University of Novi Sad
Faculty of Sciences
Department of Mathematics and Informatics

Danijel Lazarević

An evaluation of speech emotion recognition systems based on transformer and long-short term memory architectures

Master's thesis

Mentor:

Dr. Nikša Jakovljević

Novi Sad, 2024.

Table of Contents

1. Introduction

Advances in technology enabled the use of high-performance computers that influenced the further development of artificial intelligence. The development of artificial intelligence made a lot of problems easier to solve, and one such problem is the speech emotion recognition (SER) [1].

The desire for machines to be able to conduct dialogue with humans independently leads to the problem of recognizing emotions in speech. In the beginning, recognizing emotions in speech meant recognizing emotions based on extracting handmade acoustic features from the speech utterances. Now, with the advancement of technology, a deeper analysis is possible. If a machine recognizes a speaker's emotion, which carries a lot of secondary information, then the communication between a speaker and machine can be simpler [2]. Machines are considered as things that cannot understand or express emotions, but it is possible by providing information about emotion in a way that a machine can understand. An interaction between machine and human can be useful in situations where the response of the machine depends on the detected emotion. For example, the emotion detected in the speech can give us information about the mental state of the driver of the car, which will provide information to the system that will take actions if the safety of the driver is endangered. Also, therapists can use the emotion detected by machine, which then represents a diagnostic tool.

According to Paul Ekman [3], emotions are processed, where our nervous system automatically reacts to actions that are considered as important to our welfare, and that reaction is influenced by our evolutionary and personal past. A physical event, social interaction, remembering or imagining event can activate emotions. Paul Ekman's research obtained seven universal facial expressions of emotions and those are anger, contempt, disgust, enjoyment, fear, sadness, and surprise. The emotions mentioned in the previous sentence are the base of the SER model called discrete emotional model. The other widely used model is so-called dimensional model, where emotions are characterized by small number of latent dimensions. The dimensions are valence, arousal, control, and power. One of the most preferred models is a model that uses arousal dimension versus valence dimension, which is illustrated in Figure 1. The arousal provides us information about the strength of the emotion (range is from boredom to excitement) and valence is describing whether the information is positive or negative, and its range is from unpleasant to pleasant [2]. The disadvantage of the discrete model impossibility of defining complex emotions and the disadvantage of the dimensional is that cannot represent each emotion.

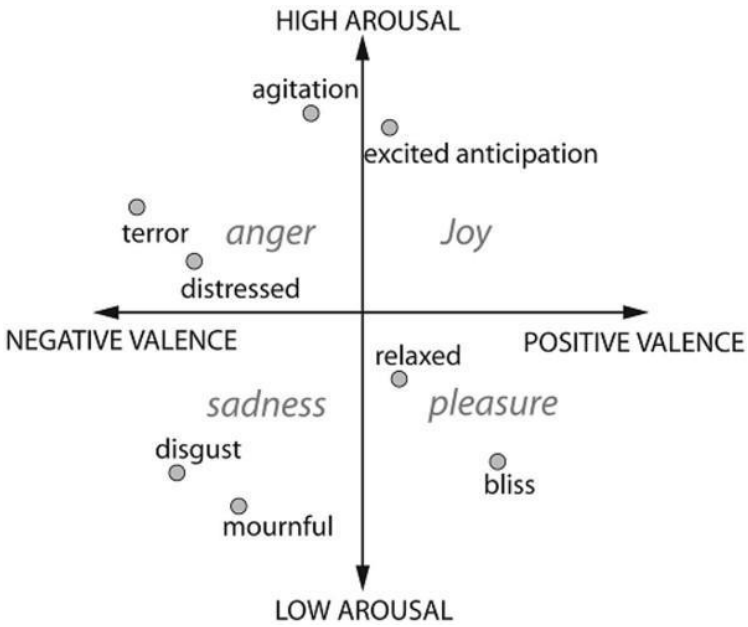


Figure 1. Arousal versus valence in the dimensional model [4]

Processing speech utterance in a SER system can be described as a collection of methodologies whose goal is detecting emotions that are embedded in the speech signals. SER systems should be trained on the labeled data that require preprocessing before feature extraction. The importance of features is extracting the most significant characteristics from the original data, where the irrelevant and redundant information are removed [5].

Figure 2 shows all elements in an SER system, organized from the left to the right to follow the steps in a system development. The first step is to choose the emotional model (dimensional or discrete), and, according to it, selecting the method for induce speech emotion utterances to create a speech dataset. Databases, according to selected emotion induction method, can be classified into following three groups:

- Natural speech database – utterances are obtained from call centers recordings, talk shows and similar sources. Processing and distributing this data make this way of obtaining speech emotion utterances harder, because of the legal and ethical problems.
- Elicited speech database – utterances are obtained by placing the speaker into the simulated emotional situation. Emotions obtained by this method still are not fully-elicited, but they are very close to them.
- Acted speech database – utterances are recorded in sound-proof studios by actors. Even if they are not so close to real emotions, the advantage is easier creation of the database, comparing it with two other methods.

After the data are collected, the preparation of speech utterances for extracting features begins in preprocessing phase, where several techniques are applied in order to remove irrelevant factors, normalize data, etc. The techniques applied in the preprocessing phase are:

- Framing – continuous speech signal is separated into fixed length segments. Using short frames (20ms to 30ms), speech cannot change emotion in that short period, because the signal is almost stationary and local features can be obtained. Also, by allowing overlapping of these segments, information and relation between frames are obtained.
- Windowing – the window function is applied to each frame. Usually, in signal processing literature framing is considered as windowing with rectangular window.
- Voice activity detection – for easier speech modeling it is necessary to remove non-voice and noise frames, which has as a result complexity decrease and accuracy increase.
- Normalization – the goal of a feature normalization is speaker and recording variability reduction (without loss of feature's discriminative strength), which helps in increasing generalization ability of features.
- Noise reduction – emotion recognition rate is affected by the noise and that is why it should be removed or reduced. Usually, it is done by minimum mean square error and log-spectrogram amplitude estimators.
- Feature selection and dimension reduction – is a very important step in reduction of the computational costs and increasing accuracy. It is a selection of a relevant and useful subset of the extracted features.

Further processing extracts local and global features from the preprocessed data, and in this way prepare data for a classifier [5]. These features can be organized in:

- Prosodic features – are related to the changes of fundamental frequency and energy in time in phonetic units such as syllables, words, phrases, and sentences plus phoneme durations. They are dealing with large unit (syllables, words, phrases, sentences) properties, thus these features are so called long-term features.
- Spectral features – contain implicit information about the shape of the vocal tract i.e. about a produced phoneme. Most commonly used spectral features are Log-mel spectrograms and Mel Frequency Cepstral Coefficients (MFCCs).
- Voice Quality (VQ) features – are related to physical properties that describe auditory coloring of individual's voice using terms such as warm, shrill, twangy, creaky, shrieky, breathy, yawny, gravelly, hoarse, ringing, dull, nasal, resonant, rough, etc. VQ is primarily driven by laryngeal and supra-laryngeal characteristics of speaker.
- Teager Energy Operator (TEO) based features – treat speech signal as amplitude-frequency modulated signal in order to model non-linear speech production mechanism. TEO captures high-frequency components of speech signal related to fast transition which are related with speaker emotional state.

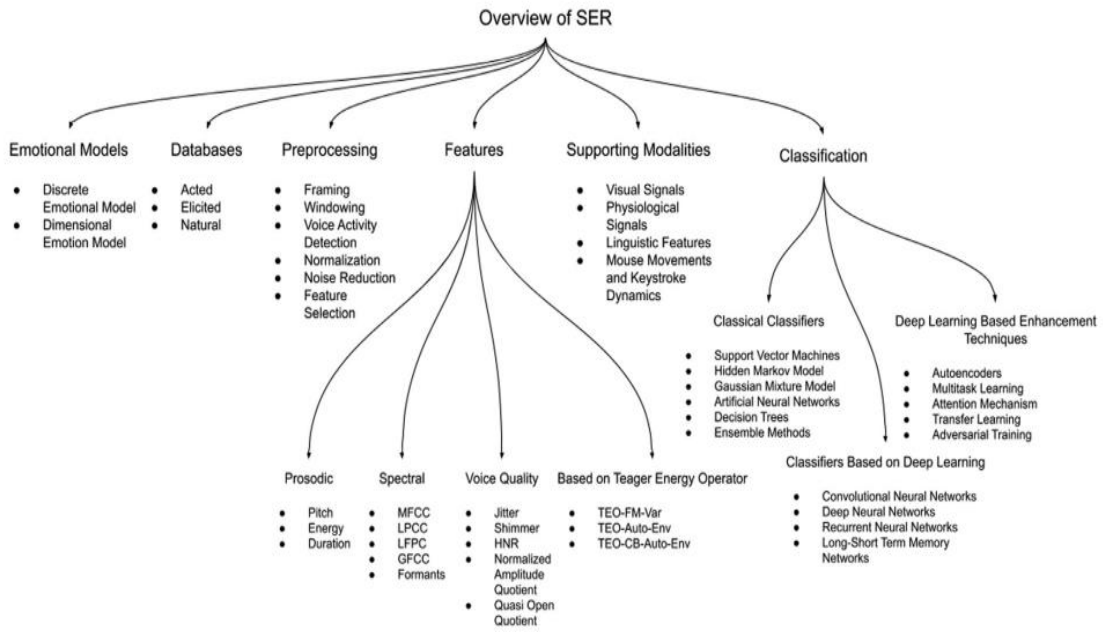


Figure 2. Overview of SER [5]

Speech emotion recognition problem is a complex task because there are so many factors that can affect an emotion. It is unknown which speech features are the most important in distinguishing emotions. To reduce the influence of irrelevant factors, which can be considered as constant in an utterance, deltas and delta-deltas features are used. An additional problem is the detection of more than one emotion in a single utterance, because each of emotions in the single utterance fits to different proportion of the utterance and it is not easy to determine bounds between those proportions. A very important thing in expressing emotions is a speaker cultural and environmental background, which is usually unknown, but it significantly affects emotion recognition [1].

As a classifier a wide range of models have been used such as Gaussian mixture models, hidden Markov models, support vector machines, decision trees, and in recent years Deep Neural Networks (DNN). Currently, DNNs are the most dominant approach, because of their huge success in image and speech recognition. High number of hidden layers allows network to extract features more adapted to a particular classification task.

The goal of this study is to evaluate the SER systems based on DNNs. Two DNN architecture were evaluated, one based on Long-Short Term Memory (LSTM) and the other based on transformers. Training, validation, and testing were carried out on two publicly available language databases: IEMOCAP (Interactive Emotional dyadic Motion Capture in English) and SEAC (Serbian Emotional Amateur Cellphone speech corpus). Additional goals are to explore the possibility of using a system trained on one language to recognize emotions in another language, as well as the possibility of transferring learned emotions across different languages.

Log-mel spectrogram, and its delta and delta-delta are used as input to our first SER model which is based on 3D Convolutional Recurrent Neural Network with Attention

Model (ACRNN). It was selected because it showed high accuracy and effective capture of information important for SER [1]. The input data are forwarded to six convolutional layers, where high-level features are being extracted. Then, the output of convolutional layers is taken by a bidirectional recurrent neural network with LSTM which produces 256-dimensional high-level feature representations. The sequence of high-level feature representations sent to attention layer. Attention layer eliminates emotion irrelevant and silent frames, thus it focuses only on relevant frames and its outputs are utterance-level features for SER [1].

The second use model is Global Aware Multi-Scale (GLAM) Neural Network, which uses Mel-frequency cepstral coefficients (MFCCs) as input features. The idea of this network is fixing lack of Convolutional Neural Networks and replacing Long Short-Term Memory and Attention Layers [6]. In GLAM model, the input MFCC is processed by spatial convolution and by temporal convolution, which are concatenated in the next layer. After concatenation, the extracted features are sent as input to Multi-Scale Block plus Max Pooling layer, two times consecutively and then, again, to the one Multi-Scale Block whose output goes to Convolutional Layer. The next step is to process the output of the Convolutional Layer in the Global-Aware Block, whose output is fed to Fully Connected Layer, which returns the emotion category [6].

The data used in this study were SEAC and IEMOCAP datasets. SEAC dataset is a dataset in Serbian that consists of male's and female's speech utterances and it contains fear, happy, sad, neutral and angry emotions [7]. IEMOCAP is a dataset in English, recorded across 5 sessions with 5 pairs of speakers, where each pair consists of a male and female speaker, where they provided angry, excited, fear, surprised, frustrated, happy, disappointed, and neutral emotions [8].

2. Related work

Speech is one of the most natural media of people communication and it is not just important because of the explicit linguistic information it carries, but also, the implicit information such as speaker's emotion [9]. That is why speech emotion recognition became a subject of interest of a lot of researchers around the world. In this section you can find a short review of the latest research in Speech Emotion Recognition (SER). The first approach used Convolutional Neural Network (CNN) and all the following studies are its gradual improvement, because each of them has something what the previous one lacked.

In [9] the idea of authors came from the fact that the images are represented by 3 dimensions, red, green and blue color channels, therefore they also used 3D input, but made of log-mels, delta and delta-delta features. Such 3D input is fed in Deep Convolutional Neural Network (DCNN), which showed good results in visual problems like object detection, thus in [10] authors explore whether DCNN can be used in the SER task. In this case, authors used DCNN as a tool for extracting segment-level features. Also, they used discriminant temporal pyramid matching to form a global feature and then it is classified by a linear Support Vector Machine (SVM). Authors in [10] concluded that this model [9] lacks the ability to perform continuous dimensional emotion recognition, therefore they thought adding long short-term memory (LSTM) to this model could help. Additionally, authors in [10] explored DCNN in combination with LSTM in an end-to-end approach. It means that the raw audio signals are fed directly in the deep learning model and the output is emotion class. Approach in [10] obtained better results than traditional designed features on the RECOLA database. As a result of this study, authors found out high correlation between gate activation and prosodic features (energy, loudness, F0) in the cells of the recurrent layers.

Additional study used LSTM is [11], where the attention mechanism based on bidirectional LSTM were investigated. Huang and Narayanan investigated an attention mechanism based on bidirectional long short-term memory (BLSTM). Such approach gave very good results in this SER problem, where the attention layer plays a big role because it can help us to consider more important frames and to not pay a lot attention to less important frames. The weighted accuracy of the model that used attention layer was and in the case of the model without attention layer.

One more research where attention mechanism was used in Speech Emotion Recognition task is [12]. This research is based on the hypothesis that if we are able to understand the internal configuration within an utterance then that will lead to reducing misclassification. Convolutional attention mechanism is integrated in the model, because it showed ability to utilize the context information. Also, CNN showed ability to exhibit some level of robustness to noise, that LSTM cannot handle.

3. Methodologies

In this section two SER architectures will be explored. The first one is based on the Long Short-Term Memory (LSTM) and attention layer, and in 3.1, the complete architecture will be described together with the input features. The same will be described in 3.2, but there will be word about the architecture without LSTM, in this case, transformers based architecture.

3.1. 3D Convolution with Attention Layer

In this section we will describe the input features that we are processing in the network and architecture of the model whose output is class of the emotion. As we have already mentioned above, the input is three dimensional, it consists of static, delta and delta-delta spectrograms. Our model is 3-dimensional Convolutional Recurrent Neural Network with Attention Model (ACRNN) and in this section you will find out what it consists of [1].

3.1.1. Input features

In speech processing tasks, spectrograms are common, because they partially simulate mechanisms in the basilar membrane of mammalian auditory system, i.e. analyze current frequency content of an audio signal. Input is three dimensional, consists of static, which is the Mel spectrogram, and its delta and delta-delta. Within the spectrograms, the horizontal and vertical axes, along with color intensity, correspond to the time axis, frequency axis, and amplitude of periodic component at a specific frequency and moment in time, respectively [13].

Prior to calculating spectrograms for speech signals, speech signals are z-score normalized (creating zero mean and unit variance) to reduce possible variations between different speakers [1]. The first step is to pass through the dataset and get parameters for the standardization. Then, pre-emphasis is applied on the input signal, which removes effects of lip radiation (by emphasizing high frequency components) [14].

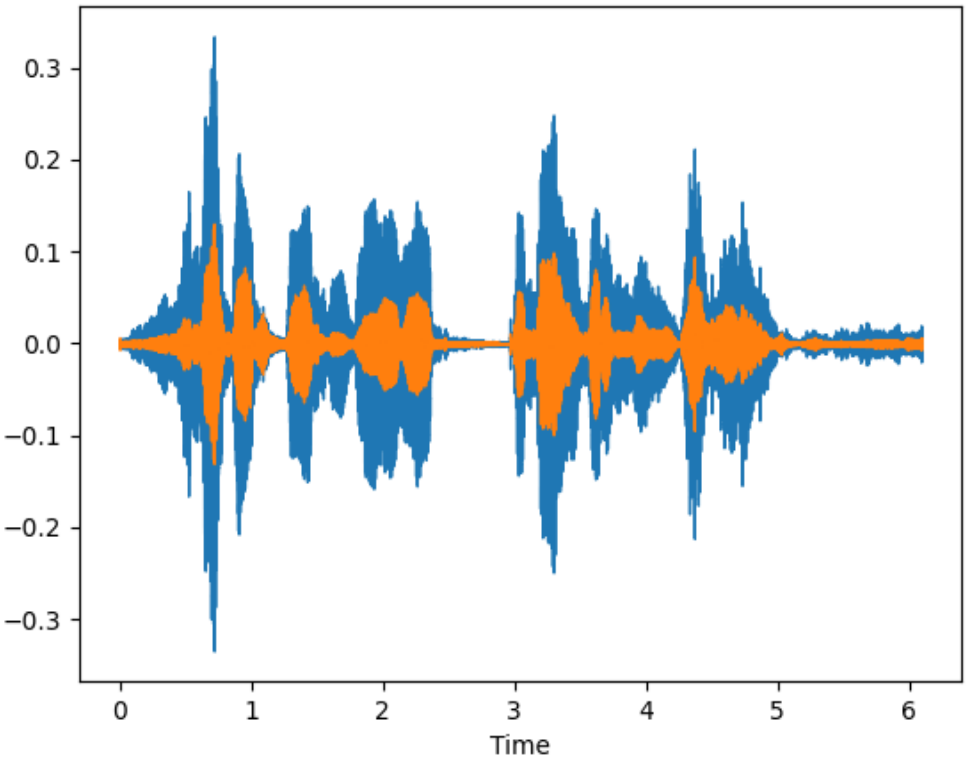


Figure 3. Waveforms of original input signal (blue line) and pre-emphasized signal (orange).

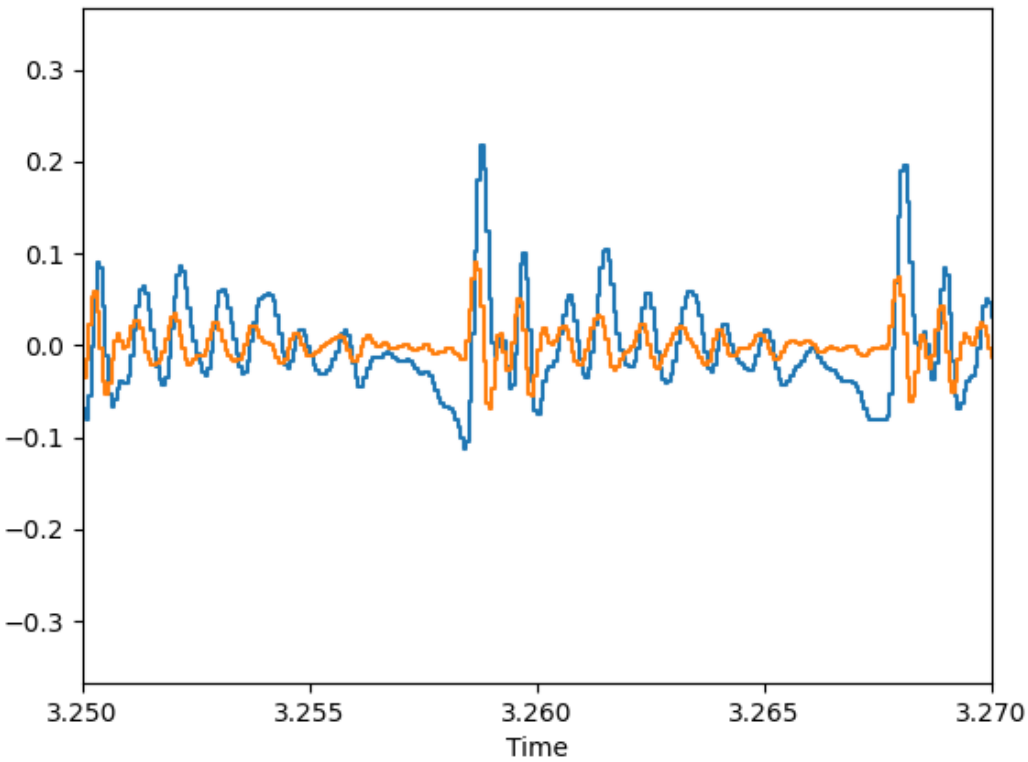


Figure 4. Waveform of signals in Fig.3 zoomed at interval $[3.250, 3.270]$ to detect changes in frequency content caused by pre-emphasis.

The next step is splitting the signal into short-time frames to capture the changes of frequency content over time, which contains information what is spoken and how it is spoken. Calculating Discrete Fourier Transform (DFT) on the whole signal, it averages energy of particular frequencies over time leading to the information loss. On the other hand, splitting signal into short-time stationary frames overcomes the problem of the information loss, but introduce low frequency resolution and spectral leakage [15]. To reduce spectral leakage different window functions are used, whose values are decreasing towards its edges. Such shape of window functions leads to reducing influence of signal samples closer to the edges on short-time spectrum, thus overlapping between successive frames is applied. In this experimental setup Hamming window length of 25 ms with shift equal to 10 ms is used. Hamming window is defined by the following function:

$$w[n] = 0.54 - 0.46\, cos\left(\frac{2\pi n}{N-1}\right) \qquad (1)$$

for $0 \leq n \leq N-1, N\ is$ the length of the window. All these steps above are preparation for DFT. DFT is selecting finite number of samples from the continuous signal, but the input signal is digital, it is not continuous and that can cause spectrum leakage, so, all steps above are taken to minimize that leakage. The number of samples in the frame is defined by a sample rate (16 kHz in this case) and the frame length. The process of framing is

described in the first section and for each window is obtained one small spectrum. By merging all local spectrums into one, Figure 5. is obtained [14,16].

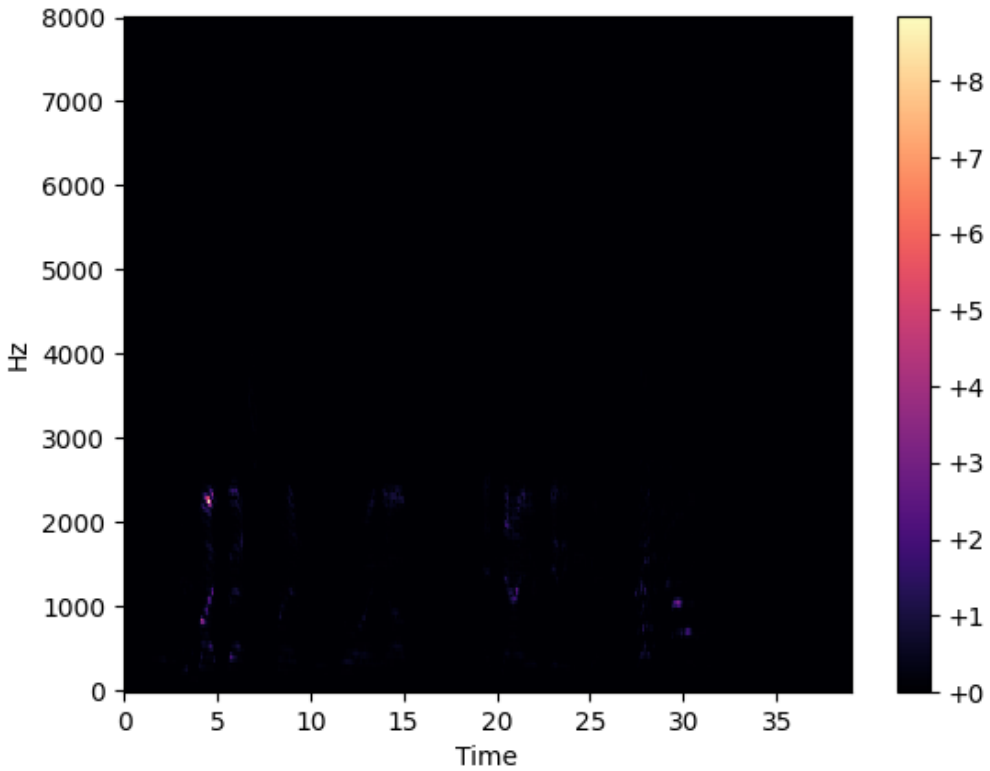


Figure 5. Spectrogram obtained after DFT

To obtain the mel-spectrogram from DFT, mel-filter banks are used. There are M (in this case M=40) triangular filters dividing mel-frequency into equal width ranges overlapped by 50% as presented in Figure 6. Weighted average energy (or amplitude) for each filter in the filter bank is transformed into decibel. The transformation from Hz to mel is done by the following equation:

$$B(f) = 2595\left(1 + \frac{f}{700}\right) \tag{2}$$

where f is the linear frequency. The triangular filter $m, m \in \{1,2,\dots,M\}$, is defined by the following equation:

$$H_m[k] = \begin{cases} 0, k < f[m-1] \\ \dfrac{2(k - f[m-1])}{(f[m+1] - f[m-1])(f[m] - f[m-1])}, f[m-1] \le k \le f[m] \\ \dfrac{2(f[m+1] - k)}{(f[m+1] - f[m-1])(f[m+1] - f[m])}, f[m] \le k \le f[m+1] \\ 0, k > f[m+1] \end{cases} \quad (3)$$

Now, let us denote f_l - lowest and f_h - highest frequencies of the filter bank in Hz, F_s – sampling frequency, M – number of filters and N – size of the DFT. Then uniformly spaced boundary points $f[m]$, in the mel-scale, are given by:

$$f[m] = \left(\frac{N}{F_s}\right) B^{-1}\left(B(f_l) + m\frac{B(f_h) - B(f_l)}{M+1}\right) \quad (4)$$

where B^{-1} is

$$B^{-1}(b) = 700\left(10^{b/2595} - 1\right) \quad (5)$$

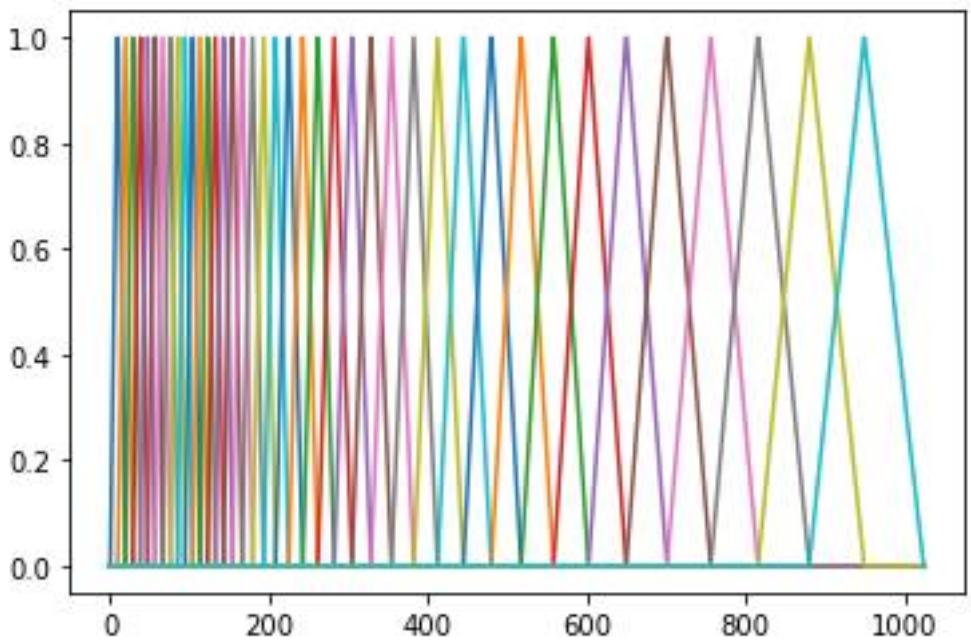


Figure 6. 40 triangle mel filters

Finally, the log-mel spectrogram is obtained by following equation

$$S_m = ln\left[\sum_{k=0}^{N-1} |X_a[k]|^2 H_m[k]\right] \quad (6)$$

where $0 \leq m < M$ [17].

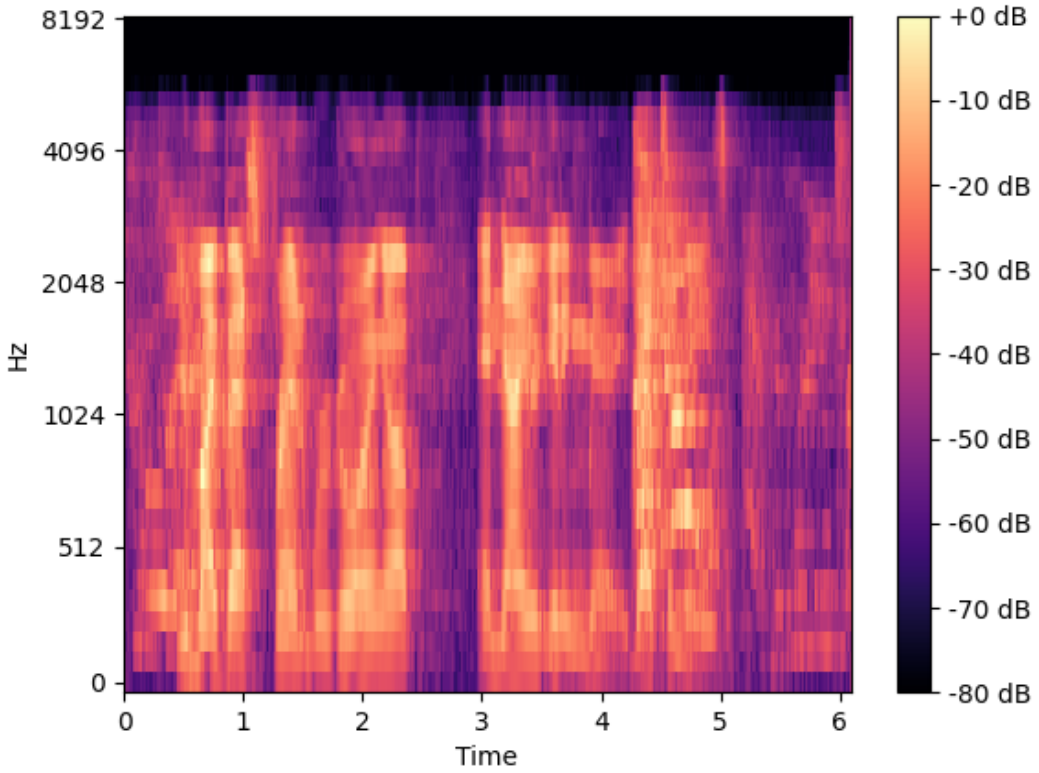


Figure 7. Log-mel spectrogram

To get information about energy changes between frames the delta and delta-delta features are used, which can be observed as approximations of the first and second derivative of the signal, respectively. These two features are adequate for machine learning algorithms because it is easy to calculate them [18]. Here is the expression for the delta feature of feature S_m:

$$\Delta_m = S_m - S_{m-1} \quad (7)$$

and for its delta-delta feature:

$$\Delta\Delta_m = \Delta_m - \Delta_{m-1} \quad (8)$$

As illustration in Figure 8 and 9 are shown delta and delta-delta features of the spectrogram in Figure 7. It can be seen that delta features emphasize changes in spectrum, and delta-delta features represent speed of that change.

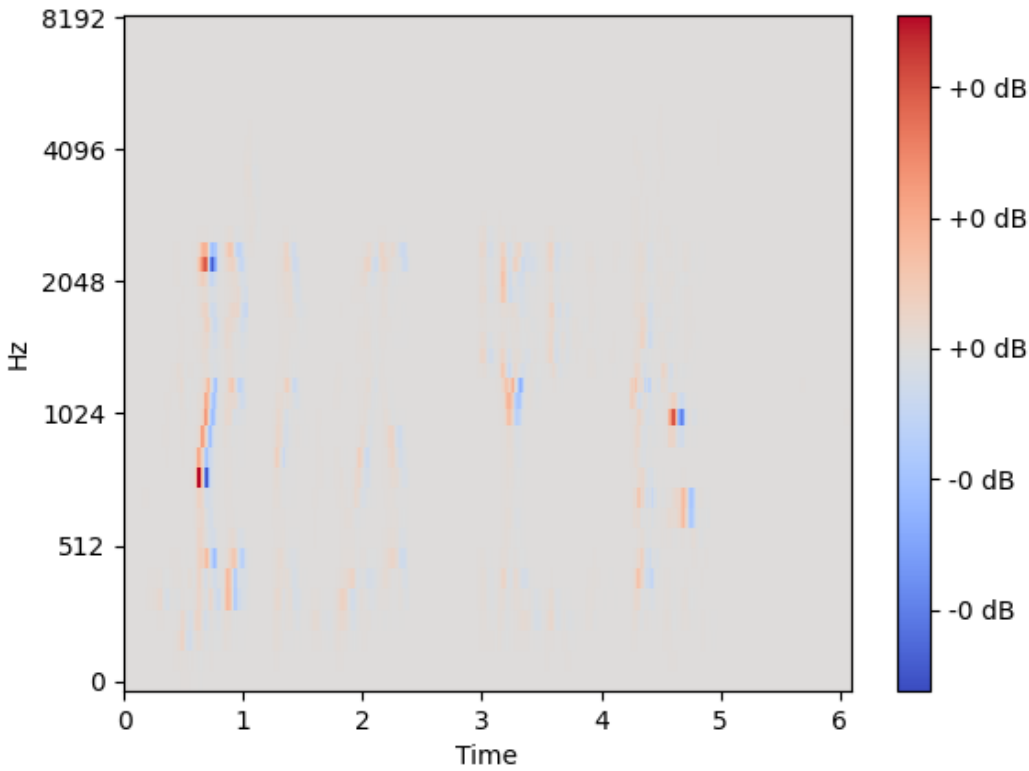


Figure 8. Delta feature

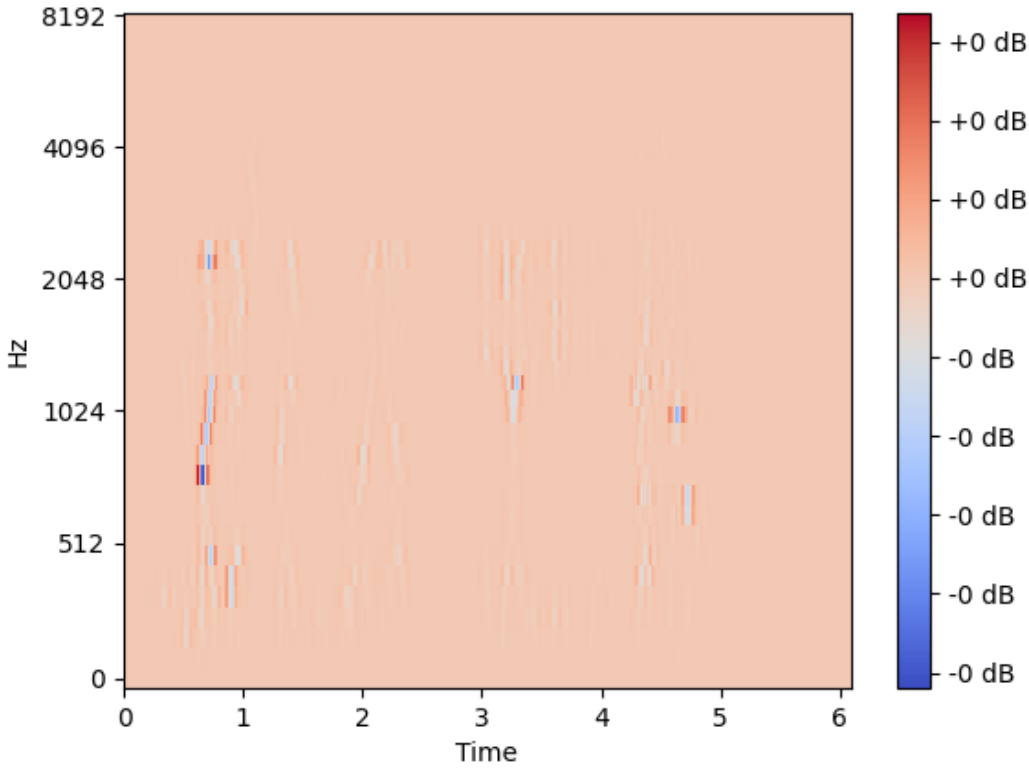


Figure 9. Delta-delta feature

3.1.2. Model architecture

In the model, the 3D input, that is described above, is processed to get some meaningful information that will be used in LSTM layer and then it is sent to final layers of the model to get the emotion for the given input. The first approach architecture consists of convolutional, max pooling, LSTM, attention and fully connected layers (Figure 10) and which will be described in this section [1].

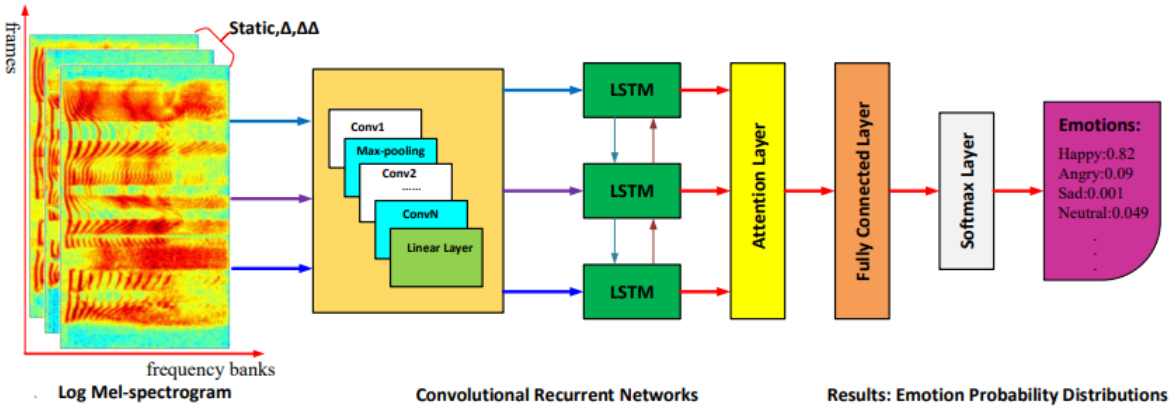


Figure 10. 3D Convolutional with Attention Layer architecture [1]

3.1.2.1. Convolutional layer

The base of the convolutional layers is the operation called convolution, which is represented by the following equation:

$$(x * w)[r,s] = \sum_{i=-h_1}^{h_1} \sum_{j=-h_2}^{h_2} x[r+i, s+j]w[i,j] \tag{9}$$

which represent the pixel on the position $[r,s]$ in the next layer, where $\boldsymbol{x}$ is the input image, $\boldsymbol{w}$ is the filter whose dimension is $[2h_1+1, 2h_2+1]$. How this operation will be performed in border parts of the input image is defined by setting padding parameter [19].

These filters are also referred as kernels. As shown in Figure 11, the selected pixel in the convolutional image is calculated as scalar product of the kernel and the corresponding pixels in the original image, then the kernel is moved for every pixel in the input and scalar product is calculated for each value in the kernel. The network will know which kernel gives us a specific feature at the given position on the image and we usually call them activations [20]. In this way P images are created, where P is the number of kernels that will be used in the convolution, where each of them will be sent to the pooling layer and that is how information is extracted from the input images. Convolutional neural networks usually extract information using multiple convolutional layers [21].

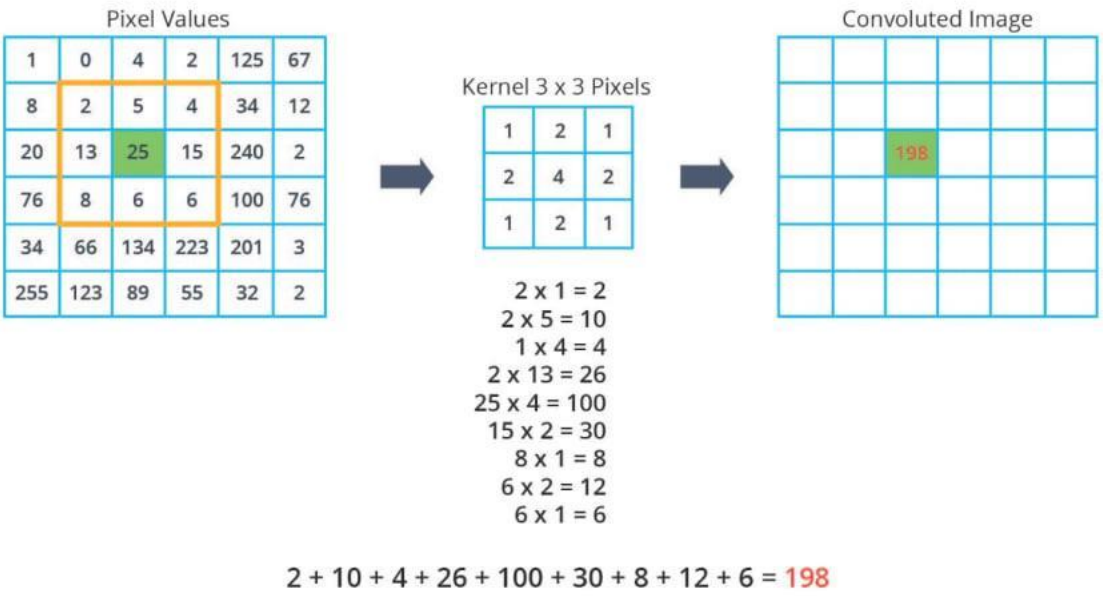


Figure 11. Convolution applied on image [22]

In this case, 3D convolution is used, i.e. 2D convolution described above is applied on each channel independently (spectrogram, delta and delta-delta). There are 6 convolutional layers in this architecture, where the first has have kernel size 5×3, the number of filters is 128 and in the others the kernel is the same size 5×3, but the number of filters is 256. The number 5 represents the size of kernel that corresponds to the time axis and 3 is the frequency axis. Stride, the parameter that defines the size of the kernel shift in number of pixels, is set to be 1 [1].

3.1.2.2. Max pooling layer

Pooling layers are important for several reasons. One of them is down-sampling features, it entails a reduction in the number of calculations in the network. More important, it overcomes the problem of location-dependency of the feature map obtained from convolutional layer. Location-dependent means that some object will be recognized only in that position and it should be recognized even if the object is a little bit shifted. Here is used max pooling after the first convolutional layer, which takes the highest value in the kernel of the size 2×2 [1, 23]. Pooling operations are visualized in Figure 12.

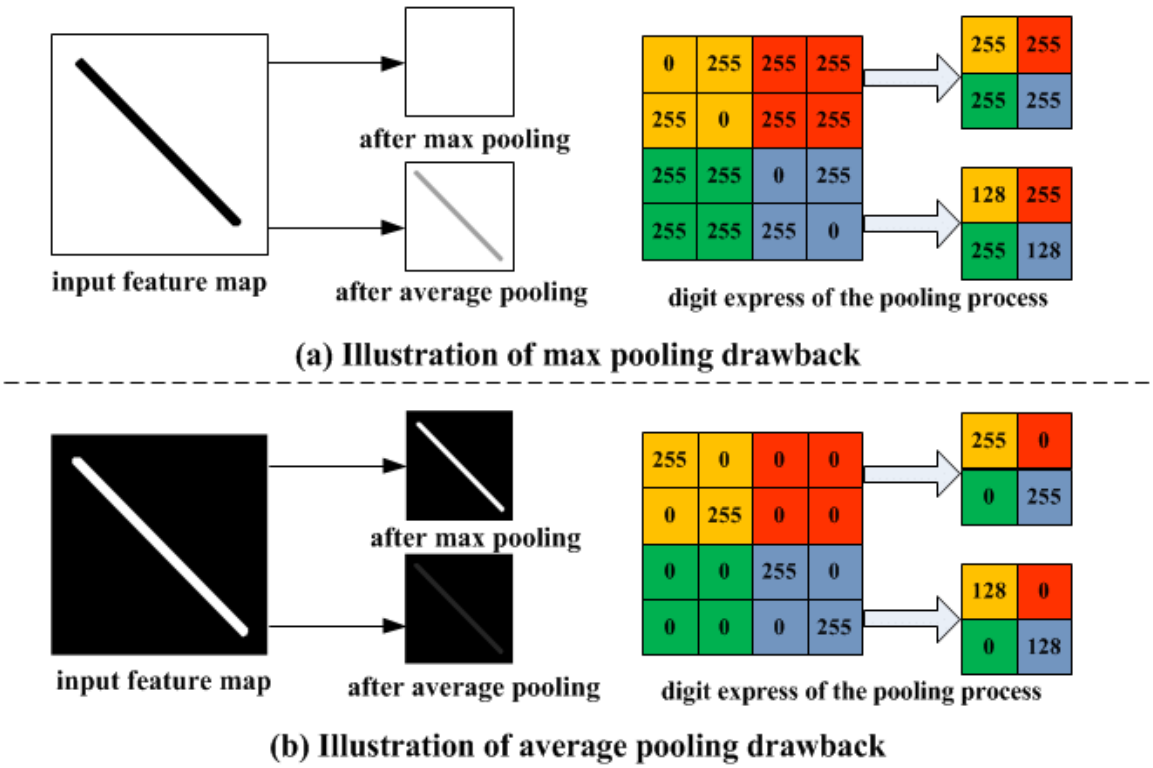


Figure 12. Performing max and average pooling [23]

3.1.2.3. Fully connected layer

The significant reduction of parameters, without producing any damage in accuracy, is possible by inserting fully connected layer bottleneck after CNN [24].

What happens in the fully connected layer is described by the following equation:

$$y(x) = f(Wx + b) \qquad (10)$$

where x is the input vector, W is the matrix of weights and b is the bias, and the latter two are learning parameters in this layer, f is an activation function, which is on our case Leaky Rectified Linear Unit (Leaky ReLU), and y is the output of the fully connected layer. In this way, every neuron from the input is connected to every neuron in the output [25], as shown in Figure 13.

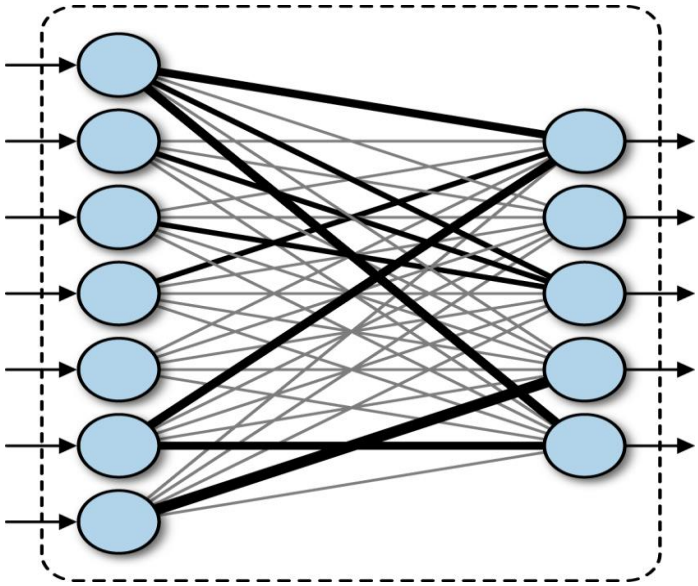


Figure 13. Fully connected layer [26]

Activation function determines whether a neuron should be activated or not in specific situation, i.e. it introduces non-linearities in the model. In this way, the activation function is helping model to separate linearly non-separable classes [27].

$$f(x) = \begin{cases} x, x > 0 \\ 0.01x, x \leq 0 \end{cases} \tag{11}$$

Leaky ReLU is the version of ReLU activation function (Figure 14), where the problem of the dying neurons is solved. The problem with ReLU is that it can turn off some neurons, which stay turned off for every next input and the leaky ReLU fixed that by allowing some small values to leak in those neurons [28]. The leaky ReLU works according to the next formula

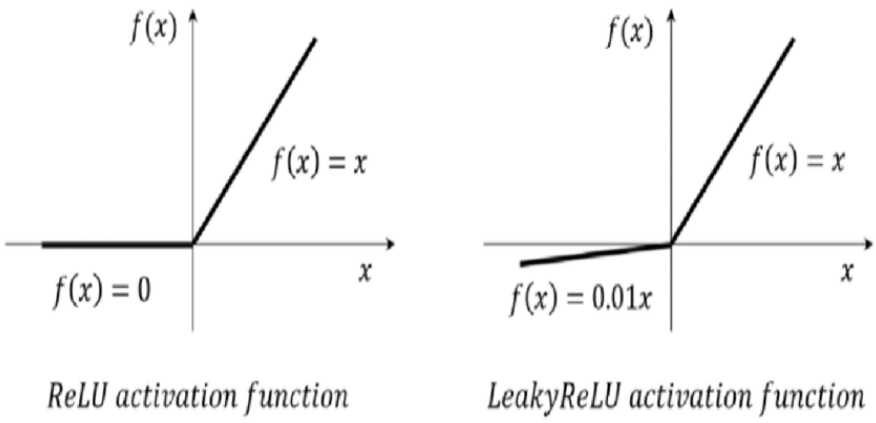


Figure 14. Graphs of ReLU and Leaky ReLU [29]

For the network, such as ours, the adequate number of units in the linear layer, in order to reduce the number of parameters without decreasing in accuracy, is equal to 768 units [1].

3.1.2.4. Bidirectional Long Short-Term Memory (BLSTM)

LSTM is a type of gated Recurrent Neural Network (RNN), which is popular in the speech processing tasks, because of its ability to deal with time contexts, and, also, long-term speech modulations can be captured by it. The vanilla RNNs have a problem with exploding and vanishing gradient during their training. This problem is overcome by using gates in RNN. In this way, it is possible to control how the information is passing through several time-steps, and the gradient can avoid some paths to overcome vanishing or exploding gradient [30].

Unrolled structure of LSTM layer is shown in Figure 15. A LSTM layer consists of three gates, i.e.: input, forget and output gate, which are outputs of sigmoid functions (σ in Figure 14) to filter the input information using those gates. First, the input x_t is combined with the output of the hidden layer h_{t-1}, where the sigmoid function, which gives us the values between *0* (nothing will pass through) and *1* (everything will pass through), gives output f_t representing the forget gate (see Figure 16) [31].

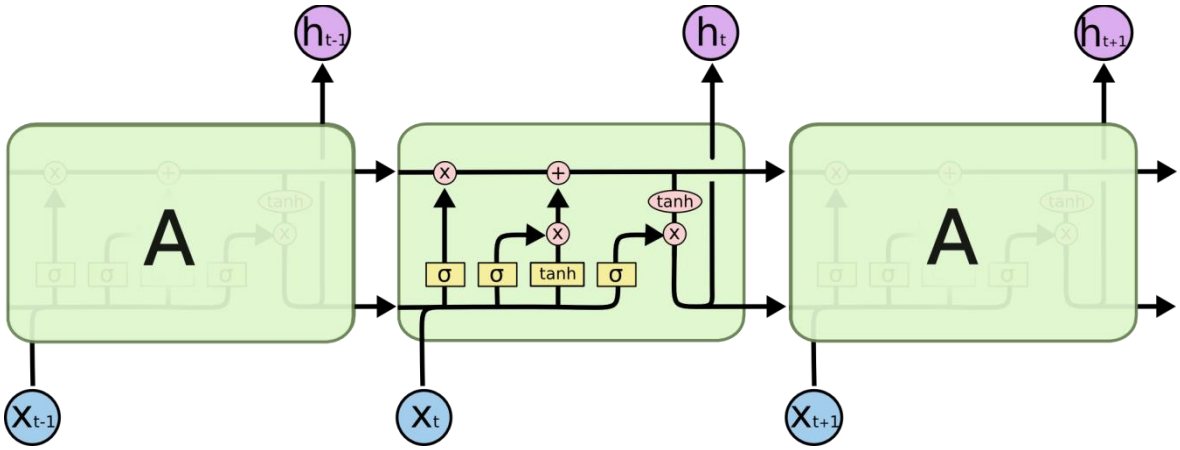
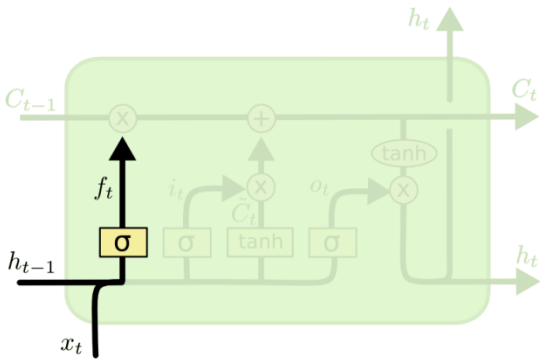


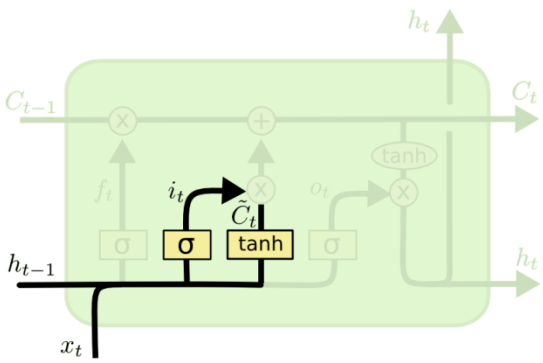
Figure 15. Part of LSTM for three units of the input vector x [31]



$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \; + \; b_f\right)$$

Figure 16. Forget gate [32]

The next step is to process $[h_{t-1}, x_t]$ in the input gate, to controls the amount of candidate values $\tilde{C}_t$ obtained as output of tanh of linear transformation of $[h_{t-1}, x_t]$ as shown in Figure 17. [31].

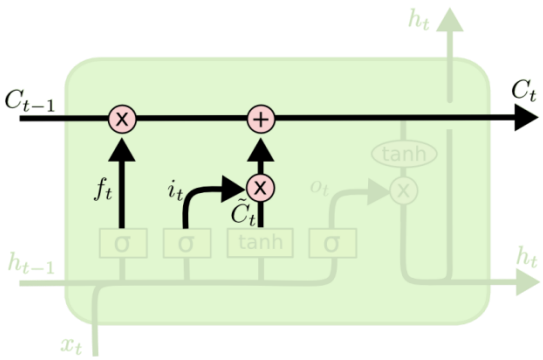


$$i_t = \sigma\,(W_i \cdot [h_{t-1}, x_t] \;+\; b_i)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \;+\; b_C)$$

Figure 17. Input gate [26]

Above calculated values $(f_t, i_t, \tilde{C}_t)$are used to update cell state C_t, which model wider context, as shown in Figure 18 [32].



$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

Figure 18. Update the old cell state [32]

The new calculated cell state C_t processed by the hyperbolic tangent and output gate controlled by $[h_{t-1}, x_t]$ are used to update the hidden state h_t, as shown in Figure 19. It should be noted that the sigmoid output range is [0, 1], thus it is used as gates (which decides amount of influence of a particular feature). On the other hand, hyperbolic tangent maps the cell state in the range from -1 to 1, therefore it is used to create new vectors h_t, and $\tilde{C}_t$ [31].

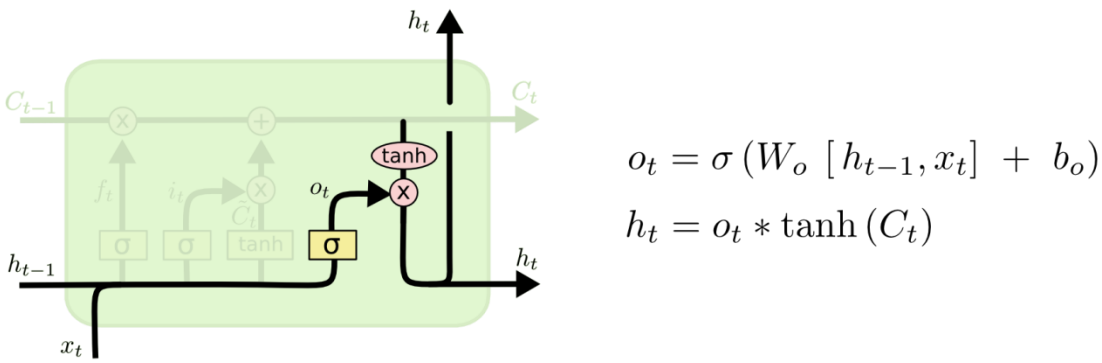


Figure 19. The output gate [31]

BLSTM consists of two LSTM layers that allow us processing information in the forward and backward direction, as shown in Figure 20. Because it is able to process the data in both ways, BLSTM is very good in analyzing the relations between elements in the whole sequence [32].

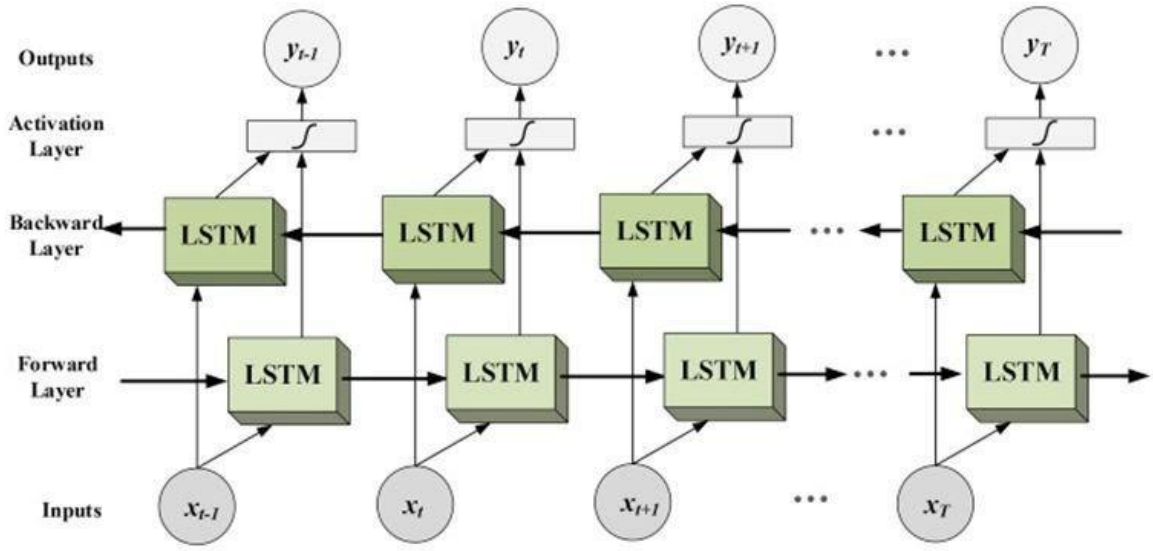


Figure 20. Bidirectional Long Short-Term Memory [33]

In our model, the 3D CNN features are sent to BLSTM, where each direction of BLSTM consists of 128 neurons. The output of BLSTM is 256-dimensional high-level feature, which is fed into the Attention Layer [1].

3.1.2.5. Attention Layer

The recurrent neural networks with the Attention Layer are present in the area of speech processing. The attention mechanism showed good results in decoding the context using the relevant encoding context vectors and reducing the influence of the irrelevant vectors [11].

As not every frame in the LSTM output holds equal significance in making decisions about emotion, these less important frames corresponding to silence should receive lower scores, while the more important frames should be assigned higher scores in this attention layer. [1].

The input into the Attention Layer is LSTM output $h_t = [\overrightarrow{h_t}, \overleftarrow{h_t}]$, where $\overleftarrow{h_t}$ represents the hidden backward layer at the time step t and $\overrightarrow{h_t}$ represents the hidden forward layer at the time step t. From such the input, the normalized weights α_t, which denote the importance of the frame t, are calculated by the softmax function

$$\alpha_t = \frac{W \cdot h_t}{\sum_{\tau=1}^{T} exp(W \cdot h_\tau)} \tag{12}$$

$$c = \sum_{t=1}^{T} \alpha_t h_t \tag{13}$$

where c represents utterance-level representation and each output from BLSTM is scaled according to its importance, more important frame is multiplied by the greater value [1]. The process is visualized in Figure 21.

The high-level representations are obtained by sending the utterance-level representation c to the fully connected layer with 64 units and that showed as helpful to the softmax, thus it can better map these representations to N classes. Also, to obtain faster training and better performance the batch is normalized after the fully connected layer [1].

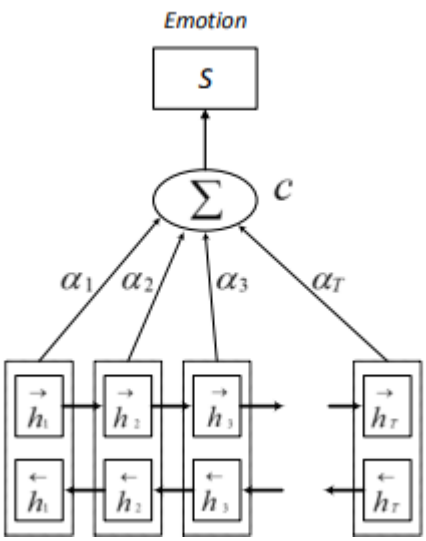


Figure 21. Attention layer [1]

3.1.2.6. Batch Normalization

The batch normalization is a normalization process, which result is faster training of Deep Neural Network (DNN). It is applied after or before nonlinearity. The batch normalization has two parameters γ, β that are trainable and the first one allows us adjusting the standard deviation and the second one adjusting the bias and shifting the

curve of the nonlinearity. The output of the batch normalization, whose input is activation vector $x^{(i)}$ obtained from the previous layer, is y^*

$$y^* = \gamma x_{norm}^{(i)} + \beta \tag{14}$$

where

$$\mu = \frac{1}{n}\sum_i x^{(i)} \tag{15}$$

$$\sigma^2 = \frac{1}{n}\sum_i \left(x^{(i)} - \mu\right)^2 \tag{16}$$

$$x_{norm}^{(i)} = \frac{x^{(i)} - \mu}{\sqrt{\sigma^2 + \epsilon}} \tag{17}$$

and ϵ is a constant, which represents numerical stability [34].

3.2. Global-Aware Multi-Scale Neural Network

The section 3.2. is the place where transformer based architecture and its input features will be presented. In the section, first, it is explained how input features are obtained. The detailed presentation of the architecture is shown in 3.2.2, and there will be word about Multi-Scale Block and Global-Aware Block.

3.2.1. Input features

The input in the GLAM model is mel-frequency cepstral coefficients (MFCCs). MFCCs are obtained by further processing of log-Mel spectrogram. After obtaining the log-Mel, the next step is to apply Discrete Cosine Transform (DCT), which is used to decorrelate highly correlated filter bank coefficients. The DCT yields compressed representations of the filter bank. The cepstrum is obtained by:

$$C\big(x(t)\big) = F^{-1}|log\ (|F[x(t)]|)\ | \qquad (18)$$

where $x(t)$ is input signal, F is DFT [14, 35]. As illustration, in Figure 22, 40 MFCCs for the signal whose mel-spectrum is presented in Figure 7. It is evident that the lower MFCCs are more prominent compared to the others. The lower MFCCs correspond to the slow changes in the envelope of mel-spectrum, that contains information about spoken phonemes.

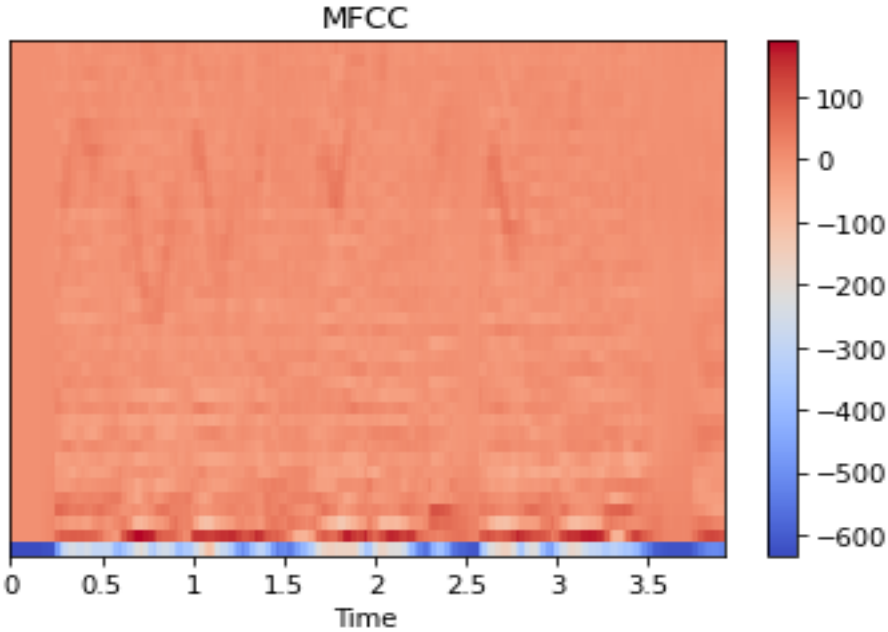


Figure 22. MFCC

3.2.2. Model architecture

In this section, the focus is on demonstrating how global emotion information can be extracted without LSTM. The proposed model used several convolutional layers and multi scales of the processed input MFCC [6].

As you can see in Figure 23, the GLAM model consists of several Max Pooling Layers, Convolutional Layers and Fully Connected Layers, which are already explained in 3.1, thus only new blocks (Multi-Scale and Global-Aware Blocks) will be explained here.

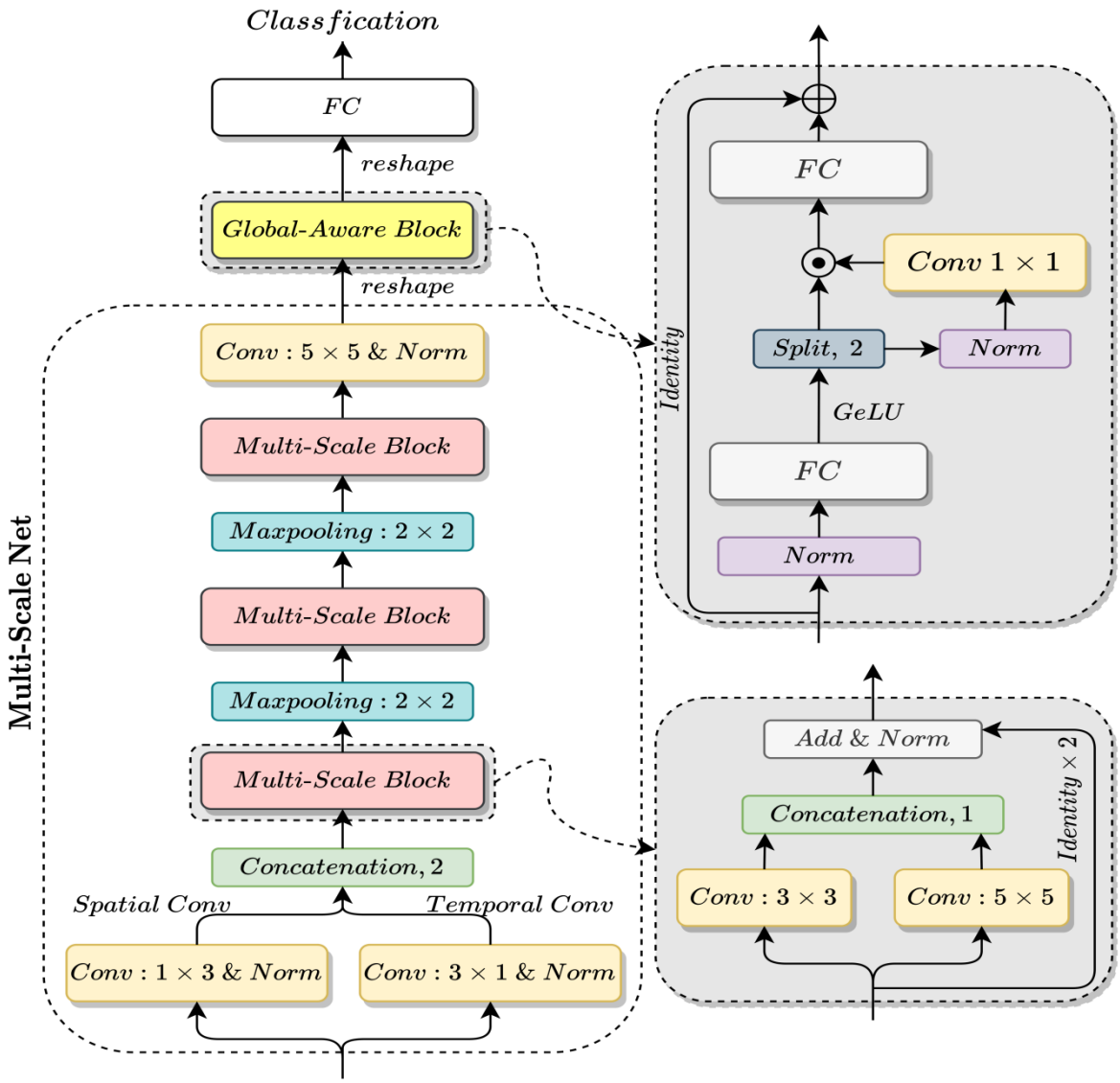


Figure 23. GLAM model [6]

Before the data arrive to the first Multi-Scale Block, they are processed with the 1×3 Convolution for obtaining better spatial features and 3×1 Convolution for better temporal features. After each Convolution Batch Normalization is applied. After Batch Normalization, both outputs have 16 channels and they are concatenated along spatial dimension and this is sent to Multi-Scale Block [6].

In the first part of the GLAM model, there are 3 Multi-Scale Blocks, first two are followed by Max Pooling and third by 5×5 Convolution and the Batch Normalization. Reshaping the output from the final Batch Normalization layer prepares the data for processing in the Global-Aware Block. The output of the Global-Aware Block is directed to the Fully Connected layer, yielding the emotion of the provided audio segment [6].

3.2.2.1. Multi-Scale Block

In this block, the input is processed with 3×3 and 5×5 convolutions, where the padding 1 and 2 are used respectively, thus, the dimensions of the input data remain the same because there are shortcut connections i.e. adding the input data to the output. This approach safeguards against the issues of vanishing gradients in layers closer to the input, that may arise during backpropagation. After adding shortcut connection to the concatenated data, Batch Normalization is applied [6, 36].

3.2.2.2. Global-Aware Block

In the top right part of Figure 22. Global Aware block is shown and the output of the last Convolution layer of the Multi-Scale Net is reshaped thus it can fit the Batch Normalization and the full connected layer.

Global-Aware Block can be represented by the following equations:

$$Z = \tau(XU) \tag{19}$$

$$\tilde{Z} = s(Z) \tag{20}$$

$$Y = \tilde{Z}V \tag{21}$$

where τ is Gaussian Error Linear Unit (GELU), s is a layer where spatial interactions are obtained, Y is the output, U and V are linear projections along the channel dimension, one at the beginning and one at the end of this block. Their purpose is scale down and up, and the part in the middle is like a bottleneck, which is good to lose the insufficient information from the data [37].

The key part of this block is done by the function s, which can be written as

$$s(Z) = Z_1 \odot f_{W,b}(Z_2) \tag{22}$$

where $\odot$ denotes elementwise multiplication, the input Z is splitted along channel dimension into independent parts Z_1 and Z_2. Here Z_1 is used to multiply the output of the gating function, whose input is Z_2 and

$$f_{W,b}(Z) = WZ + b \tag{23}$$

and in this case the gating function is convolution 1×1. Also, the input of the gated function is normalized [37].

3.2.2.3. Gaussian Error Linear Unit

Gaussian Error Linear Unit (GELU) is one of the latest activation functions which has properties of ReLU (see Figure 24), dropout and zoneout. GELU is represented as $x\Phi(x)$, where $\Phi(x)$ is the cumulative distribution function of the standard Gaussian distribution. The reason why this distribution is used is that the properties of the dropout, zoneout and

ReLU are obtained by determining zero-one mask stochastically and still it is dependent on the input. The standard normal distribution is adequate because the normalized neuron inputs tend to the normal distribution and leads to the higher probability of the dropout for such a small x values. There are two approximations of the Gaussian Error Linear Unit, where the error function comes from the cumulative distribution function of a Gaussian distribution that is usually calculated with the error function. The approximations are:

$$\mathrm{GELU}(x) = 0.5x\left(1 + tanh\left[\sqrt{2/\pi}(x + 0.44715x^3)\right]\right) \tag{24}$$

$$\mathrm{GELU}(x) = x\sigma(1.702x) \tag{25}$$

where σ is sigmoid activation. In this case (24) is using, although (25) is significantly faster for calculating [38].

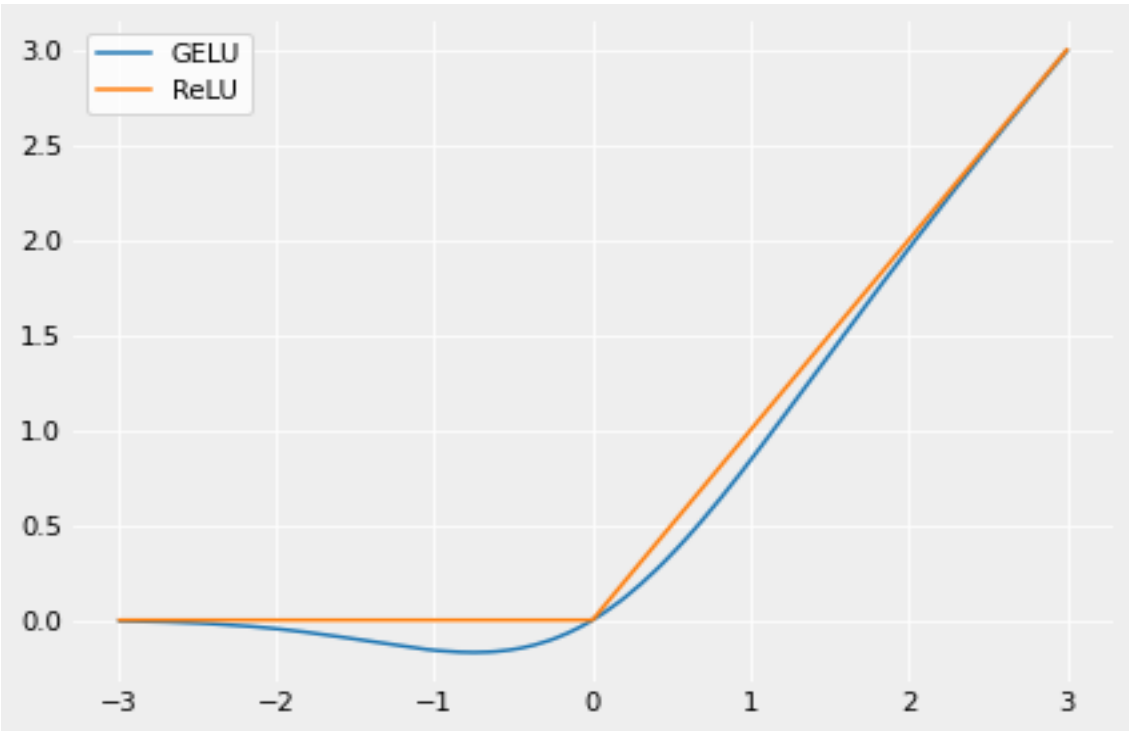


Figure 24. GELU and ReLU graphs [39]

4. Experiments

The study consists of the following experiments:

1. Model is trained on IEMOCAP (Interactive Emotional dyadic Motion Capture database in English) dataset and performance is tested on IEMOCAP.
2. Model is trained on IEMOCAP dataset and performance is tested on SEAC (Serbian Emotional Amateur Cellphone speech corpus).
3. Fine-tuning of a model trained on IEMOCAP dataset on SEAC dataset by freezing all layers except the last layer and testing the performance on SEAC.
4. Fine-tuning of a model trained on IEMOCAP dataset on SEAC dataset by freezing all layers except the last two layers and testing the performance on SEAC.
5. Fine-tuning of a model trained on IEMOCAP dataset on SEAC dataset by freezing all layers except the last three layers and testing the performance on SEAC.
6. Fine-tuning of a model trained on SEAC dataset on IEMOCAP dataset by freezing all layers except the last layer and testing the performance on IEMOCAP.
7. Fine-tuning of a model trained on SEAC dataset on IEMOCAP dataset by freezing all layers except the last two layers and testing the performance on IEMOCAP.
8. Fine-tuning of a model trained on SEAC dataset on IEMOCAP dataset by freezing all layers except the last three layers and testing the performance on IEMOCAP.

4.1. IEMOCAP dataset

IEMOCAP or interactive emotional dyadic motion capture database consists of five male and five female actors, where seven of them are professional actors and other three are senior student from the University of Southern California. There are two approaches in recording the data in this dataset, the scripted sessions and spontaneous sessions. The utterances are in English [8].

In the scripted sessions, the sessions are dialogs between two actors. The actors should memorize and rehearse on them which limits semantic and emotional contents. A theater professional selected 3 scripts with the aim to obtain happiness, anger, sadness, neutral or frustration emotion. Also, to have balanced set with the respect to gender, the plays are selected, so the pair consists of a female and a male actor [8].

The spontaneous sessions are based on improvisations of the actors, where they are free to express themselves, but before that they are polled to remember situations in the past that triggered certain emotion in them. For example, in case of sadness, the hypothetic scenarios are based on situation like loss of a family member, pet or similar situations [8].

There are 5 sessions recorded, and each session contains a male and a female actor. Session duration is approximately 6 hours long. The data are segmented and annotated manually. In a single segment, exactly one actor is speaking [8].

Working exclusively with four emotions (happy, neutral, angry, sad), the dataset in this study comprises 3784 samples (see Figure 25).

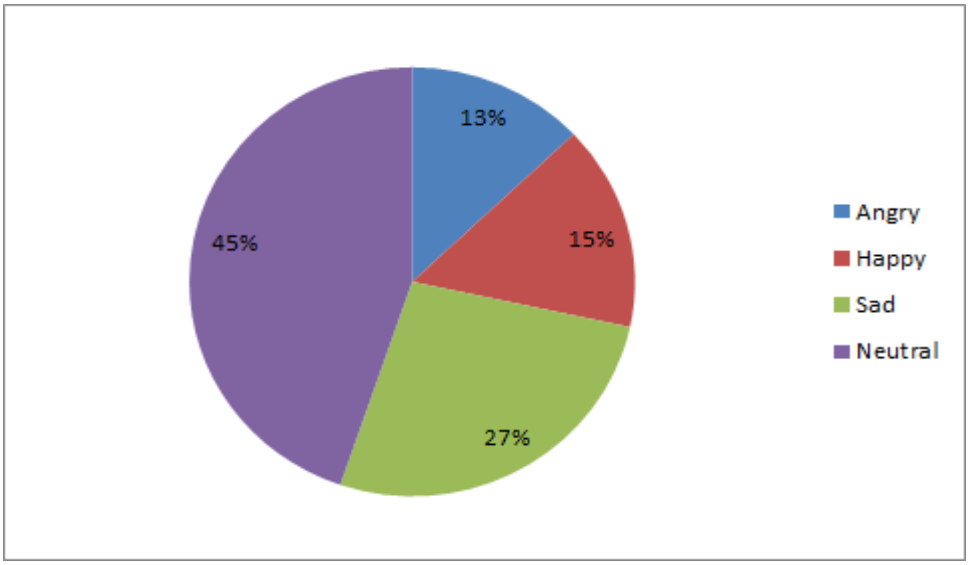


Figure 25. IEMOCAP structure

4.2. SEAC dataset

SEAC or Serbian Emotional Amateur Cellphone Speech Corpus is Serbian dataset that is released by the Faculty of Technical Sciences, University of Novi Sad. The dataset is an amateur dataset created by the students and employees of this faculty. In the dataset, 24 male and 29 female subjects recorded approximately 8 hours of speech data, where each subject recorded neutral, happy, sad, angry and fear emotion [7].

SEAC used utterances (59 to 62 utterances in 5 emotional states) from GEES [40], which is Serbian emotional speech corpus from 2003, which were selected as references for SEAC. The subjects had an Android application to record themselves producing the semantically same sentences in the required emotion. Subjects were able to repeat the recording of a single sentence as many times as they wanted. The total number of utterances obtained was 10 357 and the sampling rate was 44.1 kHz [7]. The structure of SEAC dataset, that is used in the experiment, is shown in Figure 26 and you can notice that the dataset is balanced (almost equal number of utterances in each emotion).

In this study utterances of only 25 speakers with all 4 emotions were selected, thus the total number of utterances, in this filtered SEAC dataset is 6054.

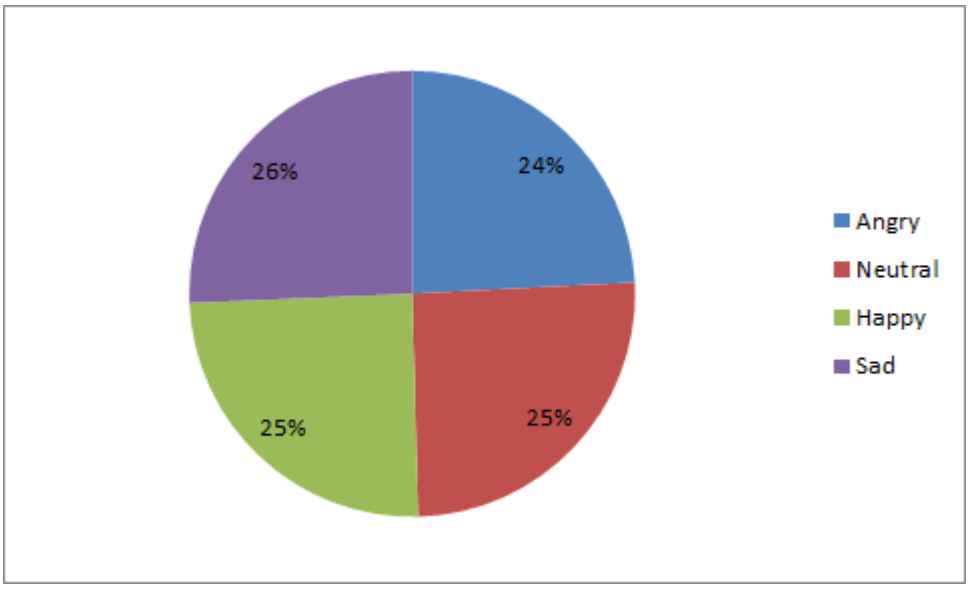


Figure 26. SEAC structure

4.3. Experiment setup

4.3.1. Experiment setup of 3D Convolutional Neural Network with Attention Model

As we have already said earlier, 10-cross fold validation is performed, such that in each fold, 8 speakers (out of 10) were selected for training, 1 speaker for validation and 1 for a test. The idea about cross-validation originates from [1], because the database IEMOCAP is not large and for performance they wanted to be as good as possible. The same experiment is done on SEAC, because we wanted to repeat the same procedure on SEAC as it is done on IEMOCAP, and even SEAC is split as it is done with IEMOCAP, all utterances of one speaker can be either in train, validation or test set, there is no possibility for one speaker to have utterances in two or even three sets. The only difference between experiments on IEMOCAP and SEAC is that SEAC is balanced dataset. Also, it is important to mention, if model is trained on one dataset and the performance tested on the other, then, the parameters of the loaded model are parameters of the fold that gave highest accuracy.

Results are obtained for each fold, and the mean value of these results is calculated. As you can see from Figure 26, IEMOCAP is unbalanced set and that is the reason why unweighted average recall (UAR) is reported, which is calculated as

$$\text{UAR}=\frac{TP}{TP+FN} \tag{26}$$

where TP is true positive (in confusion matrix it is the element on the main diagonal), FN is false negative (in confusion matrix it is the sum of elements in the corresponding row that are not on the main diagonal) [41].

The speech signal is split into 3-second segments, where zero-padding is applied in the cases where the segment is shorter than 3 seconds. The model takes each of these segments

as input, predicting the emotion conveyed by each segment. In the test phase the whole sentence is used for prediction, but max pooling of posterior probabilities is necessary to adapt dimensions. Log-mels are extracted, where the window of 25 ms is applied and the shift is 10 ms. The normalization of training and testing log-mels is done by the mean and the standard deviation, where the mean and standard deviation are calculated for the training set [1].

Optimization of the parameters of the model is done by minimization of the cross-entropy function, using Adam optimizer with the Nesterov momentum, where the momentum is set to 0.9. The size of mini-batch is 40 samples and the initial learning rate is 0.0001 [1].

4.3.2. Experiment setup of Global-Aware Multi-Scale Model

As we have already described above, GLAM model's inputs are MFCCs and we have the same four emotions, but in this case the excited emotion is used labeled as happy emotion and this can be done because happiness and excitement are similar [6]. For the training phase, in case of the IEMOCAP, the training dataset consists of 4 sessions, where one speaker from the fifth session is in the validation set and the other is in the test set. Also, in the case of GLAM, 10-fold cross validation is done. The training on SEAC dataset is done in the same way, out of 23 speakers that have all emotions recorded [40], there are 20 speakers (10 male, 10 female) randomly selected, such it is possible to split them into 5 sessions, where 4 sessions are training set and one session is test set. Also, 10-fold cross validation is done here. MFCCs are generated from the 2-second segments that are extracted from the utterance and the overlap between 2 segments is 1.6 seconds. The prediction result for some utterance is obtained as average of the prediction results of the segments that are extracted from the same utterance. Objective function used in the model is Cross-entropy, Adam optimizer with the weight decay rate equal to 10^{-6}, initial learning rate was 10^{-4} and then the exponentially decay with the multiplicative factor 0.95 is applied until 10^{-6} is not reached. 32 was the size of the batch [6].

5. Results

In this section the results of each experiment mentioned above will be shown in the confusion matrices, where you can see clearly where the classification went good and where it was bad. Also, there can be seen in how many cases a certain emotion is misclassified by some other emotion. The results written in this section are given as average accuracy, which is calculated as average value of the values on the main diagonal. If some other metric is used, it will be stated what metric is used.

5.1. ACRNN results

5.1.1. IEMOCAP training, IEMOCAP test

This experimental setup corresponds to the setup of ACRNN model on IEMOCAP dataset presented in [1]. The results obtained in our repeated experiment are presented in the form of confusion matrix in Figure 27. Relatively high recall is achieved for anger and sadness 72.32% and 76.97%, respectively. The problems are other two emotions, happiness and neutral, where huge percentages of the happiness are recognized as anger (26.76%) and neutral (22.89%), and in the case of neutral, the neutral emotion is equally recognized as neutral and happiness, but, also, not small recognition percentages are obtained for sadness and anger. This is strange because in IEMOCAP neutral is the most frequent class (45% of the dataset) and should be dominant in recognition results. Additionally, confusion matrix is not symmetric around main diagonal – i.e. if the emotions are similar and cannot be distinguished, then it is to be expected that the model mixes them equally.

The results presented in [1] are shown in Figure 28, and they significantly differ from those obtained in this study. This is very strange, because their original code was backbone for all experiments – only Python scripts that run original functions on different folds was prepared, and we applied 10-fold procedure described in [1] on the open dataset IEMOCAP. We double-checked the code, but we did not find any errors. Additionally, we contacted authors by e-mail, but we have not received any answer from them. The main differences are in scores for neutral speech and happiness. Our results are better in case of happiness (41.90% vs 29.95%), but significantly worse in case of neutral (30.12% vs 66.52%). Unfortunately, we do not have any explanation for such behaviour.

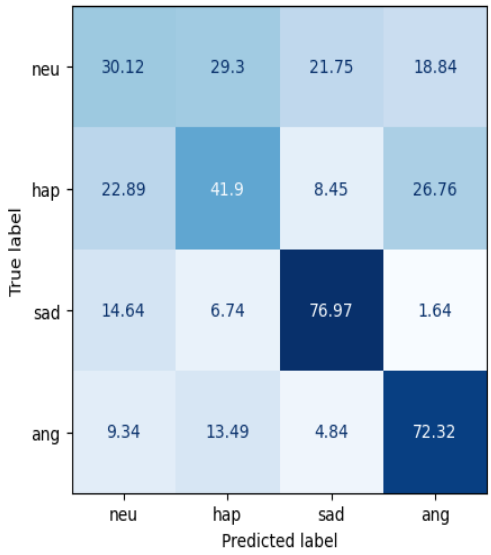


Figure 27. Confusion matrix on IEMOCAP of ACRNN

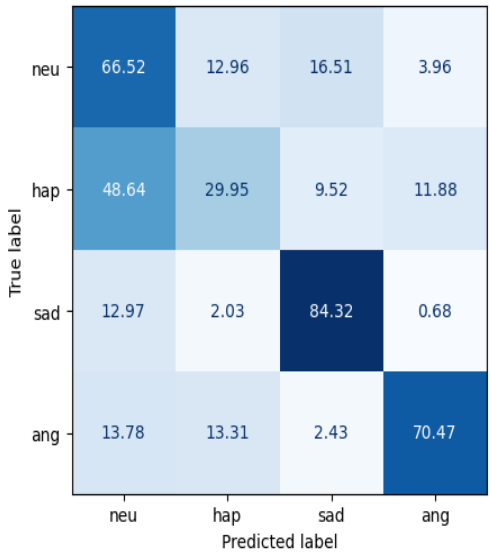


Figure 28. Confusion matrix on IEMOCAP of ACRNN reported in [1]

5.1.2. SEAC training, SEAC test

Similarly as above, but the dataset for training and test is SEAC, the corresponding confusion matrix is shown in Figure 29.

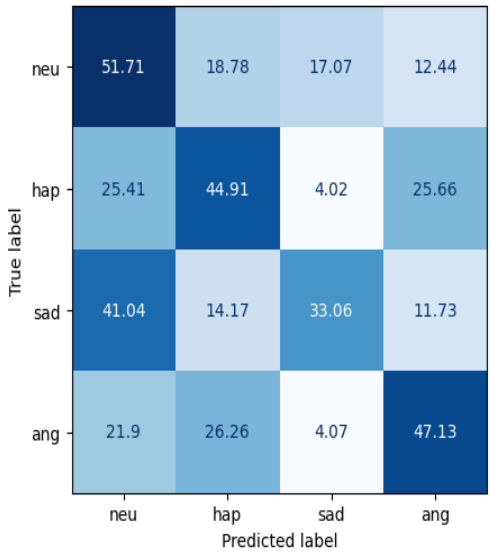


Figure 29. Confusion matrix on SEAC of ACRNN

As evident from Figure 29. the values on the main diagonal are not better than they are in Figure 27. There are misclassifications of emotions, except neutral, all other emotions are recognized correctly in less than 50% cases. The sadness has worst results, out of three sad emotions one is recognized correctly, while the great percentage is classified as neutral. In case of happiness, about one quarter of all instances is recognized as neutral and other quarter as anger. Similar situation is in case of anger, where one quarter is recognized as happiness, and one fifth as neutral. It seems that model cannot differentiate happiness and anger. The best result is for the neutral emotion, which is classified correctly in 51.71% cases.

5.1.3. IEMOCAP training, SEAC test

The experiment consists of training the model on English database and the performance is tested on the Serbian database, where there was no additional fine tuning on SEAC. The results are shown in Figure 30.

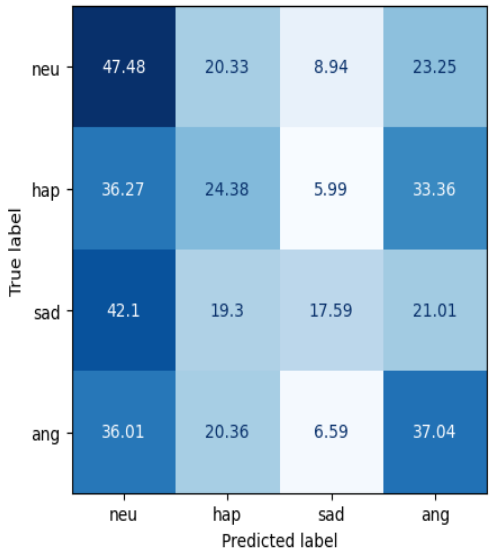


Figure 30. Confusion matrix on SEAC of ACRNN trained on IEMOCAP)

UAR of the experiment is 31.87%, where huge percentage of the emotions is classified as neutral, which is probably related with the imbalance of the English dataset (IEMOCAP) because 45% of the set is neutral emotion. The difference between these results and those from 5.1.2. is huge, where UAR (44.20%) is 12.33% greater than in this experiment, the closest recall is in neutral emotion and the difference between two recalls is 4.23%.

The results above are worse than those in Figure 29, prompting our interest in evaluating whether further adjustment of the final layers could enhance performance.

5.1.3.1. Fine tuning on SEAC

In Figure 31. and Figure 32. the increase in accurate classification of anger, sadness and happiness can be noticed, while the recall in the neutral emotion is preserved. The average recall in the test phase in the experiment where the last layer of the model is fine-tuned on SEAC is 33.93%, which is an increase in comparison to models trained on IEMOCAP and tested on SEAC (31.87%). If the last two layers in ACRNN are fine-tuned, the results are a little bit better, 34.85%, and if the last three layers, then UAR is 43.68%. Therefore, it is possible to increase the performance of the model trained in English on the Serbian data if the additional fine-tuning is applied, where the increase becomes significant if at least three last layers are additionally trained on the Serbian data. The difference in UAR between the model whose all parameters were trained, and the model whose parameters in the last 3 layers were trained is about 0.53%, where the result is obtained in 1981st iteration.

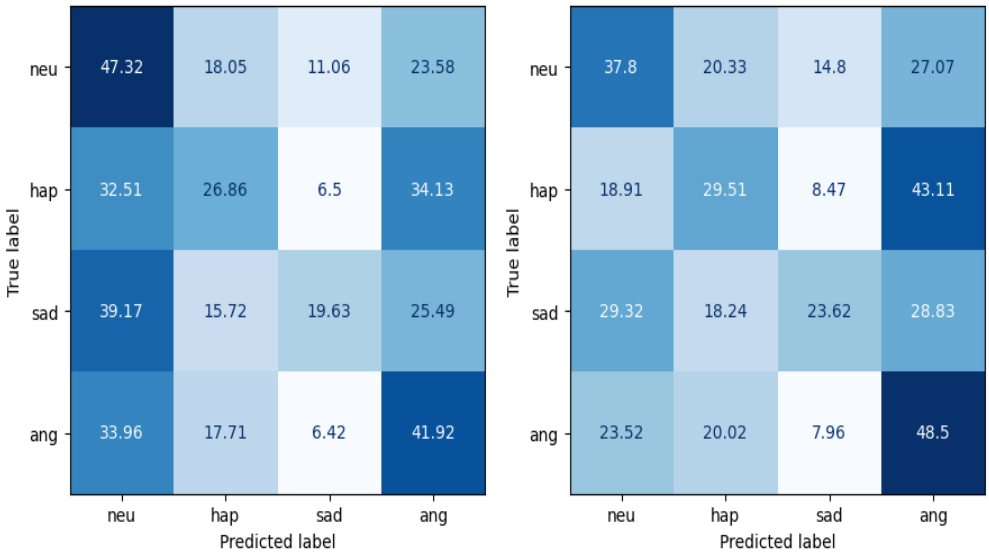


Figure 31. Confusion matrices on SEAC for ACRNN trained on IEMOCAP and fine-tuned the last layer (left) and the last two layers (right) on SEAC

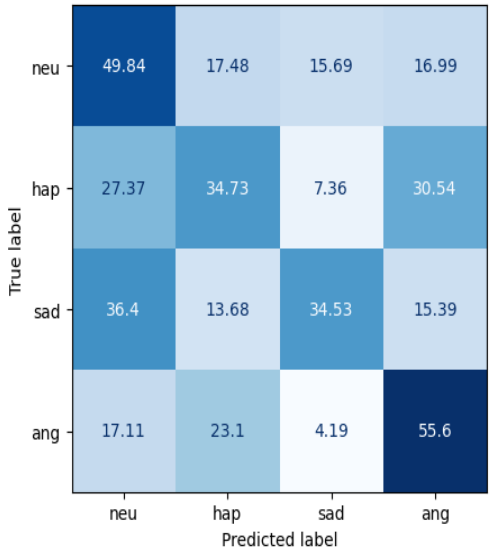


Figure 32. Confusion matrix on SEAC for ACRNN trained on IEMOCAP and fine-tuned the last 3 layer on SEAC

5.1.4. SEAC training, IEMOCAP test

The experiment is the same as previous, but train and test data are switched, the performance of the model trained on the Serbian data is tested on the English data. The results of the test phase on IEMOCAP of the model that is trained on SEAC, without the additional training of the last layers of the model are shown in Figure 33.

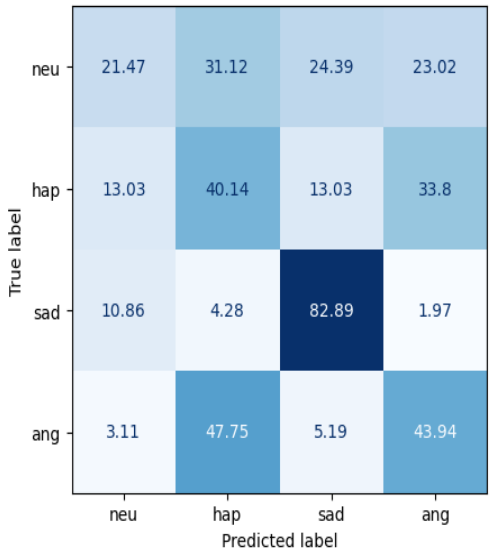


Figure 33. Confusion matrix on IEMOCAP of ACRNN trained on SEAC

Very good result is obtained for the sadness, the recall is 82.89% (which is greater than the recall in case of model trained on IEMOCAP which is 76.97%), but other emotions are misclassified in greater percentage, thus UAR is 47.13% (comparing to 55.33% of model trained on IEMOCAP). The lowest recall is in case of neutral emotion, which is classified correctly in the minimum number of cases (recall is 21.47% comparing to 30.12% of the model trained on IEMOCAP). It can be noted that model have a problem to make distinction between happiness and anger. Beside sadness, happiness class was the most frequent output, which is not the case of model trained and tested on SEAC (see Figure 29) where neutral emotion was dominant output. It is interesting that model trained on SEAC have higher recall of sadness on IEMOCAP then to its original data (82.98% vs 33.06%). This results in increase of the recall of the model from 44.20% to 47.13%, although the recall for other emotions are decreased. When the model is trained on SEAC (Serbian) and evaluated on IEMOCAP (English) the results are a little bit better than in the reverse case (47.13% vs 33.93%).

Similarly as in 5.1.3, fine-tuning on IEMOCAP of the last, last two and last three layers of the model trained on SEAC is applied. The results are presented below in Figure 34. and Figure 35.

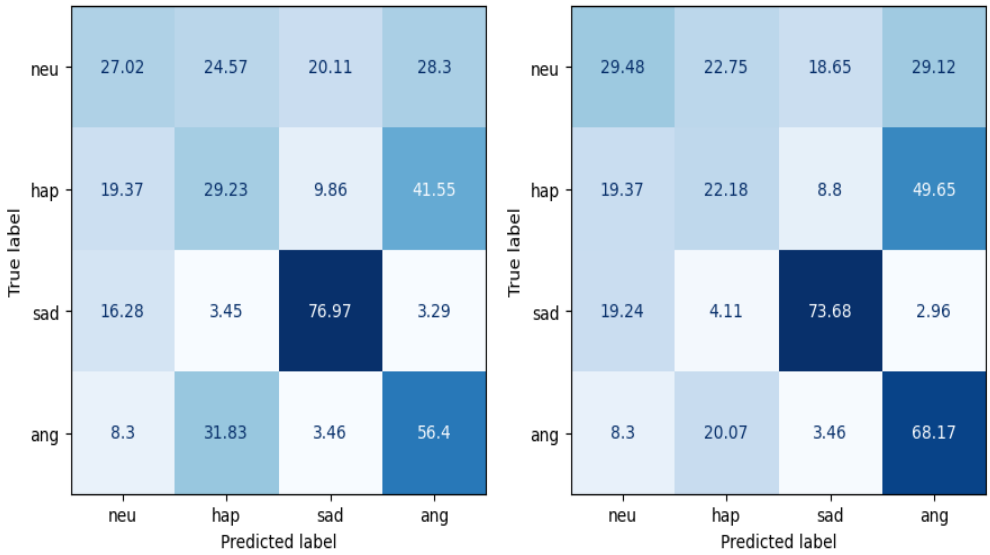


Figure 34. Confusion matrices on IEMOCAP for ACRNN trained on SEAC and fine-tuned the last layer (left) and the last two layers (right) on IEMOCAP

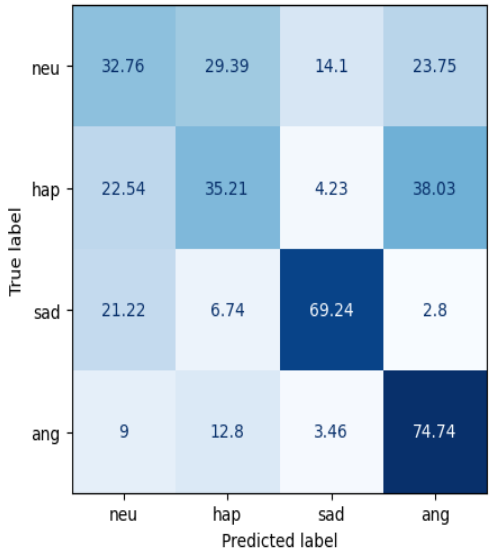


Figure 35. Confusion matrix on IEMOCAP for ACRNN trained on SEAC and fine-tuned the last three layers on IEMOCAP

Again, as it happened in 5.1.3. UAR is the highest for anger, but the neutral emotion and happiness are recognized as anger in a lot of situations as well. It is noticeable that the recall of the sadness is decreased in attempts in the additional training to correct the recall of other emotions, but the classification of the neutral emotion and happiness is still low. UAR on the IEMOCAP after the fine-tuning of the last, last two and last three layers are 47.41%, 48.38% and 52.99%, respectively. It can be noted that UAR is about 2.34% lower in case of fine-tuning comparing to a model initially trained on IEMOCAP, which is significantly higher than in case of fine-tuning of model on SEAC, where the result is obtained in 1251st iteration.

5.2. GLAM results

5.2.1. IEMOCAP training, IEMOCAP test

The first experiment with the GLAM model is to test the performance of the model trained on IEMOCAP dataset, on IEMOCAP, where 10-fold cross validation is applied in the same way as it is done in 5.1. The results of the experiment are shown in Figure 36.

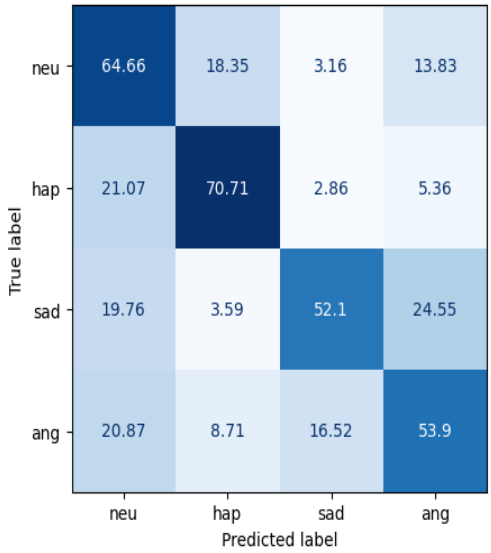


Figure 36. Confusion matrix on IEMOCAP of GLAM

UAR obtained in this experiment is 60.34%, which is higher than in the same experiment done with ACRNN (55.33%). The best performance is obtained on happiness and neutral, where is significant improvement comparing to the ACRNN model (41.90% and 30.12% comparing to 70.71% and 64.66%) and these results match the results given in [6].

5.2.2. SEAC training, SEAC test

The similar experiment as in 5.2.1 is done with the GLAM model, but in this case we were interested about the performance of the GLAM model on the Serbian data, after the training is done on the Serbian data.

There are a lot of differences between the results of this experiment and experiment from 5.1.2. with ACRNN, because the recall of the neutral emotion is worst in experiment with GLAM, but the recalls of the other three emotions are much better than in 5.1.2 with ACRNN. UAR of this experiment is 56.28%. The neutral emotion is misclassified in many cases as happiness and sadness. Additionally, model has a problem to make distinction between sadness and anger. Confusion matrices in case of model trained on IEMOCAP and SEAC (Figure 36 and Figure 37) are more similar to each other, than in case of ACRNN model.

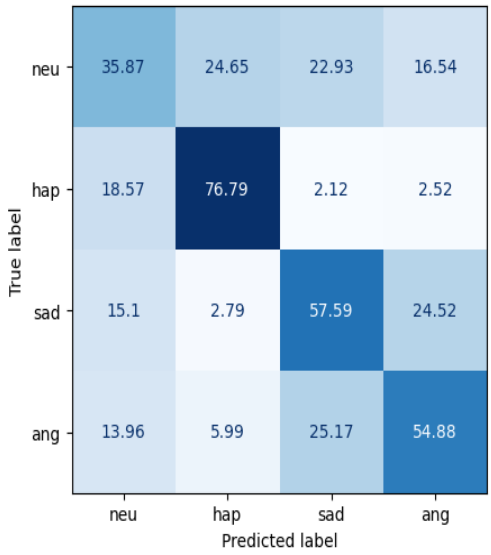


Figure 37. Confusion matrix on SEAC of GLAM

5.2.3. IEMOCAP training, SEAC testing

Here the results of 4 experiments will be showed, GLAM training on IEMOCAP and assessed on SEAC: *i*) without the fine-tuning, and with fine-tuning of *ii*) only the last layer *iii*) last two layers, and *iv*) the last three layers on SEAC.

In Figure 38. are presented results of the test phase on SEAC (Serbian dataset) of the GLAM model after the training on IEMOCAP (English dataset) and it is noticeable that the huge percentage took anger for classification the neutral emotion, sadness and anger. In about a half situations, the happiness is classified correctly and the worst recall is for sadness, which is misclassified as anger in 65.53% of situations. UAR was equal to 45.49%.

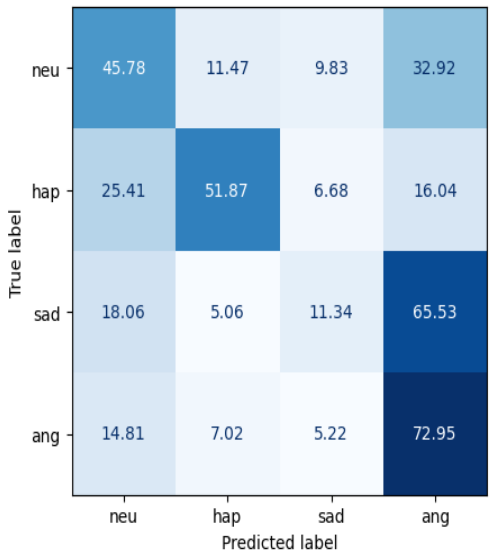


Figure 38. Confusion matrix on SEAC of GLAM trained on IEMOCAP

It can be noticed in Figure 39. that the fine-tuning affected the sadness, the percentages are becoming higher in the sadness columns, while the results in the anger are lower, but in the right confusion matrix the things are a little bit corrected in the anger column and, also, the recall for the neutral emotion and happiness is tried to be maintained how much it is possible. The average recall for the experiments where the model is trained on IEMOCAP and tested on SEAC, while the last and last two layers are additionally trained on SEAC are 42.70% and 42.36%, respectively.

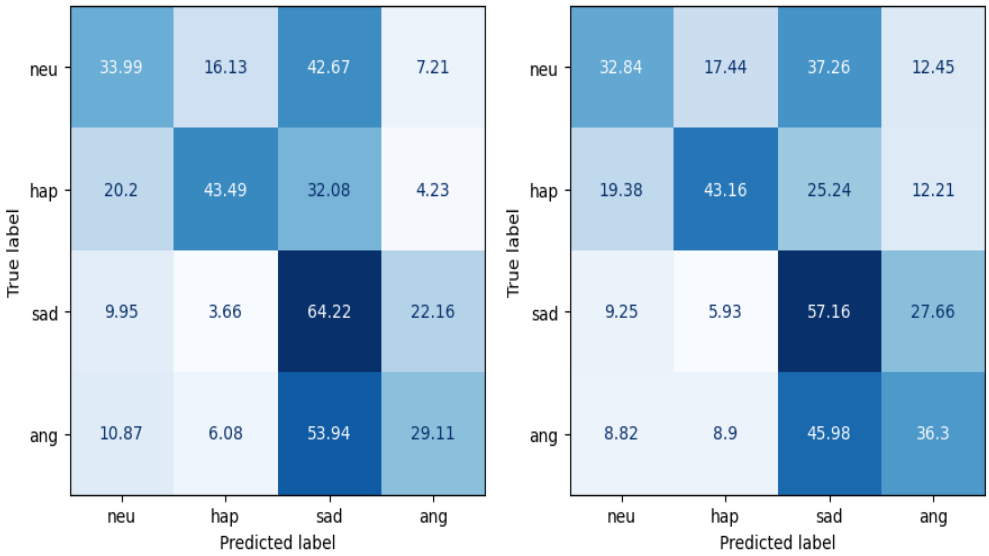


Figure 39.Confusion matrices on SEAC for GLAM trained on IEMOCAP and fine-tuned the last layer (left) and the last two layers (right) on SEAC

In Figure 40. the results of the experiment, where the GLAM is trained on IEMOCAP, tested on the Serbian and the last three layers are fine-tuned on the SEAC, are shown.

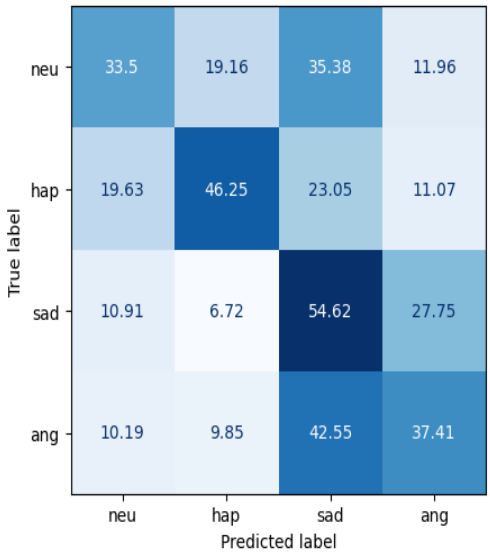


Figure 40. Confusion matrix on SEAC for GLAM trained on IEMOCAP and fine-tuned the last 3 layer on SEAC

The fine-tuning of the three last layers ended in a little bit better situation on the main diagonal in the confusion matrix, recalls over four classes is not so significantly increased. UAR of the experiment is 42.95%.

5.2.4. SEAC training, IEMOCAP testing

After the training is done on SEAC (Serbian dataset), the performance is tested on IEMOCAP (English dataset), without any additional training, and the following confusion matrix (Figure 41) is obtained.

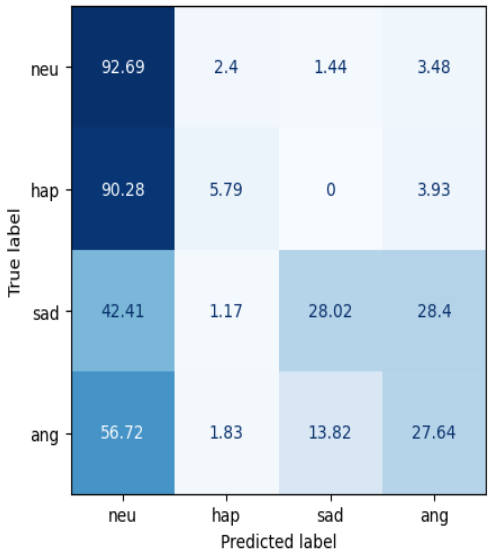


Figure 41. Confusion matrix on IEMOCAP for GLAM trained on SEAC

As you can see, a lot of utterances are misclassified as neutral emotion, especially happiness. UAR of the experiment is 38.54% and the result is even worse if you consider that the recall of the neutral emotion is 92.69%.

The next experiments consist of training the GLAM model on SEAC, freezing the layers, up to three last layers that are additionally trained on the IEAMOCAP and the performance is test on the IEMOCAP.

An improvement in Figure 42. is obvious, because in Figure 41. only the neutral emotion has high recall and all other were very low, but still the results are not good. The recall in the neutral is worse, the sadness is not significantly better, but the happiness and anger are much better. UARs of the test phase of the experiments, where the GLAM is trained on SEAC and tested on IEMOCAP, where the last and last two layers were additionally trained, are 48.24% and 52.11%, respectively.

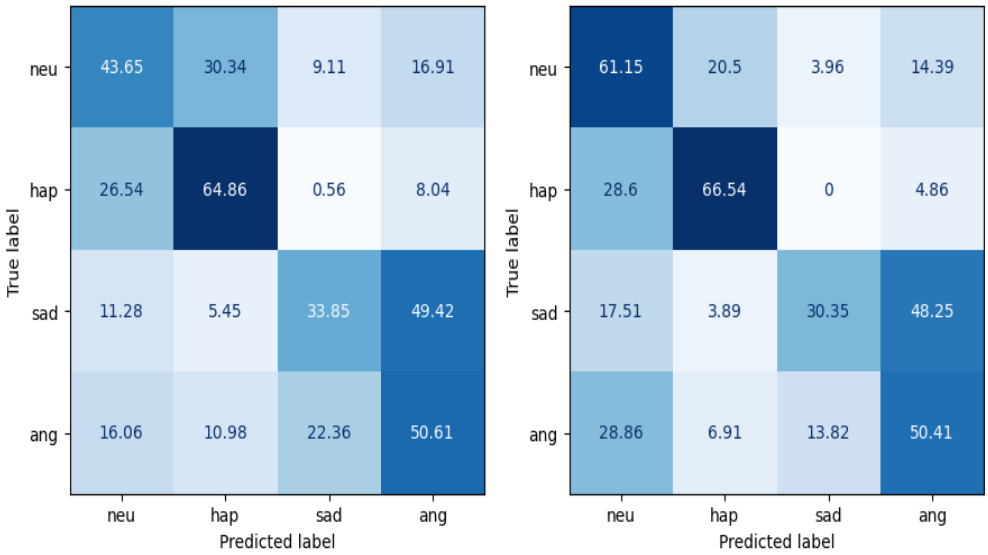


Figure 42. Confusion matrices on IEMOCAP for GLAM trained on SEAC and fine-tuned the last layer (left) and the last two layers (right) on IEMOCAP

Nothing significant has changed in Figure 43. compared to the right confusion matrix from Figure 42, UAR is now 53.12%.

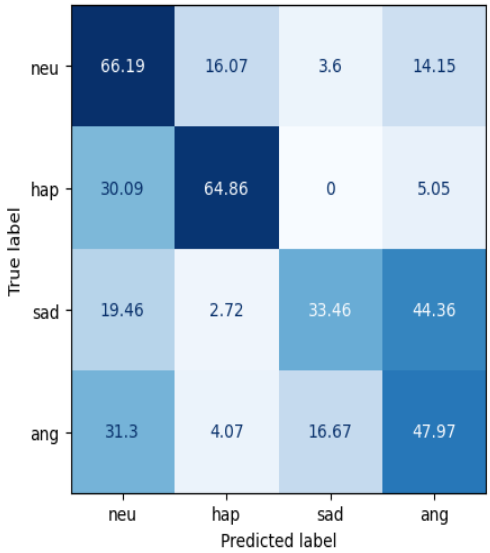


Figure 43. Confusion matrix on IEMOCAP for GLAM trained on SEAC and fine-tuned the last three layers on IEMOCAP

5.3. Modified GLAM Results

Under the assumption that Serbian language is more complex than English, in the GLAM model is added one more Multi-Scale Block, and the experiments from 5.2.2. and 5.2.4. are repeated. This has showed as good step, because results are even better.

5.3.1. SEAC training, SEAC test

The GLAM model, where one more Multi-Scale Block is involved, was trained and tested on SEAC data. The results are shown in Figure 44. below.

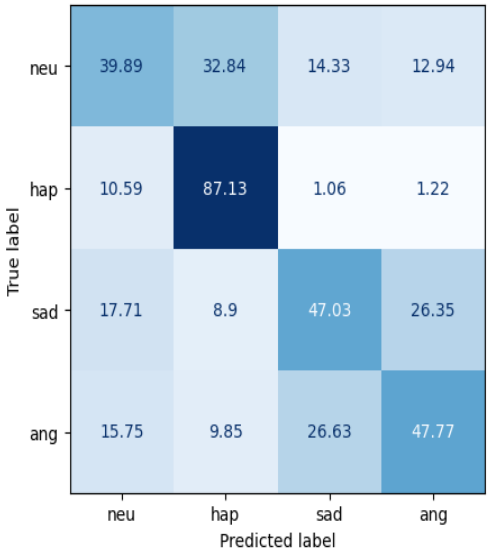


Figure 44. Confusion matrix on SEAC of the extended GLAM trained on SEAC

From Figure 44. you can notice that the recall of the happiness is significantly increased compared to the same experiment using the ordinary GLAM. Also, the neutral emotion recall is increased a little bit, but there is decrease in other two emotions and UAR is 55.46%, which is almost 1% lower compared to the same experiment which is done by the ordinary GLAM.

5.3.2. SEAC training, IEMOCAP test

The procedure is the same as in 5.2.4, four experiments, but this time with extended GLAM are done:

- Modified GLAM is trained on SEAC, then tested on IEMOCAP;
- Modified GLAM is trained on SEAC, then all layers except the last one are frozen, while the additional training of the last layer on IEMOCAP is applied and the performance is tested on IEMOCAP;
- Modified GLAM is trained on SEAC, then all layers except the last two are frozen, while the additional training of the last two layers on IEMOCAP is applied and the performance is tested on IEMOCAP;
- Modified GLAM is trained on SEAC, then all layers except the last three are frozen, while the additional training of the last three layers on IEMOCAP is applied and the performance is tested on IEMOCAP.

Figure 45. shows the improvement in recalls of all emotions, except the anger, but, neither is significant. Also, the results for the sadness are bad in both experiments. UARs of the experiments shown in the left and right confusion matrices are 51.31% and 53.54%, respectively, whiles UARs of the same experiments on the ordinary GLAM were 38.54% and 48.24%.

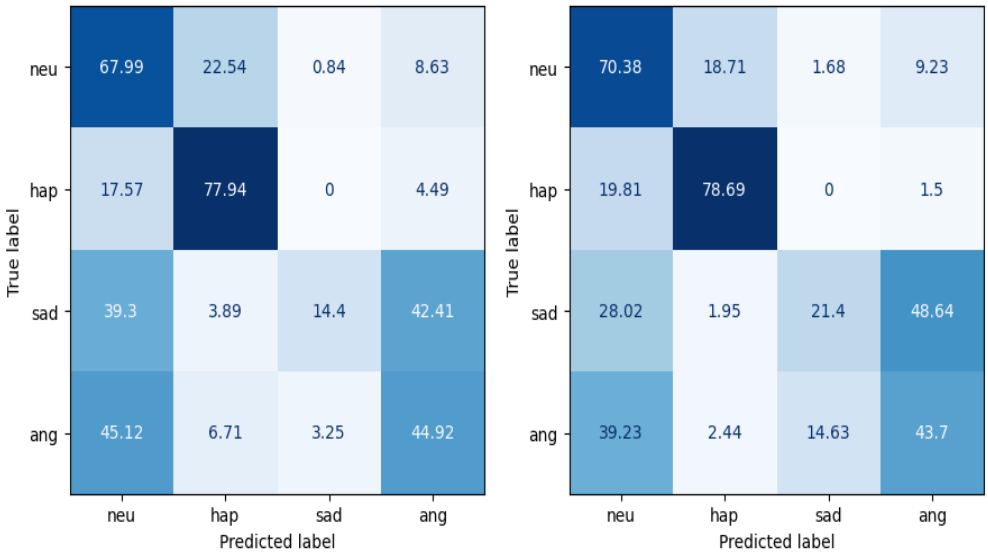


Figure 45. Confusion matrix on IEMOCAP of the extended GLAM trained on SEAC - no fine-tuning (left), fine-tuning of the last layer on IEMOCAP (right)

Figure 46. shows that there is no significant difference in the modified GLAM if you freeze last, last two or last three layers, the results are very similar. UARs of the experiments shown in the left and right confusion matrices are 53.81% and 53.6%, respectively, whiles UARs of the same experiments on the ordinary GLAM were 52.11% and 53.12%.

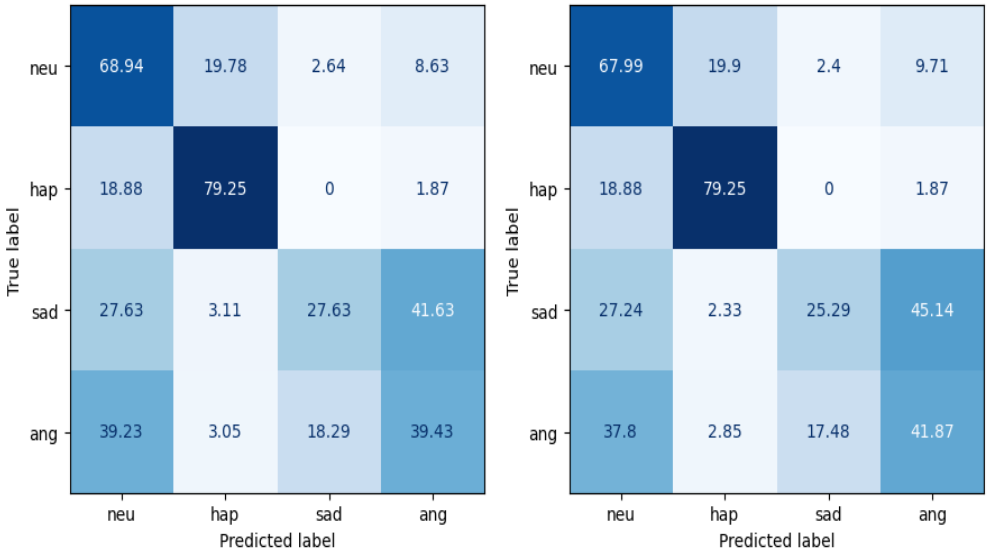


Figure 46. Confusion matrix on IEMOCAP of the extended GLAM trained on SEAC - fine-tuning on IEMOCAP of the last two layers (left) and last three layers (right)

6. Conclusion

Two DNN architectures are explored in this paper, one with 3D Convolutional Neural Network and LSTM (ACRNN), and the second without LSTM and, also, with some new parts that corrects defects of Convolutional Neural Networks (GLAM). Additionally, the extended GLAM model is evaluated in a few experiments.

Using ACRNN for the training on the English data and test on the English data, where all utterances of a single speaker are either in the train or test set, UAR of the model was 55.32%, while the same experiment with GLAM showed 60.34%. Even at this point, it was possible to expect lower score in the results of the experiments where the test set consists of the Serbian data and the training of the model was done on the English dataset, regardless of the architecture. Also, in ACRNN the sadness and anger are emotions whose recall reached more than 70% and the other two emotions are below 42%, while on the other hand, in the GLAM only emotion that reached 70% is happiness, but other three emotions are between 52% and 65%, which shows the difference between two approaches because the same experiment is done on the same dataset on the different architectures.

The training and testing of both models were done on the Serbian data, where the Serbian dataset is adapted, thus the experiments look the same done with IEMOCAP. ACRNN had only one emotion whose recall barely reached 50% and other three were in the range between 33.06% and 47.13%, while the GLAM had happiness with the recall above 70%, neutral about 35% and other two emotions were between 54% and 58%. Again, like it happened with IEMOCAP, the situation differs depending on the architecture. In these experiments, the experiment done with the modified GLAM should be mentioned as well, where the recall on the happiness was increased even more, but the other three emotions decreased.

When the models were trained on the English data and tested on the Serbian, without freezing layers and without additional training of the last layers, the performance with GLAM was much better, the anger recall was above 70%, neutral and happiness about 48% and only sadness has bad result of 11%, where 65% of sadness was misclassified as anger. In the same experiment ACRNN had the best result for neutral emotion (about 47%) and all other emotions were below 38%, where a lot of examples were misclassified as neutral.

Then, after all layers, except the last layer in each of the models were frozen and the last layers were trained on SEAC and tested on it, again, the GLAM showed as a better model with UAR equal to 42.70% against 33.93% that ACRNN obtained. When only the last two layers were additionally trained and, also, only the last three layers, still, the GLAM had better performance, but, UAR did not reach 50%.

In the experiments that are the same as previous, but the roles of the Serbian dataset and English are switched, the following results occurred. Without any freezing of the layers and additional trainings, if you compare results of the ACRNN, GLAM and modified GLAM, then you will see that the best recalls are on sadness (82%), neutral (92%) and

happiness (77%), respectively and all other emotions have recalls less than 50%, only modified GLAM has neutral above 50%, so, again, big differences in results for different architectures, even GLAM and modified GLAM differ in only one block. The best UAR has modified GLAM (51%), which is not a good result, it means about half of the emotions are classified wrongly.

The results mentioned in the previous paragraph are not significantly improved if the additional training for the last, last two or last three layers are applied in each architecture, the emotions with the highest recall scores in each model stayed the emotions with the highest scores in each of the three experiments mentioned above. Still, the highest UAR stayed for the modified GLAM and the highest recall was in the experiment in which the last three layers are additionally trained on the English data and other two models are close, where the difference in UARs are about than 0.5%, but, the recalls are reached on the difference emotions, ACRNN has highest recalls on the sadness (69%) and anger (74%), GLAM has highest recalls on the neutral (66%) and happiness (64%) and the modified GLAM has the highest recall on the neutral (68%) and happiness (79%), where happiness recall was significantly higher.

First, the results obtained above probably would be better if there is a Serbian database where voice utterances are obtained in some way similar to how it is done for IEMOCAP. From the results it is possible to see that the difference in languages is an obstacle for obtaining better results in such an experiment. For improving the results in such experiments, attention should be on the language difference, if it is possible similarities of two languages should be extracted as features that will be processed by an appropriate network, because other factors are eliminated.

Bibliography

[1] M. Chen, X. He, J. Yang, H. Zhang, „3-D Convolutional Recurrent Neural Networks with Attention Model for Speech Emotion Recogntion“. doi: 10.1109/LSP.2018.2860246.

[2] Survey on speech emotion recognition: Features, classification schemes, and databases.

[3] https://www.paulekman.com/universal-emotions/. Accessed: 26.11.2023.

[4] https://biomedical-engineering-online.biomedcentral.com/articles/10.1186/1475-925X-12-44/figures/1. Accessed: 26.11.2023.

[5] M.B. Akcay, K. Oguz, „Speech emotion recognition: Emotional models, databases, features, preprocessing methods, support modalities, and classifiers“. *Speech Comunication*, vol. 116, 56-76. 2020.

[6] W. Zhu, X. Li, „Speech Emotion Recognition with Global-Aware Fusion on Multi-Scale Feature Representation“. *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 6437-6441, May 2022.

[7] S. Suzić, T. Nosek, M. Sečujski, B. Popović, L. Krstanović, M. Vujović, N. Simić, M. Janev, N. Jakovljević, V. Delić, „SEAC: Serbian Emotional Amateur Cellphone Speech Corpus“. *Language Resources and Evaluation*, ISSN: 1574-020X.

[8] C. Busso, M. Bulut, C. Lee, A. Kazemzadeh, E. Mower, S. Kim, J.N. Chang, S. Lee, S.S. Narayanan, „IEMOCAP: interactive emotional dyadic motion capture database“. *Language Resources & Evaluation* 42.4(2008):335.

[9] Shiqing Zhang, Shiliang Zhang, T. Huang, W. Gao, „Speech Emotion Recognition Using Deep Convolutional Neural Network and Discriminant Temporal Pyramid Matching“. *IEEE Transactions on Multimedia*, vol. 20, no. 6, June 2018.

[10] G. Trigeorgis, F. Ringeval, R. Brueckner, E. Marchi, M.A. Nicolaou, B. Schuller, S. Zafeiriou, „Adieu features? End-to-end speech emotion recognition using a deep convolutional neural network“. *2016 IEEE Internacional Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5200-5204, March 2016.

[11] C. Huang, S.S. Narayanan, „Attention Assisted Discovery of Sub-Utterance Structure in Speech Emotion Recognition“. *Interspeech (2016)*, pp. 1387-1391.

[12] C. Huang, S.S. Narayanan, „Deep convolutional recurrent neural network with attention mechanism for robust speech emotion recognition“. *IEEE Internacional Conference on Multimedia and Expo (ICME)*, pp. 583-588, 2017.

[13] https://importchris.medium.com/how-to-create-understand-mel-spectrograms-ff7634991056. Accessed: 19.06.2023.

[14] https://haythamfayek.com/2016/04/21/speech-processing-for-machine-learning.html. Accessed: 19.06.2023.

[15] https://superkogito.github.io/blog/2020/01/25/signal_framing.html. Accessed: 19.06.2023.

[16] C.Wang, Y. Kang, „Feature Extraction Techniques of Non-Stationary Signals for Fault Diagnosis in Machinery Systems“. *Journal of Signal and Information Processing*, vol. 3, no. 1. 2012.

[17] X. Huang, A. Acero, H. Hon, „Spoken language processing: a guide to theory, algorithm, and system development“.

[18] https://speechprocessingbook.aalto.fi/Representations/Deltas_and_Delta-deltas.html?highlight=delta. Accessed: 19.06.2023.

[19] D. Stutz, L. Beyer, „Understanding Convolutional Neural Networks“. 2014.

[20] K. O'Shea, R. Nash, „An Introduction to Convolutional Neural Networks". arXiv: 1511.08458. 2015.
[21] H.H.Aghdam, E.J. Heravi, „Guide to Convolutional Neural Networks". doi: 10.1007/978-3-319-57550-6.
[22] https://dev.to/sandeepbalachandran/machine-learning-convolution-with-color-images-2p41. Accessed: 21.06.2023.
[23] https://towardsai.net/p/l/introduction-to-pooling-layers-in-cnn. Accessed: 25.6.2023.
[24] T. N. Sainath, V. Peddinti, B. Kingsbury, P. Fousek, B. Ramabhadran, D. Nahamoo, „Deep Scattering Spectra with Deep Neural Networks for LVCSR Tasks". *Proceedings of the Annual Conference of the International Speech Communication Association, Interspeech*, 2014.
[25] https://builtin.com/machine-learning/fully-connected-layer. Accessed: 26.6.2023.
[26] https://www.oreilly.com/library/view/tensorflow-for-deep/9781491980446/ch04.html. Accessed: 26.6.2023.
[27] https://www.geeksforgeeks.org/activation-functions-neural-networks/. Accessed: 26.6.2023.
[28] https://himanshuxd.medium.com/activation-functions-sigmoid-relu-leaky-relu-and-softmax-basics-for-neural-networks-and-deep-8d9c70eed91e. Accessed: 26.6.2023.
[29] https://www.researchgate.net/figure/ReLU-activation-function-vs-LeakyReLU-activation-function_fig2_358306930. Accessed: 26.6.2023.
[30] M. Ravanelli, P. Brakel, M. Omologo, Y. Bengio, „Light Gated Recurrent Units for Speech Recognition". *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 2, no. 2, pp. 92-102 April 2018.
[31] https://colah.github.io/posts/2015-08-Understanding-LSTMs/. Accessed: 27.6.2023.
[32] N. Minh-Tuan, Y. Kim, „Bidirectional Long Short-Term Memory Neural Networks for Linear Sum Assignment Problems". *Applied Sciences*, no. 9, c. 3470. 2019.
[33] https://analyticsindiamag.com/complete-guide-to-bidirectional-lstm-with-python-codes/. Accessed: 28.6.2023.
[34] https://towardsdatascience.com/batch-normalization-in-3-levels-of-understanding-14c2da90a338. Accessed: 24.7.2023.
[35] D.G. Childers, D.P.Skinner, R.C. Kemerait, „The Cepstrum: A Guide to Processing". *Proceedings of the IEEE*, vol. 65, no. 10, pp. 1428-1443, 1977.
[36] K. He, X. Zhang, S. Ren, J. Sun, „Deep Residual Learning for Image Recognition". *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770-778.
[37] H. Liu, Z. Dai, D.R. So, Q. V. Le, „Pay Attention to MLPs". arXiv: 2105.08050. 2021.
[38] D. Hendrycks, K. Gimpel, „Gaussian Error Linear Units (GELUs)". arXiv: 1606.08415. 2016.
[39] https://medium.com/@tariqanwarph/activation-function-and-glu-variants-for-transformer-models-a4fcbe85323f. Accessed 5.4.2024.
[40] https://catalog.elra.info/en-us/repository/browse/ELRA-S0385/. Accessed: 30.11.2023.
[41] https://www.analyticsvidhya.com/blog/2021/06/confusion-matrix-for-multi-class-classification/. Accessed: 21.04.2024.

УНИВЕРЗИТЕТ У НОВОМ САДУ • **ПРИРОДНО-МАТЕМАТИЧКИ ФАКУЛТЕТ**
21000 НОВИ САД, Трг Доситеја Обрадовића 3

КЉУЧНА ДОКУМЕНТАЦИЈСКА ИНФОРМАЦИЈА

Редни број, **РБР**:		
Идентификациони број, **ИБР**:		
Тип документације, **ТД**:		Монографска публикација
Тип записа, **ТЗ**:		Текстуални штампани материјал
Врста рада, **ВР**:		Мастер рад
Аутор, **АУ**:		Данијел Лазаревић
Ментор, **МН**:		др Никша Јаковљевић
Наслов рада, **НР**:		Евалуација система за препознавање емоција у говору заснованих на транформерима и дугорочно-краткорочној меморији
Језик публикације, **ЈП**:		Енглески
Језик извода, **ЈИ**		Енглески
Земља публиковања, **ЗП**:		Србија
Уже географско подручје, **УГП**:		Војводина
Година, **ГО**:		2024.
Издавач, **ИЗ**:		Ауторски репринт
Место и адреса, **МА**:		Нови Сад, Природно-математички факултет, Трг Доситеја Обрадовића 3
Физички опис рада, **ФО**: (поглавља/страна/ цитата/табела/слика/графика/прилога)		6 поглавља, 49 страна, 41 литературни цитат, 47 слика
Научна област, **НО**:		Математика
Научна дисциплина, **НД**:		Примјењена математика
Предметна одредница/Кључне речи, **ПО**:		Препознавање емоција у говору, неуронске мреже, дугорочно-краткорочна меморија, трасформери
УДК		
Чува се, **ЧУ**:		Библиотека Департмана за математику и информатику, Природно-математички факултет, Трг Доситеја Обрадовића 3, 21000 Нови Сад
Важна напомена, **ВН**:		
Извод, **ИЗ**:		Рад се бави евалуацијом система за препознавање емоција у говору користећи дубоке неуронске мреже. За евалуацију су кориштене двије архитектуре, једна базирана на дугорочно-краткорочној меморији, а друга базирана на трансформерима. Обука и тестирање су вршени на двије јавно доступне говорне базе: IEMOCAP (енглески језик) и SEAC (српски језик). Поред евалуације самих кориштених архитектура, евалуирали смо перформансе система обученог на једном језику за препознавање емоција на другом језику.
Датум прихватања теме, **ДП**:		9.1.2024.
Датум одбране, **ДО**:		
Чланови комисије, **КО**:	Председник:	др Душан Јаковетић, ванредни професор, Природно-математички факултет, Универзитет у Новом Саду
	Ментор:	др Никша Јаковљевић, ванредни професор, Факултет техничких наука, Универзитет у Новом Саду

Члан:	др Оскар Марко, научни сарадник, Институт БиоСенс у Новом Саду

UNIVERSITY OF NOVI SAD • **FACULTY OF SCIENCES**
21000 NOVI SAD, Trg Dositeja Obradovića 3

KEY WORDS DOCUMENTATION

Accession number, **ANO**:	
Identification number, **INO**:	
Document type, **DT**:	Monographic publication
Type of record, **TR**:	Printed text
Contents code, **CC**:	Master's thesis
Author, **AU**:	Danijel Lazarević
Mentor, **MN**:	dr Nikša Jakovljević
Title, **TI**:	An evaluation of speech emotion recognition systems based on transformer and long-short term memory architectures
Language of text, **LT**:	English
Language of abstract, **LA**:	English
Country of publication, **CP**:	Serbia
Locality of publication, **LP**:	Vojvodina
Publication year, **PY**:	2024.
Publisher, **PB**:	Author's reprint
Publication place, **PP**:	Novi Sad, Faculty Sciences, Trg Dositeja Obradovića 3
Physical description, **PD**: (chapters/pages/ref./tables/pictures/graphs/appendixes)	6 chapters, 49 pages, 41 references, 47 figures
Scientific field, **SF**:	Mathematics
Scientific discipline, **SD**:	Applied mathematics
Subject/Key words, **S/KW**:	Speech emotion recognition, neural networks, long short-term memory, transformers
UC	
Holding data, **HD**:	Library of the Faculty of Sciences Department of Mathematics and Informatics, Trg Dositeja Obradovića 3, Novi Sad

Note, **N**:		
Abstract, **AB**:		Evaluation of the emotion recognition systems in the speech using deep neural networks is covered in the paper. Two architectures were used for evaluation, one based on long short-term memory and the other based on transformers. Training and testing were conducted on two publicly available speech databases: IEMOCAP (English language) and SEAC (Serbian language). In addition to evaluating the architectures themselves, we also assessed the performance of a system trained in one language for emotion recognition in another language.
Accepted by the Scientific Board on, **ASB**:		9.1.2024.
Defended on, **DE**:		
Defended Board, **DB**:	President:	dr Dušan Jakovetić, associate professor, Faculty of Sciences, University of Novi Sad
	Mentor:	dr Nikša Jakovljević, associate professor, Faculty of Technical Sciences, University of Novi Sad
	Member:	dr Oskar Marko, scientific associate of BioSense Institute, Novi Sad