Speech Emotion Recognition Using CNN

Zhengwei Huang†, Ming Dong‡, Qirong Mao†, Yongzhao Zhan†

† School of Computer Science and Communication Engineering
Jiangsu University
Zhenjiang, Jiangsu Province, 212013, China
{mao_qr,yzzhan}@mail.ujs.edu.cn

‡ Department of Computer Science
Wayne State University
Detroit, MI 48202, USA
mdong@cs.wayne.edu
zhengwei.hg@gmail.com

ABSTRACT

Deep learning systems, such as Convolutional Neural Networks (CNNs), can infer a hierarchical representation of input data that facilitates categorization. In this paper, we propose to learn affect-salient features for Speech Emotion Recognition (SER) using semi-CNN. The training of semi-CNN has two stages. In the first stage, unlabeled samples are used to learn candidate features by contractive convolutional neural network with reconstruction penalization. The candidate features, in the second step, are used as the input to semi-CNN to learn affect-salient, discriminative features using a novel objective function that encourages the feature saliency, orthogonality and discrimination. Our experiment results on benchmark datasets show that our approach leads to stable and robust recognition performance in complex scenes (e.g., with speaker and environment distortion), and outperforms several well-established SER features.

Categories and Subject Descriptors

I.5.2 [**Computing Methodologies**]: Pattern Recognition-Design Methodology[Feature evaluation and selection]

Keywords

Speech emotion recognition; Salient feature learning

1. INTRODUCTION

As an essential way of human emotional behavior understanding, in the past decades, Speech Emotion Recognition (SER) has attracted a great deal of attention in human-centered computing. In SER, one of the central research issues is how to extract discriminative, affect-salient features from speech signals. In this direction, a number of speech emotion features have been proposed in the literature, and they can be roughly classified into four categories [1]: 1) acoustic features, 2) linguistic features (words and discourse), 3) hybrid features that use both acoustic and linguistic information, and 4) context information (e.g., subject and gender). However, it is unclear if these hand-tuned feature sets can sufficiently and efficiently characterize the emotional content of speech. Moreover, their performance varies greatly in different scenarios. Finally, automatic extraction of some of these features can be difficult. For example, existing Automatic Speech Recognition (ASR) systems cannot reliably recognize all the verbal content of emotional speech. Extracting semantic discourse information is even more challenging, which, in many cases, has to be performed manually.

Thus, in SER, it is important to explore new strategies that can obtain the optimal feature set that is invariant to nuisance factors while maintaining discriminative with respect to the task of emotion recognition. Recently, in many situations where labeled data is limited or not available, deep learning systems, such as Convolutional Neural Networks (CNNs), are shown to have the capability to infer a hierarchical feature representation that facilitates categorization (e.g., image understanding [2] and ASR [3]). For SER, while many previous works about deep neural network focused on learning discriminative features [4, 5] from input, but did not consider to disentangle factors of variation.

In the paper, we propose to learn affect-salient features for SER using semi-supervised Convolutional Neural Network (semi-CNN), in which simple features are learned in the lower layers, and affect-salient, discriminative features are obtained in the higher layers. We propose a novel objective function to train semi-CNN by encouraging the feature saliency, orthogonality and discrimination. Our experiment results on several benchmark datasets show that our approach leads to stable and robust recognition performance in complex scenes (e.g. with speaker variation and noise), and outperforms several well-established SER features.

The major contribution of this paper is:

1. To our best knowledge, this is the first paper introducing feature learning to SER, in which the optimal feature set can be effectively and automatically learned by semi-CNN with a few labeled samples.

2. By introducing a novel objective function to train semi-CNN, we can extract affect-salient features for SER by disentangling emotions from other factors such as speakers and noise. Specifically, the candidate features from unsupervised learning are divided into two blocks, related to emotion and other remaining factors, respectively. The emotion-related features are discriminative and robust, leading to great performance improvement on SER.

MM'14, November 3–7, 2014, Orlando, FL, USA.

http://dx.doi.org/10.1145/2647868.2654984 .

The rest of the paper is organized as follows. Section 2 presents our semi-CNN-based feature learning algorithm in details. Section 3 describes SER benchmark datasets and reports our experimental results. Conclusion and future direction are discussed in Section 4.

2. LEARNING SALIENT FEATURES FOR SER

The architecture of semi-CNN is shown in Fig. 1, which has an input layer, one convolutional layer, one fully connected layer and a SVM classifier. We use spectrogram of the speech signal as the input of semi-CNN. Following the hierarchy of semi-CNN, the features learned at each layer become increasingly invariant to nuisance factors while maintaining affect-salient with respect to the goal of SER.

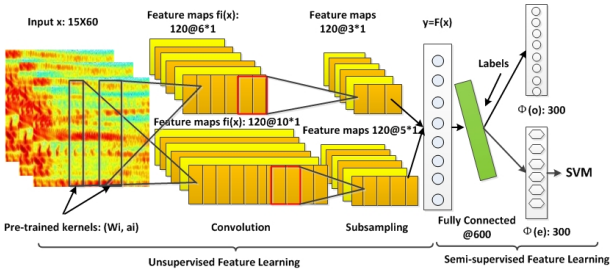

Figure 1: System Pipeline. The input is spectrogram with two different resolutions. Through unsupervised feature learning, our system obtains one long feature vector $y = F(x)$, based on which, the semi-supervised feature learning produces the affect-salient features $\phi^{(e)}$ and the nuisance features $\phi^{(o)}$. Finally, affect-salient features are fed to a linear SVM for SER.

Specifically, the input layer of semi-CNN in our system has 900 neurons to receive normalized spectrogram fragments of size 15×60. The convolutional layer consists of three consecutive operations: convolution with kernels, non-linear activation function, and pooling. The convolutional layer contains 120 kernels with two fixed sizes of 6×60 and 10×60, respectively. The kernel is pre-trained by unsupervised auto-encoder feature learning patch-wise. The size of the feature maps in the convolutional layer is 10×1 and 6×1, respectively.

The feature in our system is given as follows:

$$h(x) = sigmoid(Wx + \alpha), \tag{1}$$

where W is the weight matrix (kernel), $x \in D_x$ is a spectrogram fragment (D_x is the set of the input), and α is the bias. In feature pooling, one of the frequently used functions is down-sampling. We perform down-sampling using mean operation with a window size 2×1. The output layer has 600 fully connected neurons, each neuron corresponding to one feature, which are employed as the input of the fully-connected layer to disentangle factors of variation. Finally, the affect-salient feature block of the final feature vector is passed to a SVM classifier to determine the emotion class of the speech utterance.

2.1 Unsupervised Feature Learning

The auto-encoder has two parts: encoder and decoder [6]. The encoder learns a function h to map an input $x \in D_x$ to a feature vector $h(x) \in D_{x'}$, and the decoder reconstructs the input by minimizing the reconstruction error. Usually, the learning is done by training each layer individually and use the current layer codes to feed the next layer.

To capture the structure of the input spectrogram at different scales, we consider kernels with multiple sizes. Instead of generating kernels randomly, we pre-train kernels by sparse auto-encoder. Sparse auto-encoder is trained with patches extracted randomly at different locations, and the size of which matches that of the convolutional kernels being learned. Assuming that we have n different kernel sizes denoted as l_i-by-h_i, $(i = 1, 2, 3, ..., n)$, we can get the kernel (W^i, α^i) after we pre-train independently a sparse auto-encoder for each kernel size. In our system, we employ two kernel sizes: 6×60 and 10×60.

After we get the kernel (W^i, α^i), we compute the corresponding feature maps on the whole spectrogram by applying (W^i, α^i) to each l_i-by-h_i patch of the input spectrogram:

$$f^i(x) = s(conv(W^i, x) + \alpha^i), \tag{2}$$

where $conv()$ denotes the convolution operation. Then, we perform down-sampling using mean operation with window size $m \times n$ and get the feature map $F^i_j = mean_{k \in win_j}(f^i_k(x))$, where win_j denotes the j^{th} window. All the pooled features for kernel i are stacked into one feature vector $F^i(x)$, and then the final output feature vector y of the convolutional layer is given as:

$$y = [(F^1_1(x), ..., F^1_j(x), ..., F^1_N(x)), ..., (F^n_1(x), ..., F^n_j(x), ..., F^n_N(x))] \tag{3}$$

where N is the number of windows. y is used as the input of semi-supervised feature learning to disentangle emotion-salient factors from others.

2.2 Semi-supervised Feature Learning

While the data is encoded into a single feature vector y after unsupervised feature learning, an input is mapped into two distinct blocks of features: one $(\phi^{(e)}(y))$ that encodes affect-salient factors of its input, and one $(\phi^{(o)}(y))$ that encodes all other factors. Both feature blocks are trained to cooperate to reconstruct their common input y with a reconstruction loss function:

$$\hat{y} = g([\phi^{(e)}(y), \phi^{(o)}(y)]) = s(U^T(\phi^{(e)}(y) + \phi^{(o)}(y)) + \delta), \tag{4}$$

where U^T is the weight matrix in the fully-connected network, and δ_i is an offset to capture the mean value of y.

Given (x, z), the labeled training set with input spectrogram fragment x and emotion label z, the $\phi^{(e)}(y)$ block is also trained to predict the emotion label $z(y)$ when the label is available. The class prediction is given by the logistic regression of the discriminative block $\phi^{(e)}(y)$, which is learned by the sigmoid function over an affine transformation of the $\phi^{(e)}(y)$ block:

$$\hat{z}_i = s(A_i \phi^{(e)}(y) + \rho_i), \tag{5}$$

where the weight matrix A_i maps the $\phi^{(e)}(y)$ block to prediction for class i, and ρ_i is the class specific bias. The corresponding discriminant component of the overall loss function is:

$$L_{DISC}(z, \hat{z}) = -\sum_{i=1}^{C} z_i log(\hat{z}_i) + (1 - z_i) log(1 - \hat{z}_i), \tag{6}$$

where C is the number of emotion classes, and z and $\hat{z}$ are the ground truth and the logistic regression output, respectively.

Since a salient feature for SER is usually a sensitive feature for reconstruction error or discrimination error, the features responded strongly to this property tend to be more important. We measure the saliency for each input as the sum of its weight saliency. Specifically, the saliency for input i is defined as,

$$S_i = \sum_{k \in \varphi(i)} Saliency(w_k) = \frac{1}{2} \sum_{k \in \varphi(i)} \frac{\partial^2 MSE}{\partial w_k^2} w_k^2. \quad (7)$$

where $\varphi(i)$ is the set of weights connected input i and w_k is the k-th weight. In Equation 7, MSE denotes the Mean Squared Error. For features in $\phi^{(e)}(y)$, both the reconstruction and discrimination errors are taken into consideration, while for features in $\phi^{(o)}(y)$, only the reconstruction error is considered. The cost function is given as:

$$\mathcal{J}_{SAL} = -\frac{1}{2} \sum \frac{\partial^2 \parallel y - \hat{y} \parallel^2}{\partial w_k^2} w_k^2 - \frac{1}{2} \sum_{k \in \phi^{(e)}(y)} \frac{\partial^2 L_{DISC}(z, \hat{z})}{\partial w_k^2} w_k^2, \quad (8)$$

where the first term encourages salient features in $\phi^{(e)}(y)$ and $\phi^{(o)}(y)$ to reconstruction error, and the second term encourages affect-salient features in $\phi^{(o)}(y)$ to discriminative error. This objective is achieved through weight suppression during training.

To encourages $\phi^{(e)}(y)$ and $\phi^{(o)}(y)$ to present different directions of variation in the input y, we ask each sensitivity vector $\frac{\partial \phi_i^{(e)}(y)}{\partial y}$ of the i-th discriminant feature $\phi_i^{(e)}$ to prefer being orthogonal to every sensitivity vector $\frac{\partial \phi_j^{(o)}(y)}{\partial y}$ associated with the j-th non-discriminant feature $\phi_j^{(o)}$. This penalty component is denoted as $\mathcal{J}_{orth}$:

$$\mathcal{J}_{orth} = \sum_{i,j} \left(\frac{\partial F_i^{(e)}(y)}{\partial y} \cdot \frac{\partial F_j^{(o)}(y)}{\partial y}\right)^2. \quad (9)$$

Putting all the components of the loss function together we get:

$$\mathcal{L}(\theta) = \sum_{y=F(x)} L(y, g(h(y))) + \sum_{(x,z) \in S} L_{DISC}(z, \hat{z}) + \lambda_1 \mathcal{J}_{SAL} + \lambda_2 \mathcal{J}_{orth}. \quad (10)$$

The coefficients λ_1 and λ_2 weigh the contribution of the saliency penalty and the orthogonality penalty to the overall loss function, respectively, and they are empirically set as $\lambda_1 = 1$ and $\lambda_2 = 2$.

3. EXPERIMENTS & RESULTS

3.1 Datasets & Experimental Setup

Our affect-salient feature learning method was evaluated on four public emotional speech databases with different languages: Surrey Audio-Visual Expressed Emotion (SAVEE) Database [7], Berlin Emotional Database (Emo-DB) [8], Danish Emotional Speech database (DES) [9], Mandarin Emotional Speech database (MES) [10].

We first split the data into training dataset and testing dataset. In the unsupervised feature learning stage, we train kernels using one third of randomly selected data in the training dataset of each database. The labels are removed and not used in this stage. Later, we train CNN with the speech utterances in the training dataset. In our experiments, we first convert the time-domain signals into spectrograms. The spectrogram has a 20 ms window size with a 10 ms overlap. The spectrogram was further processed using Principal Component Analysis (PCA) whitening (with 60 components) to reduce its dimensionality.

We evaluate affect-salient features based on the classification accuracy and compare it with several other well-established feature representations: spectrogram representation ("RAW" features), acoustic features, Teager Energy Operator (TEO), and Local Invariant Features (LIF). We also compare features obtained in our system with and without affect-salient penalty (the third term in Equation 10) or orthogonality penalty (the fourth term in Equation 10), denoted as semi-CNN(no_s)) and semi-CNN (no_or) respectively). We determine the parameters C and σ for SVM by grid search. We also evaluate robustness of our method with respect to the common disturbing factors in SER, i.e., the speaker variation and environment distortion.

3.2 Performance Evaluation

3.2.1 Accuracy On Public Emotional Speech Databases

In this section, we report the recognition accuracy using 5-fold cross-validation based on features learned in different stages of semi-CNN: single non-convolutional one-layer kernels (K1, K2), LIF, affect-salient features $\phi^{(e)}$, and non-discriminative features $\phi^{(o)}$. In all cases, an SVM classifier is used for the emotion classification. These results in Figure 2 clearly show that each successive layer in semi-CNN helps to disentangle discriminative features, yielding better classification performance. Notice that on all the databases, the accuracy obtained by using affect-salient features is much higher than that obtained by the non-discriminative features, both are obtained in the last layer of semi-CNN.

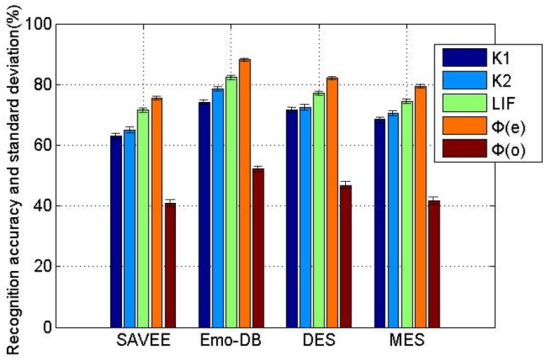

Figure 2: Average recognition accuracy and standard deviation with features learned in different stages of semi-CNN.

3.2.2 Robustness To Speaker Variation

In this section, we evaluate the features learned by semi-CNN by comparing it with other well-established feature representations with respect to speaker variance. Table 1 clearly show that all the learned features (i.e., LIF, semi-CNN(no_s), semi-CNN(no_or) and semi-CNN) outperform baseline features (RAW, TEO, and acoustic features), and semi-CNN achieves the highest accuracy in all cases (10% to 15% higher than the popular acoustic features). Notice that in the case of speaker-independent, that is, the speakers in

Table 1: SER accuracy and standard deviation with speaker variation (single speaker, speaker-dependent/independent).

speaker	database	RAW	TEO	acoustic	LIF	semi-CNN(no_or)	semi-CNN(no_s)	semi-CNN
single speaker	SAVEE	31.4±1.56	68.3±1.45	72.4±1.23	82.9±0.75	85.0±0.69	88.6±0.67	**89.7**±0.35
	Emo-DB	40.5±1.23	77.2±1.41	85.8±1.09	88.7±0.99	92.7±0.73	90.5±0.82	**93.7**±0.28
	DES	38.9±1.43	69.1±1.35	75.6±1.15	84.3±0.81	90.0±1.84	88.9±0.53	**90.8**±0.26
	MES	39.1±1.36	70.7±1.19	77.1±1.03	84.1±0.75	84.9±1.87	87.1±0.63	**90.2**±0.29
speaker-dep	SAVEE	29.4±1.64	58.8±1.47	65.0±1.21	69.4±0.79	86.7±3.75	73.3±0.66	**75.4**±0.42
	Emo-DB	37.2±1.45	66.4±1.48	72.3±1.15	80.5±0.81	74.7±1.7	86.4±0.63	**88.3**±0.31
	DES	34.6±1.42	65.3±1.41	71.1±1.25	76.8±0.93	78.9±0.57	79.9±0.58	**82.1**±0.45
	MES	35.5±1.53	60.6±1.37	67.8±1.38	72.5±0.98	77.2±2.10	76.3±0.61	**79.3**±0.50
speaker-indep	SAVEE	26.7±1.83	51.3±1.65	59.5±1.25	62.6±0.80	63.3±0.94	67.0±0.71	**73.6**±0.51
	Emo-DB	35.9±1.62	61.2±1.35	69.3±1.12	79.5±0.73	81.4±0.61	82.9±0.69	**85.2**±0.45
	DES	32.6±1.72	59.3±1.58	66.8±1.37	72.5±0.83	74.6±0.85	76.1±0.72	**79.9**±0.53
	MES	30.7±1.77	57.5±1.66	64.5±1.39	71.0±0.86	76.1±1.76	75.9±0.74	**78.3**±0.61

Table 2: SER accuracy and standard deviation with environmental distortion (noise and channel).

Environment Distortion	database	RAW	TEO	acoustic	LIF	semi-CNN(no_or)	semi-CNN(no_s)	semi-CNN
noise	SAVEE	16.2±2.02	38.5±1.93	49.5±1.96	53.6±0.88	55.7±0.89	53.8±0.91	**62.2**±0.56
	Emo-DB	28.3±1.98	47.6±1.99	55.6±1.72	68.2±0.83	74.6±0.71	76.8±0.63	**80.1**±0.53
	DES	26.7±1.96	40.6±2.01	53.3±1.68	67.8±0.91	75.1±0.98	73.2±0.80	**77.9**±0.55
	MES	28.6±1.92	43.2±1.65	50.2±1.73	64.1±1.09	69.3±1.06	70.8±0.75	**72.7**±0.70
channel	SAVEE	16.7±1.97	48.7±1.77	56.6±1.89	58.2±0.93	63.2±1.15	58.7±0.68	**69.0**±0.58
	Emo-DB	34.3±1.82	59.9±1.77	68.6±1.65	76.1±0.92	80.5±0.79	82.3±0.81	**84.5**±0.49
	DES	31.2±1.56	57.2±1.63	67.8±1.44	72.7±1.05	75.2±0.94	76.6±0.68	**77.9**±0.57
	MES	32.3±1.62	53.7±1.44	61.7±1.55	67.5±1.01	69.5±1.13	70.2±0.80	**74.8**±0.77

training set are mismatched with those in the testing set, semi-CNN has the smallest standard deviation on all the databases, which indicates that semi-CNN is able to learn a feature representation that is salient to emotion while being robust to speaker variations.

3.2.3 Robustness To Environment Distortion

Next, we report the results obtained with respect to distortions caused by the environment. In Table 2, noise means that the utterances in the test set are corrupted by the Gaussian noise of 20db. Channel means that the sampling frequency of the utterances in the test set is modified as 16 kHz, which is different from that of the samples in the training set. The accuracy of clean speech is the same as those of the second row in Table 1. Clearly, among the five methods semi-CNN achieves the highest and the most stable accuracy (the smallest standard deviation) for all the cases. In addition, feature learning methods all outperform baseline features.

4. CONCLUSION

In this paper we introduce feature learning in SER and propose to learn emotion-salient features using semi-CNN. Semi-CNN is trained in two stages: first unsupervised and then semi-supervised. In semi-supervised training, we propose a novel object function for semi-CNN that encourages the feature saliency, orthogonality, and discrimination for SER. Experiments on four public emotional speech databases show superior performance of our features with respect to speaker variation and environment distortion when compared with several well-established feature representations.

5. ACKNOWLEDGMENTS

This work was partially supported by the National Nature Science Foundation of China (No. 61272211 and No. 61170126), and by the Six Talent Peaks Foundation of Jiangsu Province (No.DZXX-027).

6. REFERENCES

[1] I. Luengo, E. Navas, and I. HernÃąez, "Feature analysis and evaluation for automatic emotion identification in speech," *IEEE Trans. Multimedia*, vol. 12, no. 6, pp. 490–501, 2010.

[2] Z. Han and J. J. et. al., "Face image super-resolution via nearest feature line," in *ACM Multimedia*, Nara, Japan, Oct. 2012, pp. 769–772.

[3] D. Yu, M. L. Seltzer, J. Li, J. T. Huang, and S. Frank, "Feature learning in deep neural networks - studies on speech recognition tasks," in *ICLR*, Scottsdale, Arizona, USA, May 2013.

[4] Y. Kim and H. L. et. al., "Deep learning for robust feature generation in audio-visual emotion recognition," Vancouver, British Columbia, Canada, 2013.

[5] D. Le and E. M. Provost, "Emotion recognition from spontaneous speech using hidden markov models with deep belief networks," Olomouc, Czech Republic, 2013.

[6] J. Lu, J. Hu, X. Zhou, and Y. Shang, "Activity-based person identification using sparse coding and discriminative metric learning," in *ACM Multimedia*, Nara, Japan, Oct. 2012, pp. 1061–1064.

[7] S. Haq, P. Jackson, and J. Edge, "Speaker-dependent audio-visual emotion recognition," in *AVSP*, Norwich, UK, Sept. 2009, pp. 53–58.

[8] F. Burkhardt, A. Paeschke, M. Rolfes, and W. Sendlmeier, "A database of german emotional speech," in *Interspeech*, Lissabon, Portugal, Sept. 2005, pp. 1517–1520.

[9] I. Engberg and A. Hansen. (1996) Documentation of the Danish emotional speech database DES. [Online]. Available: http://cpk.auc.dk/tb/speech/emotions/

[10] L. Fu, X. Mao, and L. Chen, "Speaker independent emotion recognition based on SVM/HMMs fusion system," in *ICALIP*, Shanghai, China, July 2008, pp. 61–65.