CONVOLUTIONAL NEURAL NETWORK TECHNIQUES FOR SPEECH EMOTION RECOGNITION

Srinivas Parthasarathy*[1], Ivan Tashev[2]

[1]University of Texas at Dallas, Dallas, TX, USA
[2]Microsoft Research Labs, Redmond, WA, USA
Srinivas.Parthasarathy@utdallas.edu, ivantash@microsoft.com

ABSTRACT

Affect recognition plays an important role in *human computer interaction (HCI)*. Speech is one of the primary forms of expression and is an important modality for emotion recognition. While multiple recognition systems exist, the most common ones identify discrete categories such as happiness, sadness, from distinct utterances that are a few seconds long. In many cases the datasets, used for training and evaluation, are imbalanced across the emotion labels. This leads to big discrepancies between the *unweighted accuracy (UA)* and *weighted accuracy (WA)*. Recently *Deep Neural Networks* have shown increased performance for the emotion classification task. In particular *Convolutional Neural Networks* capture contextual information from speech feature frames. In this paper we analyze various convolutional architectures for speech emotion recognition. We report performance on different frame level features. Further we analyze various pooling techniques, on top of convolutional layers, to get a utterance level representation for the emotion. Our best system provides a performance of *UA+WA* of 121.15 compared to the baseline algorithm performance of 118.10.

Index Terms— speech emotion recognition, deep neural networks, convolutional neural networks, pyramidal pooling.

I. INTRODUCTION

Affective computing is the art of recognizing emotions from various modalities. It is widely growing within the field of *Human Computer Interaction*. Speech remains a primary form of expressive communication. Predominantly, speech emotion recognition systems are built to either classify speech utterances, which typically range a few seconds in duration, into discrete categories such as sadness, anger, happiness [1], or emotional attributes such as arousal (passive vs active), and valence (negative vs positive) [2]. Complex emotions can be expressed with emotional attributes and lately, systems that recognize emotional attributes have been gaining popularity. But systems that classify utterances into a few categories remain popular due to the ease of understanding the categories. A typical framework for such system is shown in Figure 1. The entire framework can be divided into the following steps consisting a feature representation stage and a classification stage. First, low level features are extracted for every speech frame (typically $10-20$ milliseconds), or speech segment (typically $10-25$ frames). Then, a global high level feature representation, typically statistic, is learnt for the entire utterance. While the classifier could be trained to map every speech frame or segment to the emotional class, this might not be underlying case. Therefore, the high level representation provides a many to one mapping for emotion classification. Further, feature representations are modified using various algorithms. Finally, a classifier is built on top to learn the underlying emotional state. The two stages are mostly trained independently.

Recently, *Deep Neural Networks* (DNNs) [3] have shown promising performance for the emotion classification task. The simplest DNN systems for emotion recognition are feedforward networks that are built on top of the utterance level feature representations [4]. *Recurrent Neural Networks (RNN)* [5] are a class of neural networks that have cyclic connections between nodes in the same layer. These networks capture the inherent temporal context in emotions and have shown improved performance for classification task [6]. Another class of DNNs, *Convolutional Neural Networks (CNN)* [7], capture locally present context, patterns, working on frame level features. CNNs enable the training of end to end systems where the feature representations and classification are trained together using a single optimization.

Few works have analyzed the performance of CNNs for speech emotion classification [8], [9], [10]. Cummins *et al.* [11] further built image-based CNNs on spectrograms features. In this paper we experiment with different CNN architectures for solving the emotion classification problem by varying the number of convolutional layers, kernel sizes, and analyzing their effect on the accuracy. We also report performance with different combinations of low level features. First, we perform experiments with a baseline feature set including log-Mel spectrum features, F0, energy, speech presence probability. We extend the study to include

*Work on this project performed as an intern at Microsoft Research Labs, Redmond, WA.

978-1-5386-8151-0/18/$31.00 ©2018 IEEE

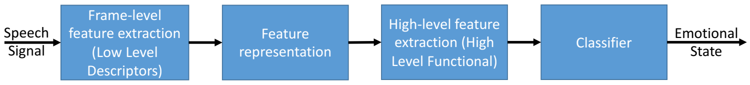
Speech Signal
Frame-level feature extraction (Low Level Descriptors)
Feature representation
High-level feature extraction (High Level Functional)
Classifier
Emotional State

Fig. 1. Typical framework for emotion classification system.

only different log Mel-spectrum bands as well as linear spectrograms. We also report performance with different pooling layers to attain utterance level statistics.

The contributions of the paper are chiefly (1) the analysis of various CNN architectures for emotion classification, (2) the analysis of pooling layers, especially the pyramidal pooling layers, for attaining utterance level representation, (3) using UA+WA as a metric for optimizing and evaluating the system, and finally (4) using the annotation distribution rather than one hot vectors as ground truth labels to train our emotional systems.

II. BACKGROUND WORK

II-A. Database

For this study we use a dataset which contains $17,048$ sentences in Mandarin from Microsoft spoken dialogue system XiaoIce [12]. Each utterance is annotated for emotional content by five human judges. Labelers chose from four main categories: happy, angry, sad, and neutral. The perceptual annotation process is fuzzy with many ways to construct the ground-truth label. Most studies use a majority vote as ground-truth, i.e. a distribution such as AAAAB or AAABC would be labeled as A. Wang and Tashev [12] only retained utterances with at least three labeler majority i.e. classes distributions such as AAABB would not get a label and used. For the rest they used a majority label as the true label. Doing so would increase the human labelers accuracy from 75% to 82.18% but at the cost of decreasing the number of available utterances to $10,527$. Note that in both cases the ground-truth label is used as a one-hot vector with the majority class. Figure 2 shows the distribution of emotion labels in this dataset. Note that there is an imbalance in the dataset as users are chat predominantly happy talking to the chatbot.

II-B. Baseline

For baseline we use the algorithm proposed in [12]. The speech signal is split into 25 ms frames at a frame rate of 100 Hz. For each frame 29 low-level features are extracted, which include 26-band log Mel-spectrum, fundamental frequency (F0), energy, and speech presence probability. Along with these the deltas were also used for each frame totaling 58 features per frame. A voice activity detector [13] was used to identify the frames with speech within the utterance. Only these frames are used for classification. A context window of 25 frames is used to final the segment level feature representation, totaling $1,450$ features per segment. Each segment is processed by a feed forward DNN with 4 hidden layers and 512 nodes at each layer. On top of this a mean pooling layer is used to get the utterance level feature representation. Finally a softmax classifier is used to train the system. Given the relatively small size of the training data, an extreme learning machine (ELM) was fitted on the final utterance level features replacing the softmax classifier. The ReLU function is used for activation. For the baseline the authors split the XiaoIce dataset into training, validation, and test sets with 70% of the data for training, 15% for validation, and 15% for testing. As a performance metric authors use weighted accuracy (WA). Given the imbalance in the dataset, the authors justify the use of WA as metric from the user standpoint as they want the majority of utterances to be better classified. III-A

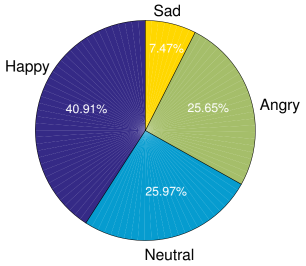

Fig. 2. Distribution of the emotion labels in XiaoIce dataset.

III. EXPERIMENTS

III-A. Preliminary experiments

In this study we propose to use the true annotator distribution rather than one hot vector to represent the ground truth. Steidl *et al.* [14] hypothesised that clasification performance should be evaluated considering the confusion between labelers. Few studies have considered methods to consider annotator distribution as soft labels [15], [16]. In our work the annotator distribution is represented as a probability vector with the probability for each class. This we believe is a better representation of the fuzzy labels. There are two main advantages of doing this. First, we capture the true variability in the emotional content. Second, we gain the use of extra samples for training, which were discarded in the baseline model. Note that to make comparisons to baseline model we have to keep the test and validation sets fixed. Therefore the gained samples are only added to the training set. We performed preliminary experiment with the baseline architecture, described in Section II-B, and true annotator distribution as ground-truth. We used 2 hidden layers and retained the traditional softmax layer for classification. Additionally, given the imbalance in classes in the dataset, we realize that both the UWA and WA are important. While UWA captures the true performance of the classifier the WA captures performance in terms of the true standpoint. Therefore we use WA+UWA as our evaluation metric for all experiments in this study.

III-B. Experiments with Convolutional Neural Networks

For all experiments with CNNs the framework for classification is as follows. First, we extract low level features

Table I. Architectures for emotion detection with CNNs * signifies a corresponding note for the architecture

Architecture	log Mel-bands	CNNs	Dense layer	Pooling	FC layer	Note
CNN-BS1	26	1	no	mean	no	
CNN-BS2	26	1	no	mean	yes	
CNN-BS3	26	3	yes	mean	no	
CNN-BS4	26	1*	no	mean	no	multiresolution
CNN-BS5	40	3	yes	mean	no	
CNN-SPECT1	26	1	no	mean	no	
CNN-SPECT2	26	1*	yes	mean	yes	multiresolution
CNN-SPECT3	256*	1	no	mean	no	spectrum
CNN-SPECT4	512*	1	no	mean	no	spectrum
CNN-PP1	26	3	yes	pyramidal	no	4x1,3x1,2x1,1x1
CNN-PP2	26	3	yes	pyramidal	no	6x1,3x1,2x1,1x1

per frame. Then we build convolutional layer(s) on top of the low level features to extract patterns from the individual frames. We may, or may not, have a dense layer on top of the convolutional layers. Next, a pooling layer is added to acquire an utterance level feature representation for emotion classification. While theoretically a classifier could be trained to learn all frames in an utterance to a particular emotional label, practically this fails as not all frames in an utterance correspond to the particular emotion, especially silence within an utterance. Therefore, the pooling layer consolidates features from all the frames within an utterance and provides a utterance level representation. In some cases we insert a fully connected layer after the pooling layer. A softmax layer is then added to provide the final classification. All systems are trained with mini-batch statistical gradient descent (SGD) with batch size of 128 utterances, using an ADAM optimizer [17] with learning rate $1e^{-4}$. The explored CNN architectures for emotion recognition task are summarized in Table I.

Baseline features + CNN: In our first set of architectures we use the same set of 58 per frame low level features as used in the baseline and preliminary framework. First we use one convolutional layer on top of the low level feature frame (CNN-BS1). Since the low level features are of different dimensions, the convolutional kernel height is fixed as 58, resulting in 1-D convolution operations. The convolutional kernels are of size ($58 \times T \times K$), where T corresponds to the width of the kernel in the time axis, and K corresponds to depth, or number of feature maps. The width of the kernel T corresponds to the context window used in the baseline architecture and preliminary experiments. It is fixed to 24, which corresponds to a 240 ms context window for emotion classification. We shift the filter by one frame with padding, therefore width of the frames is maintained. We fix the number of feature maps K to match the number of parameters in the baseline model. In the second architecture (CNN-BS2) we introduce a fully connected layer on top of the convolutional layer after the mean pooling layer. In our third architecture (CNN-BS3) we use a deeper model by placing three convolutional layers. Each layer uses a kernel of width ($58 \times 24 \times K$) as before. We also add a dense layer of size ($512 \times K$) before the pooling layer at each time frame. With our next architecture (CNN-BS4) we explore convolutional operations with kernels of different widths to capture feature patterns with varying temporal context. We use 3 convolutional kernels with size ($58 \times 16 \times K$), ($58 \times 24 \times K$), ($58 \times 32 \times K$) respectively. We expect to capture short and longer temporal patterns using these kernels. We also experimented with higher frequency resolution by increasing the number of log Mel-spectrum bands from 26 in CNN-BS3 to 40 in CNN-BS5.

Spectral features only + CNN: In another set of architectures we consider only the spectral features as low level features. In addition to evaluating the effect of the spectral features only, we can also study the effect of 2-D convolutional operations on the low level features. First we use the 26 band log Mel-spectrum features as low level features (CNN-SPECT1). We treat all frequencies independently and perform 1-D convolution on the low level features with a kernel of size ($26 \times 24 \times K$). In our second architecture we use 3 different kernels to perform the convolutions (CNN-SPECT2). With the first kernel ($1 \times 32 \times K$) we capture the temporal information without any frequency information. The second kernel ($26 \times 1 \times K$) captures the broadband frequency information without the temporal information. This operation is similar to the non-negative matrix factorization. The third kernel ($8 \times 24 \times K$) captures information from short time window, in sub-bands. We use a stride of one in time and frequency directions. A second convolutional layer with only 1-D temporal convolution is added on top of the first layer. Finally we also perform convolutions on linear spectrograms rather than log Mel-frequency scale. We use two window sizes (256 for CNN-SPECT3, and 512 for CNN-SPECT4) for calculating the spectrograms. We perform 1-D temporal convolutions, treating each frequency independently, similar to the baseline architecture CNN-BS1. Note that in the case of linear scale spectrograms we use all frames of the utterance irrespective of speech or non speech frames, which may lead to lower accuracy.

Pyramidal pooling: In this group of architectures we replace the pooling layer. Rather than doing one mean pooling we adopt the concept of pyramidal pooling [18].

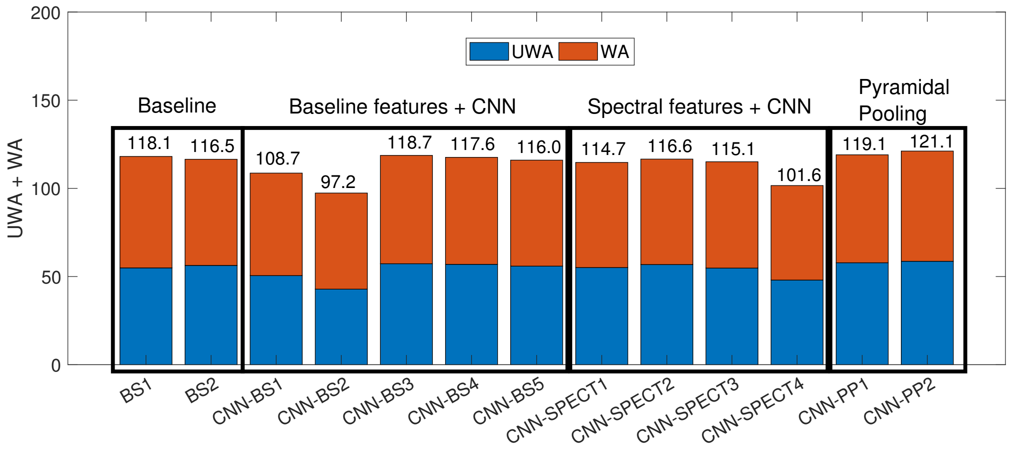

Fig. 3. Experimental results.

Here, the input signal is partitioned into smaller regions and the pooling operation is repeated on each region. The pyramidal pooling layer is invariant to the dimensions of its input as well. The number of partitions that form the pyramid stay constant and only the dimension of the partition varies with the input dimension. Further, the coarsest representation of the pyramidal layer is the single mean pooling done with the earlier experiments. The pyramidal layer also captures the statistics from features from different regions in the utterance which we believe is suited for emotion recognition. We experiment with two pyramidal mean pooling layers. The first one (PP1) has 4 pyramidal levels (4×1, 3×1, 2×1, 1×1) and the second one (PP2) has pyramidal levels (6×1, 3×1, 2×1, 1×1). Both pyramidal layers are tested on the convolutional architecture with baseline features which contain 1-D convolutional filters. The pooling layers are therefore done in the time domain.

IV. RESULTS

The results are summarized in Figure 3. As declared in section III-A the evaluation criterion is the sum of UWA and WA. Note that the results show the absolute value of the accuracies and the best value possible is 200. Performance of UWA and WA is highlighted in blue and red respectively. The baseline architecture, described in section II-B and denoted BS1, shows performance of 118.1. The performance of the baseline architecture and true annotator distribution as ground-truth, described in section III-A, is denoted (BS2) with overall UWA+WA of 116.5. We notice that although the UWA+WA is lower than that of the baseline system, we achieve better UWA, without hurting much the WA. Thus, even with smaller number of nodes (2 instead of 3 layers), we achieve similar performance to BS1 with the help of the true annotator distribution.

For the first CNN architecture with baseline features CNN-BS1 we notice a drop in performance for both the WA and UWA. The shallow convolutional layer might not be able to capture the feature representations needed to classify the emotions. The introduced fully connected layer after the mean pooling in CNN-BS2 brings more destruction which decreases performance. Adding a dense layer in CNN-BS3 works quite well and the deeper network is able to extract better features and we see an increase in performance with results comparable to our baseline. Multiresolution CNN-BS4 shows performance that is comparable to BS1. In both these cases we increase UWA without a big decrease in WA. Increasing the number of log Mel-filters in CNN-BS5 did not produce better results compared to CNN-BS3.

For CNN-SPECT1 the UWA+WA is smaller than the baseline features which shows that energy, F0, and speech presence probability features add value to the CNN architectures. CNN-SPECT2 shows better performance and it is comparable with, but lower than the CNN-BS4 architecture. Comparing CNN-SPECT3 and CNN-SPECT4 we see better performance with 256 window than with 512, and the former performs slightly better than CNN-SPECT1.

The two pyramidal pooling architectures produce the best results. They make significant gains in terms of UWA, while maintaining the WA, which was a drawback with previous works.

V. CONCLUSIONS

In this paper we analyzed the performance of various CNN-based architectures for the task of emotion recognition from speech. Overall these architectures outperform the architectures with fully connected networks, used as baseline. Critical for the success is the pooling approach, with pyramidal pooling bringing the highest accuracy. In the feature set the frame energy, F0, and speech presence probability contribute to the better accuracy. For training and evaluation we used UWA+WA as a metric, which brought good results even with imbalanced datasets. Using the annotation distribution rather than one hot vectors as a ground-truth labels increased the accuracy and led to better utilization of the labeled dataset.

VI. REFERENCES

[1] 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 chalenge,” *Speech Communication*, vol. 53, no. 9, pp. 1062–1087, 2011.

[2] F. Weninger, F. Ringeval, E. Marchi, and B. Schuller, “Discriminatively trained recurrent neural network for continuous dimensional emotion recognition from audio,” in *Proceedings of IJCAI*, 2016, pp. 2196–2202.

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

[5] G. Keren and B. Schuller, “Convolutional rnn: an enhanced model for extracting features from sequential data,” *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] Yann LeCun, Fu-Jie Huang, and Leon Bottou, “Learning methods for generic object recognition with invariance to pose and lighting,” in *Computer Vision and Pattern Recognition Conference*, 2004.

[8] Wootaek Lim, Daeyoung Jang, and Taejin Lee, “Speech emotion recognition using convolutional and recurrent neural networks,” in *Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, 2016.

[9] Abdul Malik Badshah, Jamil Ahmad, and Nasir Rahim, “Speech emotion recognition from spectrograms with deep convolutional neural network,” in *International Conference on Platform Technology and Service (PlatCon)*, 2017.

[10] Yafeng Niu, Dongsheng Zou, Yadong Niu, Zhongshi He, and Hua Tan, “A breakthrough in speech emotion recognition using deep retinal convolution neural networks,” *arXiv:1707.09917*, 2017.

[11] Nicholas Cummins, Shahin Amiriparian, Gerhard Hagerer, Anton Batliner, Stefan Steidl, and Björn W Schuller, “An image-based deep spectrum feature representation for the recognition of emotional speech,” in *Proceedings of the 2017 ACM on Multimedia Conference*. ACM, 2017, pp. 478–484.

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

[13] Ivan Tashev, Andrew Lovitt, and Alex Acero, “Unified framework for single channel speech enhancement,” in *Proceedings of IEEE Pacific Rim Conference on Communications, Computers and Signal Processing*, 2009.

[14] Stefan Steidl, Michael Levit, Anton Batliner, Elmar Noth, and Heinrich Niemann, “” of all things the measure is man” automatic classification of emotions and inter-labeler consistency [speech-based emotion recognition],” in *Acoustics, Speech, and Signal Processing, 2005. Proceedings.(ICASSP’05). IEEE International Conference on*. IEEE, 2005, vol. 1, pp. I–317.

[15] Haytham M Fayek, Margaret Lech, and Lawrence Cavedon, “Modeling subjectiveness in emotion recognition with deep neural networks: Ensembles vs soft labels,” in *Neural Networks (IJCNN), 2016 International Joint Conference on*. IEEE, 2016, pp. 566–570.

[16] R. Lotfian and C. Busso, “Formulating emotion perception as a probabilistic model with application to categorical emotion classification,” in *International Conference on Affective Computing and Intelligent Interaction (ACII 2017)*, San Antonio, TX, USA, October 2017.

[17] Diederik P. Kingma and Jimmy Ba, “Adam: A method for stochastic optimization,” *arXiv:1412.6980*, 2014.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, “Spatial pyramid pooling in deep convolutional networks for visual recognition,” *arXiv:1406.4729*, 2014.