Speech Emotion Recognition Using Segment-level Features and Utterance-level Features

Xueyao Huang[1], *Ruizhe Zhou*[1]

[1]Department of Computer Science,
The University of Chicago, Chicago, IL, 60637
{huangxy, rzhou12}@uchicago.edu

Abstract

In this project, we present a speech emotion recognition approach using segment-level features and utterance-level features trained by deep neural network (DNN) along with extreme learning machines (ELM). Our project is inspired by this work[1] at Microsoft Research. Like other speech recognition tasks, one big challenge in speech emotion recognition is that we do not know what features are useful to classify emotions. In this project, we first extract segment-level features which consist of MFCC features, pitch-based features, and delta features. Then we use DNN to get the probability distribution of an emotion state over the whole utterance. Further more, we construct utterance-level features using four statistics of the segment-level probabilities and feed the utterance-level features to the adaboosted ELM to predict the emotion of the sentence. Our results shows a good improvement compared to the baseline approach using hidden markov model (HMM).

Index Terms: speech emotion recognition, segment-level feature, utterance-level feature, adaboost, extreme learning machine

1. Introduction

The emotion state of human is an important factor in actions influencing most of communications, and speech is one of the primary expression to reflect emotions. As for a human-computer interface, it is important to recognize and respond to the emotions in speech.

The study of SER(Speech Emotion Recognition) draws an increasing attention recently, and the primary objective for this study is enhancing human-machine interaction. Despite the fact that a lot of progress has been made in the area, the major of work of detecting emotions from speech is quite limited, and still, there is a need for a better understanding of human emotions. Currently, researchers have done a lot of studies on feature analysis but still have a debate on what kind of features would dominate the performance in detecting emotions. In this study, we combine segment-level features and utterance-level features, and perform a standard machine learning approach to classify emotions. The feature extraction process consists of two steps. First, extracting acoustic features, including MFCC, harmonic-to-noise ratio and delta feature across time frames. Second, apply variant high level statistical functions to segment-level features across the whole utterance, and concatenate them into utterance-level features [2]. Table 1 shows the features in both segment and utterance levels that related to the study.

DNN has been considered as one of the best algorithm that can learn from raw data in classifying tasks. It performs very well in producing segment level features with sufficient training data and appropriate training strategies. A DNN is a feed-forward neural network that has multiple hidden layers between its input and output. During in this project, a deep neural network takes traditional acoustic features in segment level and produces probability distribution over all emotions for each segment. Next, we construct utterance level features based on the probability distribution for each segments. Since the segment-level features have represented many emotion information, the utterance-level features do not need too much training. We use a single hidden layer neural network instead of DNN to train the utterance level features, which is well known as ELM[6]. Moreover, we construct balanced training dataset and employ ensemble method to boost the ELM in order to improve the low recall of some emotion.

Segment-level features	MFCC, Mel-filterbank, formant, HNR, jitter, shimmer, etc.
Utterance-level features	mean, variance, max, min, median of segment-level features.

Table 1: Segment/Utterance-level features

2. Algorithm

2.1. Segment-level features

The first step of the algorithm is extracting segment-level features from an utterance. Most of the features can be inferred from a spectrogram representation of a signal, and thus we convert input into a frame-dimension and extract feature vector for each frame consists of MFCC related features, pitch-based features, and their delta feature across frames [1]. For each 25 msec long frame of speech, thirteen standard MFCC parameters are calculated by taking the absolute value of the STFT, warping it to a Mel frequency scale, taking the DCT of the log-Melspectrum and returning the first 13 components. Each MFCC feature vector is 167-dimensional.

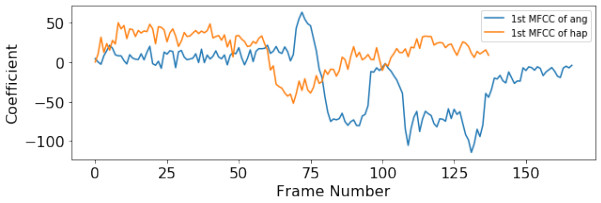

Figure 1: MFCC for two emotions

Figure 1 shows the variation of the 1st MFCC in emotional states of anger and happiness.

The pitch-based features are extracted from speech waveform. Using a frame length of 25 msec, the pitch for each frame is calculated and placed in a vector to correspond to that frame. The pitch-based features include pitch period $\tau_0(m)$ and the HNR[1], which is computed by:

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

where $ACF(\tau)$ represents the autocorrelation function at time τ, and m is the the frame number.

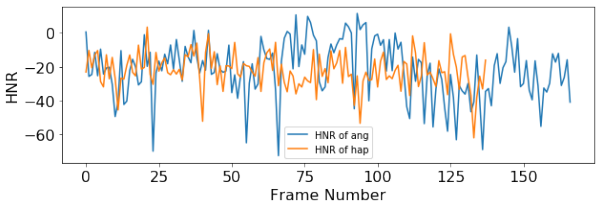

Figure 2: Pitch period for two emotions

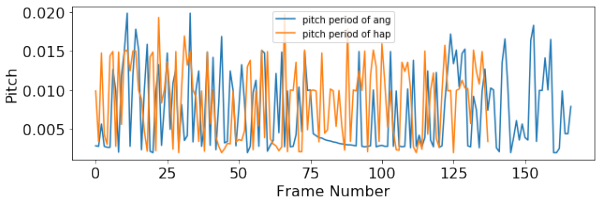

Figure 3: HNR for two emotions

Figure 2 and 3 shows the variation of pitch period and harmonic-to-noise ratio in emotional states of anger and happiness. Again each pitch based feature vector is 167-dimensional. For the segment level classification, the input fed into the DNN classifier are MFCC features, pitch-based features, and delta features. The target is the label of the whole utterance, i.e. we assign the same label for each segment.

2.2. Deep neural network for segment-level features

With segment-level features obtained from last step, we trained the DNN to produce a probability distribution of an emotion state over the whole utterance. It's worth noticing that the segmental emotion state is not necessarily identical to the emotion state of the whole utterance.

The DNN classifier is built upon fully connected layers. It takes segment level features as the input, and its input dimension is consistent with the dimension of the segment level features. Since DNN aims to find the probability distribution of a emotion over all of the segments, it uses softmax layer to produce five possible probabilities for five emotions that we care about. All the hyperparameters, such as number of hidden layers, number of hidden units in each layer, data keep probability in dropout, and optimizer choice are determined by grid search.

Figure 4 shows an example of a DNN output for an utterance over each segment for each emotion of *ang, exc, hap, neu, sad*. According to the figure, the probability for each segment

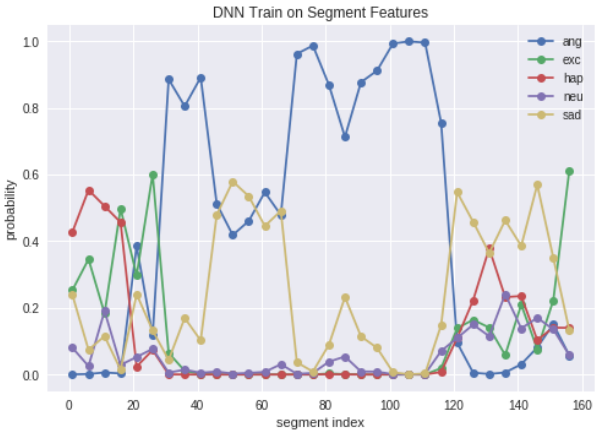

Figure 4: DNN outputs for an utterance, each line represents the probability for each emotion state

changes cross the whole utterance, and different emotion states dominate different segments. But overall, *ang* dominates most of the segments with the highest probability. Thus, the result is consistent with the ground truth utterance label *ang*, but not all of the other results perform this good.

2.3. Utterance-level features

Based on the segment-level features, we can form the utterance-level features as the input fed to ELM. Basically, the utterance-level features are the combination of four statistics of the probability distribution of an emotion state over a whole utterance. Specifically, we will use the maximal, minimal, average and the percentage above a certain threshold of the segment-level probability. Let $P_s(E_i)$ be the probability of emotion i of segment s that we obtained from the DNN output. Denote the maximal of the segment-level probability of emotion i over a whole utterance as f^i_{max}, then we have

$$f^i_{max} = \max_{s \in U} P_s(E_i)$$

where i can be excitement, anger, happiness, neutral and sadness and U stands for the whole utterance. Similarly, we define the minimal and average as

$$f^i_{min} = \max_{s \in U} P_s(E_i)$$

$$f^i_{avg} = \frac{1}{|U|}\sum_{s \in U} P_s(E_i)$$

Further more, we define f^i_{pct} as the percentage of segments which exceeds a probability threshold of emotion i as follows:

$$f^i_{pct} = \frac{|P_s(E_i) > \theta|}{|U|}$$

θ is a hyperparameter, for which we set to 0.2 as suggested in the reference[1]. At last, we stack these four vectors horizontally to form the utterance-level feature: $(f^i_{max}, f^i_{min}, f^i_{avg}, f^i_{pct})$. Each f^i has 5 dimensions which is consistent with the number of emotions. Thus, the dimension of our utterance feature is 20.

2.4. Extreme learning machine for utterance-level features

The last step of our approach is using a discriminative classifier to train and predict with the utterance-level features. We can choose from lots of classifier such as K-nearest neighbors (KNN) and support vector machine (SVM) [5]. In this project, we use extreme learning machine[6] which has been demonstrated to be very promising when the input dataset is small. ELM is very fast, thousands of times faster than back-propagation algorithm and can produce good generalization performance in most cases. Our training dataset for the last classifier contains 1400 utterance features, so it is very appropriate to use ELM.

ELM is a single hidden layer neural network with several unique properties which makes it perfect for small datasets. First, the number of hidden units is much greater than the number of input units. The input data are projected into higher dimensional space by the hidden layer weights. With a large number of hidden units, the random projection is able to represent training data and the projected data has higher chance to be linearly separable in that higher dimensional space. Second, the weights between the input layer and hidden layer is randomly generated from uniform distribution or radial basis function (RBF) distribution and then fixed during the training. Unlike the conventional back-propagation algorithm to train the deep neural network, ELM does not need any training for the hidden layer weights. The weights are generated independent of training dataset, so it has good generalization performance on new data. Further more, the weights between the hidden layer and output layer are not trained using back-propagation either. They are computed by a pseudo inverse operation instead. If the dataset is small, this pseudo inverse computation can be thousands of times faster than back-propagation.

Given a training set $S = \{\mathbf{x}_i, \mathbf{t}_i\}$, activation function $g(x)$, the ELM algorithm is summarized as follows:

1. Randomly generate weights and biases from uniform distribution.
2. Compute the hidden layer output matrix $\mathbf{H}$, $\mathbf{H} = g(\mathbf{W} \cdot \mathbf{X} + b)$.
3. Compute the output weights $\boldsymbol{\beta}$, $\boldsymbol{\beta} = (\mathbf{H}\mathbf{H}^T)^{-1}\mathbf{H}\mathbf{T}^T$, where $\mathbf{T}$ is the training label matrix.

3. Experiment

3.1. Corpus of Emotional Speech Data

The data used for this project is Interactive Emotional Dyadic Motion Capture (IEMOCAP) database which comes from Signal Analysis and Interpretation Laboratory at the University of South California. It contains 12 hours of audiovisual data, including video, speech, motion capture of face, text transcriptions[3].

The recordings consist of professional actors improvising and scripting a series of semantically neutral utterances spanning ten distinct emotional categories[3]. There were 5 female speakers and 5 male speakers. The number and count ratio of utterances that belong to each emotion category is shown in Table2

3.2. Experiment Settings

During the experiment, we chose five emotions: anger, excitement, happiness, neutral, and sadness over all of the 10 emotion states. According to the five emotions we chose, we reorganized the wav files into five different folders for each session, and, due to the compute limitation, sample the dataset with two methods: 1. random sample dataset with ratio 0.35 for each folder 2. random sample 70 examples for each folder, then construct cross validation dataset for further experiment. The size of the training set, develop set and test set are 1400, 350 and 350, respectively.

	ang	hap	exc	neu	sad
Counts	1103	595	1041	1708	1084
Ratio	19.9%	10.8%	18.8%	30.9%	19.6%

Table 2: Emotion counts and ratio in IEMOCAP

The input signal is converted into frames with a 25 msec window slides at 10 msec each time. The number of segment is set to be 25, so that the total length of segment is $10 * 25 + (25 - 10) = 265$ msec [1]. The reason of setting 265 msec each segment is that a segment longer than 250 msec contain sufficient emotion information [7].

The DNN for segment level classification has a 750 input units corresponding to the dimension of segment level features, and it contains three hidden layers with 256 hidden units for each layer activated by rectified linear function. RMSProp optimizer is chosen over Mini-batch, SGD, and Adam to learn the objective function which is cross-entropy.

The ELM for utterance level classification uses radial basis function distribution to generate the hidden layer weights and biases with the RBF width set to be 0.1. The input units for ELM corresponds to the dimensionality of the concatenated features calculated by the statistical functions over the segment level features.

3.3. Balanced dataset and adaboost

In our training dataset, 10% of them are happiness. It will not cause big problem if the number of training examples is large. However, in our first way of down sampling the original dataset, by using the extraction ratio 0.35, we have very few happiness examples, which leads to the prediction accuracy of happiness close to 0%. Thus, we construct a balanced training dataset using the second way of down sampling, i.e. sample a fixed number of examples from each emotion.

We trained the ELM using this balanced dataset and improved the prediction accuracy of happiness. However, the prediction accuracy of happiness is still lower than the other emotions. To solve this problem, we use adaboost to put more emphasis on happiness examples, using ELM as the base classifier, and this further improves the prediction accuracy of happiness. The number of hidden units of ELM is set to 300 and the number of base classifier of adaboost is set to 350 tuning on the development set.

3.4. Other classifiers for utterance-level feature

We also tried other classifiers for utterance-level feature prediction. The baseline approaches is support vector machine (SVM). Other classifiers, such as k nearest neighbors, ELM+bagging are also implemented with hyperparameter tuning, among which ELM+adaboost outperforms the rest.

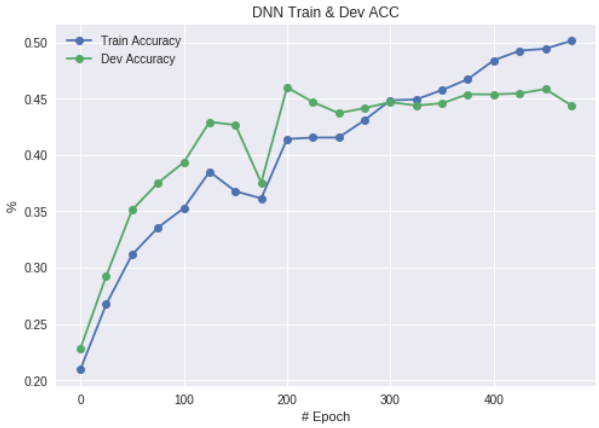

Figure 5: Segment emotion classification accuracy for training and develop dataset of DNN

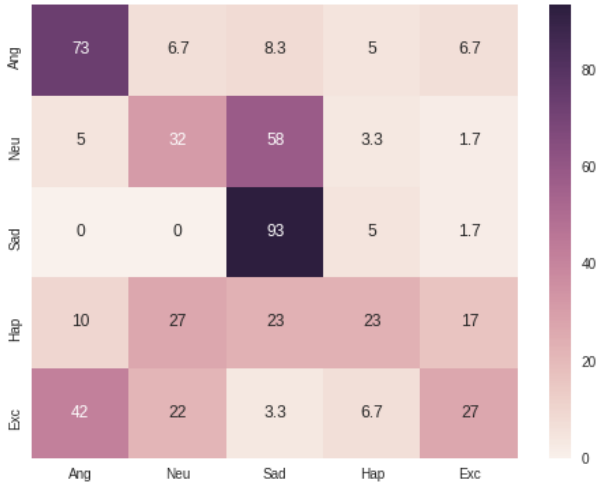

Figure 6: Confusion matrix of classification accuracy for AdaBoosted+ELM on 5 different emotion states

4. Results

The segment-level emotion accuracy of training and develop from DNN is shown in figure 5. We can see from the figure that the accuracy on develop set stays the same after 200 epochs, which means 500 epochs leads to over fitting. Thus, we use early stop and set the training epoch to 200.

Figure 6 shows the confusion matrix of the emotion recognition results of AdaBoosted+ELM tested on the test set. The row labels are the ground truth and column labels are the predicted emotions. We can see from the figure that the Adaboosted+ELM provides an outstanding classification accuracy on recognizing sadness and anger, and makes a big progress on improving the performance of classifying happiness from almost 0% to 23%.

We further compare our model with other speech emotion recognition approaches by using the classification accuracy on the whole test dataset. The baseline is Kun et al.[1]'s HMM model that trained on the segment-level features, which obtained 37.63% accuracy. The second method we compare to is traditional machine learning classification, KNN. We fit KNN model with the utterance level features, and obtained an accuracy of 35.40%. We also implemented an approach of utterance level SVM, and the approach delivers an accuracy of 38.8%. The proposed ELM approaches outperform other three methods by 10% in accuracy. We found that Adaboosted ELM makes an improvement by 5% from the performance of the ordinary ELM. The accuracy of each method is summarized in figure 7.

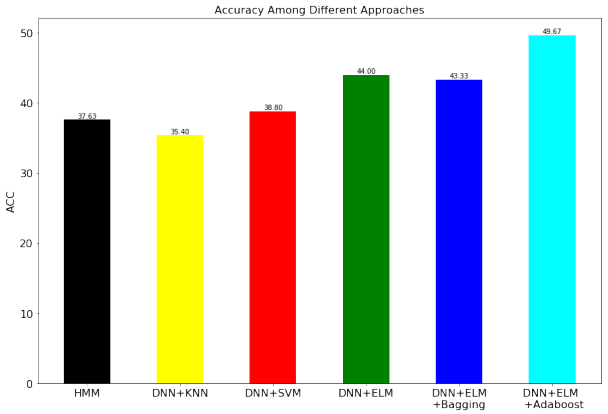

Figure 7: Comparison of different approaches in terms of accuracy. HMM method is the only method using segment level features, and other approaches are trained with utterance level features

5. Conclusions

In this project, we present a speech emotion recognition approach using segment-level features and utterance-level features. We use DNN to get the probability distribution of an emotion state over the whole utterance. Then we use ELM to predict the emotions for the utterance. We also found that constructing a balanced dataset and using adaboost can significantly improve the prediction accuracy of happiness. Our results demonstrate that this approach can substantially boost the accuracy compared to HMM.

6. References

[1] Fred G. Martin *Robotics Explorations: A Hands-On Introduction to Engineering*. New Jersey: Prentice Hall.

[2] Seyedmahdad Mirsamadi, Emad Barsoum, Cha Zhang 2017. *Automatic Speech Emotion Recognition Using Recurrent Neural Networks with Local Attention*. Washington: Microsoft Research, One Microsoft Way, 2017

[3] C. Busso, M. Bulut, C. Lee, A. Kazemzadeh, E. Mower, S. Kim, J. Chang, S. Lee, and S. Narayanan, *IEMOCAP: Interactive emotional dyadic motion capture database,* Journal of Language Resources and Evaluation, vol. 42, no. 4, pp. 335-359, December 2008.

[4] Kim, P.Georgiou, S.Lee, S.Narayanan. *Real-time emotion detection system using speech: Multi-modal fusion of different timescale features*, Proceedings of IEEE Multimedia Signal Processing Workshop, Chania, Greece, 2007

[5] E. Mower, M. J. Mataric, and S. Narayanan, *A framework for automatic human emotion classification using emotion*

profiles, Audio, Speech, and Language Processing, IEEE Transactions on, vol. 19, no. 5, pp. 10571070, 2011.

[6] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, *Extreme learning machine: theory and applications*, Neurocomputing, vol. 70, no. 1, pp. 489501, 2006.

[7] . 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.

[8] Schuller, G. Rigoll, and M. Lang, *Hidden markov model-based speech emotion recognition,* in Proceedings of IEEE ICASSP 2003, vol. 2. IEEE, 2003, pp. II1.