ISBN 978-981-14-1684-2
Proceedings of 2019 the 9th International Workshop on Computer Science and Engineering
(WCSE 2019 SUMMER)
Hong Kong, 15-17 June, 2019, pp. 491-499
doi: 10.18178/wcse.2019.06.073

Research on the Effect of Different Speech Segment Lengths on Speech Emotion Recognition Based on LSTM

Zheng Liu [1], Fuji Ren [1 +], and Xin Kang [1]

[1] School of Information Faculty of Engineering, Tokushima University, Japan

Abstract. The emergence and development of deep learning makes speech emotion recognition more crucial. For the neural network sequence model, the amount of information contained in different lengths of speech segments has different effects on the sequence model. There is no reasonable explanation for how to separate the speech as input. In this work, we used the CASIA Chinese Emotional Corpus and divided it into 5 groups that every group has different lengths between 100-500ms. Using the features extracted by OpenSmile toolkit to calculate the standard deviation of each group, we found that the features of the same dimension have a very even distribution in the 200ms segments. We used the LSTM model with different features as input, and statistically analyzed the results, the results verified that 200ms is the most suitable input for the sequence model.

Keywords: speech emotion recognition, LSTM, CASIA, feature analysis

1. Introduction

Advances in computer hardware have promoted the development of artificial intelligence in text, images, and voice [1]. In order to allow the robot to enter everyone's life, some research has made the robot have the ability to recognize objects [2] and be able to move like a human [3], but the core is to make the robot emotional [4]. The research on speech emotion recognition is mainly about capturing the information contained in the speech signal in the sound wave and distinguishing the emotion conveyed by the speech through a series of conversion processing. In the last ten years, along with the research progress of speech emotion recognition, great achievements have been made in speech emotion feature extraction and analysis, emotion corpus construction, emotion recognition model and other aspects.

In the recent years, with the rapid development of deep learning, relevant data models have been widely applied to the field of speech research. Among them, the recurrent neural network (RNN) [5] introduces the concept of time series into the network structure design, making the neural network stronger adaptability in data with time changes. However, speech is a wave with a time dimension. If the speech is divided into segments according to different lengths of time, the contained information features distribution is different, which has different effects on the sequence model.

In order to study the effect of different speech segment lengths on speech emotion recognition. In this work, we divided the dataset into 5 groups of different lengths between 100-500ms, then calculated the standard deviation of the same feature by 6000 samples, and found that the speech features are more obvious in 200ms. We used the LSTM model to verify the performance is optimal at 200ms, which confirms this finding.

More detailed experimental description will be given in this paper. Section 2 shows some research related to our experiments. Section 3 shows some of the methods used in this experiment, including speech

+ Corresponding author. Tel.: +81886569684; fax: +81886566575.
E-mail address: ren@is.tokushima-u.ac.jp.

feature analysis, and the LSTM model. Section 4 describes the process of our experiment, the experimental results and analysis of the results. Section 5 concludes the work and leads to the future direction of the work.

2. Related Work

Through the study of speech emotion recognition, we can help robots understand human emotions easily and let robots react and interact with people kindly [6]. Many of the research work at the beginning of speech emotion recognition focused on aspects of speech emotion features, such as the work of [7] [8] explored the use of principal component analysis to reduce the dimension of emotional features in speech emotion recognition. At the same time, many corpora of speech emotion recognition were constructed. The authors in [9] recorded and built the speech library IEMOCAP for emotional analysis for the purpose of academic research, which has been highly recognized by the academic circles in relevant fields. In [10], the authors combined with the MFCC and D-MFCC features in speech, achieved good results in speech emotion recognition. The authors in [11], using more features, combined with the SVM model, achieved a higher recognition rate in speech emotion recognition research. There is a problem in these models, which cannot capture timing information in speech.

In the next few years, the time series models have attracted many researchers, such as works in [12] combined with speech i-vector features, using RNN time series models to identify speech emotions. The authors in [13], combined with more features of speech segments, explored speech emotion recognition using BiLSTM. However, there was a problem in the study of speech emotion recognition using the sequence model, that they did not make a reasonable explanation for the length of the speech segment.

Eyben and others in technical university of Munich have developed an open tool kit OpenSmile for speech emotion feature extraction, which realizes batch automatic extraction of commonly used speech emotion features including energy, fundamental frequency, duration, Mel cepstrum coefficient, etc. [14], and has been widely applied in emotion extraction related research.

LSTM model [15] made up for the problems of RNN such as gradient disappearance, gradient explosion, and lack of long-term memory capability, which enables the cyclic neural network truly and effectively utilize long-distance timing information.

In our work, we used the OpenSmile tool for feature extraction. By analyzing the features and using LSTM model, we proved that the speech segments performed optimally on 200ms segments.

3. Method

3.1. Feature analysis

In this session, we divided the audio files into different speech segments between 100-500ms. Using the OpenSmile tool to extract features for each segment, we extracted 1582 features on every segment. In Figure 1, to observe each feature distribution of segments of different segment lengths, we calculate the standard deviation of each feature separately. Standard deviation calculation, shown in formula 1. We found that the standard deviation of each feature is different in different time segments.

$$\mathrm{SD} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(xi - \mu)^2} \tag{1}$$

In order to see the distribution of features more clearly, we visualized one of the feature distribution. The same features of the 100-500ms speech segment were normalized to 0-1 separately. Figure 2 shows one feature called voicingFinalUnclipped distribution histogram between the 0-1 between the five groups. It can be seen from the figure that the distribution of the same feature on different time segments is different.

For the time series neural network model, if the distribution of data features in time are denser, it is impossible to learn the difference of data during the time effectively. Hence, this feature is not a good feature. Conversely, if the data features have a very even distribution during the time, the information in the features can be more effectively captured by the time series model. This provides a basis for our subsequent experiments. The features have different distributions on different time segments and the information contained is different.

3.2. LSTM model

Recurrence neural network (RNN) can effectively process data with time series problems. The Long Short-Term Memory model we used in this paper, called LSTM for short, is a kind of recurrent neural network, evolved from RNN. Figure 3 is the basic structure of LSTM model. X_t represents the high-dimensional feature of the current speech segment. After the result of the operation of one LSTM neuron, X participates in the operation of the next neuron, thus establishing the time dimension of the data.

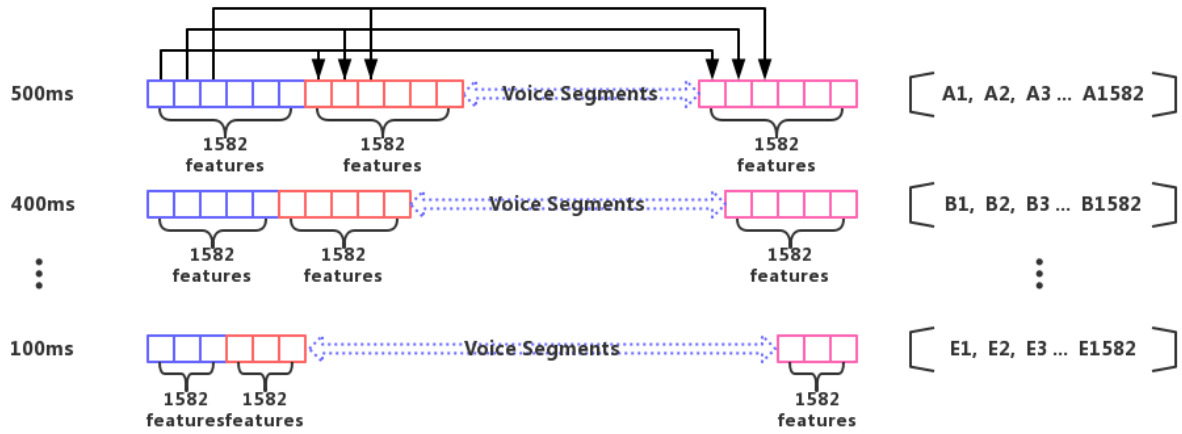


Fig. 1: Feature distribution calculation process. [A1, A2 … A1582] represents the standard deviation set of 1582 features of the 500ms segments. A1, B1 … E1 represent the feature of the first dimension at 500-100ms segments and their values are different.

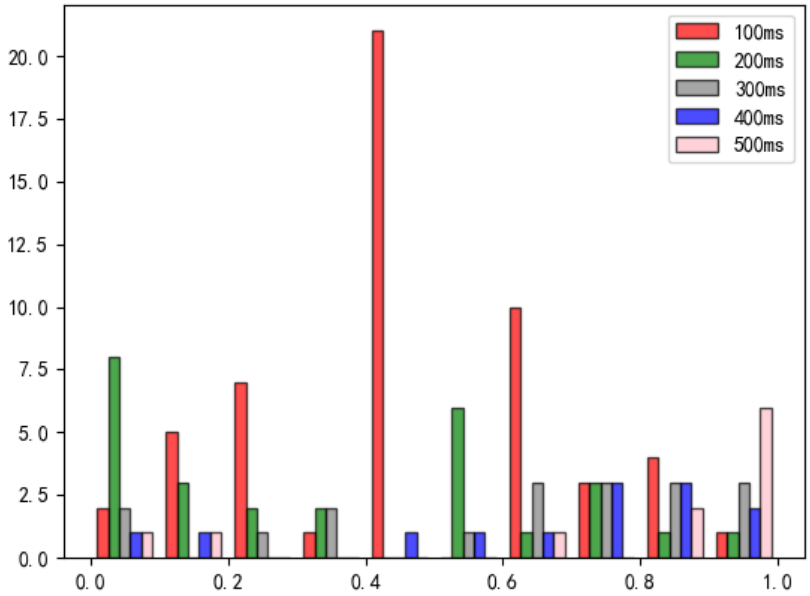


Fig. 2: The distribution histogram of voicingFinalUnclipped on different speech time segments.

LSTM is a special structural type of RNN model. Figure 4 shows the internal structure of model neurons. Compared with RNN, LSTM adds three control units: forgetting gate, input gate and output gate. With the information entering the model, neurons in LSTM will judge the information, the information conforming to the rules will be left behind, and the information not conforming will be forgotten. Based on this principle, the problem of long sequence dependency in neural network can be solved.

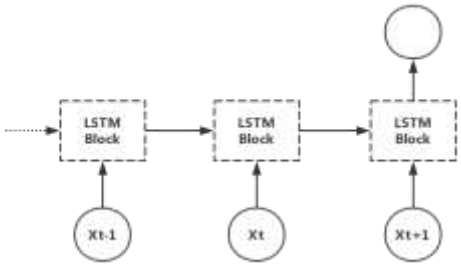


Fig. 3: LSTM structure.

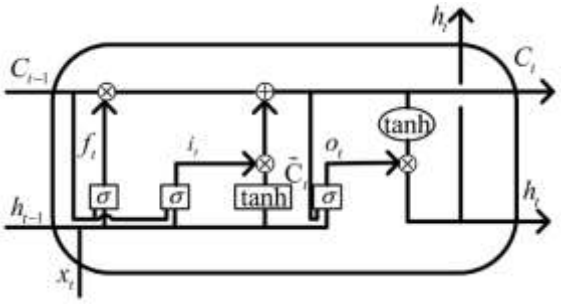


Fig. 4: Cell of LSTM.

Forgetting gate is used to discard useless information in the previous state. Its inputs include the input vector of the current time step and the output vector of the output gate in the previous time step. Mapping to 0 to 1 through a sigmoid activation function transformation. determines how much of the old state information needs to be discarded, so the directly obtained results are multiplied element by element.

$$f_t = \sigma(W_{fh}h_{t-1} + W_{fx}x_t + W_{fc}C_{t-1} + b_f) \quad (2)$$

In the step of input gate i_t new information is mainly added to the neuron state. This work includes two parts: input gate layer and activation layer. The input gate layer will determine which values in the neuron state need to be updated, and then generate a set of candidate values $\tilde{C}_t$ in combination with the tanh activation layer to replace the old values that need to be updated in the neuron state. Then the two vectors are multiplied element by element and added to neurons. The output vector of the input gate is only 0 and 1, and only those state quantities of 1 will be updated after multiplication by elements, so i_t acts as a mask.

$$i_t = \sigma(W_{ih}h_{t-1} + W_{ix}x_t + W_{ic}C_{t-1} + b_i) \quad (3)$$

$$\tilde{C}_t = tanh(W_{ch}h_{t-1} + W_{cx}x_t + b_c) \quad (4)$$

The output gate is still based on the neuron state of the current time step. This part also consists of a sigmoid activation layer and a tanh activation layer. The sigmoid layer determines which state information is to be output, and the tanh layer compresses the current neuron state into an interval of (-1, 1). Then, multiplication by element is performed to generate the output variable of this unit. The output variable is simultaneously added to the loop as h_{t-1} of the next cell.

$$o_t = \sigma(W_{oh}h_{t-1} + W_{ox}x_t + W_{oc}C_t + b_o) \quad (5)$$

$$h_t = o_t * tanh(C_t) \quad (6)$$

4. Experiments and Results

There are many popular corpora in the field of speech emotion recognition. For example, common EMO-DB German effective speech corpora [16] in Germany, Belfast English affective speech corpus [17] in Britain, Semaine corpus [18], IEMOCAP corpus, etc.

CASIA Chinese Emotion Corpus that we used in this work was constructed by the Institute of Automation of the Chinese Academy of Sciences in 2005. It is composed of four professional sound recorders, 2 men and 2 women. Under pure sound recording environment, it includes happiness, sadness, anger, fright and neutrality on the basis of five different emotions. Each recorder reads 2500 sentences of text according to the five different emotions mentioned above, totaling 9,600 sentences, and is sampled at 16kHz. In this experiment, 6,000 audio files were used as our dataset.

The basic flow of this experiment is shown in Figure 5 below. In this experiment, we used CASIA Chinese Emotion Corpus. The corpus was divided into different time segments. OpenSmile Toolkit was used to extract features from each speech segment separately. We first counted the feature distribution of all the files in the corpus, and then send the extracted features to the model for training. The results of the training

were used to verify our findings in the feature statistics. In order to reduce the accidental errors in the experiment, we also adjusted different feature dimensions and statistically processed the experimental results.

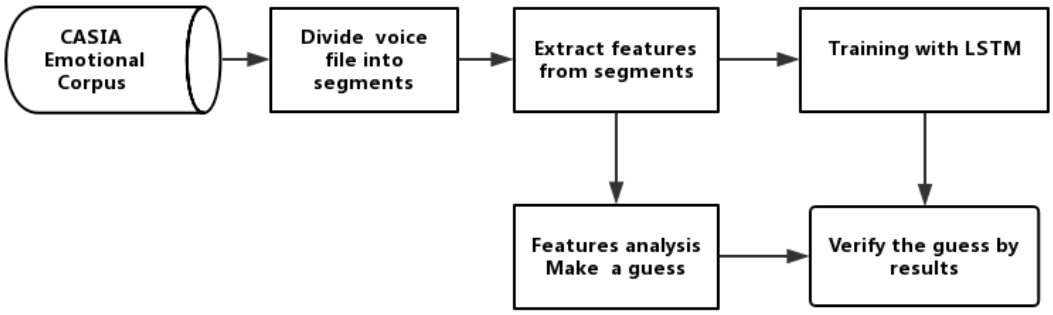


Fig. 5: Experimental process.

4.1. Feature extraction and statistic

CASIA corpus contains many audio files. After determining the speech corpus, our next step is to extract the features of the audio files. In the step of feature extraction, we used OpenSmile Toolkit to extract the features of the speech segments.

OpenSmile is an open toolkit developed by technical university of Munich for speech feature extraction, which is widely used by many companies and academic research. The basic features of voice such as cepstrum coefficient (MFCC), linear prediction coefficient (LPC), frame strength, etc. can be extracted effectively.

In order to explore the effect of segmentation time on the model, we divided each speech file into several speech segments by taking 100ms, 200ms, 300ms, 400ms and 500ms as time units. Then, emobase2010 in OpenSmile is used as the feature configuration file to extract each segment one by one, and the extracted results are saved for the input of subsequent models.

For each feature of a voice file, the distribution is different between 100-500ms. The standard deviation indicates the degree of dispersion of the data distribution. The larger the value means the wider the data distribution, the voice information is richer in the time segment. We use the standard deviation to calculate the distribution of the 1582 features of a voice file in different length segments, and define at what time segment the information of each voice file is more abundant.

As shown in Figure 6. We calculated statistics on 6000 files in the dataset and found that the distributions of the features are the most discrete on 200ms time segments.

4.2. Model training

In our work, as shown in Table 1, 5440 audio files were randomly separated as training sets, 280 files as verification sets and 280 files as test sets.

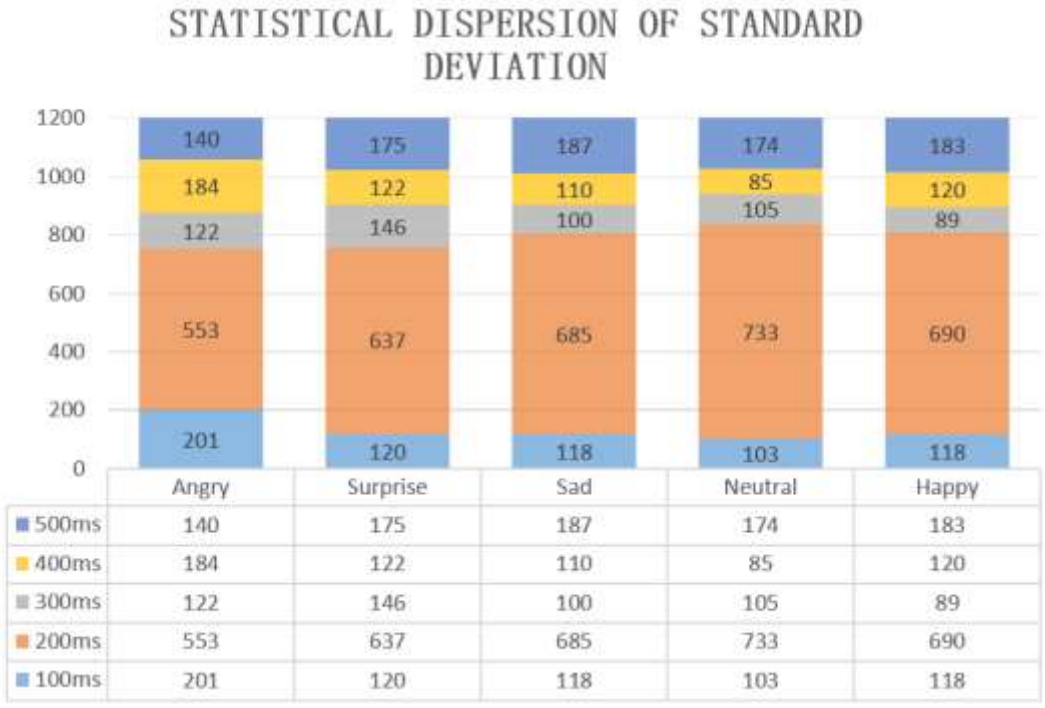
Table II shows various parameters used in LSTM model. We selected different numbers of speech features as input, so the input dimensions of the model were adjusted by 300,500,800,1000,1200 dimensions in turn. The hidden layer dimension was fixed at 256 dimensions. To fit the data better, we used dynamic learning rate. During the model training, we dynamically adjust the learning rate according to the training rounds.

Table 1. Dataset statistics.

Datasets	CASIA
#Train	5440
#Valid	280
#Test	280

Table 2. Model parameters

Parameters	Value
Input Size	300,500,800,1000,1200
Hidden Size	256
Learning Rate	0.001->0.0001->0.00001
Batch Size	1
Optimizer	Adam
Loss Function	CrossEntropy



	Angry	Surprise	Sad	Neutral	Happy
500ms	140	175	187	174	183
400ms	184	122	110	85	120
300ms	122	146	100	105	89
200ms	553	637	685	733	690
100ms	201	120	118	103	118

Fig. 6: Statistics of feature dispersion on dataset in different time segments.

In order to observe the changes in the different evaluation indicators during the training process, Figure 7 and Figure 8 show the training process in the 300 features dimension.

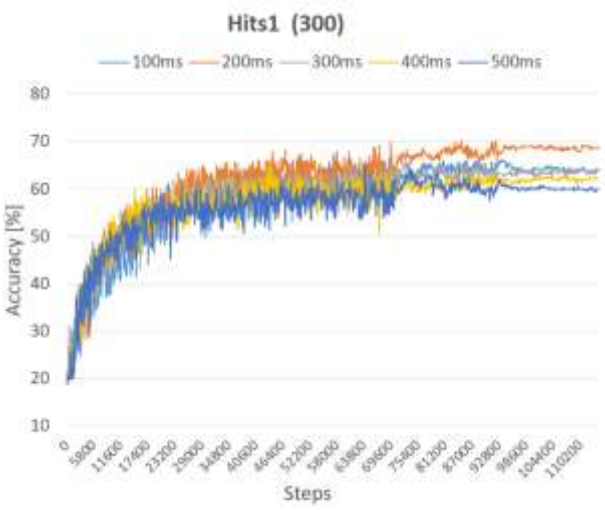


Fig. 7: Hits1 Performance on Verification Set.

Fig. 8: MRR Performance on Verification Set.

In the figures, we can see that with the increase of training times, the validation set curve converges to a fixed value, and each evaluation index performs best on 200ms time segments.

4.3. Results statistics

Tables III shows the experimental results under different time lengths. Where : (i) Hits1 indicates that the first 1 digit of the prediction result output by LSTM model contains positive solution; (ii) Hits2 indicates that the first 2 digits of the prediction result output by LSTM model contain positive solutions; (iii) Hits3 indicates that the first 3 digits of the prediction result output by LSTM model contain positive solutions; (iv) MRR represents the number of positive solutions and the mean of the reciprocal sum.

The calculation formula for MRR evaluation index is as the follow: Q is the sample query set, $|Q|$ represents the number of queries in Q, and $rank_i$ represents the number of queries in the 1st query.

$$MRR = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{rank_i} \tag{7}$$

In the table data, the 1, 2, 3 marks in the upper right corner of the number indicates the ranking of the prediction results in the current column under different time divisions for subsequent statistics.

Table 3. Results in different time lengths.

Features	Time	Hits@1	Hits@2	Hits@3	MRR
300	100	73.93[2]	**91.79**[1]	97.14[2]	85.3[2]
	200	**75**[1]	**91.79**[1]	**98.21**[1]	**85.98**[1]
	300	69.64[3]	85	95.71[3]	81.88[3]
	400	66.07	87.14[3]	95.71[3]	80.43
	500	65.71	84.29	93.21	79.57
500	100	70.36[2]	**90**[1]	**97.14**[1]	83.24[2]
	200	**72.86**[1]	**90**[1]	96.79[2]	**84.48**[1]
	300	69.29[3]	87.86	96.07[3]	82.26[3]
	400	66.07	88.57[3]	95.71	80.7
	500	68.57	84.64	95	81.17
800	100	65	85	95	79.48
	200	**72.86**[1]	**88.57**[1]	95.36[3]	**84.1**[1]
	300	71.43[2]	86.07[3]	**96.07**[1]	83.05[2]
	400	69.64[3]	**88.57**[1]	95.71[2]	82.43[3]
	500	63.93	82.14	95.36[3]	78.53
1000	100	57.14	80	90.36	74.22
	200	66.07[2]	85[2]	92.86[3]	79.89[2]
	300	**68.93**[1]	**85.36**[1]	92.5	**81.31**[1]
	400	61.79[3]	84.29[3]	93.93[2]	77.68[3]
	500	59.64	83.57	**94.29**[1]	76.55
1200	100	55	82.86[3]	92.14[3]	73.83
	200	**64.64**[1]	**85**[1]	**94.29**[1]	**79.27**[1]
	300	64.29[2]	81.79	92.14[3]	78.35[2]
	400	63.21[3]	82.86[3]	92.86[2]	78.03[3]
	500	57.5	**85**[1]	91.79	75.44

4.4. Results evaluation

To reduce the impact of contingency of the test set prediction results, we respectively make statistics on index Score results between 100-500ms, and obtain score after weighting. In table VII, Top1, Top2 and Top3 represent the number of results with the index score ranking first, second and third. We give weight to each item, and the calculation formula is as the follow.

$$Score = Top1 * 5 + Top2 * 3 + Top3 * 1 \quad (8)$$

According to Score, we conclude that speech segmentation performs best in 200ms in the time series model.

Table 4. Results statistics

Time	Top1	Top2	Top3	Score
100	3	5	2	32
200	14	4	2	**84**
300	4	4	6	38
400	1	3	11	25
500	2	3	1	20

5. Conclusion and Future Work

In this paper, to study the effect of different speech segment lengths on speech emotion recognition, we propose an analysis method for feature distribution of speech time segments. Through our method, we observed that the feature information contained was evenly distributed in 200ms. In addition, we combined the LSTM model with 100-500ms of features as input and the results showed the best performance at 200ms, which confirmed our finding.

At present, our work only based on the statistics of the overall features. In the future, we will continue to study the distribution of speech features, explore which features are more effective in time series and provide more interpretable basis for speech segmentation.

6. Acknowledgment

This research has been partially supported by JSPS KAKENHI Grant Number 15H01712.

7. References

[1] Fuji Ren : From Cloud Computing to Language Engineering， Affective Computing and Advanced Intelligence，International Journal of Advanced Intelligence， Vol.2， No.1， pp.1-14， 2010

[2] Fuji Ren, Bo Li, Qimei Chen, Single Parameter Logarithmic Image Processing For Edge detection, IEICE Transaction INF. & SYST，Vol.E96-D,No.11,pp.2437-2449,Nov. 2013

[3] Yu Gu, Jinhai Zhan, Yusheng Ji, Jie Li, Fuji Ren and Shangbin Gao : MoSense: An RF-Based Motion Detection System via Off-the-Shelf WiFi Devices, IEEE Internet of Things Journal, Vol.4, No.6, 2326-2341, 2017

[4] Fuji Ren : Affective Information Processing and Recognizing Human Emotion， Electronic Notes in Theoretical Computer Science， Vol.225， No.2009， pp.39-50， 2009

[5] Hojjat Salehinejad, Sharan Sankar, Joseph Barfett, Errol Colak, and Shahrokh Valaee .Recent Advances in Recurrent Neural Networks, 2017.

[6] Fuji Ren, Zhong Huang, Automatic facial expression learning method based on humanoid robot XIN-REN, IEEE Transactions on Human-Machine Systems, Vol.46, No.6, pp.810-821,2016

[7] Bj ̈orn Schuller, Anton Batliner, Dino Seppi, et al., “The relevance of feature type for the automatic classification of emotional user states: low level descriptors and functionals.,” in INTERSPEECH, 2007, pp. 2253–2256

[8] Changqin Quan, Dongyu Wan, Bin Zhang and Fuji Ren:Reduce the Dimensions of Emotional Features by Principal Component Analysis for Speech Emotion Recognition, 2013

[9] C.Busso,M.Bulut,C.C.Lee,A.Kazemzadeh,E.Mower,S.Kim, J. N. Chang, S. Lee, and S. S. Narayanan, “Iemocap: interactive emotional dyadic motion capture database”, 2008

[10] Kai Yang, Quan Shi and Fuji Ren: Speech emotion recognition based on MFCC and D-MFCC, 2016.

[11] Yixiong Pan, Peipei Shen and Liping Shen. Speech Emotion Recognition Using Support Vector Machine. International Journal of Smart Home Vol. 6, No. 2, April, 2012

[12] Teng Zhang, Ji Wu:Speech Emotion Recognition With I-vector feature and RNN Model, 2015

[13] Yue Xie: Long-short term memory for emotional recognition with variable length speech, 2018.

[14] Eyben F, Wollmer M, Schuller B. openSMILE—The Munich versatile and fast open-source audio feature extractor. In: Proc. of the 2010 ACM Multimedia. Firenze, 2010.1459-1462 .

[15] GRAVES A. Long short-term memory[M]. Berlin: Springer, 2012: 1735-1780.

[16] F. Burkhardt, A. Paeschke, M. Rolfes, W. Sendlmeier, and B. Weiss, “A database of german emotional speech,” in Proc. Interspeech 2005, Lisbon, Portugal, 2005, ISCA, pp. 1517– 1520.

[17] Cowie R, Douglas-Cowie E, Savvidou S, McMahon E, Sawey M, Schroder M. Feeltrace: An instrument for recording perceived emotion in real time. In: Proc. of the 2000 ISCA Workshop on Speech and Emotion: A Conceptual Frame Work for Research. Belfast: ISCA, 2000. 19-24.

[18] Gary McKeown, Michel F. Valstar, Roderick Cowie , Maja Pantic:The SEMAINE corpus of emotionally coloured character interactions, 2010.