MULTI MODAL APPROACH OF SPEECH EMOTION RECOGNITION USING MULTI LEVEL MULTI HEAD FUSION ATTENTION BASED RNN

AUTHORS:

1. **NGOC-HUYNH HO**
2. **HYUNG-JEONG YANG**
3. **SOO-HYUNG KIM**
4. **GUEESANG LEE**

PUBLISHED ON:30TH MARCH 2020

DEPARTMENT OF ELECTONICS AND COMPUTER ENGINEERING,CHONNAM NATIONAL UNIVERSITY ,GWANGJU 61186,SOUTH KOREA

OUR TEAM:

BOTTA PRUDHVEE RAJ-1806054

PRATTIPATI KRISHNA KUMAR-1806114

PADALA SHASHANK-1806119

PROBLEM STATEMENT:

Speech emotion recognition is a challenging but important task in human computer interaction (HCI). As technology and understanding of emotion are progressing, it is necessary to design robust and reliable emotion recognition systems that are suitable for real-world applications both to enhance analytical abilities supporting human decision making and to design human machine interfaces (HMI) that assist efficient communication.so the problem is to recognise emotion of a human using deep learning.

ABSTRACT:

We presents a multimodal approach for speech emotion recognition based on Multi-Level Multi-Head Fusion Attention mechanism and recurrent neural network (RNN). The proposed structure has inputs of two modalities: audio and text. For audio features, we determine the mel-frequency cepstrum (MFCC) from raw signals using Librosa library. Further, we use pre-trained model of bidirectional encoder representations from transformers (BERT) for embedding text information. These features are fed parallelly into the self-attention mechanism base RNNs to exploit the context for each timestamp, then we fuse all representatives using multi-head attention technique to predict mainly 4 emotional states (Sad,Angry,Happy,Neutral). We run our model on IEMOCAP dataset.

DATASET:

We used IEMOCAP dataset which consists of approximately twelve hours of recordings. Audio, video and facial key points data was captured during the five sessions. Each session is a sequence of dialogues between man and woman. In total, ten people split into five pairs took part in the process. After recording these conversations, authors divided them into utterances with speech. Note that audio was captured using two microphones. Therefore, the recordings contain two channels which correspond to male and female voices. Sometimes they interrupt each other. In these moments the utterances might intersect. This intersection takes about of all utterances time. It might lead to undesired results because microphones were place relatively near each other and thus inevitably captures both voices . To be comparable with related works, we divide two sets from the data: only *Improvised*, and *Mixed* (merging *Improvised* and *Scripted*) scenarios. In the *Improvised* scenario, subjects were asked to improvise based on hypothetical scenarios while in the *Scripted* one, subjects were asked to memorize and rehearse with scripts. The *Mixed* scenario means we combine both improvised and scripted scenarios to cover a full range of situations.

Emotion		Angry	Excited	Frustrated	Happy	Sad	Neurtal	Surprise	Fear	Disgust	Other
Improvised	# Utterance	289	663	971	284	608	1099	60	8	1	1
	Duration (min)	22.15	42.14	79.94	19.62	50.23	74.54	3.37	0.43	0.03	0.14
Mixed	# Utterance	1103	1041	1849	595	1084	1708	107	40	2	3
	Duration (min)	82.96	82.95	145.16	43.05	99.30	111.08	5.54	1.82	0.08	0.25

Number of utterances and duration per emotion class in the two scenarios of the IEMOCAP dataset Improvised and Mixed.

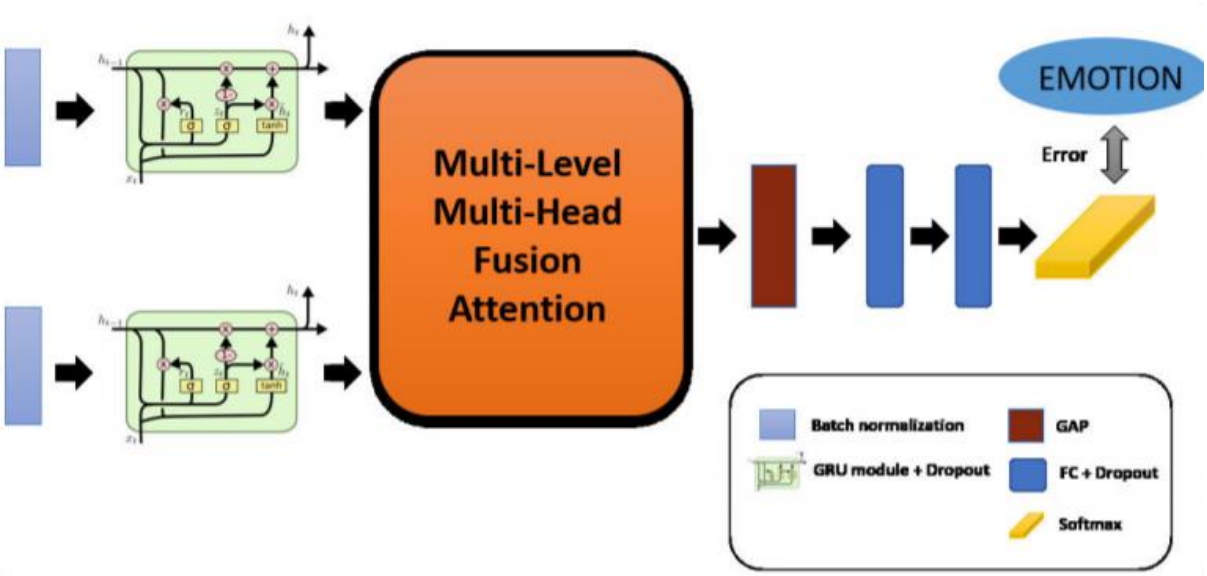
PREPROCESSING OF AUDIO DATA:

Initially we extracted audio signal from wav file using librosa at a sample rate of 16000. Later on we set audio frames of 100ms sampled at a rate of 50ms using hamming window. We set a large length of audio frame because this can decrease the difference between temporal length of audio data and textual data as the length of utterance is less than the duration of audio signal and then we extracted a total of 39 features for each step in which the first feature is the log-energy of sound. Then, it computes 12 MFCC (1-12) from 26 Mel frequency bands, and applies a cepstral liftering fillter with a weight parameter of 22. The 13-delta and 13-acceleration coefficients are appended to the MFCC. These features are mean normalized with respect to the full input sequence. The frequency range of the Mel-spectrum is set from 0 to 8 kHz.

PREPROCESSING OF TEXTUAL DATA:

To utilize information from text data, we compute the vector representation of words using BERT. BERT is an open sourced natural language processing (NLP) pre-trained model developed by researchers at Google in 2018. It is pre-trained on a large corpus of unlabelled text which includes the entire Wikipedia (about 2,500 million words)and a book corpus (800 million words). As opposed to directional models, which read the text input sequentially,BERT is considered bidirectional path.we have to tokenixze each sentence in our data and then to make sure that each sentence has equal length we have to do padding for all the sentences we can also do padding by splitting whole sentences into batches .To get rid of those extra zeroes which may affect the accuracy of the model we can use attention masking.And after then we have to convert each padded sentence into tensors to fit our data to the bert model.After we got word embeddings for each sentences each has 768 features.

MODEL:

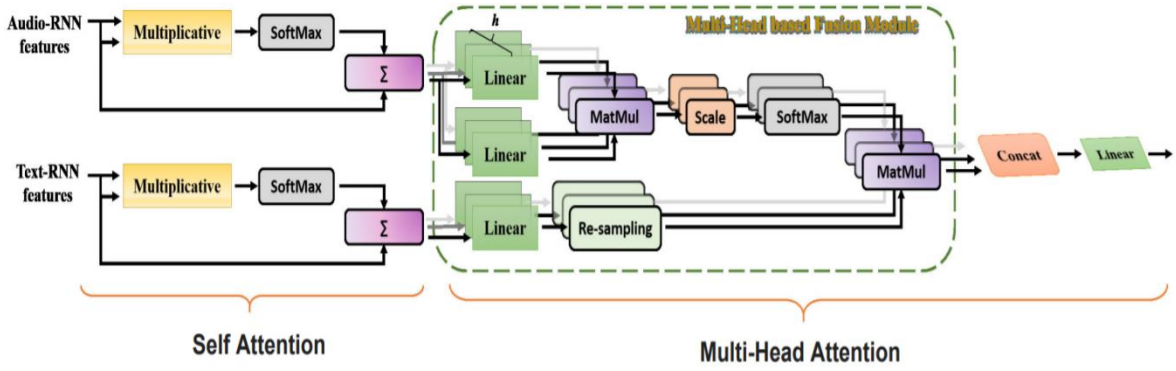


1. BATCH NORMALISATION:

Both audio and textual features are padded to the maximum length of them respectively and then they passed into a batch normalisation separately. we set batch size as 30.the batch normalisation layer helps us to prevent bias for different modalities.

2. GRU module:

The normalized data is fed into a GRU layer which is a variant of a RNN model. The problem of conventional RNN is it suffers with vanishing or explosion gradient descent problem. GRU has a cell containing multiple operations that allows it to carry forward information over many time periods in order to influence a future time period. No.of recurrent layers of GRU in text branch is 192 and no.of layers of GRU in audio branch is 78. We use *L*2 as regularization function with parameter of 1*e*-4. Recurrent dropout is set to 0.3



3. SELF ATTENTION:

This attention computes a representation at different positions of a single sequence for each audio and text RNN-features. First of all, we perform a multiplicative operation $f_{att}(x_t,x_{t'})$ of current state, x_t at current timestamp *t*, over previous states $x_{t'}$ previous timestamp t' calculate the attention alignment. Next, we determine attention scores, a_t , by applying softmax function to $f_{att}(x_t,x_{t'})$.Then, we calculate the context vector, l_t , at position *t* as an average of the previous states weighted with the attention scores a_t . The self attention can be expressed as

$$f_{att}(x_t,x_{t'}) = x_t^T W_a x_{t'} + b_a$$
$$a_t = \text{softmax}(e_t)$$
$$l_t = \sum_{t'} a_{t,t'} x_{t'}$$

where *Wa* and *ba* are weight matrix and bias value to be learned in the attention model.

4. MULTI HEAD ATTENTION:

The multi-head attention in to fuse the attention features from audio and text. Rather than only computing the attention once, the multi-head mechanism runs through the scaled dot-product attention (SDPA) multiple times in parallel. However, the original SDPA requires the same temporal lengths of inputs for computation. In fact, the temporal dimension of audio and text are always different since the length of audio depends on the recording duration while

the length of text is the number of words. To overcome this issue, we first fed the outputs of the self attention to multiple linear transformation modules, then we duplicate the audio branch and apply to them the sequential operations of dotproduct, scaling, and softmax function. On the text branch, we add 'Re-sampling' block to interpolate the text features having the same temporal size with the audio features. Then, we perform another dot-product operation to extract the sympathetic representation between multiple models. The independent attention outputs are simply concatenated and linearly transformed into the expected dimensions. The number of heads is set to 2. Moreover, we add *L*2 function as regularizer of 1*e*-3 for the Multi-Head module.

5. GLOBAL AVERAGE POOLING (GAP):

The Global average pooling layer is added to minimize over fitting problem by reducing the temporal dimensions.

6. FULLY CONNECTED LAYERS:

The reduced vector is passed to two fully connected (FC) layers for scaling and compressing feature-dimension to predict the probabilities of emotional states using 'Softmax' function. Each FC layer contains 80 perceptive units followed by a batch normalization layer and a dropout layer of 0.3 . The activation is fast Gaussian error linear units (fastGELU).

$$\text{fastGELU}(x)=\max(0,\min(1,(1.702*x+1)/2)))*x$$

TRAINING:

Loss function and optimization strategy used in this study is cross entropy and stochastic gradient descent (SGD), respectively. Learning rate is set to $1e^{-3}$ with decay of $1e^{-6}$. We evaluate the proposed model for SER using 10-fold leave-one-speaker-out cross validation, which assign a person in a session as validation while other person of the same session as test and vice verse, and the remaining sessions are used for training. Also, we assume that speaker identity information is not available in our study. The quantitative

measurement of speech emotion recognition is per-sample *Accuracy*. Additionally, other terms such as *Balanced Accuracy*, *Precision*, *Recall*, *F*1 and F_β scores are also calculated for comparison.

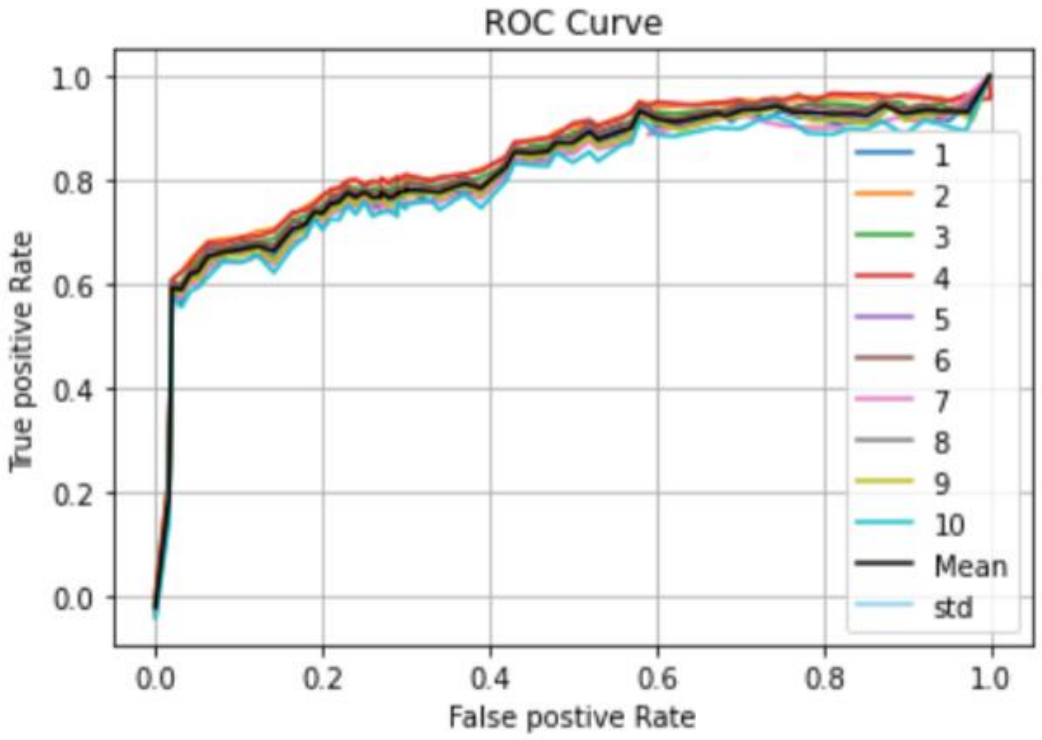
Evaluation Criteria:

Since the number of ‘‘Disgust’’ and ‘‘Other’’ samples in IEMOCAP data set are too small, we drop them and decide two experiments corresponding to two classifiers: four emotions (neutral, angry, sad, and happy/excited).For each classifier, we conduct two scenarios which are Improvised and Mixed. To evaluate the IEMOCAP data set, we configure 10-fold leave-one-speaker-out cross validation method, which assign a person in a session as validation while other person of the same session as test and vice-verse, and the remaining sessions are used for training. Also,we assume that speaker identity information is not available in our study.

Balanced Accuracy is used to deal with imbalanced data set and defined as the average of recall obtained on each class.Precision basically tells us how many positive samples classified by model are actually positive.Recall provides how many true positives are found by model.F1score takes into account both Precision and Recall as we can’t always evaluate them and then take the higher one for our model. It is the harmonic mean of Precision and Recall.Fβ score is used as a evaluation metric to assign different weights to Precision and Recall. Using the notation of true positive (TP), true negative (TN), false positive (TP), and false negative (FN)

Accuracy=TP+TN/(TP+FP+TN+FN)
Accuracy Balanced=1/2*(TP/(TP+FN)+TN/(TN+FP))
Precision=TP/(TP+FP)
Recall=TP/TP+FN
F1=2*Precision*Recall/Precision+Recall
Fβ=(1+β^2)*Precision*Recall(β^2·Precision)+Recall
In this study,β=0.5 was chosen to lend more weight to precision.

Roc curve for Multimodal:



DIFFERENCES:

1. **we have use librosa in place of open smile**
2. **Researchers has done batch wise padding but we have done padding to the whole dataset**

Researcher's Model	Accuracy	Precession	Recall	F-beta
Multi-Modal	73.23+/-4.07	74.26+/-4.22	73.23+/-4.07	73.7+/-4

Our model	Accuracy	Precession	Recall	F-beta
Multi-Modal	65.56+/-2.07	66.34+/-2.52	64.31+/-2.14	66.23+/-1.32