# Early Threat Warning Via Speech and Emotion Recognition from Voice Calls

[1]Ifaz Ishtiak, [1] Mohammad Mazedur Rahman, [1]Md.Razaul Haque Usmani and [1]Hossain Arif

[1]Department of Computer Science & Engineering,
BRAC University,
66 Mohakhali, Dhaka, Bangladesh.

E-mail: {ifazishtiak06, riad.druvo , usmanibd14}@gmail.com, hossain.arif@bracu.ac.bd

*Abstract*— **The aim of this system is to identify potential cases of threats, and provide an early warning or alert to such cases. This will be based on voice such as voice chat over telecommunication networks or social media. The intended result will be achieved in three major steps. At first, the conversion of speech to text from both real time audio recordings and from accent groups will be applied using primarily IBM Watson's Speech to Text. This will then be used to identify possible trigger words or word patterns from a classified selection of threat-related and negative words. And finally, the same audio source will be utilized for detecting emotions from the frequency shifts through vocal feature extraction from audio input and processing it using multiple classifier algorithms such as Support Vector Machines (SVMs), Random Forests and Naïve Bayes. Libraries such as LibROSA will be applied to extract primary audio features such as Mel Frequency Cepstral Coefficients (MFCC) to generate accurate predictions. The system yields a result of approximately 84% using the SVM RBF (Radial Basis Function) kernel, which highlights the accuracy of emotion detected based on the speech.**

***Keywords— Emotion Recognition; Support Vector Machines; Speech to Text; Random Forest; Feature Extraction; MFCC***

## I. INTRODUCTION

In recent years, the advancements in networking and telecommunications has proved to be imperative for the global technological progress. Enhanced communication and instant updates in messaging, video-based communication, location tracking and even traffic conditions around the user are excellent examples of how much the world wide web has contributed to create a link around the globe. However, with its benefits, there must also be some drawbacks. Among the many are the crimes that are secretly taking place through the same instant messaging and voice calls. One article [1] illustrates this clearly; telling about a German male who ended up killing two women whom he claimed to have met in online chat rooms. This is just one of the minor cases in contrast to all the other indescribable events that have occurred to this day. This calls for taking certain measures and fast; hence the aid of machines and machine learning is required, where both voice and text can be effectively used to hint at potential danger and threat before any situation escalates.

There has been great deal of advancement in technology that has made the work of humans a lot easier and reduced the need to of large labor. But some functionalities are still being worked on for the betterment of mankind. The need of recognition of human emotion has been of great need as this can be a breakthrough in the identification of human psychology. Speaking can be of different tones which can identify how a person is meaning to say that particular speech. Thus, to identify the emotion or manner of saying behind the speech is important. This is where the need of machine learning comes into play that can identify the type of emotion being applied in a particular sentence by analyzing the vocal characteristics or in other words audio features of the particular audio sample. The features that are identified can be applied into the well-known classification algorithms in machine learning which then classifies the emotions according the matching features.

The paper is divided into the following sections: Section II highlights some of the work related to the tasks relevant to our research (STT, word recognition as well as emotion classification). This is followed by Section III, which describes the proposed design created for the system, its interface and working process. Section IV explains the implementation of the system, which includes a comparative study done in order to achieve the final result. The results are obtained in Section V, and the conclusions are drawn in Section VI.

## II. RELATED WORKS

### *A. Speech To Text*

For the Speech to text conversion, multiple approaches are observed. A similar work [2] has been done where speech to text is performed for mobile devices to enable and enhance the voice recognition and command features in the newer smartphones. Through efficient algorithms and the addition of neural network, such as the General Regression Neural Network (GRNN), over 95% accuracy in speech recognition is achieved. A VAD or Voice Activity Detection algorithm pipeline is used which involves two algorithms performing distinct task. One to calculate the signal features directly from the audio energy and the other performs similarly, but is determined by the Zero Crossing Rate in the signal and Mel Frequency Cepstral Coefficient (MFCC) is used for the feature extraction. Based on these values, it provides an estimation (VAD decision variable) of speech recognition. Moreover, the introduction of Neural Network and creation of distinct identifiers in the database which allows for particularly uttered syllable detection, boost the overall performance of the system. The recognition results are effective and similar to our approach to convert text from speech.

Fortunately, in recent times, the principles of speech to text systems has been made easy and readily available in the form of applets and online services such as the IBM Watson's Speech to Text service [29]. This allows quick conversion of speech to text in real time, provided a volunteer speaker or even multiple formats of recorded audio files. Further in the discussion, many test cases are provided which highlights the accuracy, efficiency and the convenience of having a complex program readily available which is able to provide greater accuracy than the above mentioned.

### B. Trigger Word Determination

For the detection of offensive or trigger words, [5] mentions a couple ways for recognizing profane or offensive words from a text, most of them sorts the text in some way and then compares it with one or more dictionaries, i.e. brute force, along with the incorporation of some other features. The most notable of them is the lexical syntactic feature which not only checks the offensiveness of the word but also checks the offensiveness in user level. It yields with the precision of 98.24% and recall of 94.34%. For the lexical feature extraction, Bag-of-Words used to be very popular in the early research programs but using this approach yields low accuracy in subtle offensive language detection and gives high false positive rate during heated conversation. The N-gram approach, most notably the Bi-gram and Tri-gram, is a much safer and improved approach as it also includes information of the words nearby context. For the syntactic feature extraction, we introduce natural language parsers to parse sentences on grammatical structures to avoid unrelated word sets in the offensiveness detection of the user.

### C. Emotion Recognition

For the emotion detection, there has definitely been intensive work and research on this field of how to recognize human emotion via speech or other physical traits. We have studied about such work in most of these works dealt with how extract values of sound such as pitch, amplitude, frequency and time to match the corresponding specific emotion. One work [7] has been carried out but here there has not been any use of algorithm rather the comparison of some acoustic parameters. The acoustic parameters used were mean overall fundamental frequency, overall mean energy, overall mean standard deviation of energy, mean overall jitter, mean overall shimmer. The data source for this research included a 27-year-old female and a 32-year-old male. Seven different emotions were considered for this study which were happiness, sadness, cold anger, hot anger, interest, elation and neutral. The number of sentences they have taken to compare the results of all the acoustic parameters of corresponding emotion were 70 sentences. Thus, the two subjects were exposed to neutral environment and recorded each of the emotional conditions for all the 70 sentences. The values recorded against each emotional state for each parameter was used to identify the differences in values of the parameters in each emotion.

A comparable study [9] of emotion recognition has also applied multiple machine learning algorithms, such as the SVM and Random forest, to compare between the efficiency and accuracy. The Random forest algorithm proved to be the most accurate, at 81.05%, followed by Gradient Boosting at 65.23% and finally the SVM algorithm, a comparatively poor performance score of 55.89%. Like most others, the training and testing of the model is done using the Berlin Database of Emotion Speech, on a total of 535 samples which includes 7 emotions in total: Neutral, Sad, Happy, Fear, Anger, Boredom and Disgust. The features selected are the MFCC and Energy, then are extracted and applied on the aforementioned classifiers.

This particular thesis work [10] also relates to some level to what we are trying to achieve in our work. There is similarity in the use of the MFCC (Mel Frequency Cepstral Coefficients) features to detect any sort of noise that can remotely indicate possible threats. Such noise would include glass breaking, gun shots, explosions and other forms of threatening sounds. They have used both audio and visual effects of the possible threatening situation using network cameras and the samples of sound collected are then processed by main node where a decision is made using supervised algorithm to identify and give the level of danger present in the monitored area and a signal of waning in send to crisis management center. The results after application, has shown that there has been a 79% accuracy for threat identification. They have also tried to use the SVM along with PCA but that could not match the accuracy of the single frame SVM. The other algorithm used also did not give enough accuracy that is the Dynamic Time Wrapping (DTW).

### D. The Psychology of Threat and Speech Under Stress

To justify the psychological aspect of potential threat from the emotion recognition using similar features such as energy, pitch and MFCC, etc. multiple studies have been done in the past. This is termed as stress or speech under stress. The method for identifying such a case is to compare the audio features with and without stress – whether the values are impacted greatly. This book [16] suggest that psychological threat revolves around the relation between anger, fear and anxiety; and showcases a shift diagram to prove that during stress and arousal – the excitation level of a certain emotion – the results are different than otherwise. The frequency, pitch and the speaking rate changes when one is under stress and then eventually goes back to normal as stress decreases.

Using the principle of speech under stress, more studies have been done where this psychology has been considered through the extraction of audio features in digital form and classified using suitable algorithms. This work [18] in particular, is performed on the SUSAS (Speech Under Simulated and Actual Stress) database, focusing only on the stress and its related emotions labeled Neutral, Angry, Lombard (an effect where sound quality is altered to adjust with a noisy environment) [19] and Loud. Multiple stress related features such as pitch, MFCC and other spectral features are used on a multiclass SVM to achieve around a 100% accuracy rate.

## III. System Design

As aforementioned, the aim of creating this system is to propose such a study through our prototype that is able to identify potential threats and provide alerts in likely cases. Figure 4 provides a comprehensive look at the overall working process of the proposed system. Firstly, the voice call being held will be recorded in the telecommunication device and will be send over to a server as soon as an internet connection is made; optionally it can be extracted manually as well. The audio recording can now be simultaneously utilized in two ways: first, using the IBM Watson service, the text can be reliably converted from the speech, with its effective automatic speaker diarization and sentence separation. This can allow to understand the context of the conversation better, since the speaker are separated. Next, this string is passed to the trigger word detection algorithm, which provides a few language classifications, based on the words spoken – whether they are flagged as a threat, profane or slang or just define as a negative sense of the given word (The workings, utility and results algorithm are explained in Figure 6 and further in Section 5).

On the other hand, the same audio file is utilized in the LibROSA library for feature extraction and selection. For our highest achieved results, we have chosen to extract 20 MFCC features, which is normalized for optimal distribution of data and then scaled to ensure an even range and minimize any outliers – exceptionally large or small values in the data. Sequentially, the extracted features will be evaluated using a trained classifier algorithm (in this case the SVM-RBF) to identify the emotion of the speech. Finally, the classified text and the resulting emotion from the algorithm will allow one to evaluate whether there exists any potential threat in the conversation. Overall, this system is designed with the focus to aid in the investigations of such cases where there are suspicions of danger or threat. The underlying principle of the system is that to correctly understand the message, one must first accurately interpret what the other is specifically saying and how it is being conveyed. Thus, using the speech to text, the language processing and word filtering, the theme of the conversation – particularly, the level of threat – can be understood, and the emotion recognition can help estimate *how* were the words spoken. And what is the overall mood of the conversation. Both the words and the mood can prove to be a better estimation tool instead of working distinctly with either.

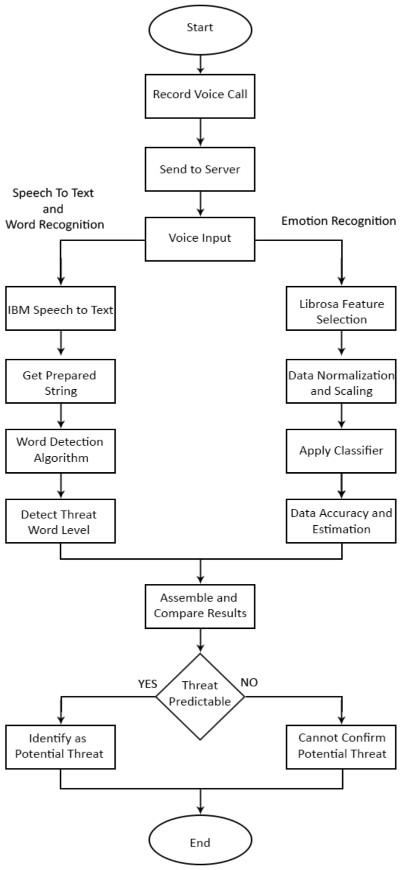

Start
Record Voice Call
Send to Server
Speech To Text and Word Recognition
Voice Input
Emotion Recognition
IBM Speech to Text
Librosa Feature Selection
Get Prepared String
Data Normalization and Scaling
Word Detection Algorithm
Apply Classifier
Detect Threat Word Level
Data Accuracy and Estimation
Assemble and Compare Results
YES
Threat Predictable
NO
Identify as Potential Threat
Cannot Confirm Potential Threat
End

Figure 4: Workflow of the Overall System

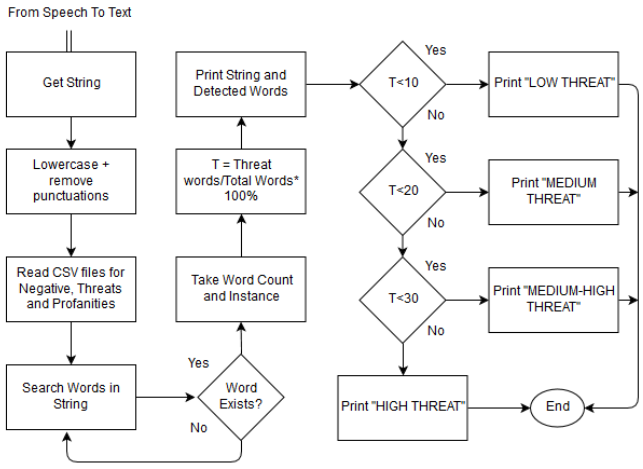

From Speech To Text
Get String
Print String and Detected Words
T<10
Yes
Print "LOW THREAT"
No
Lowercase + remove punctuations
T = Threat words/Total Words* 100%
Yes
T<20
Print "MEDIUM THREAT"
No
Read CSV files for Negative, Threats and Profanities
Take Word Count and Instance
Yes
T<30
Print "MEDIUM-HIGH THREAT"
No
Yes
Search Words in String
Word Exists?
No
Print "HIGH THREAT"
End

Figure 6: Workings of the Trigger Word Detection Algorithm

## IV. Implementation

### A. Speech To Text

To make the program versatile, the data collected is accent group recognition oriented. Vocal data from a wide variety of nationalities and dialects is sampled - both pre-recorded and spoken – and implemented with the IBM speech service to convert the audio to text form. This has led to a variety of results in terms of accuracy. The recorded data [31] consists of over two thousand voice samples reciting a few sentences (~67 words) in English.

The ones chosen for the current experiment are one from the British dialect. However, it must be considered that there are numerous accent groups of the same nationality with different backgrounds and proficiencies at the language spoken. These are the results and their accuracy comparisons of the common accents of each nationality, gathered from the IBM Watson Speech to Text service:

British: *“Please call Stella ask you to bring these things with her from the store 6 spoons of fresh snow peas 5 thick slabs of blue cheese and maybe a snack for her brother Bob. We also need a small plastic snake in a big toy frog for the kids. She can scoop these things into 3 red bags and we will go meet her Wednesday at the train station. “*

*[ 65 out of approximately 67 words correct, an above 97% accuracy]*

With this small comparison, it is shown that the IBM’s speech service alone enables users to gather satisfactory results when converting from speech to text. This data will then be extracted and implemented in an algorithm which will compare it with a set library of data to detect words and patterns in the speech which can help determine the situation of a voice conversation between people.

### B. Trigger Word Detection

The aim of the Trigger Word Detection is to detect trigger words from a string or a list of strings. The strings are obtained from the speech to text part where each line spoken by each speaker is classified as a string. The detection part is done by a brute force comparison algorithm where each word in the string is compared with a list of words, also known as dictionary, and the words that match between these two lists are listed together. This list of words shows all the trigger words in the input string. Furthermore, with the help of specific classifiers associated with each word in the trigger word dictionary, we are able to classify each of the trigger words obtained from the trigger word detection algorithm accordingly.

Refer to the algorithm given in figure 9. For the trigger word detection of the string or list of strings collected from the Speech to Text part, we are following a simple brute force comparison algorithm with a predefined set of dictionaries. The comparison is done with the help of a default function in python, called intersection, which is a lambda expression. The dictionaries consist of a list of words collected from multiple sources throughout the internet based on the desired classes; for example, a list of negative words. After collecting the lists of words, we had to refine the list from irrelevant and misleading words manually. For now, we made three classifying lists of words or dictionaries, negative word list, to find in general negative expressions; threat word list, for words that dictate actions, phrases or nouns that – commonly or otherwise – indicate threat; and profane word list, to find swears, slangs and profanities. These lists of words are converted to lists of arrays, separately, at the beginning of the algorithm. A single string is taken as input or from a list of strings and each word is separated and stored into an array. This list is, then, first compared with the list of negative words to find all the negative words within the string. This list of negative words is then displayed with a count of the total number of negative words detected.

The same process is done to the string word list with the threat word list and profane word list. The only difference is that, for the threat word list, after displaying the list of threat words detected, we calculate the percentage of threat words with respect to the total number of words in the string. This percentage is then displayed as the percentage of threat followed by an alert, which depends on the percentage of threat, where less than ten percent gives 'Low threat', less than twenty percent gives 'Medium threat', less than thirty percent gives 'Medium to High threat', and anything above that flags the result as 'High threat' alert message. The percentage of profanity is also displayed after the list of profane words detected within the word string list is displayed.

```
Step 1: Read input string
Step 2: Create list of words from input
string
Step 3: Create list of negative words
Step 4: Compare list of negative words with
list of input string words
Step 5: Display number of negative words
detected and each negative word detected
Step 6: Create list of threat words
Step 7: Compare list of threat words with
list of input string words
Step 8: Display number of threat words
detected and each threat words detected
Step 9: Calculate percentage of threat
Step 10: Define alert message according to
percentage of threat
Step 11: Display percentage of threat and
alert message
Step 12: Create list of profane words
Step 13: Compare list of profane words with
list of input string words
Step 14: Display number of threat words
detected and each threat words detected
Step 15: Display percentage of profanity
```

Figure 9: Comprehensive Methodology of the Word Detection Algorithm

### *C. Emotion Recognition*

#### *1) Feature Extraction:*

MFCC or Mel Frequency Cepstral Coefficients refers to the determining coefficients for the Mel Frequency Cepstrum. It has become one of the most widely and commonly used features in audio processing and speech recognition. This is due to its ability to identify and differentiate between logical and linguistic content between all the others influencing factors in a speech such as noise, the speaker's tone and emotions, background sound information, etc. Commonly, MFCC features are calculated in the following steps as shown in figure 12:

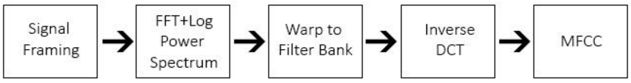


Figure 12: Process of MFCC Feature Extraction

Although 13 MFCC features are generally used we found 20 features using the LibROSA library provide better results.

Initially we import all the important libraries needed to do the feature extraction process of each audio sample from the dataset, most notably `librosa`. The first part of the program finds the path to all the audio samples of a certain emotion from the dataset using the `find_files` operation from the `librosa.util` package and stores it in a string variable. The second part of the program initializes the header for each column of each CSV files of the audio samples, in this case, from 'mfcc-1', 'mfcc-2', and so on, till 'mfcc-20'.

The third step uses the load operation from `librosa.core` package which loads the audio file and decodes it into a time series variable, in this case, y. It is stored in y as a one-dimensional NumPy floating list or array. The variable followed by y, in this case, sr, is used to store the sample rate of y, which is the number of samples per second of the audio being processed. As mentioned before, at load time, all the audio is resampled to 22050 Hz.

Using this variable y, in the fourth step, we extract the Mel-frequency cepstral coefficients, which is said to be the short term power spectrum of a signal which is derived from the linear cosine transformation of the log power spectrum on a non-linear Mel scale frequency, of each audio sample from the dataset by the help of the `mfcc` operation from the `librosa.feature` package. After running this operation, we are able to generate the `mfcc` of each audio sample in matrix form, which is a `numpy.ndarray` of size (`n_mfcc`, T), where the variable T is the duration of each track in frames and the `n_mfcc` is used to denote the number of `mfccs` to be generated, which is set to 20 by default. We can easily change this value by assigning a different number of not more than 40 to indicate the number of MFCCs that we want. Two important parameters here are hop length, which denotes the number of samples between each frames of an audio sample, and the frame length or number of Fast Fourier transform, which is the number of frames in an analysis window or frame. By default, `n_fft` is set to 2048 and the value of `hop_length` is 512 bits per sample.

Finally, after normalizing the data, this is used to sort or arrange the matrix of each audio sample in such a way that the columns represents each of the features whereas the rows represents each of the audio samples of the entire set of a particular emotion in the dataset. We take the mean or average of all the segmented samples of each audio for each feature.

#### *2) A Comparative Study*

Throughout the research, during the implementation of the tools, libraries and algorithms, many modifications have been made to each aspect of the study to achieve the optimum results. Thus, in the process, many techniques were tried, tested and altered.

Firstly, it is already mentioned that the study involved testing a couple of English language databases for the emotion recognition. The RAVDESS database is selected over TESS as it contains full sentences conveyed in one of eight different emotions, whereas the latter contains a specific word in said emotion. For the feature extraction and selection from the pyAudioAnalysis library, 34 short term features are generated in total for each audio clip. From this data set, a multitude of possible data subsets were trained and tested. These also include the utilization of manually normalizing and randomizing the data as well as the Principle Component Analysis (PCA) for feature reduction. Thus, the LibROSA library is tested next, that has parameters for hop lengths and window sizes in bits, which, according to assumption, is easily to compute rather than millisecond in time. Keeping complexity to data size ratio in mind, this time only the first 20 MFCC features are selected and extracted to be trained. The data is normalized and standardized as scale, and applied to the SVM, Random Forest and Naïve Bayes algorithms and the results produces are greatly improved (especially the SVM RBF kernel) and also in an acceptable 80%: 20% train to test ratio. In order to achieve an even higher score, the algorithm parameters were evaluated.

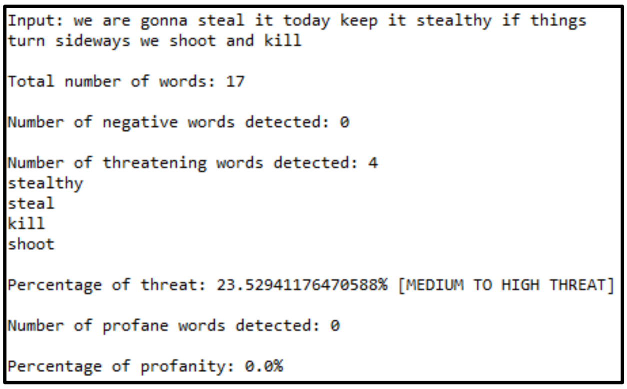
Further consideration to lower the complexity of the algorithm and raise the score led to the reduction of the number of emotions to suit the needs of the research. To specifically identify potential threat cases, and according to studies regarding speech under psychological stress, the most influential emotions were chosen – Sadness or Anxiety, Anger and Fearful. Besides these, the Calm and neutral emotions have been generalized to one and the Happy emotion is also being considered. This is for the optimization of the overall program and to identify and distinguish between cases where the threatening words may be spoken, but the emotion of the conversation shows otherwise (words uttered in a mocking, joking or light-hearted way). It ensures that the result from either word or emotion detection does not lead to completely one-sided assumptions. Hence, the quantity of emotions is reduced from 8 to 5. This change led to a greater increase in accuracy scores as shown in Figure 21. Again, the RBF kernel with C=10 is used with a random state of 0 and a gamma of 0.1 or 1 of 10. It gives an impressive 81% score.

| | Calm | Happy | Sad | Angry | Fearful |
|---|---|---|---|---|---|
| Calm | 24 | 2 | 0 | 0 | 1 |
| Happy | 0 | 36 | 1 | 0 | 4 |
| Sad | 4 | 2 | 30 | 1 | 3 |
| Angry | 1 | 6 | 1 | 38 | 0 |
| Fearful | 3 | 2 | 4 | 2 | 27 |
| Accuracy: 155/192 = 81% | | | | | |

Figure 21: Test Scores Using SVM RBF Classifier on 5 Emotions

## V. Results

To reiterate, the study proposes a prototype of a system that is divided in three parts: speech to text, word recognition and emotion recognition. At the end of the research, from varieties of options available in each part, the ones with the most accuracy, convenience and efficiency has been used. For the speech to text, a readily available system such as the IBM Watson Speech to Text Service proves to be sufficient, with as high as 95% accurate predictions of words spoken. For the word recognition, a simple program is applied that uses the text from the previous part, highlights the different criteria of words listed to be as the trigger words, as well as provides a percentage of threatening or threat related words used in the conversation and flags the level of threat. Results shown in Figure 22.



```
Input: we are gonna steal it today keep it stealthy if things
turn sideways we shoot and kill

Total number of words: 17

Number of negative words detected: 0

Number of threatening words detected: 4
stealthy
steal
kill
shoot

Percentage of threat: 23.52941176470588% [MEDIUM TO HIGH THREAT]

Number of profane words detected: 0

Percentage of profanity: 0.0%
```

Figure 21: Results Achieved from Trigger Word Detection

For the emotion recognition, after numerous changes and modifications being made and various tests of dataset combinations being conducted, we arrived at a high score of 84% accuracy in the emotion recognition. In comparison to the Random Forest, Naïve Bayes, Polynomial and Linear SVM classifiers, the Radial Basis Function (RBF) SVM performed the best. This trained model can then be used on other audio recordings directly through a library like pyAudioAnalysis to get the predicted emotion (Figure 18(ii)), or broken down into features and processed manually through the LibROSA library. Despite the ease of use of the pyAudioAnalysis library, LibROSA provided the current high score from its feature extraction. Of a wide variety of features tested on, the 20 MFCC features extracted by LibROSA has given the best results. As such, the train-test split is a standard 80-20 and the parameters tuned for the classification are: Gamma set to 0.2 or 1 of 5 features, C =10, and random state set to 0. Therefore, of the 960 feature vectors from 5 emotions, 192 were tested and 161 were correctly predicted (Figure 23).

| | Calm | Happy | Sad | Angry | Fearful |
|---|---|---|---|---|---|
| Calm | 23 | 0 | 0 | 0 | 4 |
| Happy | 0 | 34 | 1 | 2 | 4 |
| Sad | 2 | 1 | 35 | 0 | 2 |
| Angry | 1 | 5 | 0 | 39 | 1 |
| Fearful | 2 | 1 | 4 | 1 | 30 |
| Accuracy: 161/192 = 84% | | | | | |

Figure 23: Highest Accuracy Obtained for the Emotion Recognition

## VI. Conclusion

The level of criminal offences greatly rising and risking the lives and wealth of individual nations as a whole could be cut down in great proportion if prior knowledge of such actions can be recognized in advance. The use of this system can be of assistance in concluding possible criminal activities in a shorter time period by recognizing threat possibilities. The drawback of breaching personal privacy thus cannot limit the access of this system only to the government or safety and defense sector where the risk of misuse of this information is limited to the maximum. This line of work has been deeply considering in forms of human emotions but has yet to be executed in identifying threat. The thesis here tries to find the possible threat detection and also levels of threat to give an extra edge in deciding whether the speaker has any motive in executing threatening activities.

## References

[1] “Internet killer admits murdering women he met in online chat rooms,” *The Telegraph*, 15-Jan-2009.[Online].Available:https://www.telegraph.co.uk/news/worldnews/europe/germany/4243030/Internet-killer-admits-murdering-women-he-met-in-online-chat-rooms.html. [Accessed: 13-Nov-2018].

[2] R.Sandanalakshmi, P.A. Viji, M.Kiruthiga, M.Manjari and A.Sharina, “ Speaker Independent Continuous Speech to Text Converter for Mobile Application”, arXiv:1307.5736 [cs], Jul. 2013.

[3] N.Sharman and S.Sardana, “A Real Time Speech to Text Convesion System Using Bidirectional Kalman Filter In MATLAB”, In *the International Conference on Advances in Computng, Communications and Informatics (ICACCI)*, Jaipur, pp.2353-2357, 2016.

[4] S.Sultana, M.A.H. Akhand, P.K.Das and M.M.H.Rahman, “Bangla Speech-to-Text Conversion Using SAPI”, In *the International Conference on Computer and Communication Engineering (ICCCE),* Kuala Lumpur, pp.385-390, 2012.

[5] Y.Chen, Y.Zhou, S.Zhu and H.Xu, “Detecting Offensive Language in Social Media to Protect Adolescent Online Safety”, In *the International Conference on Privacy, Security Risk and Trust* and *the International Conference on Social Computing,* Amsterdam and The Netherlands, pp.71-80, 2012.

[6] S.Casale, A.Russo, G.Scebba and S.Serrano, “Speech Emotion Classification Using Machine Learning Algorithms”, In *the IEEE International Conference on Semantic Computing,* Santa Clara CA, pp.158-165, 2008.

*[7]* S.P.Whiteside, ”Simulated Emotions: An Acoustic Study of Voice and Perturbation Measures”, In *the International Conference on Speech and Language Processing (ICSLP),* Sydney, 1998.

[8] B.Reichardt, O.Julian, B.Zapata, K.Saurty and E.Fleißwasser,”Pitch Tracking – Comparison of Different Algorithms for Pitch Tracking”.[Online]. Available: https://nats-www.informatik.uni-hamburg.de/pub/SLP16/WebHome/POSTER-pitch-tracking-poster.pdf. [Accessed: 13-Nov-2018].

[9] M.Ghai, S.Lal, S,Duggal and S.Mani, “Emotion Recognition on Speech Signals Using Machine Learning”, In *the International Conference on Big Data Analytics and Computational Intelligence (ICBDAC),* Chirala, pp.34-39, 2017.

[10] A.Glowacz and G.Altman, “Automatic Threat Classification Using Multiclass SVM from Audio Signals”, In *Proceedings of the IEEE International Conference on Emerging Technologies & Factory Automation,* Krakow, 2012.

[11] Y.Lin and G.Wei, “Speech Emotion Recognition Based on HMM and SVM”, In *the International Conference on Machine Learning and Cybernetics,* Guangzhou, pp.4898-4901, Aug. 2005.

[12] P.Shen, Z.Changjun and X.Chen, “Automatic Speech Emotion Recognition Using Support Vector Machine”, In *Proceedings of the International Conference on Electronic & Mechanical Engineering and Information Technology,* Harbin, pp. 621-625, Aug. 2011.

[13] S.R.Livingstone and F.A.Russo, ”The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A dynamic, multimodal set of facial and vocal expressions in North American English”, PLOS ONE. 13. e0196391. 10.1371/journal.pone.0196391, May 2018.

[14] T.Giannakopoulos, “pyAudioAnalysis: An Open-Source Python Library for Audio Signal Analysis”, PLOS ONE. 10. e0144610. 10.1371/journal.pone.0144610, Dec. 2015.

[15] T.Giannakopoulos and S.Petridis, “Fisher Linear Semi-Discriminant Analysis for Speaker Diarization”, In *the IEEE Transactions on Audio, Speech, and Language Processing,* vol. 20, no.7, pp.1913-1922, Sept. 2012.

[16] H.Hollien, *Forensic Voice Identificaiton.* London: Academic Press, 2002, pp.51-57.

[17] J.H.L.Hansen and S.Patil, *Speech Under Stress: Analysis, Modeling and Recognition,* Chapter Speaker Classification I of the series Lecture Notes in Computer Science, vol. 4343, pp. 108-137, Jan. 2007.

[18] S.Besbes and Z.Lachiri, “Multi-class SVM for Stressed Speech Recognition”, In *the 2nd International Conference on Advanced Technologies for Signal and Image Processing (ATSIP),* Monastir, Mar. 2016.

[19] J.H.L.Hansen and S.E.Bou-Ghazale, “Getting Started with SUSAS: A Speech Under Simulated and Actual Stress Database”, In *the European Conference on Speech Communication and Technology (EUROSPEECH),* Rhodes, Sept. 1997.

[20] I.Jolliffe, “Principal Component Analysis”, *International Encyclopedia of Statistical Science,* M.Lorvic (eds), Berlin, p.22, 2011.

[21] M.S.Likitha, S.R.R.Gupta, K.Hasitha and A.U.Raju, “Speech Based Human Emotion Recognition Using MFCC”, In *the International Conference on Wireless Communication, Signal Processing and Networking (WiSPNET),* Chennai, pp.2257-2260, Mar. 2017.

[22] S.Basu, J.Chakraborty and M.Aftabuddin, “Emotion Recognition From Speech Using Convolutional Neural Network with Recurrent Neural Network Architecture”, In *the 2nd International Conference on Communication and Electronics Systems (ICCES),* Coimbatore, pp.333-336, Oct. 2017.

[23] H.Misra, S.Ikbal, H.Bourlard and H.Hermansky, “Spectral Entropy Based Feature for Robust ASR”, In *the IEEE International Conference on Acoustics, Speech and Signal Processing,* Montreal, Quebec, pp.193-196, May 2004.

[24] S.Lee, J.Kim and I.Lee, “Speech/Audio Signal Classification Using Spectral Flux Pattern Recognition”, In *the IEEE Workshop on Signal Processing Systems*, Quebec City, Quebec, pp.232-236, Oct. 2012.

[25] F.Rong, “Audio Classification Method Based on Machine Learning”, In *the International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS),* Changsha, pp.81-84, Dec. 2016.

[26] L.Grama, L.Tuns and C.Rusu, “On the Optimization of SVM Kernel Parameters for Improving Audio Classification Accuracy”, In *the 14th International Conference on Engineering of Modern Electric Systems (EMES),* Oradea, pp.224-227, Jun. 2017.

[27] A.Mertins, “Filter Banks”, In *Signal Analysis: Wavelets, Filter Banks, Time-Frequency Transforms and Applications,* New York: John Wiley & Sons, Inc., 1999, pp.143-195.

[28] E.Frank, M.A.Hall and I.H.Witten, “Data Mining: Practical Machine Learning Tools and Techniques (Online Appendix)”, *The WEKA Workbench,* 2016.

[29] IBM, *IBM Watson Speech to Text Service Demo,* 2015.

[30] Neutrino, *Neutrino Bad Word Filter API*, 2014.

[31] R.Tatman and S.Weinberger, *Speech Accent Archive,* 2013.

[32] B.McFee, C.Raffel, D.Liang, D.P.W.Ellis, M.McVicar, E.Battenberg and O.Nieto, “Librosa: Audio and Music Signal Analysis in Python”, In *Proceedings of the 14th Python in Science Conference,* Austin, Texas, pp.18-25, Jul. 2015.

[33] F.Pedregosa *et al.,* “Scikit-learn: Machine Learning in Python”, In *the Journal of Machine Learning Research*, vol.12, pp.2825-2830, 2011.

[34] J.D.Hunter, “Matplotlib: A 2D Graphics Environment”, In *Computing In Science & Engineering,* vol.9, no.3, pp.90-95, Jun. 2007.

[35] K.Dupuis and M.K. Pichora-Fuller, *Toronto Emotional Speech Set (TESS),* 2010.