Classification of Social Signals Using Deep LSTM-based Recurrent Neural Networks

Himanshu Joshi*, Ananya Verma† and Amrita Mishra*
*Dept. of ECE, DSPM International Institute of Information Technology Naya Raipur, India
{himanshuj16101, amrita}@iiitnr.edu.in
†Dept. of CSE, DSPM International Institute of Information Technology Naya Raipur, India
ananyav16100@iiitnr.edu.in

***Abstract*—Non-linguistic speech cues aid expression of various emotions in human communication. In this work, we demonstrate the application of deep long short-term memory (LSTM) recurrent neural networks for frame-wise detection and classification of laughter and filler vocalizations in speech data. Further, we propose a novel approach to perform classification by incorporating cluster information as an additional feature wherein the clusters in the dataset are extracted via a k-means clustering algorithm. Extensive simulation results demonstrate that the proposed approach achieves significant improvement over the conventional LSTM-based classification methods. Also, the performance of deep LSTM models obtained by stacking LSTMs, is studied. Lastly, for classification of the temporally correlated speech data considered in this work, a comparison with popular machine learning-based techniques validates the superiority of the proposed LSTM-based scheme.**

I. Introduction and Background Work

Human communication, comprising of both verbal and non-verbal components, plays an integral role in social correspondence. Interestingly, more than 60% of human communication is conveyed via various non-linguistic cues namely facial expressions, voice modulation, gesture, eye contact etc., thereby placing more emphasis on the manner of delivery than the words [1]. In order to better interpret the emotions associated with human behaviour, non-verbal cues such as laughter, fillers, pauses etc. play an extremely critical role. Thus, *computational paralinguistics*, the paradigm of computer-based analyses of non-verbal cues, has become an active area of research in the recent years [2], [3].

Towards this end, several prior works have focused on the detection and classification of laughter, fillers and garbage in human speech. The initial work in [4] employed the classical hidden Markov model (HMM) for detection of the non-verbal sounds in television broadcasts. Next, the authors in [5] incorporated HMM as well as other statistical tools to develop an automatic and segmented approach for segregation of audio recordings into four categories of intervals: laughter, filler, speech and silence. The work in [6] employed Mel-frequency cepstrum coefficients (MFCC) along with other features for classification of laughter and filler pauses via Gaussian mixture models (GMMs). An advancement in [7] combined the benefits of both GMM and support vector machines (SVMs) for garbage, laughter and filler classification based on the GMM scores. However, the major disadvantage associated with the HMM and GMM-based classification methods is that these schemes fail to leverage the inherent temporal correlation associated with the speech data owing to their conditional independence assumption between the different operating modules.

In this regard, Gupta et al. in [8] used probability factor for event detection and MFCCs for feature extraction, trained on models such as HMM, SVM and deep neural network (DNN), with two hidden layers and an output layer. The output layer comprised of three nodes with a sigmoid activation function representing each class i.e. laughter, filler or garbage. DNNs and convolutional neural networks were used in [9] on conversational telephony speech and UT-Opinion corpus based on a simple feature set which combines the Mel-filter bank energies and pitch information. The works in [10], [11] employed deep long short-term memory (LSTM) and bidirectional LSTM frameworks for classification of human speech.

Thus, motivated by the success of DNNs in the field of automatic speech recognition (ASR), the present work considers the Interspeech 2013 Computational Paralinguistics Social Signals Sub-Challenge dataset [12] to develop a novel approach for frame-wise classification of audio recordings into laughter and filler. The key contributions of this paper are summarized as follows.

- Motivated by the improved performance of recurrent neural networks, this work effectively employs deep LSTM networks for detection and classification of laughter and filler vocalizations in speech data.
- In conjunction to employing MFCC for feature extraction, this work proposes a novel paradigm of including cluster as a feature for performing the classification task. Towards this end, the k-means clustering algorithm is also developed.
- Extensive numerical experiments are conducted to demonstrate that the proposed approach achieves significant improvement over conventional LSTM-based classification methods. Some interesting insights regarding the accuracy of deep LSTM networks with and without clustering as a feature are also discussed.
- Lastly, an exhaustive comparison with other machine learning techniques validates the superiority of the proposed LSTM-based scheme.

978-1-7281-8895-9/20/$31.00

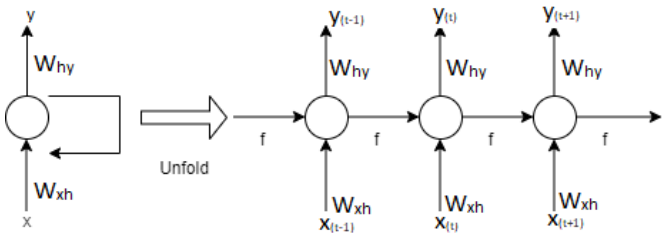

Fig. 1. Recurrent Neural Network Architecture

The content of the paper is organized as follows. Section II presents a brief introduction about recurrent neural networks and long short-term memory architecture employed in this work. Section III discusses the database and feature set for the classification problem and also describes the novel approach for considering cluster as a feature. The k-means clustering algorithm is discussed herein. The experiment, results and ensuing discussions are presented in section IV followed by the concluding remarks in section V.

Notation: The following notation has been used in the paper. The notations $\|\mathbf{h}\|_2$ and $\mathbf{h}^{(r)}$ represent the l_2-norm of the vector $\mathbf{h}$ and the estimate of $\mathbf{h}$ in the $r-$th iteration respectively.

II. Recurrent Neural Networks and LSTM

A. Recurrent Neural Networks

Recurrent neural networks (RNNs) are a class of neural networks which process sequential data such that the outputs corresponding to the previous step are fed as inputs in the current step. Let the given input sequence be denoted by $\mathbf{x} = (x_1, x_2, ..., x_T)$ where each x_t corresponds to the $t-$th real valued data. The sequence of hidden vectors $\mathbf{h} = (h_1, h_2, ..., h_T)$ and the output vectors $\mathbf{y} = (y_1, y_2, ..., y_T)$ for the time steps $t = 1$ to $t = T$ are iteratively evaluated employing the following equations [10]

$$h_t = f_{act}(W_{xh}x_t + W_{hh}h_{t-1} + b_h) \quad (1)$$
$$y_t = W_{hy}h_t + b_y \quad (2)$$

where $W_{(\cdot)}$ represents the weight matrices, f_{act} represents the hidden layer activation function such as the sigmoid or $\tanh(\cdot)$ functions [13]. A block diagram representation of the RNN architecture is demonstrated in Fig. 1. Unlike conventional artificial neural networks, although RNNs successfully capture the inherent temporal correlation which exists in sequential data, yet they are associated with numerous shortcomings. They are generally unstable and cannot be stacked into very deep models. Further, they suffer from vanishing gradient and exploding gradient problems. Since RNNs cannot keep track of long-term dependencies of the memory units, the functionalities of RNNs are limited for long-memory sequences. Thus, in order to overcome the shortcomings associated with RNNs, long short-term memory networks were developed [14]. A brief description of LSTMs is given in the next subsection.

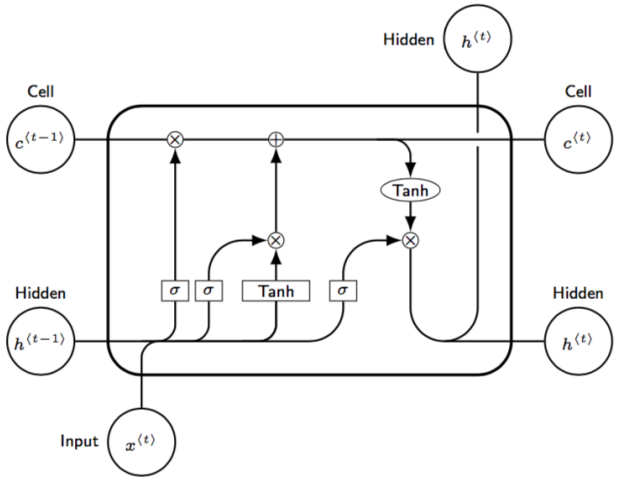

Fig. 2. LSTM Unit Architecture

B. Long Short-Term Memory

A LSTM layer comprises of recurrently connected memory blocks. Each memory block consists of one or more recurrently connected memory cell units, three multiplicative units and three gates mainly the input, forget and output gates which regulates the flow of information inside the memory block. Each cell memorizes the previous state's values over time intervals such that information inside and outside the cell is regulated via these three gates. The input/output equations corresponding to the gates in a LSTM are given as [14]

$$i_t = \sigma(w_i[h_{t-1}, x_t] + b_i) \quad (3)$$
$$f_t = \sigma(w_f[h_{t-1}, x_t] + b_f) \quad (4)$$
$$o_t = \sigma(w_o[h_{t-1}, x_t] + b_o) \quad (5)$$

where i_t, f_t, o_t represents the input, forget, output gates respectively and σ denotes the standard sigmoid activation function. The quantity $w_{(\cdot)}$ represents the weight corresponding to the respective gate neurons, x_t denotes the input at the $t-$th instant and $b_{(\cdot)}$ is the bias vector associated with the respective gates. The equations corresponding to the cell state c_t, candidate cell state $\widetilde{c}_t$ and the final output h_t corresponding to the $t-$th instant are given as [14]

$$\widetilde{c}_t = \tanh(w_c[h_{t-1}, x_t] + b_c) \quad (6)$$
$$c_t = f_t * c_{t-1} + i_t * \widetilde{c}_t \quad (7)$$
$$h_t = o_t * \tanh(c_t) \quad (8)$$

where $*$ represents the convolution operator. A diagrammatic representation of the LSTM architecture is shown in Fig. 2.

III. Database and Feature Set

A. Database

The study in this work is carried out on the SSPNet Vocalization Corpus (SVC) whose description is provided in the Social Signals sub-challenge of the 2013 Interspeech Computational Paralinguistics Challenge [12]. The primary task is to perform frame-wise detection and localisation of paralinguistic events into two classes: laughter and fillers

Algorithm 1 $k-$means Clustering for Social Signal Classification

Input: No. of clusters k, Maximum number of iterations I, Stopping threshold ϵ

Initialization: MFCC feature vectors corresponding to randomly chosen k data points initialised as the centroids $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_k$ of the k clusters

Set counter $r = 0$

while $r < I$ or $\sum_{i=1}^{k} \left\| \boldsymbol{\mu}_i^{(r)} - \boldsymbol{\mu}_i^{(r-1)} \right\|^2 > \epsilon$ **do**

Step 1: Evaluate the Euclidean distance between each MFCC feature vector and the k centroid vectors.

Step 2: Assign the data point to a particular cluster whose distance from the centroid vector is minimum.

Step 3: Recompute the centroid vectors corresponding to the newly formed clusters in the previous step.

$r \leftarrow r + 1$

end while

Output: k clusters of the dataset

namely "ahm", "eh", etc. The audio data comprises of all types vocalizations such as speech and silence as well. An in-depth analysis of the SSPNet vocalization corpus showed that non-verbal cues didn't distribute uniformly over time, but appear in bursts and more frequently by male subjects [15]. The corpus comprises of 2763 audio clips of 11 seconds time-frame, each consisting of atleast one laughter or filler event. It includes 63 females and 57 males, summing up to 120 subjects. More details about the dataset is available at [12].

B. Feature Set

The dataset is segregated into training and testing cases to perform classification of the human speech into two classes: laughter and filler. In this work, Mel-frequency cepstrum coefficients, the most commonly used features in speech recognition are employed. MFCCs are derived from the cepstral representation of an audio clip wherein a cepstrum corresponds to the inverse Fourier transform of the logarithm of the estimated spectrum of the signal. The main motivation of employing MFCC in our work is that owing to the equal spacing of the frequency bands in MFCCs, it closely approximates the human auditory system's response unlike the linearly-spaced frequency bands in the normal cepstrum [16]. In addition to MFCC, we propose to include *cluster* as a feature in this work. The main motivation behind this is, clusters effectively segregate a given set of data points into a number of groups with similar traits. Further, arranging the data points into clusters helps discover hidden patterns within the dataset. The $k-$means clustering algorithm employed in this work is summarized in Algorithm 1.

IV. Experiments and Results

The LSTMs have been demonstrated to yield better performances for long-memory sequences data such as audio/speech. This work considers different hidden layer sizes and two output nodes as target classes: laughter and filler. The first experiment was performed on the original feature set of the SSPNet vocalization corpus without considering cluster as a feature. Table I summarizes the training and testing accuracies with different LSTM network topologies. The best result in terms of testing accuracy is obtained for the network with a hidden layer size of 50.

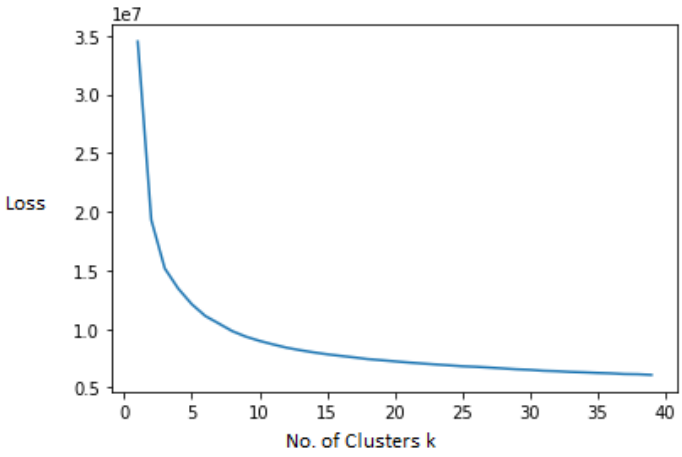

Fig. 3. Loss versus No. of Clusters k

TABLE I
LSTM Model Comparison without Clustering as a feature

LSTM Network Topology	Training Accuracy	Testing Accuracy
20-30-2	93.49%	87.06%
20-40-2	93.90%	86.17%
20-50-2	95.04%	**88.18%**
20-60-2	95.14%	86.50%
20-80-2	96.35%	86.17%

The second experiment was performed on the SSPNet dataset with clustering as a feature which resulted in the size of the input layer to be increased from 20 to 21 since the additional cluster feature information is included. The results are summarized in Table II. To fix the number of clusters k, we plot loss versus k where loss is defined as sum of Euclidean distances of samples to their closest cluster center. Fig. 3 depicts the loss versus number of clusters k plot and k is set as 14 since the loss is almost constant after that value. The maximum number of iterations is fixed as $I = 10$. On comparing the accuracies of the LSTM-based classification approaches in Tables I and II, it can be inferred that the highest testing accuracy of 89.15% is obtained by considering cluster as a feature while its analogous counterpart is 88.18% in conventional LSTM without clustering.

Next, we investigate the effect of considering cluster as a feature in *deep* LSTM networks. In order to create *deep* LSTM networks, we connect the output corresponding to

TABLE II
LSTM Model Comparison with Clustering as a feature

LSTM Network Topology	Training Accuracy	Testing Accuracy
21-30-2	93.66%	86.25%
21-40-2	94.87%	87.14%
21-50-2	95.55%	**89.15%**
21-60-2	95.07%	87.22%
21-80-2	96.35%	86.5%

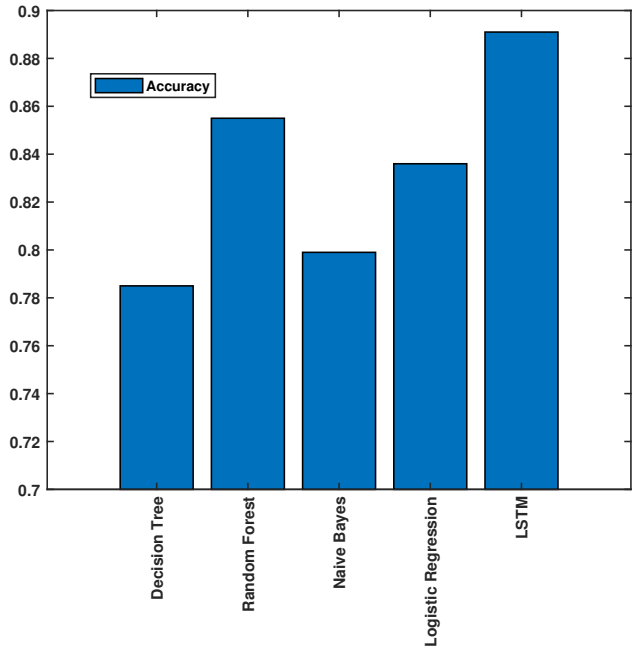

Fig. 4. Accuracy comparison between different classification models.

the first LSTM as an input to the second LSTM. Since the output layers are softmax layers, they can be interpreted as posterior probabilities of the laughter and filler classes. Thus, the second LSTM network computes an improved form of posterior probabilities. Tables III and IV respectively depict the accuracies of *deep* LSTMs with and without cluster as an additional feature. The italicized portion of the network topology represents the first LSTM that was trained on the original feature set while the output of the first LSTM is employed to train the second one. For the case without clustering as a feature, it is observed that some of the deep LSTM models yield an improvement in the testing accuracies in comparison to their regular LSTM counterparts. However, interestingly, for the deep LSTMs considering cluster as a feature, there is no improvement in the testing accuracies in comparison to the regular LSTM networks.

The performance accuracy of the proposed LSTM-based approach is benchmarked by comparison with popular classification models such as random forest, naive Bayes, logistic regression, decision tree. For the purpose of a fair comparison between the various schemes, we considered cluster as a feature for the other classification methods as well. A graphical representation of the accuracy comparisons with different classification models is depicted in Fig. 2 and Table. V provides the accuracy of different classification models along with the proposed LSTM technique in a tabular form. It can be seen that the LSTM-based classification has the highest testing accuracy of 89.1% while the decision tree-based scheme demonstrates the lowest accuracy of 78.5%.

V. Conclusion

This work proposes a long short-term memory-based approach for frame-wise classification and localization of non-verbal laughter and filler vocalizations of the SSPNet vocalization corpus. Further, the novel idea of incorporating cluster as a feature for the LSTM-based classification via a $k-$means clustering algorithm demonstrated significant improvement in both training and testing accuracies. Deep LSTM networks were formed by stacking one LSTM over another and exhibited improved performance for regular networks. However, for deep LSTMs which considered cluster as an additional feature, there was no perceivable improvement. Finally, a performance comparison with popular machine learning-based classification methods validated the superior performance of LSTM for classification of non-verbal cues.

TABLE III
Deep LSTM Model Comparison with Clustering as a Feature

Stacked LSTM Network Topology	Training Accuracy	Testing Accuracy
21-30-2-21-2	92.66%	84.32%
21-40-2-21-2	93.8%	86.09%
21-50-2-21-2	94.83%	88.59%
21-60-2-21-2	94.25%	85.53%
21-80-2-21-2	95.18%	85.13%

TABLE IV
Deep LSTM Model Comparison without Clustering as a Feature

Stacked LSTM Network Topology	Training Accuracy	Testing Accuracy
20-30-2-20-2	93.56%	86.66%
20-40-2-20-2	94.52%	87.54%
20-50-2-20-2	94.49%	87.46%
20-60-2-20-2	94.90%	86.74%
20-80-2-20-2	95.35%	85.77%

References

[1] M. Pantic, R. Cowie, F. D'Errico, D. Heylen, M. Mehu, C. Pelachaud, I. Poggi, M. Schroeder, and A. Vinciarelli, "Social signal processing: The research agenda," in *Visual analysis of humans*. Springer, 2011, pp. 511–538.

[2] A. Vinciarelli, M. Pantic, and H. Bourlard, "Social signal processing: Survey of an emerging domain," *Image and vision computing*, vol. 27, no. 12, pp. 1743–1759, 2009.

[3] A. Vinciarelli, H. Salamin, and M. Pantic, "Social signal processing: Understanding social interactions through nonverbal behavior analysis," in *2009 IEEE computer society conference on computer vision and pattern recognition workshops*, 2009, pp. 42–49.

[4] P. E. Kennedy and A. G. Hauptmann, "Laughter extracted from television closed captions as speech recognizer training data," in *Sixth European Conference on Speech Communication and Technology*, 1999.

[5] H. Salamin, A. Polychroniou, and A. Vinciarelli, "Automatic detection of laughter and fillers in spontaneous mobile phone conversations," in *2013 IEEE International Conference on Systems, Man, and Cybernetics*, 2013, pp. 4282–4287.

[6] T. F. Krikke and K. P. Truong, "Detection of nonverbal vocalizations using Gaussian mixture models: Looking for fillers and laughter in conversational speech." in *INTERSPEECH*, 2013, pp. 163–167.

[7] A. Janicki, "Non-linguistic vocalisation recognition based on hybrid GMM-SVM approach." in *INTERSPEECH*, 2013, pp. 153–157.

TABLE V
Model Comparisons with Clustering as a Feature

Model Architecture	Accuracy
Decision Tree	0.785
Random Forest	0.855
Naive Bayes	0.799
Logistic Regression	0.836
LSTM	0.891

[8] R. Gupta, K. Audhkhasi, S. Lee, and S. Narayanan, "Detecting paralinguistic events in audio stream using context in features and probabilistic decisions," *Computer speech & language*, vol. 36, pp. 72–92, 2016.

[9] L. Kaushik, A. Sangwan, and J. H. Hansen, "Laughter and filler detection in naturalistic audio," *International Speech and Communication Association*, 2016.

[10] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *2013 IEEE International conference on acoustics, speech and signal processing (ICASSP)*, 2013, pp. 6645–6649.

[11] R. Brueckner and B. Schulter, "Social signal classification using deep blstm recurrent neural networks," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 4823–4827.

[12] B. Schuller, S. Steidl, A. Batliner, A. Vinciarelli, K. Scherer, F. Ringeval, M. Chetouani, F. Weninger, F. Eyben, E. Marchi *et al.*, "The interspeech 2013 computational paralinguistics challenge: Social signals, conflict, emotion, autism," in *INTERSPEECH 2013, 14th Annual Conference of the International Speech Communication Association, Lyon, France*, 2013.

[13] B. L. Kalman and S. C. Kwasny, "Why tanh: choosing a sigmoidal function," in *International Joint Conference on Neural Networks (IJCNN)*, vol. 4, 1992, pp. 578–581.

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

[15] A. Vinciarelli, P. Chatziioannou, and A. Esposito, "When the words are not everything: the use of laughter, fillers, back-channel, silence, and overlapping speech in phone calls," *Frontiers in ICT*, vol. 2, p. 4, 2015.

[16] T. Ganchev, N. Fakotakis, and G. Kokkinakis, "Comparative evaluation of various MFCC implementations on the speaker verification task," in *SPECOM*, 2005, pp. 191–194.