Comparisons of extreme learning machine and backpropagation-based i-vector approach for speaker identification

Musab T.S. Al-KALTAKCHI1,∗, Raid R. O. AL-NIMA2, Mohammed A.M. ABDULLAH3
1Department of Electrical Engineering, College of Engineering, Mustansiriyah University, Baghdad, Iraq
2Technical Engineering College of Mosul, Northern Technical University, Mosul, Iraq
3Department of Computer and Information Engineering, College of Electronics Engineering, Ninevah University, Mosul, Iraq

Received: 18.06.2019 • Accepted/Published Online: 26.12.2019 • Final Version: 08.05.2020

Abstract: The extreme learning machine (ELM) is one of the machine learning applications used for regression and classification systems. In this paper, an extended comparison between an ELM and the backpropagation neural network (BPNN)-based i-vector is given in terms of a closed-set speaker identification task using 120 speakers from the TIMIT database. The system is composed of the mel frequency cepstral coefficient (MFCC) and power normalized cepstral coefficient (PNCC) approaches to form the feature extraction stage, while the cepstral mean variance normalization (CMVN) and feature warping are applied in order to mitigate the linear channel effect. The system is utilized with equal numbers of speakers of both genders with 120 speakers with eight dialects from the TIMIT database. The results demonstrate that the combination of the i-vector with the ELM for different features has the highest speaker identification accuracy (SIA) compared with the combination of the BPNN with the i-vector. The results also show that the i-vector with ELM approach is faster than the BPNN-based i-vector and it has the highest SIA.

Key words: Speaker recognition, extreme learning machine, TIMIT database, i-vector

1. Introduction
There are several open issues in machine learning techniques, such as intensive human interference, slow learning speed, and poor learning [1]. The extreme learning machine (ELM) appears to solve these problems [1]. The researchers in [2–5] presented a review in terms of ELM and dealt with theory, algorithms, trends, and applications. In the aforementioned papers, the mathematical model of the ELM was explained. In addition, the previous works reported that the ELM is efficient, simple, and widely used in various domains such as system identification, computer vision, biomedical engineering, robotics, and control. The ELM algorithm is utilized to attain more accurate results and to save time during regression and classification.

On the other hand, the backpropagation neural network (BPNN) is widely used as a supervised learning technique in machine learning to extract information. However, in [6] the BPNN showed poor time and space complexity. In [7], a description of the algorithm as well as an example of a nonlinear model-based BPNN classifier including the MATLAB code were given. In [8] the authors employed three methods to overcome the detection of nontechnical losses (NTLs) using an online sequential ELM (OS-ELM), BPNN, and support vector machine. A comparative study was conducted for two supervised machines learning the BPNN and the
OS-ELM. The results showed that the OS-ELM outperformed the BPNN with hit rates of 51.38% and 36.07%, respectively.

An overview in terms of the state of the art using the i-vector was presented in [9]. In addition, a survey of different applications of the i-vector approach in the speech processing domain was also given. Although the ELM was successfully employed for speaker identification in our previous studies in [10–12], the depicted models are time-consuming and comparisons with other neural networks methods were not considered.

This paper provides a combination of the i-vector approach with the ELM for speaker identification and the main motivations for this combination are as follows: This combination gives higher speaker identification accuracy (SIA) than each technique alone. In addition, the ELM has a faster training performance compared with other neural network techniques such as BPNN and deep learning, and the ELM has the ability to randomly generate the weight and the bias. Moreover, the performance with the ELM does not suffer from local error problems such as with the BPNN. Furthermore, the ELM has faster learning speed compared with the BPNN or deep learning, where it is straightforward, so no feedback is required and it thereby reduces the training and testing time required. On top of that, combining the ELM with the i-vector gives better accuracy for speaker identification. We compare two classifiers based on ELM and BPNN in terms of their speed and SIA for evaluating closed-set speaker identification performance. In summary, our contributions are as follows:

- The MFCC and PNCC are exploited to form a robust feature extractor model while the CMVN and feature warping are used for normalization.
- The ELM and BPNN-based i-vector approaches are employed for classification.
- A fair comparison is given between the BPNN and ELM-based i-vector approaches under the same conditions.

This paper is organized as follows: Section 2 discusses the acoustic features. Section 3 introduces the concept of the BPNN. Section 4 presents the concept of the ELM. Section 5 presents the main concept of the i-vector. Experiments and results with discussion are given in Section 6. Finally, Section 7 concludes this paper.

2. The main block diagram

Figure 1 shows the main block diagram using MFCC and PNCC for feature extraction, CMVN and feature warping for feature normalization, the i-vector as an acoustic model, and the BPNN and ELM as the classifiers.

In the proposed system, a preemphasis filter with 0.96 emphasis coefficients was applied for both features to compensate the high frequency that was suppressed while producing the speech signal. A Hamming windowing with 16 ms frame duration is used with 8 ms overlap between the frames. Then MFCC and PNCC features with 16 feature dimensions were used in the feature extraction stage.

The MFCC extraction operator can be divided into five stages: preemphasis, frame blocking and windowing, fast Fourier transform, mel-scaled filter bank, and cepstrum evaluation. Likewise, the PNCC implementation starts by preemphasizing the speech signal, and then a short-time Fourier transform (STFT) is performed using a Hamming window of 16 ms duration with 8 ms frame period. The squared magnitude of the STFT outputs is passed to a 40-channel Gammatone filter bank. The center frequencies of the Gammatone filter are linearly spaced in the equivalent rectangular bandwidth (ERB) auditory frequency scale. The output is then temporally masked and spectrally smoothed. Finally, the mean power is estimated for each frame and the running average for time-frequency normalization is used.
Feature normalization is also adopted by using FW and CMVN for both MFCC and PNCC features. FW and CMVN approaches are used to improve the SIA for the system and reduce the sensitivity between the types of telephone handsets and could also help to reduce the linear channel effects. The concepts of the BPNN, ELM, and i-vector are presented in the next sections, respectively.

3. The concept of BPNN

In machine learning, the BPNN is used to construct a classifier in which the set of decision rules is formulated to model the nonlinear functions [7]. The BPNN model involves three sections: the input layer, hidden layers (one or more layer), and the output layer. In the input layer, the number of neurons is equal to the number of attributes (i-vector dimension), while in the output layer, the number of neurons is equal to the number of nonlinear functions in order to construct the classifier from the decision rules. On the other hand, in the hidden layers, the number of neurons is randomly selected [7]. Figure 2 shows the structure of the BPNN.

![BPNN Diagram](image)

**Figure 2.** The general structure of backpropagation neural network (BPNN).
4. The concept of ELM

The ELM was originally proposed for a single hidden layer based on a feedforward network with randomly generated hidden neurons. In the ELM, no iterations or tuning are needed in the hidden layer [13]. In addition, all hidden nodes are independent with training data and target functions and have universal approximation capability. The output weights of the ELM may be determined with and without iterations. The ELM can be applied for multiclass classification and regression and it is efficient in sequential, batch, and incremental learning. The ELM has been widely used in different applications such as in signal processing, biometrics, image processing, bioinformatics, and brain/computer interface [13]. Figure 3 shows the structure of the ELM.

Figure 3. Structure of a single-layer feedforward extreme learning machine with d input dimension, L hidden nodes, and L outputs.

According to Figure 3, the implementation for the ELM network scheme is illustrated as shown in [1, 14, 15]. The number of input nodes is equal to the i-vector dimension. In addition, different neuron numbers are also selected when it is necessary to achieve higher performance accuracy. However, the number of output neurons is equal to the number of classes, and in this work we employ 120 classes to represent 120 speakers. The sigmoid function is used as the activation function in this paper. To calculate the output weights of the ELM, the following equations are used to find regularized values [1, 14, 15]:

\[
H\beta = \tilde{T},
\]

\[
\beta = H^\dagger \tilde{T},
\]

\[
\beta = \left( I + H^T H \right)^{-1} H^T \tilde{T},
\]

where \( \beta \) is the output weight matrix \( \beta = [\beta_1, ..., \beta_L]^T \), \( H = [h^T(x_1), ..., h^T(x_N)]^T \), \( h(x) = [g(a_1^T x + b_1), ..., g(a_L^T x + b_L)] \) is the hidden node outputs, \( x \) is the input vector, and \( g(a_j^T x + b_j) \) is the output of the \( j \)th hidden node. \( N \) is the number of training samples, \( \tilde{T} \) is the target label matrix, \( \tilde{T} = [\tilde{t}_1, ..., \tilde{t}_N]^T \), \( H^\dagger \) is the
Moore–Penrose generalized inverse of matrix $H$, and $r$ is the regularization factor. The target label matrix $\mathbf{T}$ is represented by $(\text{no. of speakers} \times \text{no. of training samples} \ (L \times N))$. Each i-vector with 100 dimensions is entered into the ELM classifier. The outputs represent the speaker classes in this work with 120 speakers. In addition, the actual outputs are real numbers, while the maximum is selected for the output vector and this maximum refers to the identified speaker position. Ultimately, the SIA is determined. 

The main steps of the ELM classifier and SIA calculation can be summarized as follow:

1. Compute the i-vectors from training and testing data.
2. Assume the number of output neurons is equal to the number of speakers.
3. Form the target matrix for training.
4. Randomly generate the input weights and biases for the hidden neurons.
5. Suppose the number of hidden neurons is equal to the i-vector dimension.
6. Calculate the hidden neurons output, matrix $H$.
7. Calculate the output weights.
8. Apply the testing data to the trained ELM.
9. Calculate the genuine speaker identified for each training example by the position of the maximum in the output of the ELM.
10. Calculate the SIA.

Table 1 shows the main differences between the BPNN and ELM.

<table>
<thead>
<tr>
<th>BPNN</th>
<th>ELM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Takes a long time for training</td>
<td>Faster learning speed</td>
</tr>
<tr>
<td>May suffer from local minimum (the point that has the lowest mean square error (MSE), where the network starts to train the neurons and the rate of MSE decreases)</td>
<td>Does not suffer from local minimum</td>
</tr>
<tr>
<td>Iterations required during training</td>
<td>No need to iterate or tune</td>
</tr>
<tr>
<td>Cannot easily add or remove trained subjects</td>
<td>Can easily add or remove trained subjects</td>
</tr>
</tbody>
</table>

5. The concept of i-vector

The main stages of constructing the i-vector are as follows [16]: first, forming the UBM from training data using the expectation maximization (EM) algorithm and Gaussian mixture components for the speakers; second, extracting the sufficient statistics for the training features using the Baum–Welch (BW) algorithm; and third, learning the total variability subspace. Finally, we extract the i-vector. The i-vector (identity vector) has a fixed length with a low dimension, which is used for modeling speakers for both MFCC and PNCC features. The coupling between the i-vector and ELM and BPNN is utilized in this paper in order to improve the identification rate. The mathematical model to establish the i-vector is proposed in [17] and is explained in equations (4), (5), and (6), where equation (4) is the model of joint factor analysis and equation (5) illustrates the relation between the total factor and total variability matrix. In addition, equation (6) is used to calculate the i-vector.
\[ S = \mu + Ux + Vy + Dz, \quad (4) \]

\[ S = \mu + TVi, \quad (5) \]

where \( S \) is a dependent supervector for both speaker and channel; \( V, U, \) and \( D \) are the speaker, channel, and diagonal residual matrices, respectively; and \( \mu \) is the independent supervector for the speaker and channel. In addition, factors \( y, x, \) and \( z \) represent the speaker, channel, and residual factor, respectively. Furthermore, the total factor identity vector is \( i \) and \( TV \) is the total variability low rank matrix. Finally, i-vectors can be determined as in [17] and are described in (6):

\[ i = (I + (TV)^T \Sigma^{-1} \hat{N}(u) TV)^{-1} (TV)^T \Sigma^{-1} \tilde{F}(u), \quad (6) \]

where \( i \) is the identity vector (i-vector); \( I \) is the identity matrix; \( \Sigma \) is the \((CF \times CF)\) diagonal covariance matrix, where \( C \) is the number of mixture components; \( \hat{N}(u) \) is a diagonal matrix of dimension \((CF \times CF)\); and the dimension of the feature vectors is \( F \). Moreover, \( \tilde{F} \) is the \((CF \times 1)\) dimension supervector, which is acquired by concatenating all first-order Baum–Welch statistics, and \( u \) is the given speech utterance and \((.)^T\) denotes transpose.

6. Experimental results and discussion

In this paper, 120 speakers with equal numbers from both genders are selected randomly from eight dialects of American English for the TIMIT database. The data source was microphone speech with sampling frequency equal to 16 kHz. The dialect regions (DR) include New England, Northern, North Midland, Southern, New York City, Western, and Army Brat, corresponding to: DR1, DR2,..., DR8, respectively. Each speaker has ten speech utterances, five of them for training and the remaining used for testing. For the 120 speakers, 720 utterances are used for training while 480 utterances are used for testing. Hence, the i-vector is constructed and classified using the ELM to identify speakers by employing 120 classes to represent 120 speakers.

In addition, instead of the ELM, BPNN is used for comparison purposes. Furthermore, the best hidden layer is selected to achieve higher SIA, which is proved experimentally to be 100. In addition, the best SIA is obtained with UBM mixture size of 256. Table 2 shows the SIA for the i-vector ELM approach where the best results are empirically attained under 100 hidden neurons with 100 i-vector dimensions where the sizes of the UBM are 128, 256, and 512. The highest SIA is reported to be 92.95% using the i-vector with CMVN-MFCC features and mixture size of UBM equal to 256. In addition, Table 2 illustrates the processing time of the current method, which is 40-47 s. Furthermore, the results with MFCC are better than the corresponding results with MFCC. On the other hand, Table 3 shows the results for the i-vector using the BPNN approach with UBM mixture sizes of 128, 256, and 512. The results show that the highest SIA obtained is equal to 86.235% with CMVN MFCC features. It has been noticed that the result with MFCC also outperformed the results of PNCC features. Moreover, the comparisons between the ELM and BPNN and the i-vector are explained through comparisons between Table 2 and Table 3. The results obtained by employing the ELM are better than those obtained with the BPNN. In addition, the timing requirements of the BPNN are about 10 times larger than the
### Table 2. The SIA for the i-vector based on ELM Approach at UBM Mixture Sizes 128, 256, 512 with 100 hidden neurons and 100 i-vector dimension.

<table>
<thead>
<tr>
<th>Method</th>
<th>Features</th>
<th>UBM Mixture size</th>
<th>SIA</th>
<th>Consuming Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>i-vector-ELM</td>
<td>MFCC+Feature Warping</td>
<td>128</td>
<td>91.13%</td>
<td>42.168 Sec</td>
</tr>
<tr>
<td>i-vector-ELM</td>
<td>PNCC+Feature Warping</td>
<td>128</td>
<td>85.41%</td>
<td>40.346 Sec</td>
</tr>
<tr>
<td>i-vector-ELM</td>
<td>MFCC+CMVN</td>
<td>128</td>
<td>92.77%</td>
<td>40.233 Sec</td>
</tr>
<tr>
<td>i-vector-ELM</td>
<td>PNCC+CMVN</td>
<td>128</td>
<td>87.95%</td>
<td>46.94 Sec</td>
</tr>
<tr>
<td>i-vector-ELM</td>
<td>MFCC+CMVN</td>
<td>256</td>
<td></td>
<td>92.95%</td>
</tr>
<tr>
<td>i-vector-ELM</td>
<td>PNCC+CMVN</td>
<td>256</td>
<td></td>
<td>47.156 Sec</td>
</tr>
<tr>
<td>i-vector-ELM</td>
<td>MFCC+Feature Warping</td>
<td>256</td>
<td>92.29%</td>
<td>41.4 Sec</td>
</tr>
<tr>
<td>i-vector-ELM</td>
<td>PNCC+Feature Warping</td>
<td>256</td>
<td>87.16%</td>
<td>49.468 Sec</td>
</tr>
<tr>
<td>i-vector-ELM</td>
<td>MFCC+Feature Warping</td>
<td>512</td>
<td>89.42%</td>
<td>42.625 Sec</td>
</tr>
<tr>
<td>i-vector-ELM</td>
<td>PNCC+Feature Warping</td>
<td>512</td>
<td>84.72%</td>
<td>46.103 Sec</td>
</tr>
<tr>
<td>i-vector-ELM</td>
<td>MFCC+CMVN</td>
<td>512</td>
<td>91.75%</td>
<td>42.599 Sec</td>
</tr>
<tr>
<td>i-vector-ELM</td>
<td>PNCC+CMVN</td>
<td>512</td>
<td>84.71%</td>
<td>46.841 Sec</td>
</tr>
</tbody>
</table>

### Table 3. The SIA for the i-vector based on BPNN Approach at UBM Mixture Sizes 128, 256, 512 with 100 hidden neurons and 100 i-vector dimension.

<table>
<thead>
<tr>
<th>Method</th>
<th>Features</th>
<th>UBM Mixture size</th>
<th>SIA</th>
<th>Consuming Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>i-vector-BPNN</td>
<td>MFCC+Feature Warping</td>
<td>128</td>
<td>64.69%</td>
<td>179.849 Sec</td>
</tr>
<tr>
<td>i-vector-BPNN</td>
<td>PNCC+Feature Warping</td>
<td>128</td>
<td>59.375%</td>
<td>256.029 Sec</td>
</tr>
<tr>
<td>i-vector-BPNN</td>
<td>MFCC+CMVN</td>
<td>128</td>
<td>69.375%</td>
<td>286.954 Sec</td>
</tr>
<tr>
<td>i-vector-BPNN</td>
<td>PNCC+CMVN</td>
<td>128</td>
<td>58.75%</td>
<td>371.909 Sec</td>
</tr>
<tr>
<td>i-vector-BPNN</td>
<td>MFCC+Feature Warping</td>
<td>256</td>
<td>74.437%</td>
<td>311.694 Sec</td>
</tr>
<tr>
<td>i-vector-BPNN</td>
<td>PNCC+Feature Warping</td>
<td>256</td>
<td>71.375%</td>
<td>524.828 Sec</td>
</tr>
<tr>
<td>i-vector-BPNN</td>
<td>MFCC+CMVN</td>
<td>256</td>
<td>86.235%</td>
<td>414.908 Sec</td>
</tr>
<tr>
<td>i-vector-BPNN</td>
<td>PNCC+CMVN</td>
<td>256</td>
<td>73.75%</td>
<td>491.187 Sec</td>
</tr>
<tr>
<td>i-vector-BPNN</td>
<td>MFCC+Feature Warping</td>
<td>512</td>
<td>82.23%</td>
<td>914.975 Sec</td>
</tr>
<tr>
<td>i-vector-BPNN</td>
<td>PNCC+Feature Warping</td>
<td>512</td>
<td>80.312%</td>
<td>985.730 Sec</td>
</tr>
<tr>
<td>i-vector-BPNN</td>
<td>MFCC+CMVN</td>
<td>512</td>
<td>84.27%</td>
<td>776.145 Sec</td>
</tr>
<tr>
<td>i-vector-BPNN</td>
<td>PNCC+CMVN</td>
<td>512</td>
<td>81%</td>
<td>813.46 Sec</td>
</tr>
</tbody>
</table>

Timing corresponding to the ELM. Hence, using the ELM gives faster learning performance while maintaining classification accuracy.

Table 4 shows the SIA for clear speech of 120 speakers from the TIMIT database with a wide range of Gaussian mixture components. In addition, Table 5 depicts the SIA for noisy speech of 120 speakers from the TIMIT database using the i-vector-ELM approach. Figure 4 shows the SIA versus Gaussian mixture components for both the i-vector-ELM approach and i-vector-BPNN approach. It can be seen from Figure 4 that the i-vector ELM approach outperforms the I-vector BPNN approach. In summary, all the parameters of the setups are illustrated in Table 6.
Table 4. The SIA for Clean Speech of 120 Speakers from TIMIT Database with 100 Dimension of I-vector-ELM Approach.

<table>
<thead>
<tr>
<th>Mix 8</th>
<th>Mix 16</th>
<th>Mix 32</th>
<th>Mix 64</th>
<th>Mix 128</th>
<th>Mix 256</th>
<th>Mix 512</th>
</tr>
</thead>
<tbody>
<tr>
<td>35%</td>
<td>61.67%</td>
<td>87.5%</td>
<td>90%</td>
<td>93.33%</td>
<td>95.83%</td>
<td>95.83%</td>
</tr>
</tbody>
</table>

Table 5. The SIA for Noisy Speech of 120 Speakers from TIMIT Database with 100 Dimension of I-vector-ELM Approach.

<table>
<thead>
<tr>
<th>AWGN</th>
<th>0dB</th>
<th>5dB</th>
<th>10dB</th>
<th>15dB</th>
<th>20dB</th>
<th>25dB</th>
<th>30dB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>5%</td>
<td>7.5%</td>
<td>19.17%</td>
<td>33.33%</td>
<td>47.5%</td>
<td>59.17%</td>
<td>76.67%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Street Traffic NSN</th>
<th>13.33%</th>
<th>23.33%</th>
<th>41.67%</th>
<th>61.67%</th>
<th>77.5%</th>
<th>81.67%</th>
<th>87.5%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>53.33%</td>
<td>63.33%</td>
<td>78.33%</td>
<td>85.83%</td>
<td>86.67%</td>
<td>90.83%</td>
<td>92.5%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bus Interior NSN</th>
<th>11.67%</th>
<th>23.33%</th>
<th>47.5%</th>
<th>64.17%</th>
<th>74.17%</th>
<th>84.17%</th>
<th>85.83%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>90.83%</td>
<td>92.5%</td>
<td>95.83%</td>
<td>100%</td>
<td>128%</td>
<td>200%</td>
<td>256%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Crowd Talking NSN</th>
<th>11.67%</th>
<th>23.33%</th>
<th>47.5%</th>
<th>64.17%</th>
<th>74.17%</th>
<th>84.17%</th>
<th>85.83%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>90.83%</td>
<td>92.5%</td>
<td>95.83%</td>
<td>100%</td>
<td>128%</td>
<td>200%</td>
<td>256%</td>
</tr>
</tbody>
</table>

Figure 4. Comparisons between the i-vector ELM and BPNN approaches in terms of the Gaussian mixture components.

7. Conclusions

In this paper, a fair comparison between the ELM and BPNN based the i-vector was given in terms of a closed-set speaker identification task using the TIMIT database. The results showed that the ELM approach achieved better timing performance with higher SIA compared to the BPNN. In addition, the results obtained using the MFCC features gave better identification rates compared with the PNCC features. Moreover, various mixture sizes of UBM are considered and the highest accuracy is achieved using the UBM mixture of size equal to 256.
Table 6. Parameters and setup used for 120 speakers selected from TIMIT database.

<table>
<thead>
<tr>
<th>Aspects</th>
<th>Parameters and experimental setup</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sampling frequency</td>
<td>16000</td>
</tr>
<tr>
<td>Window type</td>
<td>Hamming</td>
</tr>
<tr>
<td>Frame length</td>
<td>16 ms</td>
</tr>
<tr>
<td>Frame shift</td>
<td>8 ms</td>
</tr>
<tr>
<td>Preemphasis factor</td>
<td>0.96</td>
</tr>
<tr>
<td>Databases</td>
<td>TIMIT</td>
</tr>
<tr>
<td>Number of speakers</td>
<td>120 speakers for each database, total 480 speakers for all databases</td>
</tr>
<tr>
<td>Total speech utterances used</td>
<td>1200 for each database</td>
</tr>
<tr>
<td>Language</td>
<td>English</td>
</tr>
<tr>
<td>Data source(s)</td>
<td>Microphone speech for TIMIT</td>
</tr>
<tr>
<td>No. of samples per speaker</td>
<td>10 for each of TIMIT</td>
</tr>
<tr>
<td>Testing samples for each database</td>
<td>Total 480 utterances</td>
</tr>
<tr>
<td>Training samples for each database</td>
<td>Total 720 utterances</td>
</tr>
<tr>
<td>Average sample duration</td>
<td>8 seconds in length, 129250 (for each speech utterance in both training and testing); all speech samples were taken with fixed length of 129250 samples; concatenation is applied where necessary</td>
</tr>
<tr>
<td>Features</td>
<td>MFCC and PNCC</td>
</tr>
<tr>
<td>Feature dimensions</td>
<td>16</td>
</tr>
<tr>
<td>Feature normalization</td>
<td>FW and CMVN</td>
</tr>
<tr>
<td>Modeling</td>
<td>i-vector</td>
</tr>
<tr>
<td>Classifier</td>
<td>ELM and BPNN</td>
</tr>
<tr>
<td>UBM mixture sizes</td>
<td>{8, 16, 32, 64, 128, 256, 512}</td>
</tr>
<tr>
<td>System environment</td>
<td>Original speech recordings, AWGN with street traffic, bus interior, and crowd talking NSN</td>
</tr>
<tr>
<td>SNR levels in dB</td>
<td>{0, 5, 10, 15, 20, 25, 30}</td>
</tr>
</tbody>
</table>

References


