Sentence-Level Emotion Apprehension Through Facial Expression & speech verification Analysis

Md. Mohaimanul Haque
Department of CSE
American International University, Bangladesh
h.mohaimanul@gmail.com

Abu Fuzail polin
Department of CSE
American International University, Bangladesh
fuzail.polin@gmail.com

Victor Stany Rozario
Assistant Professor(CS)
American International University, Bangladesh
stany@aiub.edu

***Abstract*—The importance of Emotional state apprehension is widely perceived in social interaction and social intelligence. Since the nineteenth century, this has been a popular research subject [10]. In human-to-human communication, the understanding of facial expressions forms a communication carrier that offers vital data about the mental, emotional and even physical state of the persons in conversation [10]. Inevitably user's emotional state plays an important role not only in human associations with other people but also in the way a user uses computers. As the emotional state of a person may determine consistency, task solving, and decision-making skills [12]. Facial expression analysis, as used in this research, refers to computer systems that try to automatically predict user emotional state by analysing and identifying facial motions and facial feature changes from visual data. Though situations, body gesture, voice, individual diversity, and cultural influences, as well as facial arrangement and timing, all aid in interpretation [6]. Facial expression analysis tools will be used in this research to analyse facial actions regardless of context, society, gender, and so on.**

I. Introduction

A. Background of the study

This region includes a complete review of the strategies beforehand used to identify the emotion of a sentence. It can be shown that D.Das and S.Bandyopadhyay [4] used Conditional Random Fields in their ML approach (CRF). It is a two-step procedure in which the first action is to generate an emotion for each word in a sentence using WordNet's Affect List, and the second action is to determine the dominant sentiment of each sentence using the impact points of each word in the sentence. The first step employs CRF for word-level commentary, which evaluates information in the sentence that is already in the Affect list and returns idiom tagging of feelings. The next step is to apply these word-level preferences to attain the overall sentence level feeling based on the weight scores of each word in a sentence.

It has a high accuracy of 87.65%, which is constrained by the Affect list's Synonyms set (SynSet). This effort to identify words that do not exist in the said list failed.

The authors of Paper [5] conducted an exploratory analysis of various practices to address the issue of sentence-level emotion tagging. The first tactic of knowledge-based concepts, WordNet-Affect Presence, was used to deal with comprehension changes in a text based solely on the presence of words from the WordNet Affect vocabulary. Whereas in the following method, the LSA discovered the similarity between the given text and each emotion separately.

The third approach employs synonyms from the WordNet SynSet. Finally, LSA all emotion words serves the preceding set by blending words from all SynSets associated with a given sentiment, as does WordNet Affect List. This paper also employs the Corpus linguistics-based approach [5]. This method is more realistic and has been used in our model as well.

The authors of paper [6] discuss a design for obtaining emotion tags using two systems: keyword spotting and lexical resemblance. Several techniques use the current lexical corpus to find words related to a specific emotion.

B. Statement of the problem

The market for emotion-detection technology is currently worth $21.6 billion, and it is expected to more than double by 2024, reaching $56 billion. Businesses will buy systems to assist them in screening job applicants, analyzing ads for sentimental value, and testing criminal defendants for signs of fraud [19]. Suwa et al. adopted the facial expression identification system in 1978, manifesting an attempt to automatically understand facial expressions by monitoring the motion of 20 identified points on an image series [10]. The impact of the facial expression system on social interaction and social intelligence is widely recognized. Estimates of facial motion and identification of expression are all part of facial expression analysis.

Paul Ekman and Friesen carrying out more in-depth Darwin studies have concluded that there are six transcultural and prototypic emotional expressions [7]. These basic expressions are happiness, anger, sadness, surprise, disgust, and fear. Some authors consider the neutral face as a seventh expression [7]. It is known that facial expression changes through a set or subset of prototypic emotional expressions.

Given the enormous effect that Ekman's work had on the science of emotion, his views are prevalent in emotion-detection technology, influencing many algorithms, including those marketed by Microsoft, IBM, and Amazon.

Identify applicable funding agency here. If none, delete this.

C. Goals and Objective

Our primary aim is to create a model that can separate the fundamental emotions: angry, sad, surprised, disgusted, happy, neutral, and dread, and to achieve an accuracy higher than the standard 14.29 percent. We are now determined to examine our model's output in terms of the accuracy of each depicted class. Further development of the model is likely to result in more detailed classification with a more complex variance of the condition than lab condition images. We are determined to further develop the system that might include the frequency of human speech as well as gesture recognition.

II. Literature Review

In the Emotion detection research field, several domains contribute like machine learning, natural language, neuroscience, etc. In previous researches, facial expression, voice features, or textual data are used separately to classify emotions. Emotion can be classified into several static classifications like happiness, sadness, disgust, anger, fear, and surprise [20]. It can be further improved by combining the image, voice, and textual data. This combination of data gives further improved results.

A. Convolutional Neural Network:

The emotion recognition problem, like any other classification problem, necessitates the use of an algorithm to complete feature extraction and categorical classification. To classify an emotion, we must extract specific features from data and construct a model that can classify the input based on the feature. The process is summarized below:

- Data Pre-processing:
  - The data pre-processing is to standardize the data. The typical way is to set the mean to 0 and to also divide the data by the standard deviation.
- Feature Extraction:
  - The typical method is to detect the face and extract the Units from the face, and certain emotions contain the combination of AUs code [21].
- Model Construction:
  - The conventional classifier can be either a supervised or unsupervised algorithm. A typical example of a supervised algorithm is Support Vector Machine [5], and the examples of the unsupervised algorithm include Principal Component Analysis (PCA) and Linear Discriminant Analysis.
- Label or Result Generation:
  - The typical way to generate a label or result is to find which decision boundary has the minimum Euclidean distance from the data.

B. The issues with conventional method

Classification of facial expression can sometimes be difficult, surprisingly for humans also. Several studies have shown that people can identify different feelings in the same facial expression. It's still more difficult for AI. A variety of reasons make emotion recognition difficult. These elements can be classified as technical and psychological.

C. Technical challenges

Emotion detection has many issues detecting moving object in the video: detecting a target, uninterrupted detection, deficient or unforeseeable actions and many more. We have to overcome these challenges to obtain optimum model for our work. The emotion detection model examine facial features like eyes, noses, eyebrow position, chins, mouths and other features as actuation points. Occasionally, this detection intricate due to:

- The distance between features
  - There might be some functionality in the system that treats the average distance between landmarks as an epitome and compare them only within the range.
- Feature size
  - Sometimes the model might encounter some issues with detecting irregular features, like slender or pale lips, narrow eyes and many more.
- Skin color
  - In some cases, a solution may misclassify a feature due to skin color.

According to some papers, adopting a part-based model that separates facial features into several different parts based on the physical structures of the face. The model then feeds the features into the networks with the appropriate labels [22].

1) Data: When using machine learning or deep learning algorithms problems requires large amount of training data. Our data has videos with inconsistent frame rate, various angles and backgrounds, people of different genders and nationalities.

- Training Dataset
  - The training data (fer2013) contains 48x48 pixel grayscale images of different faces. The data has been automatically registered so that the faces are nearly centered and occupies similar space in each image.
- Test Dataset
  - For the Test Dataset we have used CMU-MOSEI multimodal Dataset. We had to pre-process the video file according to our needs.

2) Face occlusion and lighting issues: Occlusion happens when the target's posture is changed. This is frequent problem in motion detection in a video, especially if the video is raw or unprepared. We can dissipate the problem by formalizing the system that is responsible for detecting the facial features in the video, identifying the features and deduce them to a 3D model of human face.We can further increase the efficiency of the model by introducing the illumination normalization algorithms.

3) Face occlusion and lighting issues: Most algorithms are designed to recognize high-intensity expressions. This might results in an inconsistent recognition of emotions when

examining people from cultures with traditions of emotional suppression.

D. CNN

CNN models are composed of three distinct substructures: convolutional layers, pooling layers, and fully connected layers [23].

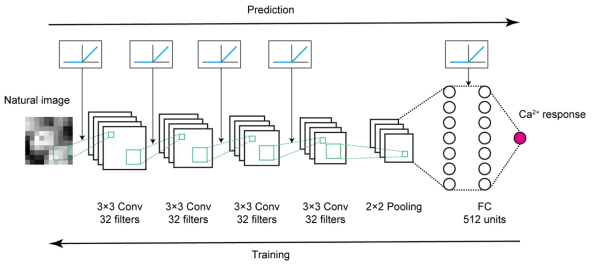

Fig 1: Convolutional Neural Network

Convolutional layer consists of many features maps. Local receptive field might be slithered when creating a new feature map over the input [24]. Different variations of data are used including images, text when performing convolution. Individual neurons in the layers may not be connected to all nodes or neurons in the preceding layer, but rather to nodes in a specific region known as the local receptive field.

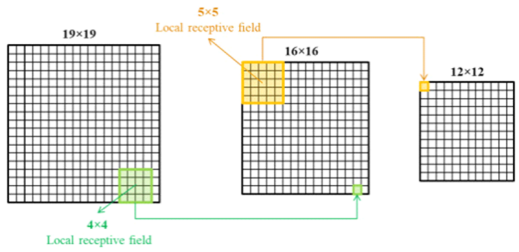

Fig 2: Local receptive field

To simplify the information, layer pooling is launched. These are commonly referred to as the subsampling layers. Pooling can be used to solve a variety of equations, including harmonic average, geometric average, and maximum pooling. The most common pooling procedures are max-pooling and average-pooling. These layers are mandatory to decrease the computational time.

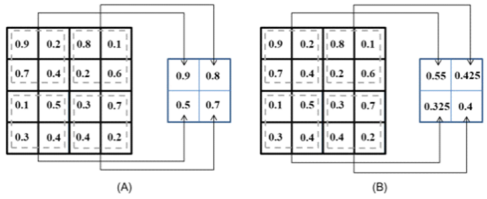

Fig 3: Max-pooling

Fully linked layers are the final layers of the CNN model, following the convolution and pooling layers. This layer allows the feature vector to be used as input data for certain machine learning problems such as prediction and classification[23]. SoftMax Classifier is another name for the final layer of completely connected layers.

$$\sigma(\mathbf{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \text{ for } i = 1, \ldots, K \text{ and } \mathbf{z} = (z_1, \ldots, z_K) \in \mathbb{R}^K$$

$$P(y = j \mid \mathbf{x}) = \frac{e^{\mathbf{x}^\mathsf{T}\mathbf{w}_j}}{\sum_{k=1}^{K} e^{\mathbf{x}^\mathsf{T}\mathbf{w}_k}}$$

Fig 4: SoftMax function

The max-pooling layer down samples the image or feature map, minimizing computation and directing the subsequent layers to concentrate on more detailed features. Finally, these three kinds of layers contribute to the convolution block's uniqueness. The completely connected layers can act as a classifier in addition to the convolutional block. The weight matrix is contained in each unit of the layer, and the output is transformed and activated to become the input of the subsequent layers of units through the linear transformation and activation function [9]. Unlike a classical linear transformation, the activation function ReLU will behave in the same manner as the convolutional layers to help the system discern the feature more easily. Output = max (input, 0)

Furthermore, the final layer is generally a SoftMax classifier, and the result is traditionally the one with the highest probability in a multinomial distribution.

classI Multinomial(i)

label = arg max.

E. Speech Process

There are two types of data for computer- structured and unstructured. The structured data is highly organized and structured for processing and analysis within databases or spread sheets. Computers handle structured data very well. However, dealing with unstructured data or data that does not have a pre-defined structure or format, such as human language, is a challenge for them.

1) Stop words: Words that are super common and doesn't carry that much of a meaning, they just connect the important words of a sentence are called stop-words. For example, in English "a", "the", "are", "is" etc. are very common in pretty much every English sentence. So, the prescience is that by removing these words, one can focus on the words that carry more prominence in a sentence or carry more information about the overall corpus. Different languages have different stop words. The Bengali language is a grammatically complex language. It has stop-words that needs to be removed for extracting meaning from a corpus. This technique commonly used in topic extraction, keyword searching, NLP classification tasks and so on.

2) Tokenization: Tokenization is the method of splitting a text stream into symbols of words phrases and other significant items called tokens of symbols may be individual phrases

of words or even whole phrases. Tokens can be phrases, individual words or even entire sentences. Any characters that are not words like punctuation marks, are discarded in the process of tokenization. This tokenized word becomes input for things such as parsing and text mining.

Thus we have used Text2Emotion for recognition of the context. Text2emotion is the python package developed with the clear intention to find the appropriate emotions embedded in the text data. The research concludes that when a person is in the thinking process and is clear approximately his statement then he will express his emotions in the right context of manner. Therefore it will be properly aligned. It gives an output as a dictionary labeling context into 5 basic emotion categories such as **Happy, Angry, Sad, Surprise**, and **Fear**.

III. Methodology

A. Workflow

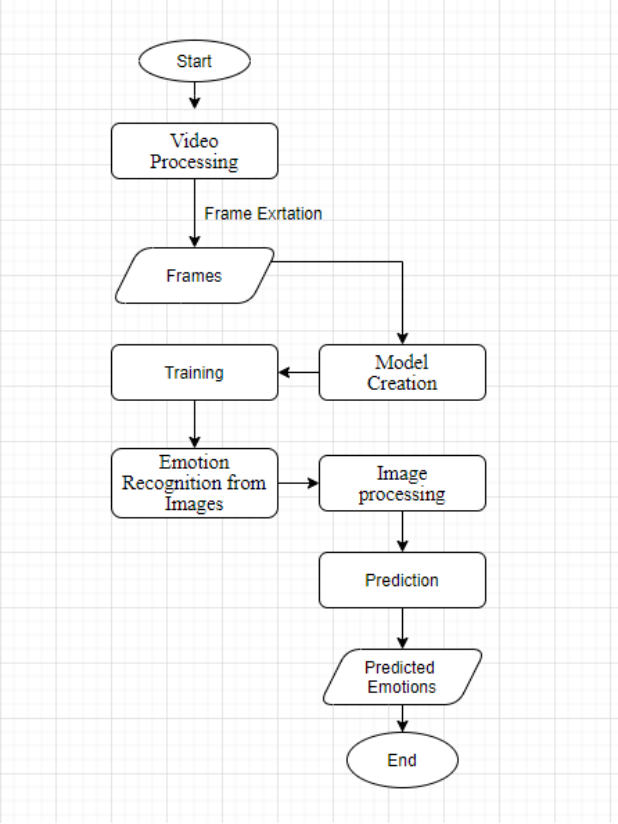

Fig 5: Work Flow (Frame recognition)

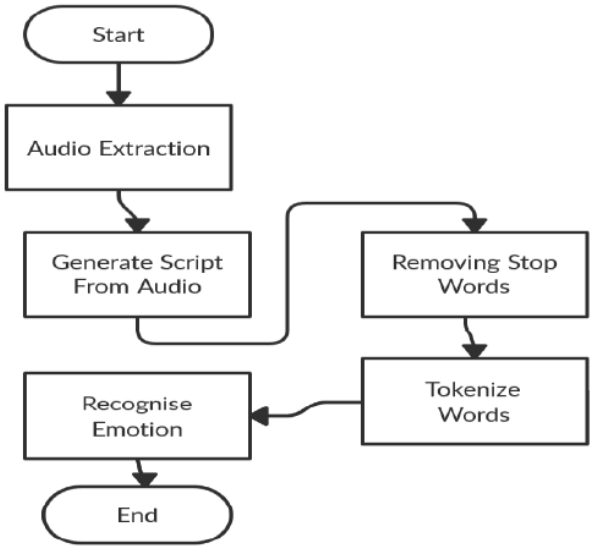

Fig 6: Work Flow (speech Recognition)

B. Video Processing

1) Video to Frame: First, we extract frame from the mp4 video which is located in Google drive. For extracting frame from video, we imported scikit-image, OpenCV-python library. First challenge is to have consistent data. Main rules in the dataset creation were:

- light conditions: records during the day
- video quality at least 640 width or above
- removed any cover with titles
- Face must be in the middle

Then we make each frame grayscale and resize it to 640. After that we stored every frame in Google Drive.

2) Algorithm: Image should go Here

3) Video to Audio: Initially, we have extracted audio from Video using MoviePy. MoviePy can edit all the several popular audio and video forms, including GIF, and runs on Windows/Mac/Linux, with Python 2.7+ and 3.

4) Reducing Noise: The default feature given by speech recognition which is (adjust for ambient noise) was used to reduce the noise of the audio.

5) Audio to Text: We have used and compared multiple voice recognition technology to achieve the best outcome. The technologies include google Text to speech, Microsoft Bing Speech, Wit, SoundHound for generating the script. On a given audio file, Google, Wit had shown similar results which are about 45% of the whole audio was extracted correctly where Bing showed comparatively better output about 61% of the whole transcript correctly. But finally, Soundhound was our API of choice for recognizing speech which is 80

C. Model Creation

For creating the model we use the Sequential API from keras library. Our Model contains four Convolutional Layers, two dense layer and one hidden layer.

1) Convolutional Layers: First Convolutional layer has 64 filters, size is 3x3, output matrix and input matrix are same. Second layer contains 128 filter, third layer contains 256 filters and last layer contains 512 filters. Other parameters were same.

Model is given bellow:

```
def createModel():
    model = Sequential()

    model.add(Conv2D(64, (3, 3), padding='same', input_shape=(48,48,1)))
    model.add(BatchNormalization())
    model.add(Activation('relu'))
    model.add(MaxPooling2D(pool_size=(2, 2), strides=None, padding='same'))
    model.add(Dropout(0.25))

    model.add(Conv2D(128, (3, 3), padding='same'))
    model.add(BatchNormalization())
    model.add(Activation('relu'))
    model.add(MaxPooling2D(pool_size=(2, 2), strides=None, padding='same'))
    model.add(Dropout(0.25))

    model.add(Conv2D(256, (3, 3), padding='same'))
    model.add(BatchNormalization())
    model.add(Activation('relu'))
    model.add(MaxPooling2D(pool_size=(2, 2), strides=None, padding='same'))
    model.add(Dropout(0.25))

    model.add(Conv2D(512, (3, 3), padding='same'))
    model.add(BatchNormalization())
    model.add(Activation('relu'))
    model.add(MaxPooling2D(pool_size=(2, 2), strides=None, padding='same'))
    model.add(Dropout(0.25))

    model.add(Flatten())

    model.add(Dense(512))
    model.add(BatchNormalization())
    model.add(Activation('relu'))
    model.add(Dropout(0.25))

    model.add(Dense(256))
    model.add(BatchNormalization())
    model.add(Activation('relu'))
    model.add(Dropout(0.25))

    model.add(Dense(7))
    model.add(Activation('softmax'))

    return model
model = createModel()
```

Fig 7: Model

D. Training

In the case of Recognition from Video Frames, the task is to categorize each face into one of seven categories based on the feeling displayed in the facial expression (0=Angry, 1=Disgust, 2=Fear, 3=Happy, 4=Sad, 5=Surprise, 6=Neutral). The educational set contains approximately 28,709 examples, while the public test set contains approximately 3,589 examples.

In case of Audio Recognition, the task is to categorize each word based on the meaning into one of five categories on the scale 0 to 1. Which are: Angry, fear, Happy, Sad and Surprise.

We train the model for 30 epoch and batch size of 8. Which game the model to have 71.5% accuracy.

E. Emotion Recognition from Images

For recognizing the images that were saved during video to frame conversion need to have some modification to fit into the model.

1) Image Processing: Firstly we need to detect the faces from the images. For face detection we use python keras.preprocessing library. Then the images were converter to have 48x48 shape.

2) Prediction: We used model.predict() function to predict the Emotion of the processed images taken form the videos. The predict() function takes an array of one or more data instances and enables us to predict the labels of the data values on the basis of the trained model. Then we take the max index of the predicted data and select the emotions.

F. Emotion Recognition from Audio

1) Removing Stop words: Words that are super common and doesn't carry that much of a meaning, they just connect the important words of a sentence are called stop-words. These should be removed too optimize and reduce valuable processing time. Here we have user NLTK (Naturak Language Toolkit) to Remove stop words from the Extracted Context.

Thus, the prescience is that by removing these words, one can focus on the words that carry more prominence in a sentence or carry more information about the overall corpus.

2) Prediction: We have used Text2Emotion python package to recognize the Emotion from the processed text extracted from the given video. Text2Emotion is a python package which automatically process texts, tokenize and extracts the emotions (Angry, fear, Happy, Sad and Surprise)

IV. Result Discussion

The Video we used, was compiled into frames of every second and text was extracted from video

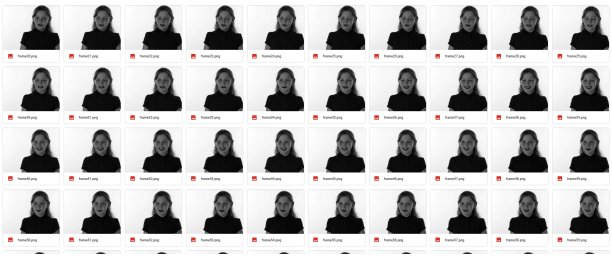

Fig 8: Frames taken from the video

100 Frames were taken throughout the whole video. Then The frames were resized into 48x48 grayscale image which was required according to our test Dataset. Using SoundHound Speech recognition Api support the recognized audio file was later transcript into a text as,

"what's the best thing about a gypsie on her period when you finger her you get here palm red for free biggest slut in history miss pac-man for twenty-five cents that is a pap smear called a pap smear because girls wouldn't do if it was called scrape the short side and while what do you call a cheap circumcision difference between a walrus and a lesbian."

After Removing stop words from the raw transcript it changed significantly into,

" what's best thing gypsie period finger get palm red free biggest slut history miss pac-man twenty five cents pap smear called pap smear girls called scrape short side call cheap circumcision difference walrus lesbian "

Thus, our text is ready for processing. Tokenization was automatically done through the process of Text2Emotion as its own feature.

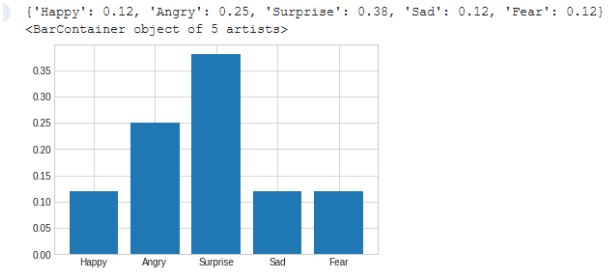

Fig 9: Results extracted from the video

The Above image shows the Result Found from the extracted context from the video. On the Other hand, from the video recognition we get,

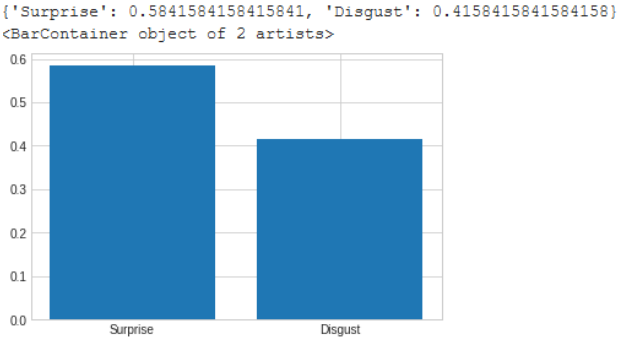

Fig 10: Video Recognition

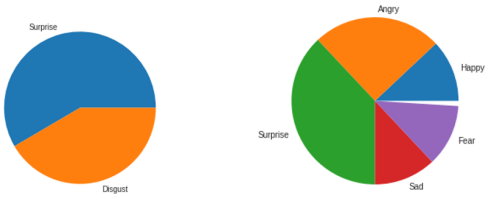

Fig 11: Pi-Chart

Then we Merged Both result Dictionaries into one final dictionary given priority to the image recognized result. Which gives us a final picture of the recognition.

'Happy':	0.12
'Angry':	0.25
'Surprise':	0.5841584158415841
'Sad'	0.12
'Fear'	0.12
'Disgust'	0.4158415841584158

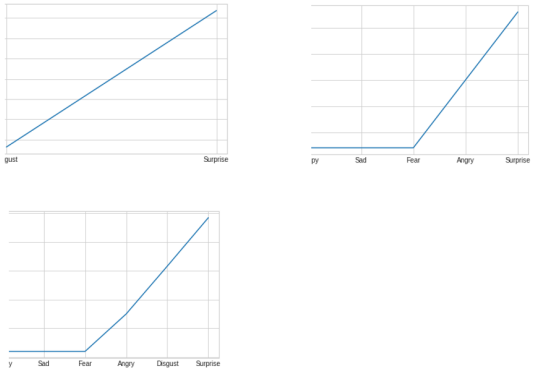

Fig 12: Merged Final Result

V. Future Work

Due to a lack of time, various tests and experiments have been postponed (i.e. the experiments with real data are usually very time consuming, sometimes requiring even days to finish a single run). Future work will focus on a more in-depth examination of specific processes, as well as pure curiosity.

There are some improvements we can do to further improve our system. For our training data set in the future, we can use a custom-made dataset rather than using preexisting data we have taken from others. We can also use the custom model for text recognition. We can also include gesture detection to further increase the accuracy of our model. But the most important aspect of our work is to decrease the cost of implementing the model in practical problems.

References

[1] Awad, W. A., ELseuofi, S. M. (2011). MACHINE LEARNING METHODS FOR SPAM E-MAIL CLASSIFICATION. International Journal of Computer Science and Information Technology (IJCSIT), 4(5), 2352–2355.

[2] Almeida,tiago. Almeida, Jurandy.Yamakami, Akebo " Spam filtering: how the dimensionality reduction affects the accuracy of Naive Bayes classifiers" Journal of Internet Services and Applications, Springer London , February 2011 pp.68–73.

[3] Trivedi, Shrwan Kumar. "A Study of Machine Learning Classifiers for Spam Detection." 2016 4th International Symposium on Computational and Business Intelligence (ISCBI), Sept. 2016, doi:10.1109/ISCBI.2016.7743279.

[4] Barbara Kitchenham, O. Pearl Brereton, David Budgen, Mark-Turner,John Bailey, Stephen Linkman(2009), Systematic literature reviews in software engineering – A systematic literature review, Elsevier.

[5] Yuchun Tang, Sven Krasser, Yuanchen He, Weilai Yang, Dmitri Alperovitch "Support Vector Machines and Random Forests Modeling for Spam Senders Behavior Analysis" IEEE GLOBECOM, 2008

[6] Jane Webster, Richard T. Watson(2002), Analyzing The Past to Prepare For the Future: Writing A Literature Review, MIS Quarterly Vol. 26 No. 2, pp. xiii-xxii.

[7] Shripriya Dongre, Prof. Kamlesh Patidar "E-Mail Spam Classification Using Long Short-Term Memory Method." International Journal of Scientific Research and Engineering Trends, vol. 5, no. 5, 2019, pp. 1659–1665.

[8] Enrico Blanzieri,Anton Bryl(2009), A survey of learning-based techniques of email spam filtering, Artif Intell Rev, DOI 10.1007/s10462-009-9109-6.

[9] Asghar, Muhammad Zubair, et al. "Sentence-Level Emotion Detection Framework Using Rule-Based Classification." Cognitive Computation, vol. 9, no. 6, 2017, pp. 868–894., doi:10.1007/s12559-017-9503-3.

[10] Samson, Andrea C., et al. “Eliciting Positive, Negative and Mixed Emotional States: A Film Library for Affective Scientists.” Cognition and Emotion, vol. 30, no. 5, 2015, pp. 827–856., doi:10.1080/02699931.2015.1031089.
[11] Strapparava, C., Mihalcea, R.: SemEval-2007 task 14: affective text. In: Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007), pp. 70–74 (2007)
[12] Ekman, P.: An argument for basic emotions. Cogn. Emot. 6(3–4), 169–200 (1992)
[13] International Survey on Emotion Antecedents and Reactions data set. https://www. unige.ch/cisa/index.php/download file/view/395/296/
[14] Das, D., Bandyopadhyay, S.: Sentence level emotion tagging. In: 2009 3rd International Conference on Affective Computing and Intelligent Interaction and Workshops, pp. 1–6. IEEE (2009).
[15] Strapparava, C., Mihalcea, R.: Learning to identify emotions in text. In: Proceedings of the 2008 ACM Symposium on Applied Computing, pp. 1556–1560 (2008).
[16] Francisco, V., Gerv´as, P.: Exploring the compositionality of emotions in text: word emotions, sentence emotions and automated tagging. In: AAAI-06 Workshop on Computational Aesthetics: Artificial Intelligence Approaches to Beauty and Happiness (2006).
[17] Shaheen, S., El-Hajj, W., Hajj, H., Elbassuoni, S.: Emotion recognition from text based on automatically generated rules. In: IEEE International Conference on Data Mining Workshop, pp. 383–392 (2014)
[18] www.theneweconomy.com/technology/the-problem-with-emotion-detection-technology
[19] X. U. Feng and J.-P. Zhang, “Facial microexpression recognition: A survey,” Acta Automatica Sinica, vol. 43, no. 3, pp. 333–348, 2017.
[20] M. S. Özerdem and H. Polat, “Emotion recognition based on EEG features in movie clips with channel selection,” Brain Inf., vol. 4, no. 4, pp. 241–252, 2017.
[21] S. K. A. Kamarol, M. H. Jaward, H. Kälviäinen, J. Parkkinen, and R. Parthiban, “Joint facial expression recognition and intensity estimation based on weighted votes of image sequences,” Pattern Recognit. Lett., vol. 92, pp. 25–32, Jun. 2017.
[22] Hongli Zhang , Alireza Jolfaei , and Mamoun Alazab, “A Face Emotion Recognition Method Using Convolutional Neural Network and Image Edge Computing,” IEEE ACCESS, Nov, 2019. 2949741
[23] H. Ma and T. Celik, “FER-Net: Facial expression recognition using densely connected convolutional network,” Electron. Lett., vol. 55, no. 4, pp. 184–186, Feb. 2019.
[24] A. V. Savchenko, “Deep neural networks and maximum likelihood search for approximate nearest neighbor in video-based image recognition,” Opt. Memory Neural Netw., vol. 26, no. 2, pp. 129–136, Apr. 2017.