Speech Emotion Recognition based on Gaussian Mixture Models and Deep Neural Networks

Ivan J. Tashev
Microsoft Research Labs
Redmond, WA 98051, USA
ivantash@microsoft.com

Zhong-Qiu Wang
Dept. of Computer Science and Engineering
The Ohio State University,
Columbus, OH 43210, USA
wangzhon@cse.ohio-state.edu

Keith Godin
Microsoft Corporation
Redmond, WA 98051, USA
kegodin@microsoft.com

***Abstract*—Recognition of speaker emotion during interaction in spoken dialog systems can enhance the user experience, and provide system operators with information valuable to ongoing assessment of interaction system performance and utility. Interaction utterances are very short, and we assume the speaker's emotion is constant throughout a given utterance. This paper investigates combinations of a GMM-based low-level feature extractor with a neural network serving as a high level feature extractor. The advantage of this system architecture is that it combines the fast developing neural network-based solutions with the classic statistical approaches applied to emotion recognition. Experiments on a Mandarin data set compare different solutions under the same or close conditions.**

***Index Terms*—emotion detection, deep neural networks, Gaussian mixture models, Extreme learning machine.**

I. Introduction

Spoken dialog systems can be enhanced with an emotion recognition component. It can be deployed as part of the interaction system's design, where it may influence the interaction system's reaction to the user's spoken utterances, or may otherwise enhance the user interface. It may also be applied in an ongoing assessment of system performance, to provide system operators with information on user enjoyment or utility of the service. With better understanding of the human and the emotion in spoken query, the spoken dialog system can thus achieve a better user experience.

There are two main approaches for emotion classification: dimensional and categorical. The dimensional approach provides values of arousal and valence, and the recognized emotion is actually a point in this plane. More common are categorical emotion detectors, which assign to the speech utterance one emotion, such as happy, stressed, sad, neutral, etc. In the second case, the input of the speech emotion recognition is a variable length sequence and the output is a single label, which converts emotion recognition into a classification problem. This makes it similar to speaker identification [1], gesture recognition, sentiment analysis in natural language processing [2], and other classification tasks. We limit our investigation in this paper to categorical emotion detection.

A major problem in emotion detection from speech is labeling - even humans cannot agree on the exact emotion in the spoken utterance. This is mitigated by using several judges to label each one of the utterances in the data set. In addition, the training data for building emotion recognizers is normally highly imbalanced, further increasing the difficulty of making accurate predictions.

The emotion detection systems typically start with extraction of low-level features. Some of them are in time domain, but most are obtained after splitting the input audio signal into overlapping frames, applying weighting window and converting into frequency domain. This research area is emerging and there is still no consensus on what features we should use for the task. In many studies, people just combine a lot of features for classification [3].

There are several approaches for building a classifier for emotion recognition, based on hypotheses about the character of the human emotions in short utterances. One approach processes the utterance frame-by-frame. As the duration of one audio frame is short, typically 10–30 ms, each frame provides little information on speaker emotion. An emotion decision can only be reliably made by considering all of the frames in an utterance. The most conventional method to aggregate frame scores is to train one Gaussian Mixture Model (GMM) - Hidden Markov Model (HMM) system for every emotion, and assign the emotion label according to the one giving largest likelihood at the test stage [4]. Recurrent Neural Networks (RNN) with long-short term memory (LSTM) may also be applied to this task. In [5], the same utterance-level label is assigned to every frame for LSTM training. At the test stage, frame-level predictions are averaged to make final prediction.

Assigning the same label to every frame, especially to silence frames, may incur some problems. One problem is that the data sets for emotion recognition are normally imbalanced. In such cases, many silence frames would be labeled as the majority classes, and therefore the predictions on new silence frames would be highly biased towards majority classes. Another problem is that, even if the utterance is labeled as one emotion, it does not necessarily mean that every frame should be labeled as that emotion. To deal with this problem, Lee and Tashev [6] propose an RNN-CTC (Connections Temporal Classification) approach, in which they assume that different frames should have different labels, and the label sequence should be alternating between the utterance-level label and a newly-introduced NULL state. In their study, expectation maximization (EM) algorithms are used for inferring the uncertainties in the label sequence.

A second approach is to group consecutive frames in segments, and use a classifier for evaluation of the emotional content of the speech in each segment. A Support Vector Machine (SVM) [7], or a Deep Neural Network (DNN) [8] may serve as a primary classifier. The result of this high-level feature extractor is a set of variable length vectors, one for each of the emotions. The number of elements is the number of segments, and each element is the probability of having this emotion in the segment. A second stage classifier is required to form a decision from the set of variable length vectors. Statistics over the variable length vectors including mean, median, standard deviation, and others, are computed, and the result is a fixed-length feature vector. The second-stage classifier is applied to the fixed-length vector, which can be an SVM [9], ELM [10], or kernel ELM [11], with variations of adding a soft-max block at the end [12].

All of these architectures require selection of speech-containing segments, and removal of the silence and low energy frames. This can be implemented as selection of the segments with top 15% energy [8], or by running a Voice Activity Detector (VAD) and using only voiced frames for training and evaluation [12].

A third approach is to process statistically the entire voice query first, and perform final classification into emotion categories second. A typical statistical frontend is to use a GMM in the same way they are used for speaker verification [13]. The voice query is compared with the GMM of the speaker to be verified, and with the GMM of the Universal Background Model (UBM). The decision is taken based on which one is closer, in addition applying some combined weight for false positives and false negatives. This classic approach is applied for emotion recognition from speech signals in [14]. The classic GMM approach assumes that all of the input features (MFCC filters, pitch, deltas, and delta-deltas) have equal weight during computation of the log-likelihood. This is most probably not true for both tasks. Speaker specific information is carried mostly in the upper part of the frequency band, and pitch carries more information than the lower part of the frequency band. This is also true for emotion recognition, plus the higher importance of the speech dynamics, i.e. deltas and delta-deltas. Another GMM-based approach is to have the distances between the voice query and the GMM signatures of different emotions computed per feature, and then use a secondary classifier which will learn the different weights of the features during the training, and will produce the final classification. In all cases only voiced frames are used for building the GMM signatures and computing the distance.

In this paper we compare aspects of the second and third approaches. The general difference between them is that the second approach does emotion classification first and statistics second, while the third approach does statistics first and emotion classification second. Under as close as possible conditions we compare the performance of these two approaches. The rest of the paper is organized as follows. In Section II we describe the algorithms to be compared, in Section III are described the data corpus and the evaluation parameters. Section IV provides the results, while Section V draws some conclusions.

II. Algorithms Description

We use a voice activity detector to select active frames. Then the same set of features from the voiced frames are fed into all compared classifiers.

A. Voice Activity Detection

The voice activity detector used in our study is a statistical activity classifier with HMM smoothing of decisions [15], [16]. To refine the VAD results, we apply a hangover scheme and throw away the segments with less than five consecutive active frames to remove sudden bursts, the sound of puff of air, or potential clicks at the beginning and at the end of each utterance. Another advantage of using VAD is that even if there is long silence at the beginning or at the end of an utterance, the behavior of the classifier would not be negatively influenced.

B. Classic GMM approach

This is a straightforward implementation of the GMM-based speaker identification algorithm. During the training phase we create one GMM-signature for each of the emotions in the same way the UBM model is created. During the evaluation phase the log-likelihood distance between the voice query and each of the emotions is computed, treating equally the different features and their derivatives. The classification decision is for the emotion the voice query is closest to.

C. GMM-ELM approach

The problem we address here is that different features (each of the Mel filters, pitch, their deltas) may carry more or less information about the emotion. During the training phase we create one GMM-signature for each of the emotions in the same way as above. During the evaluation phase the log-likelihood distance between the voice query and each of the emotions is computed for each one of the features and deltas separately. This new vector of features is the input of an ELM, which produces the final classification. During the training phase ELM learns the different weights and dependencies of the distances between low-level features and each of the emotions.

D. GMM-DNN approach

This is the same structure as the previous one, except that the secondary classifier is a DNN. In this case we expect the better abstraction capabilities of the DNN to provide better classification rate if we do have the information in the per-feature distances.

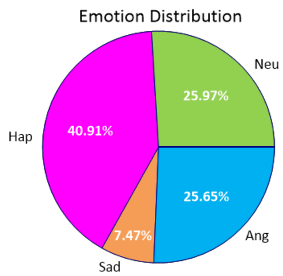

Fig. 1. Class distribution for the emotion recognition task.

E. DNN-ELMK algorithm

While all of the previous three algorithms were using statistics for the entire utterance, extracted by the first level classifier, this is the algorithm which extracts the low-level features for each segment and then computes the statistics for the entire utterance. This a subset of the DNN-ELMK algorithm, described in [8]. Every utterance is a mini-batch for the DNN training. The two-layer network is trained jointly, i.e. each iteration consists of training of the DNN, extracting the statistics, and then training the ELMK.

III. Experimental Setup

We have collected $17,408$ real-traffic Mandarin utterances from a Microsoft spoken dialogue system. Each utterance is labeled by five crowdsourcing judges using the Microsoft UHRS labeling system. All of the judges are native Mandarin speakers. There are four meta-categories for the emotion recognition task, i.e. neutral (no clear emotion), happy (excited/interested/happy/funny/flirting), sad (depressed/bored/tired/sad/frustrated) and angry (disgust/impatient/offended/angry). We have finer categories for each meta-category when prompting the crowdsourcing judges to label the data, mainly because speech emotion itself is very fuzzy.

Although a large number of labeled utterances can be quickly obtained using crowdsourcing judges, the labels are less reliable compared with professional and serious annotators. Therefore, utterance selection is of great importance for building good emotion recognition classifiers. In our study, we only retain utterances with labels like AAABC, AAAAB, and AAAAA, as the annotations with less than three agreements are considered unreliable. For these retained utterances, we use majority voting to label each utterance. The performance of human labelers in this subset is 82.18%. Note that utterances with labels like AABB are not considered, because the underlying emotion could be A, but also with high probability it could be B. If we incorporate these, the performance of human labelers drops to around 75%.

From the initial set of $17,408$ utterances, $10,527$ utterances (approximately 10 hours) are left after filtering using the aforementioned criterion. As we can see from Fig. 1, the class distribution is very imbalanced, which is typical for many other data sets for this task. Nonetheless, we think that the distribution is reasonable from a practical perspective. More details about the utterances selection can be found in [12].

In our experiments, we randomly choose 70% of the data for training, 15% for validation, and the remaining 15% for testing. We use the validation set to perform hyper-parameter tuning and early stopping.

We use weighted accuracy and un-weighted accuracy to measure the performance, as it is done in many other studies. The weighted accuracy is just the classification accuracy over the entire test set. The un-weighted accuracy is the average of the classification accuracies for each class, which accounts for the imbalanced nature of the data. In this study, we focus more on the weighted accuracy, as it represents the percentage of users we can satisfy and it is self-weighted from the usability standpoint. We have tried to give larger weights to minority classes so that un-weighted accuracy can be improved, but in such cases the weighted accuracy drops substantially.

IV. Evaluation Results

We compare the four proposed approaches keeping the feature sets and the complexity of the neural networks as close as possible. The low level feature set consists of energy, pitch, voice probability, and 26-dimensional log Mel-spectrogram features extracted from each frames. For the log Mel-spectrogram features, we perform utterance–level mean normalization as this can alleviate the channel effects of different microphones. The delta component of all the features mentioned above is added to consider the dynamics of these features. The number of features extracted from each frame is 58. Only frames with voice presence probability larger than 0.95 are used for training, validation, and testing.

For the GMM-based algorithms we use fitting with 16 Gaussians for each of the four emotions. For training were used all of the utterances in the training set, split on separate for each emotion groups. The GMM means, variances, and weights were estimated using HTK [17]. The frame presence probability and the frame energy do not make sense for the GMM algorithms, so they were excluded from the feature set. For the classic GMM algorithm the voiced frames are evaluated against each of the emotions and the overall log-likelihood is computed. The emotion with highest log-likelihood is the classification decision. For the GMM-ELM and GMM-DNN algorithms we compute the log-likelihood for each of the 54 features and four emotions. This leads to 216 features vector, which is the input for the ELM and DNN networks.

The number of neurons in the hidden layer of the ELM is set to be five times the size of the input vector, i.e. 1080. The implementation is pretty standard and according to [10]. For training of the ELM was used the evaluation set.

There are four hidden layers in all DNNs, each with 128 rectified linear units. The network is trained with mini-batch stochastic gradient descent (SGD) algorithm with momentum to minimize the cross-entropy criterion. The 58 features used in our study are concatenated to form a symmetric 25–frame context window to obtain our final frame–level features. The input feature dimension is therefore $1,450$. The output of the

TABLE I

ACCURACY IN % OF THE PROPOSED METHODS ON THE SPEECH EMOTION RECOGNITION TASK

Algorithm	Val. WA	Val. UA	Test WA	Test UA
GMM	38.02	38.03	38.44	39.98
GMM–DNN	50.12	40.33	48.00	41.46
GMM–ELM	51.64	38.57	51.61	39.97
DNN–ELMK	55.73	47.63	57.95	50.42
Human			82.184	80.26

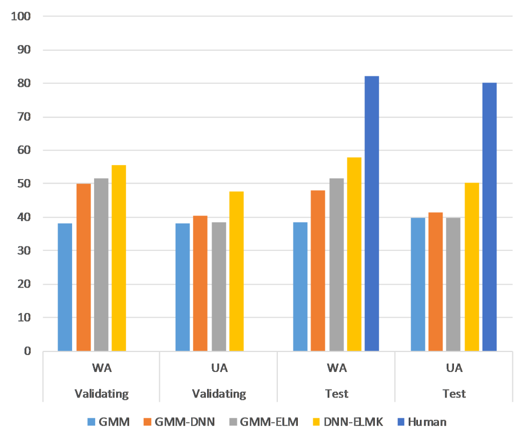

Fig. 2. Accuracy in % of the proposed methods on the speech emotion recognition task.

DNN for each segment is four numbers - the probabilities for each of the emotions.

The number of these four probabilities is equal to the number of segments. From this variable length vectors are computed mean, max, min, and variance for each emotion. This fixed length vector of 16 features is the input of kernel ELM algorithm, implemented according to [10]. It is trained together with the DNN on each iteration. The output is a vector of four variables - the probability for each emotion. The utterance is labeled as having the emotion with the highest probability. The weighted and un-weighted accuracies for the validation and testing data sets are presented in Table I and illustrated in Fig. 2, where we have added for reference the accuracy of the human labelers. We will discuss and compare the numbers of the weighted accuracy from the test data set, the trends for the un-weighted accuracy and for the validation data sets are the same. The classic GMM approach has the lowest performance of 38%. Using a DNN or ELM to apply different weights to different features increase the accuracy to 48% and 51.6% respectively. Apparently we do not need the abstraction power of a DNN with four hidden layers, the single hidden layer ELM does a sufficient job (actually slightly better). With the 57.9% accuracy the DNN-ELMK remains the best performing algorithm. There is a substantial difference in the weighted and un-weighted accuracies due to the imbalanced data set. Still, weighted accuracy reflects better the usability of the proposed algorithms.

V. CONCLUSIONS

We have compared the performance of several GMM-based algorithms, which estimate the statistics of the entire utterance first and perform classification second, with our state of the art DNN-ELMK algorithm, which performs classification on segments first, and computes statistics second. Overall the GMM-based algorithms cannot produce accuracy comparable to the DNN-ELMK algorithm performing emotion detection on segments of 0.25 sec. In the GMM-based algorithms computing the log-likelihoods for each of the features separately and combining them with different weights through a simple ELM provided very good improvement. Logical next steps are to evaluate a segment-based GMM algorithm for the needs of emotion recognition and to evaluate the improved GMM-ELM algorithm against the classic for this algorithm scenarios for speaker identification and verification.

REFERENCES

[1] J. Hansen and T. Hasan, "Speaker recognition by machines and humans: A tutorial review," *IEEE Signal Processing Magazine*, pp. 74–99, 2015.

[2] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov, "Bag of tricks for efficient text classification," in *arXiv preprint arXiv: 1607.01759*. 2016.

[3] F. Eyben, M. Wllmer, and B. Schuller, "OpenEAR—introducing the Munich Open-source Emotion and Affect Recognition Toolkit," in *Affective Computing and Intelligent Interaction and Workshops*, 2009, pp. 1–6.

[4] B. Schuller, G. Rigoll, and M. Lang, "Hidden markov model-based speech emotion recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2003.

[5] G. Keren and B. Schuller, "Convolutional RNN: an enhanced model for extracting features from sequential data," in *arXiv preprint arXiv:1602.05875*. 2016.

[6] Jinkyu Lee and Ivan J. Tashev, "High-level feature representation using recurrent neural network for speech emotion recognition," in *Interspeech*, 2015.

[7] H. Hu, M. Xu, and W. Wu, "GMM supervector based SVM with spectral features for speech emotion recognition," in *ICASSP*, 2007.

[8] Kun Han, Dong Yu, and Ivan J. Tashev, "Speech emotion recognition using deep neural network and extreme learning machine," in *Interspeech*, Sept. 2014.

[9] 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.

[10] G. Huang, H. Zhou, X. Ding, and R. Zhan, "Extreme learning machine for regression and multiclass classication," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 42, no. 2, pp. 513–529, 2012.

[11] Z. Bai, G. Huang, W. Wang, H. Wang, and M. Westover, "Sparse extreme learning machine for classication," *IEEE Transactions on Cybernetics*, 2014.

[12] Zhong-Qiu Wang and Ivan J. Tashev, "Learning utterance-level representations for speech emotion and age/gender recognition using deep neural networks," in *ICASSP*, 2017.

[13] D. Reynolds, T. Quatieri, and R. Dunn, "Speaker verication using adapted Gaussian Mixture Model," *Digital Signal Processing*, July 2000.

[14] N. Thapliyal and G. Amoli, "Speech based emotion recognition with Gaussian Mixture Model," *International Journal of Advanced Research in Computer Engineering & Technology*, 2012.

[15] J. Sohn, N. Kim, and W. Sung, "A statistical model based voice activity detector," *IEEE Signal Processing Letters*, vol. 6, pp. 1–3, 1999.

[16] Ivan Tashev, Andrew Lovitt, and Alex Acero, "Dual stage probabilistic voice activity detector," in *NOISE-CON 2010 and 159th Meeting of the Acoustical Society of America*, 2010.

[17] S. J. Young, G. Evermann, M. J. F. Gales, T. Hain, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. C. Woodland, *The HTK Book, version 3.4*, 2006.