance.

In this paper, we studied the possibility of using neural networks for speech identification. In particular, standard approaches to speech recognition were considered, the concept of an artificial neuron as an object used in speech identification was defined. A speech recognition option using a neural network was investigated, and steps were presented to perform this task. Accuracy using neural networks with limited learning data and a higher i-vector dimension is superior to others with a score of 92.1%. From this study, we can conclude that the size of the UBM and the dimension of the i-vector affect the accuracy of voice identification based on the i-vector.

Keywords: voice identification, i-Vector, deep neural network.

1. Introduction. The task of voice identification (VID) includes the automatic identification of the language in which a given speech utterance was spoken. Voice identification systems are used in a variety of applications: multilingual language translation, emergency or consumer call routing, surveillance and security applications [1, 2]. Recently, voice identification has received considerable attention, primarily due to the several NIST language Recognition Evaluations (LRE) and also partly due to programs such as the DARPA Robust Automatic Transcription of Speech (RATS).

The low-dimensional representation of speech in the so-called common variability space (TVS) obtained by factor analysis (FA) in the middle space of the Gaussian mixture model (GMM) vectors has been popular in modern voice verification systems. This representation, which is widely known as the i-vector, maps utterances of arbitrary duration into a space of small and fixed dimension [3]. Recently, I-Vector-based technologies have become the latest technologies in the field of voice identification, closely following similar developments in the field of speaker identification (SID) [4-6]. More recently, voice identification approaches based on the Deep Neural Network (DNN) and Convolutional Neural Network (CNN) have become increasingly popular, and have been reported to offer comparable, and in many cases higher performance compared with the I-Vector-based VID methods formulated using a Gaussian Mixture Model Universal background Model Framework (GMM-UBM) based framework [7-9, 28].

The I-vector contains a significant amount of information, and therefore it has been found useful in various applications. Language and especially phonetic details are among the most important information contained in this vector. Thus, several methods used this vector to identify the language [4-8]. The use of this vector for age [9, 10] and emotion estimation [11] along with accent recognition [12, 13, 25] are among other applications of this type of representation. All of these applications, as well as its main voice recognition application, show the large amount of information it contains, so it seems reasonable to reduce the unrelated information of this vector for better speaker recognition. Various methods have been proposed for modeling speakers in i-vector space [14, 15]. The Gaussian PLDA is the most common that ignores the process by which i-vectors are extracted (that is, a point estimate of hidden variables in the FA model) and instead pretends that they are random vectors generated using the PLDA method. Although in most cases, PLDA provides higher accuracy. A number of channel compensation methods such as SN-LDA, SN-WLDA, and WLDA have been proposed to improve the performance of the i-vector based on the CSS system [16]. In General, the best modeling technique for voice recognition is one that takes into account only the information related to the speaker. We propose to use the training i-Vectors to train the initial DNN to identify the voice. Next, we present a novel strategy for using this initial DNN to strip the language labels of the inappropriate set from the development data. The final DNN for voice identification is trained using the original training data and the estimated out-of-set language data. We show that augmenting the training set with out-of- set labels leads to a significant improvement in voice identification performance.

Our approach makes it possible to obtain very competitive costs (defined by NIST) in the amount of 26.56 and 25.98, respectively, for the subgroups of progress and task estimates. Our approach makes it possible to obtain very competitive costs (defined by NIST) in the amount of 26.56 and 25.98, respectively, for the subgroups of progress and task estimates. Since the amount of training data is very limited (300 i-Vectors per language), this study outlines a successful recipe for DNN-based VID using very limited resources.

This study proposes a new approach to VID using DNN. We suggest training the initial DNN for VID using i-Vectors. The rest of the paper is organized as follows: Section 2 explains parts of the i-vector based voice verification system, and then the next section briefly describes some of the related work. In 3 - the problem of learning machine LRE. Section 4 describes Deep neural network (DNN). The experiments and results are presented in section 5, and finally the conclusions are presented in Section 6.

2. Voice identification using the i- vector.

2.1. i-Vector extractor. The i-vector extractor is a system that converts a speech utterance of arbitrary duration into a vector of fixed length [1]. For this purpose, the Baum-Welch statistic should be extracted from the universal background model (UBM), which can be a GMM model or a hidden Markov model (HMM). In this system, the average super vector for an utterance can be modeled as follows:

$$s = m + Tw\ . \tag{1}$$

where m is the session-independent and channel-independent components of the mean supervector obtained froFm UBM, T is the basis matrix covering the subspace encompassing the important (both for the speaker and the session) in the super vector space, and w is the standard, normally distributed hidden variable. For each observation sequence representing an utterance, our i-vector is a point estimate of the A-a posteriori maximum (MAP) for the hidden variable w. Our I-vector extractor training procedure is based on the effective implementation proposed in [24, 26].

First it is necessary to build UBM using the utterances in the development set to use this method. Then, using this model, the zero and first order statistics (the Baum – Welch statistics) are calculated from the developmental utterances and used to evaluate the T matrix. The average UBM super vector is typically used for the super vector m, and this is not necessary for evaluation. After training the model parameters, the calculated statistics from each statement are used to extract the corresponding i-vectors to be used for the next steps. Unlike the JFA method, in the i-vector method, there is no circuit for removing channel effects. Therefore, it is necessary to remove channel effects separately after extracting the i-vector. Various methods have been adopted for this purpose, and the two most common methods are described in the next section.

2.2. DNN tandem extraction. The Deep Neural Network system serves as an acoustic modeling network used to extract the tandem function of a phonetic level. First, the DNN acoustic model is trained using acoustic characteristics and phonetic label data. Then the MFCC function is assigned to the DNN model, and we can extract the given probabilities of the associated states of triphons. We use VAD method on top of it to create low-dimensional tandem functions. Finally, the tandem function is combined with the MFCC function to generate a hybrid function.

The number of input - x_n and output- Y neurons is known. Each of the number of input neurons corresponds to one set of numbers. And on the output layer there is only one neuron, the output of which corresponds to the voice recognition value. Given n data points $\{x_1, x_2, .. x_n, \}$ sampled from the underlying submanifold Y, it is possible to construct acoustic modeling in the DNN system, in figure 1. x_n – n-th input value.

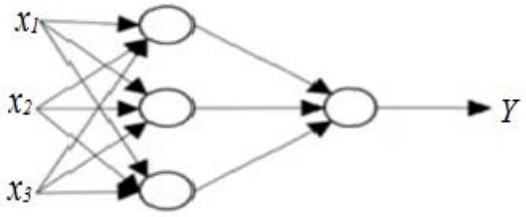

Figure 1 – Neural network with one feedback DNN system

2.3. Related works. As a rule, there are two types of errors in voice verification systems. First, it is a false reject error. The main reason for this type of error is inconsistency, which can be caused by various reasons, a change in the state of the speaker (e.g. stress [17], haste, etc.) or even intentional changes in the pronunciation, such as whispering. All these differences in speech conditions during testing and training increase the variance within the class. This leads to an increase in false deviation errors. Various compensation methods have been proposed to reduce the impact of such changes:

1. NAP: Nuisance Attribute Projection (NAP) removes nuisance subspaces.

The orthogonal projection, which depends only on the dynamics, is performed in the extra channel space using the projection matrix. This method was used both in the average super vector and in the i-vector spaces [3,18, 19] to minimize false deviations and false errors of adoption at the training stage of the support vector machine (SVM). In other words, this method scales the space to remove dimensions with a high intraclass dispersion. Consequently, this method reduces the intraclass variance [3, 20, 21].

2. JFA: The purpose of the joint factor analysis (JFA) method is to model channel effects and dynamics separately in two different subspaces. In this method, the average super vector is decomposed into two separate super vectors, where one relates to the dynamics and the other to the channel. After this decomposition, the super vector representing the channel effects is discarded and, therefore, the channel effects are excluded from the solution [22, 23, 27].

3. LDA: this method is used in pattern recognition to reduce differences between classes and to increase differences between classes. This method is also used to reduce the dimension in the classification. Since this method reduces intra-class variations, it can be used to reduce channel effects [3].

4. PLDA: this is a probabilistic method whose purpose is to decompose the i-vector space into two separate subspaces for the dynamics and the channel. This method is similar to JFA in some respects.

The second type of errors in voice verification systems is a false acceptance error. The main reason for this error is the presence of similarities between the vectors of different speakers. These similarities have different causes and can be grouped into two categories: the first is a set of similarities associated with the speaker. This type of similarity allows us to identify the speaker by his vectors. In the case of two loudspeakers with very similar voices (whether in a natural way or by emulating / converting a voice), the similarity of interconnected voices between their respective vectors increases, and this, in turn, leads to an increase in the frequency of erroneous acceptance errors. The second category of similarities is a set of non-speaking vector similarities that can be associated with different sources, such as channel similarities, a microphone, or similarities in a spoken text. As far as we know, no method has been proposed to reduce the effects of this kind of similarity. This study aims to study information fully connected direct neural network using i-Vector to reduce their impact.

3. LRE I-Vector machine learning challenge. The NIST LIR i-Vector Machine Learning challenge is aimed at developing new VID techniques employing i-Vectors for conversational/narrow-band broadcast speech [12]. The challenge consists of 3 distinct data sets: a training set with 300 i-Vectors per language, corresponding to each of the 50 insert target languages , a test set and a development set. The speech utterances corresponding to the training-set i-Vectors were chosen so that their durations exhibit a log-normal distribution with a mean duration of 35.15s. The development and test-set were unlabeled, and also contained i-Vectors corresponding to out-of-set languages.

The primary task of the challenge is to identify the corresponding language of a test i-Vector, or to assign it as an "out of set" (a single label corresponding to the out-of-set languages), if the i-Vector is deemed not to correspond to any of the 50 in-set languages. The test-set was divided randomly into progress subset (30% of the test-set) and evaluation subset (70% of the test-set). The challenge rules did not allow using the outputs corresponding to other test i-Vectors to be used in any way in evaluating the output of a given test i-Vector [12]. The performance was assessed using the following cost function defined by NIST,

$$C_{cost} = \frac{(1-P_{OOS})}{n} * \sum_k^n P_{error}(k) + P_{OOS} * P_{error}(OOS) \quad (2)$$

$$P_{error}(k) = \left(\frac{no.of\ errors\ for\ class\ k}{no.of\ trials\ for\ class\ k}\right), n = 50, P_{oos} = 0{,}23 \quad (3)$$

The cost for progress subset and evaluation subset were evaluated by NIST.

4. Deep Neural Network (DNN) for the VID using i-VECTORS. In [8], large amounts of labeled training data were used to initially train an ASR system, which was then used to generate the senone alignments to train a CNN. In this framework, the CNN/DNN replaced a GMM-UBM to compute the posteriors needed for i-Vector extraction, and subsequent steps of i-Vectors based VID essentially remained unchanged. The front-end for CNN/DNN training utilized filter-bank outputs. A more direct approach to VID using DNNs was outlined in [10], where the output layer nodes correspond to the in-set languages along with a single node for out-of-set languages. This approach also utilized large amounts of labeled training data and employed PLP features. Our study proposes to train a DNN for VID using i-Vectors unlike existing CNN/DNN based VID techniques. To account for the out-of-set data present in the test-set, we adopt a novel 2-step DNN training strategy, where the initial DNN is trained using only in-set labeled training data. The initial DNN is then used to estimate out-of-set labels from the development data. Next, we train a second DNN for VID with both in-set and estimated out-of-set labels. Moreover, since the amount of training data is very limited, we also investigate and comment on the efficacy of some popular techniques to overcome the issue of limited training data. We have used the PDNN toolkit for the experiments reported in this study [16]. The following subsections describe details of the proposed DNN based approach for VID using i-Vectors.

4.1. DNN training for in-set languages using i-Vectors. We use a fully connected feed-forward neural network for VID in the experiments reported here. The hidden-layer units use a sigmoid activation function. The output layer is a softmax layer with output nodes corresponding to the in-set languages. Let the target classes be represented as Y , W and b be the weight matrix and bias vector respectively. The output at the ith node of the output layer, corresponding to the input vector x can be expressed as,

$$P(Y = i|x, W, b) = softmax_i(W_x + b) = \frac{e^{W_i x + b_i}}{\sum_j e^{W_j x + b_j}} . \quad (4)$$

Next, the predicted language Y_{pred} is evaluated as:

$$Y_{pred} = argmax_i P(Y = i|x, W, b) \quad (5)$$

The highest score corresponding to the label evaluated in (5) can be obtained using

$$Y_{max} = max_i P(Y = i|x, W, b) \quad (6)$$

the number of tutorial data labeled is very limited, to explore several techniques to examine if they offer any improvement in the performance of our DNN based VID system. Of particular interest are techniques of dropout, and unsupervised generative pretraining, which have been reported to be very

effective when training DNN with limited amounts of data [17, 18]. In the dropout technique, certain units of the hidden layers together with their connections are dropped randomly. This, in turn, minimizes the overfitting in the DNN by reducing the co-adaptation of the network parameters [17]. Unsupervised generative pretraining allows the DNN to use more information from the training data than contained within the labels alone [18]. It has been reported to prevent overfitting by introducing regularization [19], and has been widely used in DNN based ASR techniques [18, 20]. Additionally, L2-norm regularization is also applied since it reduces overfitting by preventing the network weight parameters from assuming very large values.

4.2. Estimating out-of-set labels for training from development data. The DNN trained on the in-set languages is used to estimate labels corresponding to out-of-set languages from the development data. Specifically, corresponding values of $У_{max}$ are computed using (6) for the i-Vectors of the development set. Next, all i-Vectors with the corresponding scores ymax for some suitable threshold θ (computed using the development set) are assigned the label "out of set" (label corresponding to out-of-set data).

4.3. DNN training with "in-set" and "out-of-set" labels. In the second stage, a new DNN is trained using the i-Vectors for the in-set languages and the i-Vectors corresponding to the out-of-set languages estimated from the development data. Thus, this DNN has an extra node in the output layer compared to the initial DNN to account for the out-of-set languages. The same strategies as mentioned previously in Section 4.1 are applied to make optimal use of the limited training data. The language labels for the test i-Vectors were assigned using (5). Figure 2 presents an overview of the proposed DNN based VID approach. Since the amount of total training data available increased after including the labels for out-of-set languages, we also explored varying the architecture of the DNN compared to what was used for the initial DNN mentioned in Section 4.1. Specifically, we explored using more units in the hidden layers as well as deeper networks.

5. Experiments, results and discussions. An initial DNN for VID was trained using the i-Vectors of the training set. We employed the mini-batch Stochastic Gradient Descent (SGD) algorithm with a mini-batch size of 256, and backpropagation to train the DNN [21]. The DNN had 2 hidden layers with 1024 units each. The input layer had 400 nodes corresponding to the 400-dimensional i-Vectors provided by NIST for the challenge. A dropout factor of 0.2 was used for each of the 2-hidden layers of the DNN. The output layer had 50 nodes corresponding to all the in-set languages. For each of the in-set languages, 10% of the labeled training data (30 i-Vectors per language) was randomly set aside as a held-out set to monitor DNN training.

Table 1 – Cost obtained using the initial DNN (DNN1) trained on the in-set languages for output-labels without (No out of set), and with the out-of-set (With out of set) labels

LID System	Output-label	Cost (progress subset)
DNN1	No-out of set	37.38
DNN1	With-out of set	32.71

As can be observed from results in Table 1, detecting out-of-set languages in the output offers a big improvement by lowering the cost by 4.67% absolute (12.49% relative).

5.1. Effect of out-of-set detection on the cost. From (1), it is clear that detecting "out of set" labels correctly is critical to obtaining a competitive cost function value on the test-set. In table 1, the results for two different sets of output-labels are shown for the progress subset.

Two-step DNN training for VID obtained using the same initial DNN (DNN1) trained with the in-set languages. The output-labels for "No out of set" were obtained using (5) by using the estimated in-set output- labels for the test i-Vectors. Next, using (6), y_{max} was estimated for each i-Vector of the test-set. To obtain the output-labels for "With out of set", any $У_{pred}$ with the corresponding $У_{max} <$, for some suitable threshold estimated using the development data, was assigned the output-label "out of set".

Comparing results of tables 1 and 2, including the estimated out-of-set labels in DNN training offered a signifi- cant reduction in cost by 5.89% absolute (18% relative), when the cost obtained using DNN2 1024 (26.82) (table 2) is compared against the "with out of set" DNN1 from Table 1. Increasing the units in the hidden layers offered a marginal re- duction in cost as evident by the results for DNN2 2048. Both

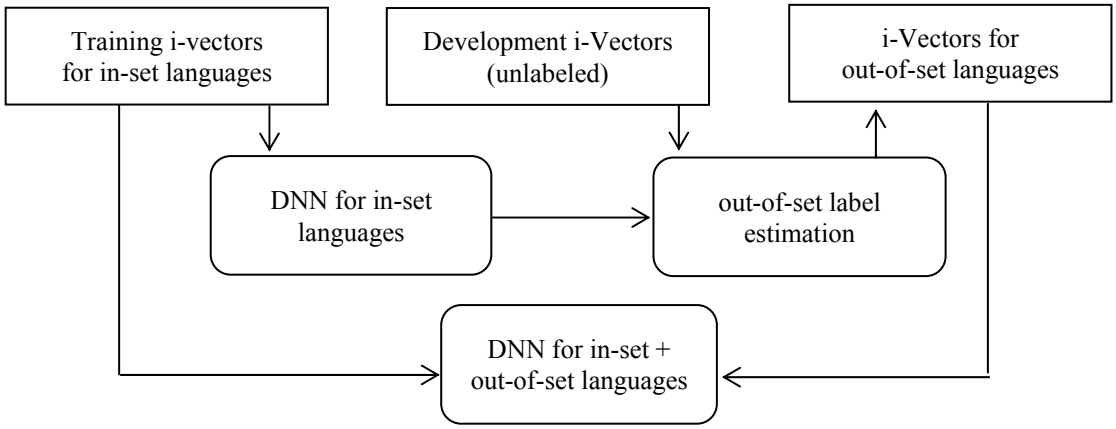

Figure 2 – 2-step DNN training for VID using i-Vectors

Table 2 – Cost obtained using the DNN with 1024 (DNN2 1024),
and 2048 (DNN2 2048) units per hidden-layer trained using the augmented (in set + estimated out-of-set)
training set compared against a CDS baseline system

LID System	Cost (progress subset)
DNN2 1024	26.82
DNN2 2048	26.56
CDS (baseline)	39.59

DNN2 1024 and DNN2 2048 are significantly better than the CDS baseline system by over 32% (relative). DNN2 2048 obtained a cost of 25.98 on the evaluation subset of the NIST LRE i-Vector ML challenge. The results obtained using the proposed DNN based VID approach are comparable to SVM based VID techniques, and offer further improvements in system fusion of the two approaches [22]. We also explored using more than 2 hidden layers for training the DNNs. Adding more layers to the DNN caused degradation in VID performance since the limited training data was insufficient to estimate the new additional parameters.

5.3. Investigating the efficacy of dropout and generative pretraining for DNN training. We investigated the use of dropout and unsupervised generative pretraining for DNN training with limited resources. It was observed that VID performance improved with progressively higher values of the dropout factor from 0.1 to 0.5, after which it started to degrade. A dropout factor of 0.5 for each of the 2-hidden layers achieved the DNN results shown in Table 2. When DNN2 1024 (trained on in-set and estimated out-of-set labels) was retrained after applying unsupervised generative pretraining (using the unlabeled development set), the cost on the progress subset degraded from 26.82 to 28.46. Evidently, pretraining did not offer any improvement to the pro- posed DNN based approach for VID. Unlike optimal acoustic features like filter-bank outputs, i-Vectors offer a more compact representation of a speech utterance. We hypothesize that this causes i-Vectors to lose much of the additional information compared to acoustic features such as filter-bank outputs. Since unsupervised generative pretraining works by utilizing the additional information contained within the features [18], it probably fails to access such information when i-Vectors are used. The limited amount of available development data could be another reason why unsupervised generative pre-training failed to offer any improvement.

6. Conclusions. This study presented a novel approach to VID using very limited training data. To explicitly detect the out-of-set lan- guages, we proposed a novel 2-step DNN training strategy, in which a DNN for VID trained using the in-set labeled training data was used to estimate out-of-set labels from an unlabeled development set. The training set augmented with the out-of-set labels was then used to train a second DNN for VID that could also detect an out-of-set language in addition to the in-set languages. This was shown to offer significantly better VID performance than a DNN utilizing only the in-set labeled data

for training. Also, the proposed approach significantly outperformed a CDS based baseline system, and obtained very competitive costs on the progress and evaluation subsets of the LRE i-Vector Machine Learning Challenge.

This study has therefore outlined a successful recipe for DNN based VID using very limited resources.

Acknowledgements. This work carried out in the framework of the project IRN AP05131207 “Development of technologies for multilingual automatic speech recognition using deep neural networks”.

О. Ж. Мамырбаев[1], Мохмед Отман[3], А. Т. Ахмедиярова[1], А. С. Кыдырбаева[3], Н. О. Мекебаев[3]

[1]Института информационных и вычислительных технологий, Алматы, Казахстан,
[2]Казахский национальный университет им. аль-Фараби, Алматы, Казахстан,
[3]Университета Путра, Малайзия

ВЕРИФИКАЦИЯ ГОЛОСА С ИСПОЛЬЗОВАНИЕМ I-ВЕКТОРЫ И НЕЙРОННЫХ СЕТЕЙ С ОГРАНИЧЕННЫМИ ДАННЫМИ ОБУЧЕНИЯ

Аннотация. В этом исследовании предлагается подход к идентификации голоса на основе нейронных сетей (DNN) для i-Vector. Современные системы идентификации голоса на базе DNN используют большие объемы помеченных данных обучения. Используя LRE i-Vector, Machine Learning Challenge ограничивает доступ только к готовым к использованию i-Vector для обучения и тестирования системы идентификации голоса. Это создает уникальные проблемы при разработке систем идентификации голоса на основе DNN, поскольку оптимизированные внешние интерфейсы и сетевые архитектуры больше не могут использоваться. Мы предлагаем использовать обучающие i-Vectors для обучения начального DNN для идентификации голоса. Далее мы представляем новую стратегию использования этого начального DNN, чтобы лишить языковые метки несоответствующего набора из данных разработки. Окончательный DNN для идентификации голоса обучается с использованием исходных данных обучения и оценочных данных языка, не установленных. Мы показываем, что добавление тренировочного набора с несоответствующими метками приводит к значительному улучшению производительности идентификации голоса.

В данной работе была исследована возможность применения нейронных сетей для идентификации речи. В частности, были рассмотрены стандартные подходы к распознаванию речи, определено понятие искусственного нейрона, как объекта, используемого в идентификации речи. Был исследован вариант распознавания речи с помощью нейронной сети, и представлены шаги для выполнения этой задачи.

Ключевые слова: идентификация языка, i-Vector, глубокая нейронная сеть.

Information about authors:

Mamyrbayev O. Zh., institute of Information and Computational Technology, Almaty, Kazakhstan; morkenj@mail.ru; https://orcid.org/0000-0001-8318-3794

Othman M., mothmanupm@gmail.com; University Putra, Malaysia;

Akhmediyarova A. T., institute of Information and Computational Technology, Almaty, Kazakhstan; aat.78@mail.ru; https://orcid.org/0000-0003-4439-7313

Kydyrbekova A. S., al-Farabi Kazakh National University, Almaty, Kazakhstan; kas.aizat@mail.ru;

Mekebayev N. O., al-Farabi Kazakh National University, Almaty, Kazakhstan; nyrbapakas.aizat@mail.ru; https://orcid.org/0000-0002-9117-4369

REFERENCES

[1] Ambikairaj E., Li H., Van L., Yin B., Sethu V. Language Identification: A Tutorial // IEEE Circuit and Systems Journal. 2011. Vol. 11, N 2. P. 82-108.

[2] Mutusami J.C., Barnard E., Cole R. A Review of Automatic Language Identification // Signal Processing Log. IEEE. 1994. Vol. 11, N 4. P. 33-41.

[3] Dehak N., Kenny P., Dehak R., Dumushel P., Owel P. The frontal analysis of factors for the verification of speakers // IEEE Trans Audio, Speech, Lang Process 2011. 19: 788-98.

[4] Dehak N., Torres-Carrasquillo P.A., Reynolds D.A., Dehak R. Language recognition through i-vectors and reduction of dimension // Interspich. 2011. P. 857-60.

[5] D'Haro Enriquez L.F., Glembek O., Plchot O., Mateyzhka P., Sufifar M., Cordova Errald Rd and others. Recognition of phonotaxic language using i-vectors and the number of posterioriogram phonemes. 2012.

[6] D'Haro L., Cordoba R., Caraballo M., Pardo J.M. Recognition of a low-resource language using a combination of phonogram phonograms, acoustic and glottal i-vectors // Q: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 2013. P. 6852-6.

[7] Lee M., Narayanan S. Simplified controlled i-vector modeling with application for reliable and effective language identification and speaker verification // Comput Speech Lang. 2014. 28: 940-58.

[8] Martinez D., Pleshot O., Burget L., Glembek O., Mateika P. Language recognition in the space of i-vectors // Proceedings of Interspeech. Florence, Italy, 2011. P. 861-4.

[9] Bahari M.Kh., Maclaren M., Van Leeuwen D. Estimation of age from telephone speech using i-vectors // Interspich, 2012. P. 506-9.

[10] Bahari M.H., McLaren M., van Leeuwen D.A. Estimation of the age of speakers using i-vectors // Eng Appl Artif Intell. 2014. 34: 99-108.

[11] Xia R., Liu Ya. Using the i-vector space model for recognizing emotions // Interspich. 2012.

[12] Bahari M.Kh., Saeidi R., Van Leeuwen D. Accent recognition using the i-vector, Gaussian mean supervisor and Gaussian rear supervisor of probability for spontaneous telephone speech // Q: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 2013. P. 7344-8.

[13] Behravan H., Hautamäki V., Kinnunen T. Detecting foreign accent in spoken Finnish using i-vectors // Interspich. 2013. P. Fourteenth.

[14] Kenny P. Bayesovsky speaker checks with heavy-tailed priors // Odyssey 2010 – Language Recognition and Speaking Workshop, Brno, Czech Republic. 2010.

[15] Burget L., Plchot O., Kumani S., Glembeck O., Mateik P., Brammer N. Discriminatory trained probabilistic linear discriminant analysis for speaker verification // Q: 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 2011. P. 4832-5.

[16] Kanagasundaram A., Din D., Sridharen S., Maclaren M., Vogt R. I-vector speaker recognition using advanced methods for compensating channels // Comput Speech Lang. 2014. 28: 121-40.

[17] Hansen J.H. Analysis and Compensation of Speech Under Stress and Noise for Environmental Sustainability in Speech Recognition // Speech Commun. 1996. 20: 151-73.

[18] Campbell V.M., Sturim D.E., Reynolds D.A., Solomonov A.A. Verification of SVM loudspeakers using the GMM supervector core and NAP variability compensation // Q: 2006 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2006 Materials. Toulouse; 2006. P. II.

[19] Solomonov A., Quillen S., Campbell V.M. Channel compensation for recognition of SVM dynamics // Odyssey 04; 2004. P. 219-26.

[20] Solomonov A., Campbell V.M., Boardman I. Achievements in the field of compensation channels for the recognition of speakers SVM // B: ICASSP. 2005. Vol. 1. P. 629-32.

[23] Hatch A.O., Kayarekar S.S., Stolke A. Intraclass covariance normalization for recognition of speakers based on SVM // Interspich. 2006.

[21] Dehak N., Kenny P., Dehak R., Glembek O., Dyumuchel P., Burget L. and others. Support vector machines and joint factor analysis for speaker verification // IEEE International Conference on Acoustics, Speech and Signal Processing. 2009. ICASSP 2009; 2009. P. 4237-40.

[22] Kenny P., Bulian G., Owel P., Dumushel P. Dynamicity and variability of the session when testing the speaker on the basis of GMM // IEEE Trans Audio, Speech, Lang Process. 2007. 15: 1448-60.

[23] Kenny P., Oueleth P., Dehak N., Gupta V., Dumushel P. Study on the variability of inter-speaker speakers when verifying speakers // IEEE Trans Audio, Speech, Lang Process. 2008. 16: 980-8.

[24] Ondrey Glembeck, Lukas Burghet, Pavel Mateika, Martin Carafiat and Patrick Kenny. (2011) "Simplification and optimization of i-vector extraction" // ICASSP, IEEE Int. Conf. Acoust. The process of the speech signal. P. 4516–4519.

[25] Mamyrbayev O.Z., Kunanbayeva M.M., Sadybekov K.S., Kalyzhanova A.U., Mamyrbayeva A.Z. Ont of the methods of segmentation of speech signal on syllables // Bulletin of the National academy of sciences of the Republic of Kazakhstan. 2015. Vol. 2. P. 286-290.

[26] Ali B.B. , Wojcik W., Mamyrbayev O., Turdalyuly M., Mekebayev N. Speech Recognizer-Based Non-Uniform Spectral Compression for Robust MFCC Feature Extraction // Przeglad Elektrotechniczny. 2018. Vol. 94, Publ. 6. P. 90-93.

[27] Mamyrbaev O.Z., Muhsina K.Z. Analysis of existing systems for determination of tonnity of text // News of the National academy of sciences of the Republic of Kazakhstan. Series physico- mathematical. 2017. Vol. 5, Publ. 315. P. 149-155.

[28] Mamyrbayev O., Turdalyuly M., Mekebayev N., Alimhan K., Kydyrbekova A., Turdalykyzy T. Automatic Recognition of Kazakh Speech Using Deep Neural Networks // Intelligent Information and Database Systems 11th Asian Conference, ACIIDS, Yogyakarta, Indonesia, Proceedings. Part II, April 8–11, 2019. P. 465-474.

[29] Tasmambetov Zh.N., Rajabov N., Issenova A.A. The Construction Of A Solution Of A Related System Of The Laguerre Type // Of the National Academy of sciences of The Republic of Kazakhstan. Series Physico-Mathematical. 2019. Vol. 1(323). P. 38-45. https://doi.org/10.32014/2019.2518-1726.5