Threat Detection System Using Data-science and NLP

Saundarya Yerawar[1], Vishal Khokad[2], Sagar Miraje[3], Miss. Aprupa Pawar[4]

[1,2,3]B.Tech Student, Department of Computer Science and Engineering, Walchand College of Engineering, Sangli, Maharashtra, India - 416415
[4]Faculty, Department of Computer Science and Engineering, Walchand College of Engineering, Sangli, Maharashtra, India – 416415

ABSTRACT:

We know that intelligence department works hard to identify potential threats for integrity of our country. They identify suspsects by various. sources one of which is tapping phone calls of suspsects which are identified by them and detect potential threat caused by them for national .security. Now increase in the population makes this task very difficult identifying all the suspect and detecting threats through their conversation is too much of task to do. So, to simplify this task we propose a threat detection system which will take audio signals and identify threatful. Conversation from it. Here, we have used various techniques as a different modules in the project to achieve our goal of giving intelligence department. A tool which can be used efficiently track threats for our country.

1. INTRODUCTION:

In 2001 Parliament House of India was under attack of terrorists which belonged to Lashkar-e-Toiba and Jaish-e-Mohammad which are the two Pakistan raised terrorist organisations. All terrorists were killed and led to nine deaths(including 6 police, two parliament security service personnel and gardener). These attacks conducted with the help of people who live in India. This attack may not have huge number of casualty but capital being attacked is huge hole in security. This attack was possible only due to some anti nationals provided information to terrorist organisation. So to identify such peoples we have proposed our system which could identify the such anti national peoples. Also get threatful conversations made and identify threat well before attacks. The Intelligence department does a tremendous job to secure the people from potential threats by identifying suspicious people. Identifying suspicious people is not an easy job in a country of 1.3Bn people and keeping an eye on each conversation of these people is also a tremendous job. Our proposed model can simplify the above task and efficiently identify suspicious people. Hence with our idea we can help security departments to identify potential threats and counter them. Our aim is to build a model which will help our country in the above mentioned manner. Presently, computers have already replaced a tremendous number of humans in many creative professions. Therefore, Artificial Intelligence areas are composed of Machine Learning, Natural Language Processing, Computer Vision and Robotics. Similarly, speech recognition can be predicted by using computers. Previous research uses deep learning to be used in speech recognition by using an audio library from Google which has 66.22% accuracy. Based on the experimental results, this research can be applied to speech recognition. In other ways, this research will make the computer more intelligent and capable. Researchers used data from the Google audio Set, which is Google's voice data warehouse to be used as training and test data. Natural language processing (NLP) has recently gained much attention for representing and analysing human language computationally. It has spread its applications in various fields such as machine translation, email spam detection, information extraction, summarisation, medical, and question answering etc. The paper distinguishes four phases by discussing different levels of NLP and components of Natural Language Generation (NLG) followed by presenting the history and evolution of NLP, state of the art presenting the various applications of NLP and current trends and challenges. The proposed system is unique in its concept. As there are many text sentiment analysis models and speech sentiment models developed but are not very effective and efficient. Our proposed model will identify audio threat signals and only those can be put under observations and this will direct the security agency in the right direction and minimise overhead of the intelligence department.

2. BACKGROUND AND RELATED WORK:

Natural language processing applications are text processing ,classification and clustering applications. The simple text analytics application is tweet classification application. The tweets are classified as positive , negative and neutral based on the content of that tweet.

With help of Term frequency and Inverse document frequency we can determine most relevant from the dataset.The count of words in a text document is known as term frequency and how frequent the word present in the

document is known as document frequency.The relevance of words from text document is calculated using TF-IDF. Words with high TF-IDF numbers imply a strong relationship with the document they appear in, suggesting that if that word were to appear in a query, the document could be of interest to the user. This algorithm has been proposed to increase the relevance for a particular query. Machine Learning algorithms such as Naive Bayes classifier, Decision Trees, Random Forest, Support Vector Machines and boosting algorithms are proven to be successful in text classification problems.

This paper is on "Detecting potential threat signals from text documents and classifying them on severity of threat'' uses a Support Vector Machine (SVM) classifier to perform the text document classification. Support Vector Machine classifier uses optimal hyperplane to perform the classification task. This paper aims at converting audio signal into text transcript then perform sentiment analysis on it and then classify it based on severity of threat.

Text data is unstructured data. Before passing text to machine learning algorithm, text should be converted into vector format. Text data converted into vector format by various techniques Some techniques use word occurrence while some other techniques use word frequency.

2.1 Algorithms

2.1.1 Machine Learning Algorithms:

Machine Learning algorithms are wont to perform classification and regression. The algorithms used for classification are logistical Regression for binary classification, call Trees, Random Forest Classifier, Support Vector Machines, Naive mathematician Classifier, and Boosting Algorithms.

2.1.2 Logistic Regression:

Logistic Regression may be a binary classification formula wont to predict output as o or one. The output of logistical regression continuously states the likelihood of being one. logistical regression works constantly because the sigmoid activation performs wherever the output falls during a vary of zero and one.

2.1.3 Naive Bayes Classifier:

Naïve mathematician formula may be a supervised learning formula, that is predicated on mathematician theorem and used for determination classification issues. It is primarily utilized in text classification that has a high-dimensional coaching dataset. Naïve mathematician Classifier is one every of the straight forward and only Classification algorithms that help in building the quick machine learning models which will build fast predictions. It is a probabilistic classifier, which implies it predicts on the premise of the likelihood of the associate object. Some standard samples of the Naïve mathematician formula are spam filtration, Sentimental analysis, and classifying articles. Bayes' theorem is additionally referred to as Bayes' Rule or Bayes' law, which is employed to work out the likelihood of a hypothesis with previous data. It depends on the chance.

The formula for theorem is given as:

$$P(A|B)= \frac{P(B|A)P(A)}{P(B)}$$

Where,

P(A|B) is Posterior probability: the likelihood of hypothesisA on the determined event B.

P(B|A) is chance likelihood: likelihood of the proof *provided* that the probability of a hypothesis is true. P(A) is previous likelihood: Probability of hypothesis before perceptive the proof.

P(B) is Marginal likelihood: Probability of proof.

2.1.3 Support Vector Machine:

Support Vector Machine or SVM is one every of the foremost standard supervised Learning algorithms, that is employed for Classification still as Regression issues. However, primarily, it's used for Classification issues in Machine Learning.

The goal of the SVM formula is to form the simplest line or call boundary which will segregate n-dimensional area into categories so that we will simply place the new information within the correct class within the future. This best call boundary is termed a hyperplane. SVM chooses the intense points/vectors that facilitate making the hyperplane. These extreme cases are referred to as support vectors, and thus the formula is termed as Support Vector Machine.

3. PROPOSED SYSTEM:

Our system comprises of total four steps which will be executed sequentially in order to detect threat patterns in the input sample and then classifying them into flags based on the severity of threat. Below is the flowchart showing flow of the entire system.

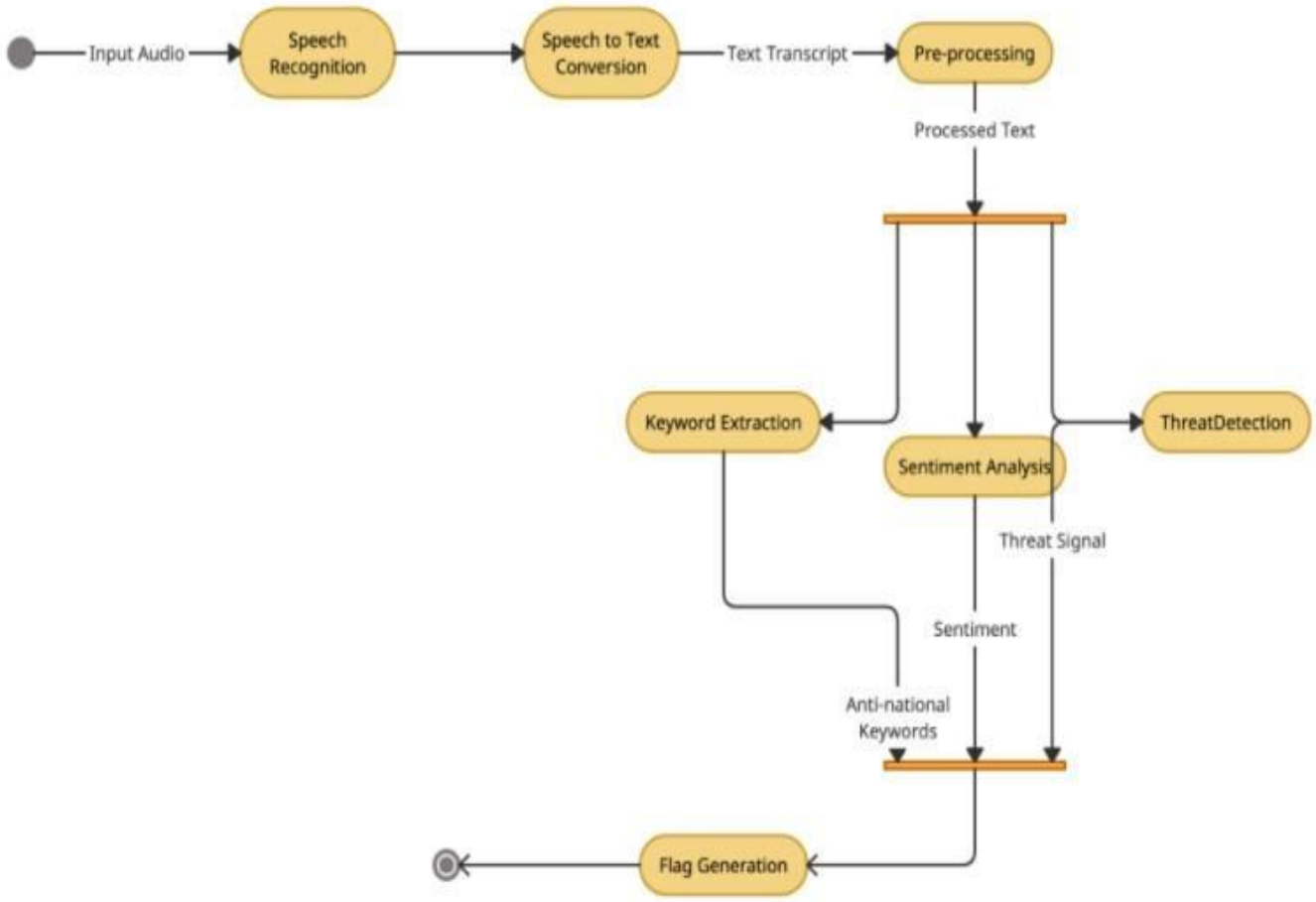

Chart-1: Flowchart

Step 1: Speech Recognition

In the first step, we take input from the user in the form of an audio/text file. Processing audio data can be a much tedious task and processing text data will be much efficient. If the input file is a text file then the process will start from the second step and if the input is an audio file then in the first step it will be converted into a text file and then it will be given to the second step as an input. For this conversion of audio to text, we have used the SpeechRecognition package in python. This package helps in recognising input audio. It consists of a recogniser class whose instance will be used. Each instance of recogniser class consists of various methods which can read various audio sources. They use various API's based on the

user requirements. Here we are using Google speech recogniser API for converting input from speech to text as it is the best accuracy among other API's. In the speech also we have two types of inputs one as audio file another is direct audio from the microphone which is being recorded live. Your microphone input required pyaudio. pyaudio consist of listen method which takes the input from the microphone. Microphone input can be taken in case of real-time monitoring.

Step 2: Keyword Extraction

The second module is keyword extraction here we will be extracting frequently occurring Anti National keywords from the input text and maintaining their account. this keyword extraction process consists of various types such as preprocessing where we remove punctuation marks and special characters from the text then there is stemming where it removes suffixes and prefixes from the word roots and the last one is limited edition air it maths remaining root forms back to the actual words. after this all preprocessing where we remove all these stop words, punctuations and convert to words back to its root forms and map them to its original meaning we all get the word which will give some meaning to the sentence. then with the help of the dataset, we will choose all the anti-national keywords and the code words were used by the terrorist to communicate between them. Only detecting those keywords will not help so we have to detect that if those keywords are being used in the context of national threat, so we will detect this by maintaining a list of all the major cities and all the countries with their capitals and all the words referring to country Nation or city and finding those words in the sentence in which the anti-national words are identified. After identifying those words we will maintain their record and will be using them in the last in the classification of the input record in different flags.

Step 3: Sentiment Analysis

the third step is sentiment analysis, this step takes the output of the first step as an Input and performs sentiment analysis on it. The basic task of sentiment analysis is to classify the text based on its polarity as positive negative or neutral. For sentiment analysis we have used NLTK which means natural language Toolkit it contains packages to make the machine understand human languages and also it has the most powerful Natural Language Processing libraries. so how this sentiment analysis is performed as we have already said that NLTK has a rich set of libraries which includes all the required data sets and function so in that libraries only it maintains set of all the words that depict some emotion and link those words to the emotion.

This way while processing when we encounter the word which depicts any emotion it will be captured and the emotion with that word is representing will be added to the list of sentiments. now all of the sentiments/emotions are already labelled as positive or negative sentiments/emotions like the words such as enjoy happy good denote positive sentiment and the words such as hate or beat denotes negative sentiments. Now will use a polarity score checker it calculates the probability of positive sentiments negative sentiment and neutral sentiment into the text. after that for classification of text into positive negative or neutral, it checks if the probability of positive sentiments is greater than the probability of negative sentiments and if it is so then the entire text will be classified as positive. And vice versa, if the probability of negative sentiments is greater than the probability of positive then the entire text will be classified as negative. and if the probability of text being positive is equal to the probability of text being negative then the text will be classified as a neutral text.

Step 4: Threat Detection

In the fourth step we are classifying the text into threatful or not threatful category. here we are using a machine-learning algorithm to classify it. as you can see in the experimental results we have tried Logistic regression, multinomial naive Bayes algorithm and support vector machine algorithm. out of these three algorithms, we got the highest accuracy with the support vector machine algorithm So we have decided to go with it. we have used the dataset that contains the text from Wikipedia talk page edits and divided the 80% of the dataset for training and 20% of the dataset for testing. testing on this data set our SVM model got an accuracy of 97%. so we have built the model of this and using this model to classified our text. support vector machine is a supervised machine learning algorithm that can be used for both classification or regression challenges. the objective of the support vector machine is to find a hyperplane in an n-dimensional space that distinctly classifies the data points where n is the number of features. To separate the two classes of data points there are many possible hyperplanes that could be chosen.

Our objective is to find a plane that has the maximum margin i.e the maximum distance between data points of both classes.

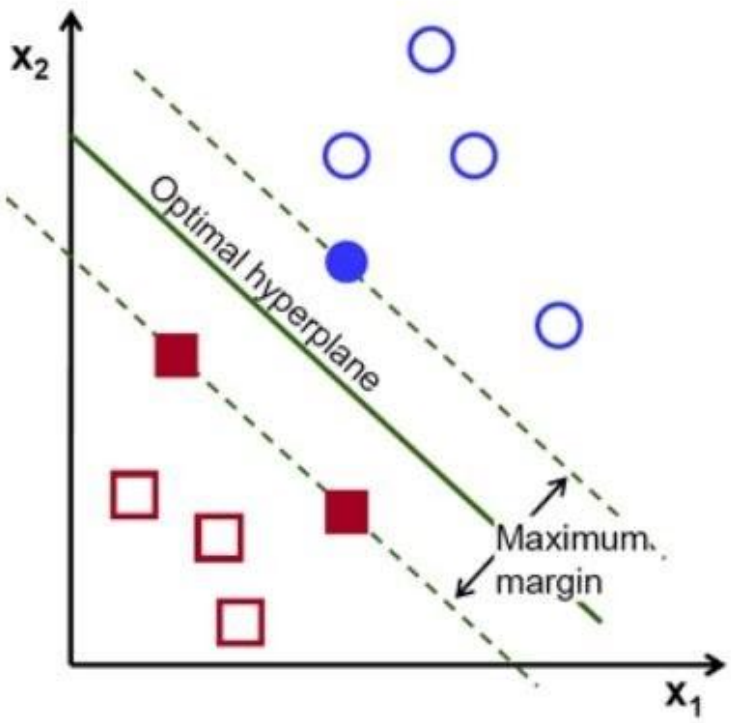

Fig-1: Optimal Hyperplane

Maximum margin distance provides some reinforcement so that future data points can be classified with more confidence. we have used SVM because its accuracy is more than other algorithms as well as it works really well with a clear margin of separation. also, it is more effective in 2-dimensional spaces and it is much memory efficient. so after successful training and testing of the model when we give our text file which is generated as an output of the first module as an input to the model for classification, it accurately classifies the text as threatful or not threatful. After the completion of these four steps, based on the output of the steps, we classify text into three flags green, yellow and red. like if the text contains anti-national keywords, its sentiment is negative and it is classified as a threat full text it will be labelled as a Red flag, if the text does not contain a single anti-national keyword, its sentiment is positive and it is classified as not threatful then it will be labelled as green flag otherwise it will be labelled as yellow flag.

4.EXPERIMENTAL RESULTS:

The machine learning models are tuned and evaluated on cross-validation data,It is part of training data.By performing Exploratory data analysis(EDA) the models are tuned with parameters.The models proposed with the current hyper-parameters provide better evaluation results than the machine learning algorithms. For each of the modelling techniques used the results are captured.

Table-1: Accuracy of different classification algorithms

Classification Algorithm	Accuracy of model
Logistic Regression	92.33
Naive Bayes	87.55
Support Vector Machine	95.33

The Support Vector Machine model tends to outperform the machine learning models. The below metric shows how the accuracy of the Support Vector Machine model is changing by changing the hyper-parameters.

Table-2: Accuracy of the Support Vector Machine model with different hyper-parameters.

Kernel	C	Gamma	Features	Accuracy
poly	1	1	5000	89
poly	0.1	1	5000	86
poly	0.1	10	5000	89.5
linear	0.1	10	5000	91.3
linear	1	1	5000	96.16
linear	1	1	10000	96
linear	1	1	20000	95.83
linear	1	1	89618	95.33

5. CONCLUSIONS:

While researching for this project we found very less research done regarding this project as a whole. This issue is not yet addressed efficiently. So, We have proposed a model which can identify the severity of threat in the audio conversation. For this, we have included four modules. These modules include audio to text transcript then processing these text files to extract keywords then performing sentiment analysis and finally categorising into flags to show the severity of threat in conversation.

Also as a future work we will be trying to generalise the system for different applications just by doing some small modifications in the classification rules and by replacing the dataset according to requirements.

6. REFERENCES:

https://wordnet.princeton.edu/

https://research.google.com/audioset/

https://www.researchgate.net/post/How-do-you-extract-keywords-from-text-Which-good-NLP-tools-are-available

BatoulAljaddohu, Nishith Kotak, Document Text Classification Using SVM (2020).

Shweta Mayor, Bhasker Pant, Document Classification Using SVM (2012).

Saurav Sahay, Support Vector Machines and Document Classification (2004).

The dataset contains text from Wikipedia's talk page edits. https://bit.ly/3uKYJQK