Deep Neural Networks for Anger Detection from Real Life Speech Data

Jun Deng[1], Florian Eyben[1], Björn Schuller[1], Felix Burkhardt[2]
[1]*audEERING GmbH, Gilching, Germany*
[2]*Deutsche Telekom AG, Telekom Innovation Laboratories, Berlin, Germany*
Email {jdeng, fe, bs}@audeering.com, felix.burkhardt@telekom.de

***Abstract*—There has been a lot of previous work on deep neural networks for automatic speech recognition, however, little emphasis has been placed on an investigation of effective deep learning architectures for anger detection from speech. In this paper, inspired by the state-of-the-art deep learning algorithms, we propose a variant of Deep Long Short-Term Memory (*LSTM*) Recurrent Neural Networks (*RNNs*), Convolution Neural Networks (*CNNs*) with 3×3 kernels, and LSTM RNNs combined with CNNs, in conjunction with log-mel filter bank features and brute forced low-level-descriptors from the standardised ComParE set for speech anger detection. We extensively evaluate the deep networks on a big real-life speech corpus of 26 970 utterances with utterance-level labels collected from a German voice portal, finding that our proposed neural networks significantly outperform traditional modelling algorithms for speech anger detection.**

1. Introduction

No humans are ever non emotional. We speak emotional, perceive others emotions and communicate emotional. Despite this, contemporary human machine dialog systems always speak with the same unmoved voice and ignore customers irony, anger or elation. This is partly due to insufficient technological performance with respect to recognition and simulation, and partly to a gap with respect to the necessary artificial intelligence to support emotional behavior [1].

In the context of Interactive Voice Response (*IVR*) portals it can be helpful to detect potential problems that arise from an unsatisfactory course of interaction with the system to help the customer by either offering the assistance of a human operator or trying to react with appropriate dialog strategies. An important decision criterion for such changes in the call flow is the automatic detection of anger from the caller's voice that can be monitored during the entire dialog. A respective technology module can be introduced in the IVR system running in parallel to the Automatic Speech Recognition (*ASR*) component.

In [2] we described an investigation to automatically annotate anger in tuning data recordings from a real life automated customer help voice portal. As there is no objective measure for anger, we labelled the data with three labellers, two women and one man. We compared between Gaussian Mixture Models (*GMMs*) and Support Vector Machines (*SVMs*), the latter gave better results. Now, about ten years later, we wondered about the possibility to gain in recognition accuracy based on modern machine classification technologies like deep neural nets.

Despite that deep neural networks have been dominating the current ASR research and industry areas [3], [4], [5], [6], there has been a little related work on investigating them for speech anger detection [7], [8], [9], [10], [11]. In [7], the authors used convolutoinal neural networks to learn salient features from data for speech emotion recognition. [10] proposed to leverage a deep convolutional recurrent network to learn from the raw speech waveform instead of hand-crafted acoustic features, leading to the performance as competitive as other traditional classifiers.

Motivated by the increasing development of deep learning architectures, this paper first focuses on exploring the context information from sequential speech data for detecting an angry state from speech. To this end, we make use of the state-of-the-art Long Short-Term Memory (*LSTM*) Recurrent Neural Networks (RNNs) including Bidirectional LSTM (*BLSTM*) in conjunction with Low-Level Descriptors (LLDs) such as the log-mel filter bank features. Moreover, inspired by the success of the VGG convolutional neural network originally proposed for image classification, we adapt this network for speech anger detection, resulting in a deep architecture with 7 hidden convolutional layers and 1 fully connected layer. To the best of our knowledge, we are the first to publish results of VGG-inspired networks applied to speech anger detection. Finally, we propose a deep network using both convolutional networks and LSTM RNNs, in an attempt to learn salient representations by the bottom convolutional layers and leverage the sequential modelling by the following LSTM layers.

The rest of the paper is structured as follows: Section 2 presents the details of the speech corpus used for the speech anger detection experiments; In Section 3, we discuss the algorithmic details of the proposed deep neural networks. We further show the LLD features and network training in Section 4. Finally, we present the experimental results in Section 5 before making the conclusions and pointing out the future directions in Section 6.

2. Selected Data

The data used for the following experiments was collected from a German voice portal where customers report problems with their phone connection and get preselected by an automated voice dialogue before being connected to an agent [2]. We used annotated anger in tuning data recordings from a real life automated customer help voice portal. The recordings were done during ten working days distributed widely in 2007. The data amounted to 26970 utterances in 4683 dialogues, i. e., about 5.8 utterances per dialogue. Most of the dialogues are very short: more than 50 % contain at most three utterances.

We labelled the data with three labellers, two women and one man. In order to achieve a consistent rating behaviour, the labellers received written label instructions and took part in a common session where some examples were discussed. For each utterance, the labellers had the choice to assign an anger value between 1 and 5 (1:not angry, 2: not sure, 3: slightly angry, 4: clear anger, 5: clear rage), or mark the utterance as "non-applicable" (garbage). Garbage utterances included a multitude of utterances that could not be classified for some reason, e. g., Dual-Tone Multi-Frequency (*DTMF*) tones, coughing, baby crying or lorries passing by. We unified the ratings by using majority voting and mapping them to two and three classes, forming two anger detection tasks, which will be considered for the experiments(i. e., Section 5).

The two classification tasks, including three-class and two-class tasks, are defined on the base of the different use of discrete ratings assigned by the three labellers. For the three-class task, a final label l of an utterance is computed as follows:

$$l = \begin{cases} 1 & \text{if } 1 \leq l_{\text{mean}} \leq 2, \\ 2 & \text{if } l_{\text{mean}} > 2, \\ 3 & \text{otherwise,} \end{cases} \tag{1}$$

where l_{mean} represents the average value of the three ratings.

The two-class task is defined in a way that [2] did. A label l is firstly assigned as follows:

$$l = \begin{cases} 1 & \text{if } 0.5 \leq l_{\text{mean}} < 1.5, \\ \text{unsure} & \text{if } 1.5 \leq l_{\text{mean}} < 2.5, \\ 2 & \text{if } l_{\text{mean}} \geq 2.5, \\ \text{garbage} & \text{otherwise.} \end{cases} \tag{2}$$

Then, we exclude all the utterances assigned as "unsure" and "garbage" to form the binary classification task.

3. Methods

3.1. RNNs

An RNN is a powerful type of deep neural networks, which has been widely used to resolve difficult machine learning problems that involve sequence inputs. RNNs can propagate information from a previous time step to the current time step as equipped with a mechanism of recurrent feedback. Consequently, RNNs learn context information across sequences of inputs rather than isolated patterns.

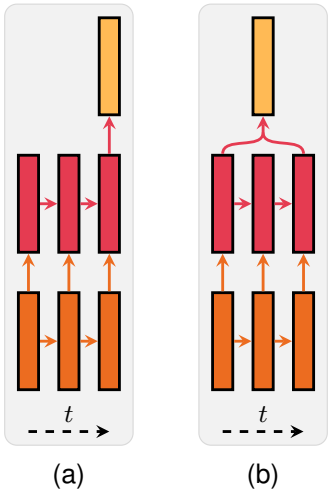

Figure 1. Illustration of two different methods to summarise a sequence in RNNs and produce a fixed-length feature vector used as input for later processing for anger detection. Figure 1a: Last Pooling chooses the last time frame as input to the objective function; Figure 1b: Mean pooling computes the average over the outputs of the last RNN layer.

Mathematically, for an RNN network with one hidden layer, the hidden layer output vector $\mathbf{h}_t$ at time t is a function of the input vector $\mathbf{x}_t$ at time t, and the hidden layer output $\mathbf{h}_{t-1}$ at the previous time step $t-1$:

$$\mathbf{h}_t = f(\mathbf{x}_t, \mathbf{h}_{t-1}), \tag{3}$$

where f represents a non-linear activation function (i. e., the sigmoid logistic function or the tanh function).

RNNs are prone to suffer from the so-called vanishing gradient problem when learning from long sequences. One effective solution is to use LSTM architectures, which are augmented by recurrent gates. The LSTM model used in a stacked form has been shown to beneficially exploit long contextual information, achieving state-of-the-art results on the magnitude of diverse problems from automatic speech recognition [4], voice activity detection [12], to computational paralinguistics [13], [14].

The LSTM RNN model is basically made of one self-connected linear memory cell $\mathbf{c}$ and three multiplicative gates containing an input gate $\mathbf{i}$, a forget gate $\mathbf{f}$, and an output gate $\mathbf{o}$ [15]. Given an input $\mathbf{x}_t$ at the time step t and the hidden output $\mathbf{h}_{t-1}$ at the previous time step, the corresponding activations of the memory cell and the three internal gates are computed as follows:

$$\mathbf{i}_t = \text{sigm}(\mathbf{W}_{ix}\mathbf{x}_t + \mathbf{W}_{ih}\mathbf{h}_{t-1} + \mathbf{b}_i), \quad (4)$$
$$\mathbf{f}_t = \text{sigm}(\mathbf{W}_{fx}\mathbf{x}_t + \mathbf{W}_{fh}\mathbf{h}_{t-1} + \mathbf{b}_f), \quad (5)$$
$$\mathbf{o}_t = \text{sigm}(\mathbf{W}_{ox}\mathbf{x}_t + \mathbf{W}_{oh}\mathbf{h}_{t-1} + \mathbf{b}_o), \quad (6)$$
$$\mathbf{g}_t = \tanh(\mathbf{W}_{gx}\mathbf{x}_t + \mathbf{W}_{gh}\mathbf{h}_{t-1} + \mathbf{b}_g), \quad (7)$$
$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t, \quad (8)$$
$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t), \quad (9)$$

where $\mathbf{W}$ denotes a weight matrix of the mutual connections; $\mathbf{b}$ indicates the bias parameter; $\mathbf{h}_t$ is the output of the hidden block; and $\odot$ represents the element-wise multiplication operation.

The focus of this paper is placed on a speech anger problem, where each utterance is assigned one label indicating the anger level. Such a problem can be viewed as the "many to one" sequence learning problem. After learning LLDs features, therefore, the proposed LSTM anger detection network summarizes them over an utterance, which eventually returns an utterance-level feature vector for further processing. Here, to achieve an utterance-level feature vector, we apply two different techniques on top of the last LSTM RNN layer, illustrated in Figure 1. The first one selects the last frame of the sequence to feed forwards to further processing, which will be referred to as *Last Frame* in the following. In contrast, the second one applies a time pooling over the time frames and produces a fixed-length vector. In this paper, we adopt temporal mean pooling to summarise an utterance, which is referred to as *Mean Pooling*.

3.2. Bidirectional LSTM RNNs

Unlike conventional RNNs that are only able to exploit previous context, Bidirectional LSTM (*BLSTM*) RNNs [4] process sequential data in both previous and future context directions with two separate hidden layers (i. e., a forward layer and a backward layer). In this way, BLSTM RNNs are capable of finding and exploiting the past and future context in an input sequence. In analogy with the proposed LSTM architecture for speech anger detection (cf. Section 3.1), we apply temporal mean pooling over BLSTM hidden outputs, which concatenate outputs from both forward and back layers at each time step.

3.3. Convolutional Neural Networks

The deep CNN we describe here is deeply rooted in the work of the particular convolutional net, widely called *VGG*, which was originally proposed for image classification in the Imagenet 2014 competition [16]. Recently, the VGG-inspired networks have been successfully adapted to ASR [5], [17] and large-scale audio classification [18]. The fundamental idea of the VGG net is to use small 3×3 convolutional kernels with Rectified Linear Unit (*ReLU*) non-linear functions without pooling between these layers. We apply this principle to constructing a VGG-inspired CNN network with 7 hidden convolutional layers and 1 fully connected layer for speech anger detection. As in [16], [17], the full configuration of the deep CNN is shown as follows: conv(1, 64), conv(64, 64), pool(2, 2), conv(64, 128), conv(128, 128), pool(2, 2), conv(128, 256), conv(256, 256), pool(2, 2), conv(256, 256), softmax. For the sake of simplicity, here, we ignore the ReLU layers following each convolutional layer. Further, the convolutional layers are written as conv(input feature maps, output feature maps) where each kernel size is set to be 3×3. The pooling layers are written as pool(time, frequency) where max pooling with a kernel of size 2×2 and stride 2 is applied.

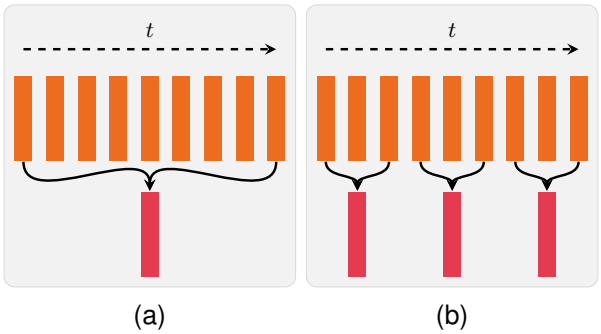

Figure 2. Illustration of mean pooling and temporal k-mean pooling, which are used for length normalisation in order to provide fixed length sequences for a later deep neural network classifier. Figure 2a depicts mean pooling over audio frames while Figure 2b depicts the temporal 3-mean pooling.

TABLE 1. COMPARE ACOUSTIC FEATURE SET: 65 LOW-LEVEL DESCRIPTORS (LLD).

4 energy related LLD	**Group**
RMS energy, zero-crossing rate	Prosodic
Sum of auditory spectrum (loudness)	Prosodic
Sum of RASTA-filtered auditory spectrum	Prosodic
55 spectral LLD	**Group**
MFCC 1–14	Cepstral
Psychoacoustic sharpness, harmonicity	Spectral
RASTA-filt. aud. spect. bds. 1–26 (0–8 kHz)	Spectral
Spectral energy 250–650 Hz, 1 k–4 kHz	spectral
Spectral flux, centroid, entropy, slope	Spectral
Spectral Roll-Off Pt. 0.25, 0.5, 0.75, 0.9	Spectral
Spectral variance, skewness, kurtosis	spectral
6 voicing related LLD	**Group**
F_0 (SHS and Viterbi smoothing)	Prosodic
Prob. of voicing	Voice qual.
log. HNR, jitter (local and δ), shimmer (local)	Voice qual.

3.4. LSTM RNNs Combined with CNNs

We finally investigate the combination of CNNs and RNNs since CNNs have translation invariance characteristics and RNNs are good at learning temporal information. Such a network, which is usually made of convolution layers at the bottom layers and multiple RNN layers on the top of the convolution layers, is currently being attracted to the speech processing community [6]. For speech anger detection, we found that using a network with two convolutional layers and two LSTM layers works best on our preliminary experiments, and chose 200 hidden units per LSTM layer.

TABLE 2. HYPER-PARAMETERS USED FOR THE PROPOSED DEEP NETWORKS IN THE EXPERIMENTS.

Net Type	# Hidden Layers	# Hidden Units	Learning Rate	k-Mean Pooling
LSTM (Last Frame)	4	200	$1e-4$	89
LSTM (Mean Pooling)	4	200	$1e-4$	89
BLSTM (Mean Pooling)	4	200	$1e-4$	89
VGG-Inspired CNN	7	–	$2e-5$	89
CNN + LSTM	4 (2 $\times$ Conv. + 2 $\times$ LSTM)	200	$1e-4$	300

Note that, unlike the VGG-inspired CNN networks with a small kernel size as mentioned above, the two convolutional layers apply a bigger stride and wider context to speed up training as fewer time steps to model a given utterance, facilitating the following LSTM layers. Specifically, the two convolutional layers were configured with a kernel of size 41×11 and stride 2, and a kernel of size 21×11 and stride 2, respectively, where the ReLU layer is used after each convolutional layer.

3.5. Temporal k-mean pooling

In the preliminary experiments, we observed that the vanishing gradient problem for the deep neural networks arose, sharply degrading the performance. We suspected, it is due to the highly diverse length of each utterance: the average, maximum, and minimum utterance duration of the selected corpus are 2.8, 62.4, and 0.36 seconds, respectively; the standard deviation is big (2.2). In order to solve this problem by providing reasonable length sequences as input to RNNs or CNNs, the temporal k-mean pooling, which was successfully adopted for video emotion recognition [14], was performed on the raw LLD data before these data are fed into deep neural networks.

The temporal k-mean pooling, shown in Figure 2, is applied to the frame-level features (e. g., LLD features) where the whole frame sequence is divided into k sub-sequences in a temporal manner and a mean pooling step is used over frames in each sub-sequence. Note that, temporal k-mean pooling corresponds to global mean pooling when k is equal to 1.

4. Experimental Setup

4.1. LLD Features

We investigate two common types of LLD features in the following experiments. The first one is the log-mel filter bank features, referred to as *Log-Mel*, which are popularised in various speech processing systems using deep neural networks. The log-mel filter bank features include 40 log-mel filter bank features and energy and their first deltas, ending up as a frame-level feature vector of 82 dimensions. The other one is brute forced LLDs from the standardised ComParE set, which was introduced for speech emotion recognition [19]. Such brute forced features, referred to as *ComParE*, contain 130 different types of acoustic features including Mel-frequency cepstral coefficients (*MFCCs*), Root Mean Square (*RMS*) energy, zero-cross rate, etc. The full details of the ComParE LLDs are given in Table 1. All the LLD features are extracted by the open source tool openSMILE [20] with a frame window of 0.025 seconds and a frame step of 0.01 seconds.

4.2. Network Training

Mean subtraction and standard deviation division were performed within each utterance to ensure it is of zero mean and unit variance. Furthermore, in order to mitigate the class imbalance, we apply downsampling technique for the training data, which downsamples the over-represented classes such that each class has similar amount of data.

We evaluate the performance by Unweighted Average Recall (*UAR*) using a five-fold cross validation strategy on the selected data (cf. Section 2). All the hyper-parameters such as the number of hidden units, the number of hidden layers, the learning rate, and the mini-batch size, are tuned when the maximum validation UAR is reached. Table 2 present the hyper-parameters selected in the five-fold cross validation experiment. In addition, mini-batch training [21] with a batch size of 50 utterances with the Adam optimization method [22] was adopted. All the experiments are implemented by TensorFlow [23].

5. Results

Table 3 presents the performance achieved by our proposed deep neural networks using five-fold cross validation in terms of UAR for the two-class and three-class tasks. It can be easily seen from Table 3 that, for the two-class problem, the proposed networks outperform the GMM-based and SVM-based systems with a reduced feature set [2] by a noticeable margin. It is worth noting that the BLSTM network with mean pooling always obtains the maximum UARs of 79.4 % and 80.1 % for the log-mel filter bank and ComParE feature sets, demonstrating the great benefit of exploring the bidirectional context information for speech anger detection. Moreover, in a direct comparison to the modern SVM model with the ComParE feature set, which is a representative method for various computational paralinguistics tasks, we find that the deep neural networks slightly perform better than it.

As the three-class task was not investigated in [2], we compare our methods to the SVM model with the original ComParE feature set. It can be observed from Table 3 that

TABLE 3. UARs [%] ACHIEVED BY DEEP NEURAL NETWORKS USING THE FIVE-FOLD CROSS VALIDATION FOR THE TWO-CLASS AND THREE-CLASS TASKS ON THE ANGER DETECTION DATABASE.

UAR [%]	2-class	3-class
Supra-features:		
GMM [2]	61.0	–
SVM [2]	69.0	–
SVM (ComParE)	78.1	71.0
LLD (Log-Mel):		
LSTM (Last Frame)	78.9	70.2
LSTM (Mean Pooling)	78.5	70.4
BLSTM (Mean Pooling)	**79.4**	70.8
VGG-Inspired CNN	66.7	67.4
CNN + LSTM	78.6	**71.0**
LLD (ComParE):		
LSTM (Last Frame)	78.8	68.9
LSTM (Mean Pooling)	78.9	71.8
BLSTM (Mean Pooling)	**80.1**	**72.2**
VGG-Inspired CNN	75.6	67.3
CNN + LSTM	79.5	70.8

the proposed deep neural networks perform above the SVM model. Again, the BLSTM with mean pooling gives the best performance. In addition, the combination of CNNs and LSTM RNNs achieves the best results when using the log-mel filter bank features.

It is worth noting that the VGG-Inspired CNN (cf. Section 3.3) generally does not perform as well as other deep neural networks containing LSTM RNNs. On the one hand, we attribute it to the lack of adequate training data for the CNN with 7 hidden convolutional layers. On the other hand, this result also provides weak support to the notion that leveraging the temporal information with RNNs units can help sequential learning tasks like speech anger detection.

6. Conclusions

In this paper, we have extensively investigated three state-of-the-art deep neural networks for anger detection from speech. The experiments on a big real speech corpus show that the deep neural networks such as LSTM RNNs and CNNs with low-level descriptors features significantly surpass the earlier SVM-based recognition framework. In comparison to the results in our original investigation from 2009 [2], the results gained about ten percent in average recall (80.1 % UAR with a bidirectional LSTM network and mean pooling on the ComParE feature set vs 69.0 % UAR with SVMs on a reduced feature set). Furthermore, we found that LSTM RNNs play a vital role in a feasible deep network for speech anger detection.

To further improve the performance, one potential direction is to use an adaptive pooling that is capable of focusing on important regions of an input speech signal. Additionally, we will investigate the adaptation of Inception [24] and ResNet [25] with LSTM RNNs to various computational paralinguistics tasks.

References

[1] F. Burkhardt, M. V. Ballegooy, K. Engelbrecht, T. Polzehl, and J. Stegmann, "Emotion detection in dialog systems: Applications, strategies and challenges," in *Affective Computing and Intelligent Interaction, Third International Conference and Workshops, ACII 2009, Amsterdam, The Netherlands, September 10-12, 2009, Proceedings*, 2009, pp. 1–6.

[2] F. Burkhardt, T. Polzehl, J. Stegmann, F. Metze, and R. Huber, "Detecting real life anger," in *Proc. Of ICASSP*, Taipei, Taiwan, 2009, pp. 4761–4764.

[3] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *Signal Processing Magazine, IEEE*, vol. 29, no. 6, pp. 82–97, 2012.

[4] A. Graves, N. Jaitly, and A. Mohamed, "Hybrid speech recognition with deep bidirectional LSTM," in *2013 IEEE Workshop on Automatic Speech Recognition and Understanding*, Olomouc, Czech Republic, 2013, pp. 273–278.

[5] G. Saon, T. Sercu, S. J. Rennie, and H. J. Kuo, "The IBM 2016 english conversational telephone speech recognition system," in *Proc. of INTERSPEECH*, CA, USA, 2016, pp. 7–11.

[6] D. Amodei, R. Anubhai, E. Battenberg, C. Case, J. Casper, B. Catanzaro, J. Chen, M. Chrzanowski, A. Coates, G. Diamos, E. Elsen, J. Engel, L. Fan, C. Fougner, A. Y. Hannun, B. Jun, T. Han, P. LeGresley, X. Li, L. Lin, S. Narang, A. Y. Ng, S. Ozair, R. Prenger, S. Qian, J. Raiman, S. Satheesh, D. Seetapun, S. Sengupta, C. Wang, Y. Wang, Z. Wang, B. Xiao, Y. Xie, D. Yogatama, J. Zhan, and Z. Zhu, "Deep speech 2 : End-to-end speech recognition in english and mandarin," in *Proc. ICML*, NY, USA, 2016, pp. 173–182.

[7] Q. Mao, M. Dong, Z. Huang, and Y. Zhan, "Learning salient features for speech emotion recognition using convolutional neural networks," *Multimedia, IEEE Transactions on*, vol. 16, no. 8, pp. 2203–2213, 2014.

[8] B. Schuller, "Deep Learning our Everyday Emotions – A Short Overview," in *Advances in Neural Networks: Computational and Theoretical Issues Emotional Expressions and Daily Cognitive Functions*, ser. Smart Innovation Systems and Technologies, S. Bassis, A. Esposito, and F. C. Morabito, Eds. Berlin Heidelberg: Springer, 2015, vol. 37, pp. 339–346.

[9] J. Deng, Z. Zhang, F. Eyben, and B. Schuller, "Autoencoder-based unsupervised domain adaptation for speech emotion recognition," *Signal Processing Letters, IEEE*, vol. 21, no. 9, pp. 1068–1072, 2014.

[10] G. Trigeorgis, F. Ringeval, R. Brückner, E. Marchi, M. Nicolaou, B. Schuller, and S. Zafeiriou, "Adieu features? end-to-end speech emotion recognition using a deep convolutional recurrent network," in *Proc. of ICASSP*, Shanghai, China, 2016, 5 pages.

[11] G. Keren and B. Schuller, "Convolutional RNN: an enhanced model for extracting features from sequential data," in *Proc. of IJCNN*, Vancouver, Canada, 2016, pp. 3412–3419.

[12] F. Eyben, F. Weninger, S. Squartini, and B. Schuller, "Real-life voice activity detection with LSTM recurrent neural networks and an application to hollywood movies," in *Proc. of ICASSP*, Vancouver, Canada, 2013, pp. 483–487.

[13] M. Wöllmer, M. Kaiser, F. Eyben, B. Schuller, and G. Rigoll, "LSTM-modeling of continuous emotions in an audiovisual affect recognition framework," *Image and Vision Computing, Special Issue on Affect Analysis in Continuous Input*, vol. 31, no. 2, pp. 153–163, February 2013.

[14] J. Deng, N. Cummins, J. Han, X. Xu, Z. Ren, V. Pandit, Z. Zhang, and B. Schuller, "The University of Passau Open Emotion Recognition System for the Multimodal Emotion Challenge," in *Proc. of CCPR*. Chengdu, P. R. China: Springer, November 2016, pp. 652–666.

[15] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[16] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[17] T. Sercu, C. Puhrsch, B. Kingsbury, and Y. LeCun, "Very deep multilingual convolutional neural networks for LVCSR," in *Proc. of ICASSP*, Shanghai, China, 2016, pp. 4955–4959.

[18] S. Hershey, S. Chaudhuri, D. P. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold *et al.*, "CNN architectures for large-scale audio classification," in *Proc. Of ICASSP*, New Orleans, USA, 2017, pp. 131–135.

[19] B. Schuller, S. Steidl, A. Batliner, A. Vinciarelli, K. Scherer, F. Ringeval, M. Chetouani *et al.*, "The INTERSPEECH 2013 Computational Paralinguistics Challenge: Social Signals, Conflict, Emotion, Autism," in *Proc. of INTERSPEECH*. Lyon, France: ISCA, 2013, pp. 148–152.

[20] F. Eyben, F. Weninger, F. Gross, and B. Schuller, "Recent developments in openSMILE, the munich open-source multimedia feature extractor," in *Proc. of ACM Multimedia 2013*. Barcelona, Spain: ACM, 2013, pp. 835–838.

[21] G. E. Hinton, *A Practical Guide to Training Restricted Boltzmann Machines*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 599–619.

[22] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. ICLR*, San Diego, USA, 2015.

[23] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: A system for large-scale machine learning," in *Proc. of OSDI*, Berkeley, CA, USA, 2016, pp. 265–283.

[24] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. of CVPR*, Las Vegas, NV, USA, 2016, pp. 2818–2826.

[25] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. of CVPR*, Las Vegas, NV, USA, 2016, pp. 770–778.