Multi-modal Hate Speech Detection using Machine Learning

Fariha Tahosin Boishakhi
Computer Science and Engineering
BRAC University
Dhaka , Bangladesh
fariha.tahosin.boishakhi@g.bracu.ac.bd

Ponkoj Chandra Shill
Computer Science and Engineering
BRAC University
Dhaka, Bangladesh
ponkoj.chandra.shill@g.bracu.ac.bd

Md. Golam Rabiul Alam
Computer Science and Engineering
BRAC University
Dhaka , Bangladesh
rabiul.alam@bracu.ac.bd

2021 IEEE International Conference on Big Data (Big Data) | 978-1-6654-3902-2/21/$31.00 ©2021 IEEE | DOI: 10.1109/BigData52589.2021.9671955

***Abstract*—With the continuous growth of internet users and media content, it is very hard to track down hateful speech in audio and video. Converting video or audio into text does not detect hate speech accurately as human sometimes uses hateful words as humorous or pleasant in sense and also uses different voice tones or show different action in the video. The state-of-the-art hate speech detection models were mostly developed on a single modality. In this research, a combined approach of multi-modal system has been proposed to detect hate speech from video contents by extracting feature images, feature values extracted from the audio, text and used machine learning and Natural language processing.**

***Index Terms*—Audio hate Speech, Video hate Speech, Hate Speech detection, Machine Learning, Multi-modal Hate Speech detection.**

I. Introduction

In this era of digital communications, hatred data is not only in social media comments and posts or text messages but also in voice messages and video content as well [1]. Content like this causes cyberbullying, rioting, fraud, loss of respect, and even murder. (Hate Crime Statistics, 2019) shows 15588 law enforcement agencies reported crimes, suspects, offenders, and hate crime zones. These groups reported 7314 hate crimes with 8,559 offenses. The findings include 57.6%race/ethnicity/ancestry/bias, 20.1 % religion, and 16.7 % sexual orientation. [2] In another research, online hate crimes frequently start online and affect us offline. They were also mistreated online, according to the report's victims. The research shows that hate data is linked to voice tone and facial emotions. Voice tones and facial expressions are crucial aspects to discern hatred. In some cases, only the text data, facial expressions or vocal information as audio data is not enough individually to detect the hateful conversations. Almost all current research on hate speech identification uses text data. This study suggests integrating audio, video, and text elements to detect hate speech. The following research is implemented by taking all the modes available to deliver hate speech into account and designed to detect hate speech more precisely. The final conclusion of hate speech is determined by merging the results of a hard voting ensemble or majority voting where we combine all model result image, audio, and text to determine the final output.

The major contributions of this research can be summarized as follows -

- Data has been prepared with both hateful and non-hateful speech video from different sources such as YouTube, EMBY mostly taken from movies or series. Afterwards, image, audio and text data has been extracted from the video contents.
- Feature Extracted from images, audio and text individually. For feature selection, Recursive Feature Selection (RFE), Maximum Relevance - Minimum Redundancy (MRMR) is used to take the most relevant features.
- A hate speech detection model has been developed with images, audio and text separately and then combining the results with a hard voting ensemble model to determine the final outcome of hate speech.
- The performance of seven different classifiers has been studied on the hate speech data set to show the comparative study of different classical machine learning models in hate speech detection.

II. Literature Review

Warner and Hirschberg [3], for detecting hate speech from websites containing hateful words, epithets, phrases, stereotypical thought, anti-racial content used SVM and with an accuracy of 94% in classifying anti-semantic speech. The dataset was collected from Yahoo!! and American Jewish Congress.

Kshirsagar et al.(2018) [5] uses deep learning to categorize online hate speech in general as well as a specific racist and sexist speech using pre-trained word embedding and max/mean from basic fully connected embedding transformations. A new data set that better represents nuanced kinds of hate speech is recommended by the research.

Arora and Singh (2012), mentioned three different approaches for speech recognition, which are acoustic-phonetic approach, pattern recognition, and Artificial intelligence and the paper provided a comprehensive survey of research in speech recognition systems [6]. After investigating an application of deep neural network architectures to do hate speech detection from tweets, Badjatiya et al.(2017) experimented with multiple classifiers like Logistic regression, SVM, Random forest, and

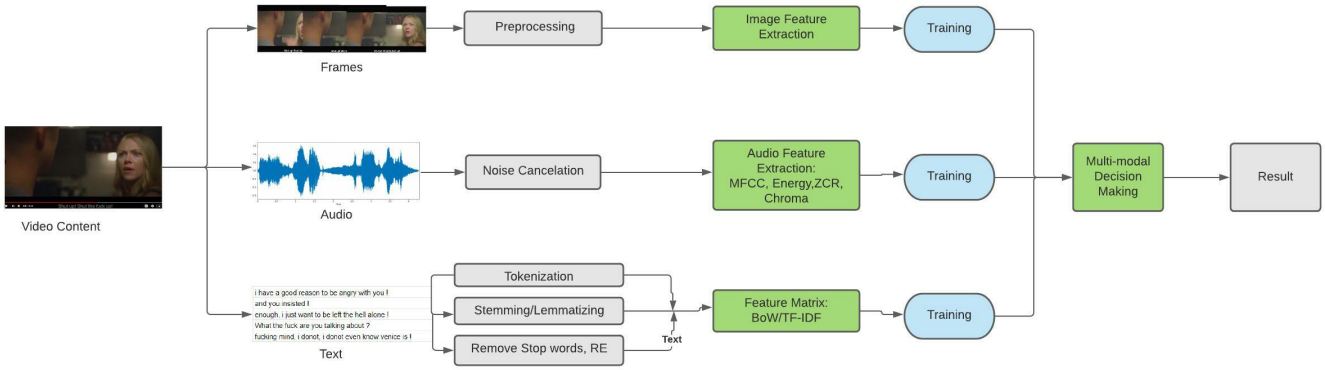

Fig. 1. Methodology for Detecting Hate Speech

deep neural networks [9]. LSTM along with Random embedding learned from deep neural network models combined with gradient boosted decision trees led to the best accuracy values. Rybach et al(2009) [10] worked with some well-known segmentation methods, focusing on the Log-linear model to determine segmentation from audio streams. GIS(Generalization Iterative Scaling) was used for optimization to detect speech using 3-state HMMs for the signal type speech, non-speech, pure music, and silence presenting a MAP decoder framework for audio segmentation. Among 15s segments, NIST, ASR-based, and MAP, MAP is the best for audio signals.

All the previous work discussed are on single mode or uses pre-trained model such as BERT. Our Approach shows importance of features from multiple mode and train on simple machine learning model to detect hatred.

III. Methodology

The system model of the proposed multi-modal hate speech detection is presented in Fig. 1. After Collecting the video data we are pre-processing the data by converting the data into image, audio, and text and use image resizing audio noise reduction. Then features from each of the data are extracted separately from each of the data. Both time domain and frequency domain features are extracted from audio data. Each of the extracted features are passed into one single algorithm and fit the values of the features. Each of the features, image, audio, and text will be passed separately to find out the accuracy of the model. The model responds to hate and non-hate for each of the data. After that, the prediction decision-making will be a majority voting ensemble to predict the final output which means two or more modes have to be hate to make the final output to be hate.

A. Data Description and Pre-processing

For our research, we have collected video data of two categories as Hate and Non-Hate. Basically, the hatred facial expressions are associated with the negative emotions of anger, fear, aggressiveness, dislikes, disgust found in hateful conversations or can be a sense of injury or violence at the extreme level [7]. On the other hand, non-hatred or positive emotions are the feeling of pleasant or desirable behaviors such as joy, amusement, satisfaction, amusement etc. [8]. we have collected data from different sources such as movies, web series that contain hate speeches along with the images of hatred facial expressions and hatred talks. Non-Hate category, we have considered the videos of positive emotions and extracted data in the same way as hatred data. In consequence, we have prepared a total of 1051 video data for our work. Among the 1051 video data, we have used 80% data as train data and 20% data as test data. The procedures of extraction of each data are as follows:

TABLE I
THE DISTRIBUTION OF HATE AND NON-HATE IN THE DATA

Label	percentage in data
Hate	60%
Non Hate	40%

- **Image:** Each video has been taken at a length of 3 to 5 seconds. OpenCV, a python library, has been used to extract the images, and we have taken converted image data of 30 frames per second. The images have been labelled as Hate and Non-Hate according to their contents.
- **Audio:** As Audio is unstructured data and, by default, audio signals are non-stationary, meaning their properties differ over time (usually rapidly) it is more efficient to break the record into short segments and calculate one (average) strength value per segment. This is also the central principle behind short-term processing. The video data are converted into audio using the python MoviePy library. and we have reduced the background noise and removed the unnecessary noises.
- **Text:** The Audio data were converted into Text using Google Speech Recognition. Each of the data was uploaded to the cloud and converted into text and labelled as Hate and Non-Hate accordingly. Then we have gone through the major text data pre-processing steps such as removing stop words, unnecessary special characters,

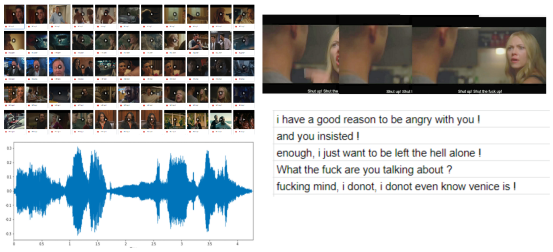

Fig. 2. Video,Image, Audio and Text data after pre-processing

URLs, double spaces and tokenization has been done for data mining.

B. Feature Extraction and Feature Selection

- **Image:** For the image data, as the images are categorized into two categories Hate and Non-Hate. We converted the data into pickle files where each of the data is converted into an array.Images of the videos are taken and resized 50 * 50. As the images are nothing but series of array. We are taking all of the images from a video and sorted and label accordingly to extract the image features.
- **Audio:** In the audio data, a sampling rate is determined for the number of samples taken from a continuous signal per second (or per unit) to make a discrete or digital signal. Frequencies are calculated in hertz (Hz) or cycles per second for time-domain signals such as the wave forms for sound (and other audio-visual content types). The Sample rate for the following task is 22050 as we found in Google Speech recognition and they suggested a frequency more than 16 KHz. For audio feature extraction two types of features are extracted from the audio signal. **Time domain features:** Time domain features are extracted from audio. we took hop length of 256 and frame length of 512 to divide the signal for sampling. From each of the frames we took the following feature and Calculated the mean of the output .

  Energy can be described as the signal's strength. It is possible to calculate the signal power by its energy. Energy is determined by the sum of signal squares, normalized by frame values the signal's duration.

$$E(i) = \sum_{n=1}^{W_L} |x_i(n)|^2 \tag{1}$$

  Zero-Crossing Rate is the rate of signal sign-changes during the frame. In other terms, it is the number of times that the Signals shift the value, divided by the frame length, from positive to negative and vice versa.

  Frequency Domain Features:

  Spectral Centroid is the spectrum's 'gravity' nucleus. It indicates where the" center of mass" for a sound is located and is calculated as the weighted mean of the frequencies present in the sound. **Spectral Rolloff** measures the frequency below which a specified (usually 90%) percentage of the total spectral energy. **Mel Frequency Cepstral Coefficient -** are a limited group of features that describe the overall shape of a spectral envelope. MFCC uses triangular overlapping windows to map the powers of the spectrum onto the mel scale. It takes the discrete cosine transform of the list of mel log powers as if it were a signal. The MFCCs are the resultant spectrum amplitudes. **Chroma Vector** is a 12-element representation of the spectral energy. The chroma vector is computed by grouping the DFT coefficients of a short-term window into 12 bins. Each bin represents one of the 12 equal tempered pitch classes of the input data. Each bin produces the mean of log-magnitudes of the respective DFT coefficients. Entropy of Energy , Spectral Spread, Spectral Entropy, Spectral Flux are also extracted.
- **Text:** First of all, tokenization steps are taken to tokenize the speech into single sentences.In this step we use sentence tokenizer. After tokenizing the whole speech or sentences we eliminate some regular expressions or special characters (like . , _ @ space etc). We just take uppercase and lowercase alphabet (a-z and A-Z). We make all the words into lowercase this time. After pre-processing the data we then used the following ways to convert the data to vectors-

  Counter Vectorizer: The model of bag-of-words (BOW) is a representation that transforms arbitrary text into vectors of fixed length by counting the number each word appears. Here, based on the label of the data and each word of a single data or sentence are given values.

  Term Frequency-Inverse Document Frequency (TF-IDF): TF-IDF is a statistical test that assesses in a series of documents how important a term is to a text. The formula that is used to compute the tf-idf is :

$$TF - IDF = TF * IDF \tag{2}$$

  where, TF = (Number of time the word occurs in the text) / (Total number of words in text) IDF = log(Total number of documents / Number of documents with word t in it)

After feature extraction, the most relevant characteristics were chosen using Recursive Feature Selection (RFE) and Maximum Relevance - Minimum Redundancy (mRMR). Recursive Feature Selection is a feature selection approach that wraps around a filter-based feature selection process. RFE works by searching for a subset of features in the training data, starting with all of the features and successfully deleting them until the target number remains. Maximum Relevance - Minimum Redundancy is a "minimum optimum" feature selection technique that aims to get the greatest possible prediction performance from a group of features given a certain number of features. Entropy of Energy , Spectral Spread, Spectral Entropy, Spectral Flux are eliminated by the feature selection.

IV. EVALUATION AND RESULTS

Following machine learning models have been used for evaluation of the performance: **Support Vector Machine** finds the decision boundary that optimizes the distance from the

nearest data points. **Random Forest** algorithm produces decision trees on data samples and then gets the prediction from each of them and selects the best solution by voting. It is an ensemble technique that is better than a single decision tree because by combining the result, it decreases the over-fitting. **Logistic Regression:** A statistical model used to evaluate if an independent variable has an effect on a binary dependent variable is logistic regression. This implies that given an input, there are only two possible results. **AdaBoost Classifier** is a meta-estimator that starts with one classifier and adds more to the same data set, but modifies the weights of wrong classifications to focus on more complex situations later. Adaboost [11] arbitrarily picks a training subset. A training set is chosen based on a forecast for the Adaboost machine learning model's last training. It lends additional weight to incorrectly classified observations so they are categorised in the next rotation. **K-nearest neighbors** is a Lazy algorithm implies that model generation does not require any training data points. The core decision factor is the number of neighbors. K is normally an odd number, because the number of classes is 2. **Naive Bayes Classifier** are a set of Bayes Theorem-based classification algorithms.It is a grading methodology based on the Bayes theorem, which implies that predictors are independent.The Gaussian model assumes that a normal distribution is followed by features. This means that the model assumes that the values are sampled from the Gaussian distribution if predictors take continuous values instead of discrete ones. **Decision Tree** starts with a single node that splits into possible outcomes. Any result leads to additional nodes that connect with other alternatives. It resembles a tree. An automated statistical model for machine learning, data processing, and statistics may be built using a decision tree. **Multi-modal approach** After all the individual classification done though the classical machine learning algorithm, a ensemble learning hard voting method or majority voting is used to get the final output combining the three individual detection. If the detection shows two or more than two individual outputs are positive to hate then the video is detected as hate otherwise it is detected as non hate speech.

From table II we can see that the multi modal approach performed well on AdaBoost and Naive Bayes where both achieved accordingly 87% and 75% accuracy than image, text and audio individually. SVM also performed 74% accuracy and 81.3% precision which is less than each of the mode separately.

TABLE II
RESULT ANALYSIS FOR IMAGE, AUDIO, TEXT AND MULTI-MODAL

Algorithm	Data	Precision	Recall	F1Score
SVM	Image	0.8959	0.8972	0.8965
	Audio	0.8047	0.81	0.84
	Text	0.775	0.813	0.793
	Multi-modal	0.74	0.813	0.875
Random Forest	Image	0.80	0.80	0.80
	Audio	0.874	0.875	0.874
	Text	0.841	0.70	0.764
	Multi-modal	0.65	0.81	0.718
Logistic Regression	Image	0.897	0.8976	0.8973
	Audio	0.874	0.80	0.887
	Text	0.740	.80	0.851
	Multi-modal	0.708	0.85	0.772
Adaboost	Image	0.841	0.82	0.787
	Audio	0.86	0.848	0.874
	Text	0.81	0.571	0.727
	Multi-modal	0.87	0.761	0.780
k-NN	Image	0.828	0.797	0.777
	Audio	0.787	0.748	0.760
	Text	0.818	0.857	0.837
	Multi-modal	0.8	0.761	0.780
Naive Bayes	Image	0.736	0.858	0.792
	Audio	0.804	0.827	0.85
	Text	0.707	0.7333	0.709
	Multi-modal	0.756	0.846	0.784
Decision Tree	Image	0.79	0.80	0.85
	Audio	0.726	0.704	0.715
	Text	0.775	0.713	0.793
	Multi-modal	0.74	0.813	0.775

V. CONCLUSION

The spread of hate must be stopped. This research detected hate and offensive content by analyzing audio, text, and video. The studied machine learning models achieved the highest 87% accuracy using the carefully extracted discriminative features.

REFERENCES

[1] Walter, E. (2012). Cambridge advanced Learner's dictionary. Cambridge Univ. Press.

[2] FBI National Press Office. (2020, November 16). FBI releases 2019 hate crime statistics. FBI. https://www.fbi.gov/news/pressrel/press-releases/fbi-releases-2019-hate-crime-statistics.

[3] W. Warner,J. Hirschberg "Detecting Hate Speech on the World Wide Web" Columbia University, Department of Computer Science, New York, NY 10027D

[4] S. Macavaney,Hao-Ren Yao-E.Yang,K. Russell, N. Goharian, O. Frieder - PLOS ONE "Hate speech detection: Challenges and solutions"- 2019 Master Business Analytics Department of Mathematics Faculty of Science August, 2018

[5] R. Kshirsagar,T. Cukuvac, K. Mckeown, S. Mcgregor "Predictive Embeddings for Hate Speech Detection on Twitter" - Proceedings of the 2nd Workshop on Abusive Language Online (ALW2) - 2018

[6] S. J. Arora, R. Pal "Singh Automatic Speech Recognition: A Review" International Journal of Computer Applications 60(9):34-44 - 2012

[7] Radhakrishnan, R., 2021. Why Do People Hate?. [online] Medicinenet.com. Available at:¡https://www.medicinenet.com/why-do-people-hate/article.htm¿ [Accessed 28 October 2021].

[8] PositivePsychology.com. 2021. What are Positive and Negative Emotions and Do We Need Both?. [online] Available at: ¡https://positivepsychology.com/positive-negative-emotions/¿ [Accessed 28 October 2021].

[9] P. Badjatiya, S. Gupta, M. Gupta and V. Varma "Deep Learning for Hate Speech Detection in Tweets" of the 26th International Conference on World Wide Web Companion -WWW '17 Companion Processing - 2017

[10] D. Rybach, C. Gollan, R. Schluterand H. Ney "Audio segmentation for speech recognition using segment features" 2009 IEEE International Conference on Acoustics, Speech and Signal Processing - 2009

[11] R. Nishii and S. Eguchi, "Supervised image classification based on AdaBoost with contextual weak classifiers," IEEE International IEEE International IEEE International Geoscience and Remote Sensing Symposium, 2004. IGARSS 04. Proceedings. 2007.

[12] M. Granik and V. Mesyura, "Fake news detection using naive Bayes classifier," 2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON), 2017.