Speech Emotion Recognition Using Deep Neural Network and Extreme Learning Machine

Kun Han[1⋆], *Dong Yu*[2], *Ivan Tashev*[2]

[1]Department of Computer Science and Engineering,
The Ohio State University, Columbus, 43210, OH, USA
[2]Microsoft Research, One Microsoft Way,
Redmond, 98052, WA, USA
hank@cse.ohio-state.edu, {dong.yu, ivantash}@microsoft.com

Abstract

Speech emotion recognition is a challenging problem partly because it is unclear what features are effective for the task. In this paper we propose to utilize deep neural networks (DNNs) to extract high level features from raw data and show that they are effective for speech emotion recognition. We first produce an emotion state probability distribution for each speech segment using DNNs. We then construct utterance-level features from segment-level probability distributions. These utterance-level features are then fed into an extreme learning machine (ELM), a special simple and efficient single-hidden-layer neural network, to identify utterance-level emotions. The experimental results demonstrate that the proposed approach effectively learns emotional information from low-level features and leads to 20% relative accuracy improvement compared to the state-of-the-art approaches.

Index Terms: Emotion recognition, Deep neural networks, Extreme learning machine

1. Introduction

Despite the great progress made in artificial intelligence, we are still far from being able to naturally interact with machines, partly because machines do not understand our emotion states. Recently, speech emotion recognition, which aims to recognize emotion states from speech signals, has been drawing increasing attention. Speech emotion recognition is a very challenging task of which extracting effective emotional features is an open question [1, 2].

A deep neural network (DNN) is a feed-forward neural network that has more than one hidden layers between its inputs and outputs. It is capable of learning high-level representation from the raw features and effectively classifying data [3, 4]. With sufficient training data and appropriate training strategies, DNNs perform very well in many machine learning tasks (e.g., speech recognition [5]).

Feature analysis in emotion recognition is much less studied than that in speech recognition. Most previous studies empirically chose features for emotion classification. In this study, a DNN takes as input the conventional acoustic features within a speech segment and produces segment-level emotion state probability distributions, from which utterance-level features are constructed and used to determine the utterance-level emotion state. Since the segment-level outputs already provide considerable emotional information and the utterance-level classification does not involve too much training, it is unnecessary to use DNNs for the utterance-level classification. Instead, we employ a newly developed single-hidden-layer neural network, called extreme learning machine (ELM) [6], to conduct utterance-level emotion classification. ELM is very efficient and effective when the training set is small and outperforms support vector machines (SVMs) in our study.

In the next section, we relate our work to prior speech emotion recognition studies. We then describe our proposed approach in detail in Section 3. We show the experimental results in Section 4 and conclude the paper in Section 5.

2. Relation to prior work

Speech emotion recognition aims to identify the high-level affective status of an utterance from the low-level features. It can be treated as a classification problem on sequences. In order to perform emotion classification effectively, many acoustic features have been investigated. Notable features include pitch-related features, energy-related features, Mel-frequency cepstrum coefficients (MFCC), linear predictor coefficients (LPC), etc. Some studies used generative models, such as Gaussian mixture models (GMMs) and Hidden Markov models (HMMs), to learn the distribution of these low-level features, and then use the Bayesian classifier or the maximum likelihood principle for emotion recognition [7, 8]. Some other studies trained universal background models (UBMs) on the low-level features and then generated supervectors for SVM classification [9, 10], a technique widely used in speaker identification. A different trend for emotion recognition is to apply statistical functions to these low-level acoustic features to compute global statistical features for classification. The SVM is the most commonly used classifier for global features[11, 12]. Some other classifiers, such as decision trees [13] and K-nearest neighbor (KNN) [14], have also been used in speech emotion recognition. These approaches require very high-dimensional handcrafted features chosen empirically.

Deep learning is an emerging field in machine learning in recent years. A very promising characteristic of DNNs is that they can learn high-level invariant features from raw data [15, 4], which is potentially helpful for emotion recognition. A few recent studies utilized DNNs for speech emotion recognition. Stuhlsatz *et al.* and Kim *et al.* train DNNs on utterance-level statistical features. Rozgic *et al.* combine acoustic features and lexical features to build a DNN based emotion recognition system. Unlike these DNN based methods, which substitute DNNs for other classifiers such as SVMs, our approach exploits

⋆Work done during a research internship at Microsoft Research.

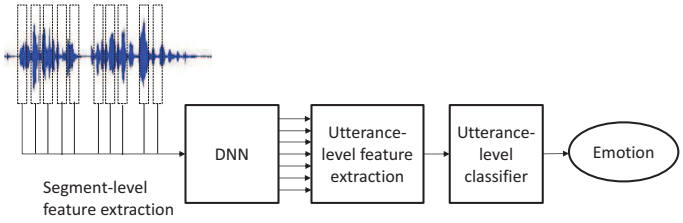

Figure 1: Algorithm overview

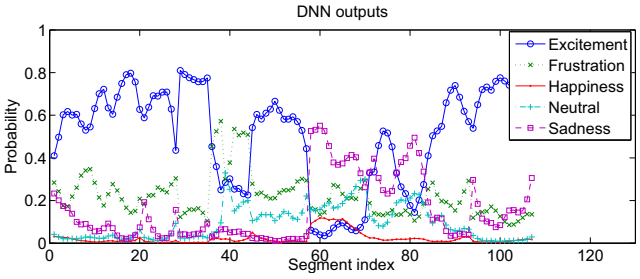

Figure 2: DNN outputs of an utterance. Each line corresponds to the probability of an emotion state.

DNNs to extract from short-term acoustic features the effective emotional features that are fed into other classifiers for emotion recognition.

3. Algorithm details

In this section, we describe the details of our algorithm. Fig. 1 shows the overview of the approach. We first divide the signal into segments and then extract the segment-level features to train a DNN. The trained DNN computes the emotion state distribution for each segment. From these segment-level emotion state distributions, utterance-level features are constructed and fed into an ELM to determine the emotional state of the whole utterance.

3.1. Segment-level feature extraction

The first stage of the algorithm is to extract features for each segment in the whole utterance. The input signal is converted into frames with overlapping windows. The feature vector $\mathbf{z}(m)$ extracted for each frame m consists of MFCC features, pitch-based features, and their delta feature across time frames. The pitch-based features include pitch period $\tau_0(m)$ and the harmonics-to-noise ratio (HNR), which is computed as:

$$\text{HNR}(m) = 10 \log \frac{ACF(\tau_0(m))}{ACF(0) - ACF(\tau_0(m))} \qquad (1)$$

where $ACF(\tau)$ denotes the autocorrelation function at time τ. Because the emotional information is often encoded in a relatively long window, we form the segment-level feature vector by stacking features in the neighboring frames as:

$$\mathbf{x}(m) = [\mathbf{z}(m-w), \ldots, \mathbf{z}(m), \ldots, \mathbf{z}(m+w] \qquad (2)$$

where w is the window size on each side.

For the segment-level emotion recognition, the input to the classifier is the segment-level feature and the training target is the label of the utterance. In other words, we assign the same label to all the segments in one utterance. Furthermore, since not all segments in an utterance contain emotional information and it is reasonable to assume that the segments with highest energy contain most prominent emotional information, we only choose segments with the highest energy in an utterance as the training samples. In addition, motivated by the recent progress in speech recognition [16, 17], we have attempted to train the DNN directly using the filterbank or spectral features, but the performance is not satisfactory.

3.2. Deep neural network training

With the segment-level features, we train a DNN to predict the probabilities of each emotion state. The DNN can be treated as a segment-level emotion recognizer. Although it is not necessary true that the emotion states in all segments is identical to that of the whole utterance, we can find certain patterns from the segment-level emotion states, which can be used to predict utterance-level emotions by a higher-level classifier.

The number of input units of the DNN is consistent with the segment-level feature vector size. It uses a softmax output layer whose size is set to the number of possible emotions K. The number of hidden layers and the hidden units are chosen from cross-validation.

The trained DNN aims to produce a probability distribution $\mathbf{t}$ over all the emotion states for each segment:

$$\mathbf{t} = [P(E_1), \ldots, P(E_K)]^T \qquad (3)$$

Note that, in the test phase we also only use those segments with the highest energy to be consistent with the training phase.

Fig. 2 shows an example of an utterance with the emotion of excitement. The DNN has five outputs corresponding to five different emotion states: excitement, frustration, happiness, neutral and sadness. As shown in the figure, the probability of each segment changes across the whole utterance. Different emotions dominate different regions in the utterance, but excitement has the highest probability in most segments. The true emotion for this utterance is also excitement, which has been reflected in the segment-level emotion states. Although not all utterances have such prominent segment-level outputs, we can use an utterance-level classifier to distinguish them.

3.3. Utterance-level features

Given the sequence of probability distribution over the emotion states generated from the segment-level DNN, we can form the emotion recognition problem as a sequence classification problem, i.e., based on the unit (segment) information, we need to make decision for the whole sequence (utterance). We use a special single-hidden-layer neural network with basic statistical feature to determine emotions at the utterance-level. We also indicate that temporal dynamics play an important role in speech emotion recognition, but our preliminary experiments show that it does not lead to significant improvement compared to a static classifier, which is partly because the DNN provides good segment-level results which can be easily classified with a simple classifier.

The features in the utterance-level classification are computed from statistics of the segment-level probabilities. Specifically, let $P_s(E_k)$ denote the probability of the kth emotion for the segment s. We compute the features for the utterance i for all $k = 1, \ldots, K$

$$f_1^k = \max_{s \in U} P_s(E_k), \qquad (4)$$

$$f_2^k = \min_{s \in U} P_s(E_k), \quad (5)$$

$$f_3^k = \frac{1}{|U|} \sum_{s \in U} P_s(E_k), \quad (6)$$

$$f_4^k = \frac{|P_s(E_k) > \theta|}{|U|}, \quad (7)$$

where, U denotes the set of all segments used in the segment-level classification. The features f_1^k, f_2^k, f_3^k correspond to the maximal, minimal and mean of segment-level probability of the kth emotion over the utterance, respectively. The feature f_4^k is the percentage of segments which have high probability of emotion k. This feature is not sensitive to the threshold θ, which can be empirically chosen from a development set.

3.4. Extreme learning machine for utterance-level classification

The utterance-level statistical features are fed into a classifier for emotion recognition of the utterance. Since the number of training utterances is small we use a recently developed classifier, called extreme learning machine (ELM) [6, 18] for this purpose. ELM has been shown to achieve promising results when the training set is small.

ELM is a single-hidden-layer neural network which requires many more hidden units than typically needed by the conventional neural networks (NNs) to achieve considerable classification accuracy. The training strategy of ELM is very simple. Unlike conventional NNs whose weights need to be tuned using the backpropagation algorithm, in ELM the weights between the input layer and the hidden layer are randomly assigned and then fixed. The weights between the hidden layer and the output layer can be analytically determined through a simple generalized inverse operation of the hidden layer output matrices.

Specifically, given training data $(\mathbf{x}_i, \mathbf{t}_i)$, $i = 1, \ldots, N$, $\mathbf{x}_i \in \mathbb{R}^D$ is the input feature, and $\mathbf{t}_i \in \mathbb{R}^K$ is the target, the ELM can be trained as follows:

1. Randomly assign values for the lower layer weight matrix $\mathbf{W} \in \mathbb{R}^{D \times L}$ from an uniform distribution over [-1,1], where L is the number of hidden units.
2. For each training sample $\mathbf{x_i}$, compute the hidden layer outputs $\mathbf{h_i} = \sigma(\mathbf{W}^T \mathbf{x_i})$, where σ is the sigmoid function.
3. The output layer weights $\mathbf{U}$ are computed as $\mathbf{U} = (\mathbf{H}\mathbf{H}^T)^{-1}\mathbf{H}\mathbf{T}^T$, where $\mathbf{H} = [\mathbf{h}_1, \ldots, \mathbf{h}_N]$, $\mathbf{T} = [\mathbf{t}_1, \ldots, \mathbf{t}_N]$,

Generally, the number of hidden units is much larger than that of input units, so that the random projection in the lower layer is capable to represent training data. The lower layer weights $\mathbf{W}$ randomly project the training data to a much higher dimensional space where the projected data are potentially linearly separable. Further, random weights are chosen independent of the training set and thus can generalize well to new data. The training for ELMs only involves a pseudo-inverse calculation and is very fast for a small dataset. Another variant of the ordinary ELM is the kernel based ELM [6], which defines the kernel as the function of the inner product of two hidden layer outputs, and the number of hidden units does not need to be specified by the users. We will compare both ELMs in the experiments.

We use the utterance-level features to train the ELM for the utterance-level emotion classification. The output of the ELM for each utterance is a K-dimensional vector corresponding to the scores of each emotion state. The emotion with the highest ELM score is chosen as the recognition result for the utterance.

4. Experimental results

4.1. Experimental setting

We use the Interactive Emotional Dyadic Motion Capture (IEMOCAP) database [19] to evaluate our approach. The database contains audiovisual data from 10 actors, and we only use audio track for our evaluation. Each utterance in the database is labeled by three human annotators using categorical and dimensional labels. We use categorical labels in our study and we only consider utterances with labels from five emotions: excitement, frustration, happiness, neutral and surprise. Since three annotators may give different labels for an utterance, in our experiment, we choose those utterances which are given the same label by at least two annotators to avoid ambiguity.

We train the model in the speaker-independent manner, i.e., we use utterances from 8 speakers to construct the training and the development datasets, and use the other 2 speakers for test. Note that, although previous study showed that normalizing features on a per-speaker basis can significantly improve the performance [20], we do not use it because we assume that speaker identity information is not available in our study.

The input signal is sampled at 16 kHz and converted into frames using a 25-ms window sliding at 10-ms each time. The size of the segment level feature is set to 25 frames, including 12 frames in each side. So the total length of a segment is $10 \text{ ms} \times 25 + (25 - 10) \text{ ms} = 265 \text{ ms}$. In fact, emotional information is usually encoded in one or more speech segments whose length varies on factors such as speakers and emotions. It is still an open problem to determine the appropriate analysis window for emotion recognition. Fortunately a speech segment longer than 250 ms has been shown to contain sufficient emotional information [14, 21]. We also tried longer segments up to 500 ms, and achieved similar performance. In addition, 10% segments with the highest energy in an utterance are used in the training and the test phase. The threshold in Eq. (7) is set to 0.2.

The segment-level DNN has a 750-unit input layer corresponding to the dimensionality of the feature vector. The DNN contains three hidden layers and each hidden layer has 256 rectified linear hidden units. Mini-batch gradient descend method is used to learn the weights in DNN and the objective function is cross-entropy. For ELM training, the number of hidden units for ordinary ELM is set to 120, and the radius basis function is used in the kernel ELM. All parameters are chosen from the development set.

4.2. Results

We compare our approach with other emotion recognition approaches. The first one is an HMM based method. Schuller *et al.* [7] used pitch-based and energy-based features in each frame to train an HMM for emotion recognition. We replace these features by the same segment-level features used in our study which are found to perform better in the experiment. We mention that Li *et al.* [22] use DNN to predict HMM states for emotion estimation. We have attempted to implement the algorithm, but the performance is similar to the conventional HMM based method. Another approach is a state-of-the-art toolkit for emotion recognition: OpenEAR [11]. It uses global statistical features and SVM for emotion recognition. We used the pro-

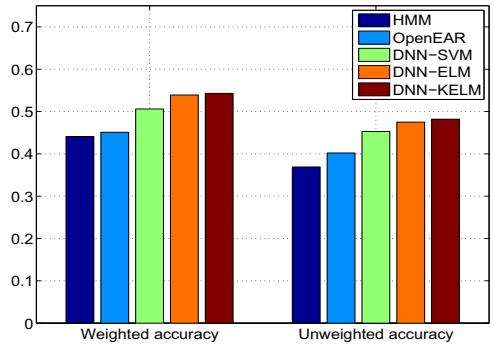

Figure 3: Comparison of different approaches in terms of weighted and unweighted accuracies. "HMM" and "OpenEAR" denote the two baseline approaches using HMM and SVM respectively. "DNN-SVM", "DNN-ELM", and "DNN-KELM" denote the proposed approach using segment-level DNN and utterance-level SVM, ELM, and kernel ELM, respectively.

vided code to extract a 988-dimensional feature vector for each utterance for SVM training. In addition, in order to analyze the performance of the ELM, we also use the proposed DNN method to generate the segment-level outputs and then use an SVM to predict utterance-level labels. We use two measures to evaluate the performance: weighted accuracy and unweighted accuracy. Weighted accuracy is the classification accuracy on the whole test set, and unweighted accuracy is an average of the recall for each emotion class, which better reflects overall accuracy in the presence of imbalanced class.

Fig. 3 shows the comparison results in terms of weighted and unweighted accuracies. Overall, the proposed DNN based approaches significantly outperform the other two with 20% relative accuracy improvement for both unweighted accuracy ($0.402 \rightarrow 0.482$) and weighted accuracy ($0.451 \rightarrow 0.543$). We found that the ordinary ELM and the kernel ELM perform equally well, both outperform SVM by around 5% relatively. It is also worth mentioning that the training time of ELMs is around 10 times faster than that of SVMs in our experiments.

5. Conclusion

We proposed to utilize a DNN to estimate emotion states for each speech segment in an utterance, construct an utterance-level feature from segment-level estimations, and then employ an ELM to recognize the emotions for the utterance. Our experimental results indicate that this approach substantially boosts the performance of emotion recognition from speech signals and it is very promising to use neural networks to learn emotional information from low-level acoustic features.

6. References

[1] M. El Ayadi, M. S. Kamel, and F. Karray, "Survey on speech emotion recognition: Features, classification schemes, and databases," *Pattern Recognition*, vol. 44, no. 3, pp. 572–587, 2011.

[2] B. Schuller, A. Batliner, S. Steidl, and D. Seppi, "Recognising realistic emotions and affect in speech: State of the art and lessons learnt from the first challenge," *Speech Communication*, vol. 53, no. 9, pp. 1062–1087, 2011.

[3] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[4] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[5] G. E. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 20, no. 1, pp. 30–42, 2012.

[6] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: theory and applications," *Neurocomputing*, vol. 70, no. 1, pp. 489–501, 2006.

[7] B. Schuller, G. Rigoll, and M. Lang, "Hidden markov model-based speech emotion recognition," in *Proceedings of IEEE ICASSP 2003*, vol. 2. IEEE, 2003, pp. II–1.

[8] C. M. Lee, S. Yildirim, M. Bulut, A. Kazemzadeh, C. Busso, Z. Deng, S. Lee, and S. Narayanan, "Emotion recognition based on phoneme classes," in *Proceedings of Interspeech*, 2004, pp. 889–892.

[9] H. Hu, M.-X. Xu, and W. Wu, "GMM supervector based SVM with spectral features for speech emotion recognition," in *Proceedings of IEEE ICASSP 2007*, vol. 4. IEEE, 2007, pp. IV–413.

[10] T. L. Nwe, N. T. Hieu, and D. K. Limbu, "Bhattacharyya distance based emotional dissimilarity measure for emotion classification," in *Proceedings of IEEE ICASSP 2013*. IEEE, 2013, pp. 7512–7516.

[11] F. Eyben, M. Wollmer, and B. Schuller, "OpenEAR - introducing the Munich open-source emotion and affect recognition toolkit," in *Proceedings of ACII 2009*. IEEE, 2009, pp. 1–6.

[12] E. Mower, M. J. Mataric, and S. Narayanan, "A framework for automatic human emotion classification using emotion profiles," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 5, pp. 1057–1070, 2011.

[13] C.-C. Lee, E. Mower, C. Busso, S. Lee, and S. Narayanan, "Emotion recognition using a hierarchical binary decision tree approach," in *Proceedings of Interspeech*, 2009, pp. 320–323.

[14] Y. Kim and E. Mower Provost, "Emotion classification via utterance-level dynamics: A pattern-based approach to characterizing affective expressions," in *Proceedings of IEEE ICASSP 2013*. IEEE, 2013.

[15] D. Yu, M. L. Seltzer, J. Li, J.-T. Huang, and F. Seide, "Feature learning in deep neural networks-studies on speech recognition tasks," *arXiv preprint arXiv:1301.3605*, 2013.

[16] J. Li, D. Yu, J.-T. Huang, and Y. Gong, "Improving wideband speech recognition using mixed-bandwidth training data in CD-DNN-HMM," in *Proceedings of SLT*, 2012.

[17] L. Deng, J. Li, J.-T. Huang, K. Yao, D. Yu, F. Seide, M. Seltzer, G. Zweig, X. He, J. Williams *et al.*, "Recent advances in deep learning for speech research at Microsoft," in *Proceedings of IEEE ICASSP 2013*, 2013.

[18] D. Yu and L. Deng, "Efficient and effective algorithms for training single-hidden-layer neural networks," *Pattern Recognition Letters*, vol. 33, no. 5, pp. 554–558, 2012.

[19] C. Busso, M. Bulut, C.-C. Lee, A. Kazemzadeh, E. Mower, S. Kim, J. N. Chang, S. Lee, and S. S. Narayanan, "IEMOCAP: Interactive emotional dyadic motion capture database," *Language resources and evaluation*, vol. 42, no. 4, pp. 335–359, 2008.

[20] C. Busso, A. Metallinou, and S. S. Narayanan, "Iterative feature normalization for emotional speech detection," in *Proceedings of IEEE ICASSP 2011*. IEEE, 2011, pp. 5692–5695.

[21] E. Mower Provost, "Identifying salient sub-utterance emotion dynamics using flexible units and estimates of affective flow," in *Proceedings of IEEE ICASSP 2013*. IEEE, 2013.

[22] L. Li, Y. Zhao, D. Jiang, Y. Zhang, F. Wang, I. Gonzalez, E. Valentin, and H. Sahli, "Hybrid deep neural network–hidden Markov model (DNN-HMM) based speech emotion recognition," in *Proceedings of ACII*. IEEE, 2013, pp. 312–317.