Novel Approach to Detect Hate Speech and Profanity on Online Platforms

Samin Islam, Barha Meherun Pritha, Tabassum Alam
Md Rayhan Kabir, Nazmus Sakeef

Department of Computer Science and Engineering
BRAC University

***Abstract*—Hate speech is becoming more prominent and dominant in the virtual world, with the popularity of social media increasing day by day. People nowadays have various online platforms where they can express their hatred and write offensive speech in the safety of their home. They could even spread false rumors and incite hatred out of nothing. Cyberbullies often verbally attack the sentiments of people with different race, nationality, gender, beliefs and political views. They could also target young children and teenagers. It is also important to note that profane language or some sensitive topic may be bothersome when reached in front of young children and teenagers. It has become necessary for modern technology to detect all those profane and hate speeches so that they can be filtered or removed automatically before they can appear in front of young children or hurt the sentiments of targeted people. However, even though it is easy to detect profanities, it could be difficult to detect all the hate speeches which do not have any offensive or sensitive keywords. It is possible to spot all sorts of hate speeches on social media through the application of machine learning, neural networks and natural language processing. In our study, to identify and recognize hate speeches we will use various models and algorithms. Then we will design and implement an algorithm which will be able to detect hate speech and profane language more efficiently.**

***Keywords*—Detection, Hate Speech, Profanity, Vectorization, Machine Learning, Word-embedding, BiLSTM**

I. Introduction

HATE speech and abusive language has been ever present and persistent to humankind for the longest time. It was there from the earliest times of slavery; it is here today with the advancement of modern technology, and it will be also be present in the undefined future. Profanity has also been present for a long time, but it is becoming more prominent. Profanities are now used increasingly in movies, pop songs and rap music.

The concept of hate speech differs. Although there is no standardized concept of hate speech, many people claim that they can identify “hate speech” when they see it [1]. The term hate speech is determined as per the UN as any sort of conversation, conductivity or writing that targets slur or excessive profanity with respect to an individual or a group based on who they are, or in other words, on the basis of their nationality, origin, faith, ancestry, sex or other identification features [2]. All have the right to opinion and speech, as stated in Article 19 of the United Nations, 1984 and The United General Assembly, 1966, which also includes the right to freedom of opinion without interruption and to search, obtain and impart knowledge and skills through any platform without being dependent on boundaries. It demonstrates that even though hate speech is seen as an act of discrimination, it is not considered illegal and is protected by the fundamental human rights. Although it had been widely discussed in the legal field and with context to offensive speech on school campuses, the first amendment of the U.S. constitution also protects the right to free speech, religious belief and the news media. Fortunately, there are regulations banning hate speech against ethnic minorities in countries like the United Kingdom, Canada and France and those accused of using hate speech will also face substantial penalties and even incarceration [3].

The definition of profanity has changed through time. Before profanity referred to showing disrespect to religious values. Now, which is used in our research, profanity means offensive language, cursing or swearing.

II. Motivation

Usage of the internet and other technologies is increasing tremendously every day. Through that the practice of using social media is also accumulating. Nowadays, most of the people who have access to the internet use a number of online platforms including Twitter, Facebook, Instagram and YouTube. With the aid of such platforms, people can leave hateful or offensive speech through posts, comments, chats, etc. from the safety of their homes. It has become a challenge for networking sites to establish a platform where no hurtful, contempt or profane language can take place. Online media are establishing and updating their policies regarding what kind of content or comments can be shared, but even then, it seems that not all hate speech can be filtered out. Even though human beings can identify hate speech through words, tones and sentence structure, it is a bit difficult for artificial intelligence to evaluate if anybody encourages hatred or just explains what has happened to them [4]. It is important to note that the tone of sarcastic speech can be similar to the tone of hate speech. Both Facebook and Twitter have faced a lot of criticism for not doing sufficient to stop hate speech on their sites. The Chief Executive Officer and Co-founder of Facebook, Mark Zuckerberg once stated that hate speech has no space on Facebook [5].

The theme of offensive speech is becoming high in recent times yet further progress has been made to discover how to better recognise and discriminate between hate speech and other normal speech. We can trace profanity or receptive conversation via the use of clear and unambiguous keywords but not necessarily hate speech. It is pretty much guaranteed that racial and homophobic course will be categorised as hate speech, while sexist speech is typically categorised as profane [3]. Words like “fuck” can be implied profane but it may not be used to convey hate speech. Furthermore, certain words like “gay” can be considered sensitive, but then again, it can also be used in a positive manner. Besides, a hate speech may not contain any offensive keywords at all; in which case it will become difficult to correctly identify them.

There are a number of methods for sentiment analysis and due to the wide availability of diverse viewpoints on the data of social media, this is indeed a tough challenge, and these methods mostly rely on lexicons. Lexical methods in detection tasks have been a common feature for explicitly identifying any predefined word, and this feature extraction alone can only be good for text classification with low accuracy.

Hateful contents on the internet can be diverse like the dataset in [13]. That might seem like a challenge, which is somewhat true but not impossible to do at all. Moreover, there may arise a question regarding the dataset, e.g. for high efficiency, what amount of training data is required? According to [14] the authors set a different bar to experiment the accuracy of the model and came up with the conclusion that the more recent data is preferable than larger dataset. Performance degrades over time if the model is not updated with new features; and based on three experiments on a temporal dataset it has been proven delicately in [14].

III. Related Works

There have been numerous studies, experiments, and surveys conducted about hate speech and profane language over the past decade. Several models and algorithms have been implemented in order to detect abusive language including natural language processing, neural networks, machine learning, lexical analysis and sentiment analysis.

The research of [6] was one of the first to talk about the abusive language on the web. In their experiment, they used n-gram, sentiment analysis and contextual features to determine the offensiveness of previous sentences, to determine if the sentence is abusive or not.

In one of the researches [7], they explained why detecting hate speech is difficult to find. The reasons they stated are:

1) It is not possible to detect hate speech using keyword spotting.
2) All ethnic and cultural slurs may be difficult to define, since any meaning that is offensive to one community might be perfectly fine for some other communities.
3) Hate speech can have grammatical or spelling errors, and it can also have no grammatical or spelling errors.
4) Abusiveness can be beyond the sentence boundaries.
5) Sarcastic speeches can have the same tone as hate speech.

According to the paper [8], they determined that a speech is considered abusive if it i) a sexist or racial insult is used ii) assaults, critiques or threats to censor a minority iii) endorses criminal offence or hate speech iv) deliberately pursue to misrepresent the truth on a minority; v) supporting controversial hashtags, e.g. “#BanIslam”, “#whoriental”, “#whitegenocide”; vi) protects misogyny or xenophobia vii) stereotyping a minority unfavourably.

In the research [9], they used profane words to identify hate speech. Here are some of the offensive keywords and the category of the targeted discrimination that they found in their research:

1) Sexual Orientation: gay, lesbian, faggot
2) Physical dysfunction: douchebag, fucktard, dumbfuck, shithead
3) Gender: cunt, bitch, pussy, dick, cock, bull
4) Religious belief: jesus, islam, god king
5) Ethnic group: sandnigger, nigga, nigger
6) Class: bastard, sucker, fucker, motherfucker.

There are numerous methods for text classification, and according to any specific task data need to be flagged based on the right context. A survey paper [10] provides a succinct description of the automated identification of hate speech that thoroughly discusses the latest methods, concentrating on extraction of features in particular. Different features have been taken under consideration while working on this task whether the approach should be predictive or not, for instance Simple Surface Feature, Word Generalization, Sentiment Analysis, Linguistic Features, Meta information, Knowledge based features, Multi-modal information (mentioned in this paper). Although the set of traits studied in the various work differs widely, the approaches of classification relies largely on supervised learning. Character-level methods perform different leading approaches than approaches at the token level. Classification can be supported by lexical tools such as a list of slurs, but normally only in conjunction with other types of functionalities. Different dynamic traits using too much linguistic skills, sarcastic statements, non textual contents (multi-modal features can be considered) are also cue suggesting the presence of hate speech. To perform experiments, a decent amount of dataset is needed from social medias like- Twitter, Facebook, Instagram, Yahoo, YouTube, ask.fm. They seem to have unique features, because these platforms have indeed been built for particular reasons and may also exhibit multiple subtypes of hate speech. When evaluating the usefulness of such functions or techniques added to them, the scale of a dataset must be taken into account.

The classic methods of ML alone cannot track down precisely all kinds of offensive speech for ambiguity. Feeling the necessity to establish accurate and automated models to identify abusive language online; the authors of [14] came forward with new features of some powerful NPL, text classification and task specified embeddings. The methodologies they have followed includes dependency parser, semantic distributional features, linguistic features and N-gram traits. They handled tactfully the noisy data; 80% of the dataset was trained and 20% tested, combining with all features and individual features in different models. The model with all features outperforms the model with a single feature. Character n-grams have the greatest contribution in terms of human characteristics and thus their future studies include extraction of comment thread and use it as a background to evaluate each comment.

Similarly, in [7] another fresh strategy to deep learning architecture to detect hate speech has been created in tweets using word embedding. The dataset of 16K tweets have been labeled as sexist, racist and neither sexist or racist. The authors analyzed both classic and deep learning methods; different semantic tweet embeddings and three neural network architectures - FastText, CNNs and LSTMs. The results were categorized into three parts as per the combination of the methods. The results of Part A are baseline methods while Part B involves approaches that use neural networks only and Part C incorporates Deep Neural Networks learned average word embeddings as features for Gradient Boosted Decision Trees. On average Part C has the best performance among all three of them, then Part C and lastly the baseline approaches.

Researchers on [14] targeted internet protection of adolescents and came up with the idea of filtering out the contents by parents or teachers before appearing on a web browser. Using lexical and parser features they proposed an approach which is noxious expression recognition on YouTube comments. Utilizing Support Vector Machines which includes features like automatically generated blacklists, n-grams, manually created regular expression and dependency parsing features, the analysis creates a supervised classification approach and through this they gain 98.24% precision success on the role of inflammatory sentence identification and 94.34% recall.

The distinction seen between [7] work and this one is that before extracting the feature they try to spell correct and stabilize noisy text. But authors in [7] found noise a theoretically strong harassment detection signal and therefore has functionality to catch multiple formed up noise. Both of the works have dependency features, [7] consisting of a far larger collection of tuples than [14].

Another paper [8] identifies the difference between racist and sexist slurs using a character n-gram based approach. The model has been trained on some extra linguistic features; such as gender, religion, location and length which is the highlight of this research. The authors searched all probable feature set groupings. They found that the word n-grams performed better than n-grams by 5 F1 scores in minimum. The problem faced during this research is when location, gender and length are trained altogether, the performance diminishes and due to lack of coverage demographic information, apart from gender, brings little improvement. But overall the authors could successfully present a model to identify racist and sexist slurs.

The conceptual framework proposed in this paper characterize conditions between sentiment analysis and other NLP tasks, and express the dependencies in first order logic rules which aims at exploiting information outside the document to improve sentiment analysis. The authors have focused on two types of knowledge which includes intra document knowledge and extra document knowledge. The external knowledge defined in this paper exploits knowledge outside the sentence and outside the document. Not only that, the framework of this paper allows two types of evidence against the rules. The first case is when the event is involuntarily conducted and the second case is when an event is accidental. In spite of the fact that this paper is a conceptual framework, it bridges together various jobs of sentiment analysis and numerous jobs in natural language processing to deliver a complete tactic to sentiment analysis and others [15].

The paper [13] proposes an unused challenge on detecting hate speech in multimodal memes with hateful memes as a dataset. It did not train models from scrape. It adjusted and verified large scale multimodal models that were previously trained. The authors reconstructed basis memes from scrape by means of a customized tool. They had third-party annotators, who consumed about 27 minutes for each subsequent meme in the dataset. The memes are reconstructed using Getty images which allows several benefits like avoiding potential noise from optical character recognition (OCR) and reducing all errors that could be present in the graphic modality. The paper got some potential downsides too. Better multi-modal systems, for example, could lead to job automation in the future and be exploited for censorship or other undesirable ends. These dangers can be minimized in part by building AI systems to counteract them [13].

IV. Data Collection and Pre-processing

We have collected our data from Twitter as it is a very popular social media platform and people use it to express their daily thoughts or feelings. We have collected our data using the Python library tweepy in the year 2020. We at first, requested Twitter for authentication, and got consumer key and access token from them. Then in our Python code, we used the OAuthHandler function to pass the consumer key and access token and got authorization from Twitter. Later, we used the Stream submodule to filter the tweets using keywords and then store them in a CSV file. We have fetched recent most tweets from random people in the feed

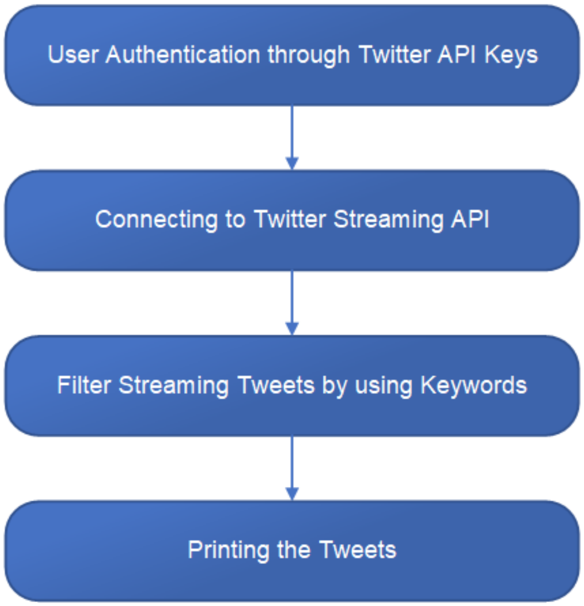

Fig. 1. The Process of our Data Collection

using the keywords. We had collected about 45k tweets. The process of data collection from Twitter is shown in Fig. 1.

The keywords we used to filter the tweets are: 'food', 'election', 'media', 'competition', 'vlog', 'travel', 'USA', 'US', 'economy', 'politics', 'programming', 'social', 'climate', 'game', 'tournament', 'movie', 'culture', 'torture', 'trump', 'biden', 'show', 'finance', 'stories', 'marketing', 'media', 'twitter', 'facebook', 'research', 'disaster', 'weather', 'life', 'motivation', 'fitness', 'science', 'goals', 'technology', 'festival', 'concert', 'song', 'review', 'hate', 'love', 'romance', 'beautiful', 'scenario', 'place', 'football', 'cricket', 'computer', 'religion', 'feminism', 'job', 'study', 'worst', 'shut', 'racism', 'kill', 'slang', 'gun', 'murder', 'suicide', 'racist', 'shit'.

Tweets are usually unstructured types of data. To work with the dataset efficiently aiming for a better accuracy, we needed to clean the dataset first as shown in Fig. 2. At first, we have removed all the non-English Tweets. Then we have removed tweets containing only urls, emojis, mentions, numbers, punctuation and other special characters. Then we have removed all the retweets. Finally, we have tagged them separately on the basis of - i) if they contain hate speech or not, and ii) if they contain any profanity or not. We have processed and tokenized our Tweets using Python code. After cleaning and removing all the duplicate tweets, we remained with about 1.5k dataset.

V. Data Exploration

A. Data Inspection

Fig. 3 is the list of our last 10 tweets in our dataset. We have tagged our data separately in the columns Hatespeech and Profanity (1 for presence and 0 for absence). We also

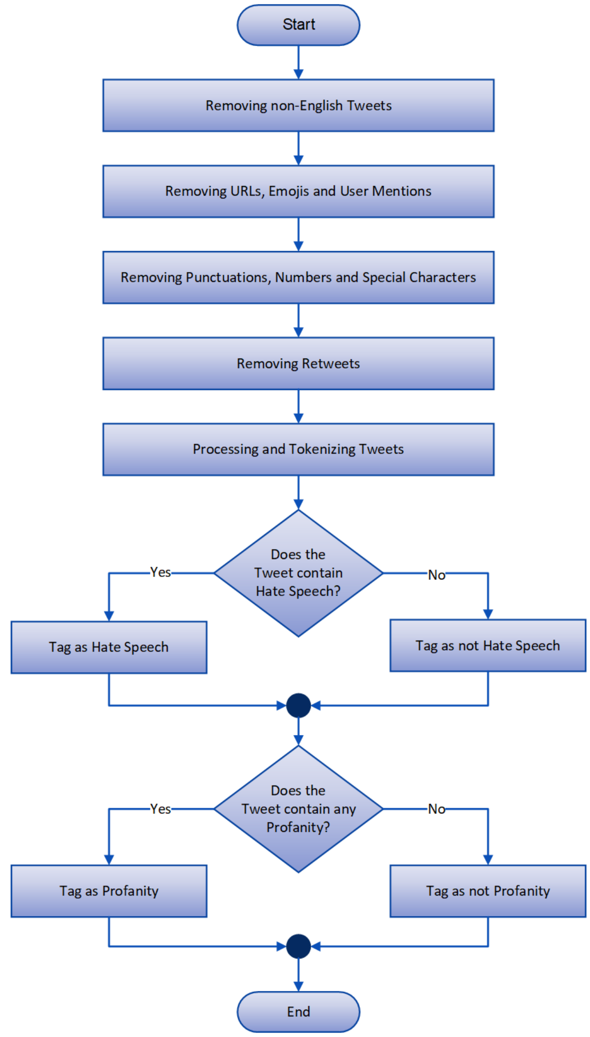

Fig. 2. Dataset Cleaning and Tagging

processed and tokenized our data with the help of Python programming.

Fig. 4 represents the top 15 targets which are referring to hate speech in our dataset. The sentence count is the number of sentences in our dataset containing the target words. The highest target is "trump" which is included in 23 sentences which represents a hate speech. The other target words are: "ass", "people", "man", "bitch", "thing", "@realdonaldtrump", "nigga", "dem", "joe", "stupid", "biden", "hell", "living" and "short".

B. Data Description

We have divided our dataset into 4 different categories: Both (Hatespeech + Profanity), Hatespeech, Profanity and None. The tweets which had no profanities and cannot be considered a hatespeech is categorized as "None". The tweets that contained profane language, but cannot be considered as a hatespeech, are categorized into "Profanity". The tweets that can be considered a hatespeech but did not contain any

	Tweets	Processed Tweets	Hatespeech	Profanity
1487	i really put 2 & 2 together and be right. we not the same ??	realli put amp togeth right	0	0
1488	Unprecedented aircraft movement in the last 10 days, 4 X the average, especially tonight, Sat night.??\n\nMonkey Werks talks about lots of aircraft moving a lot of troops just outside of Las Vegas...	unpreced aircraft movement last day x averag especi tonight sat night monkey werk talk lot aircraft move lot troop outsid la vega right recommend watch	0	0
1489	Proving again that drunk or sober Matt Gaetz is a flaming asshole.\n'You are not welcome in New Jersey': Governor slams Rep. Gaetz for attending maskless Republican gala https://t.co/9Fc8BYJ2uj vi...	prove drunk sober matt gaetz flame asshol welcom new jersey governor slam rep gaetz attend maskless republican gala via	1	1
1490	Mount Washington,NH (MWN) ASOS reports gust of 73 knots (84.0 mph) from NW @ 0756Z -- KMWN 060756Z 32058G73KT 1/16SM SN BLSN FZFG VV001 M15/M15 RMK VRY LGT ICG	mount washington nh mwn aso report gust knot mph nw z kmwn z g kt sm sn blsn fzfg vv rmk vri lgt icg	0	0
1491	literally, I say this all the time. pickle all my shit https://t.co/zhgRZ06n1H	liter say time pickl shit	0	1
1492	Finished my Cowboy Bebop re-watch. Last time was 10 years ago when I was living in Japan. Of course, still a masterpiece. And that final showdown... I want to play that. Such a beautiful use of mu...	finish cowboy bebop watch last time year ago live japan cours still masterpiec final showdown want play beauti use music cinematographi theme	0	0
1493	Just posted a photo @ Stand Up Live - Phoenix https://t.co/9xHw7AMhbT	post photo stand live phoenix	0	0
1494	@RealCapnCrunch I love the crunch crunch crunch sound ?? #CapnCrunchSweater #Sweepstakes	love crunch crunch crunch sound capncrunchsweat sweepstak	0	0
1495	@1965_superfly @CodeMonkeyZ Audited by who? I assume you mean a financial audit but that wouldn't hold true. Goods in transit are included. Also, why did y'all have truckloads of goods that wer...	audit assum mean financi audit hold true good transit includ also truckload good inventori ever happen run also none appli vote audit	0	0
1496	And outdoor #7footApart activities, like #7footApartHikes available to people, especially active Sr.s &\n50+ more at risk pop. who\nkeep health UP by being safe distance	outdoor footapart activ like footaparthik avail peopl especi activ sr amp risk pop keep health safe distanc apart outsid low risk activ like san diego day	0	0

Fig. 3. Last 10 Tweets of our Dataset

	Sentence Count	Hate Target Token
0	23	trump
1	16	ass
2	13	people
3	13	man
4	11	bitch
5	11	thing
6	9	@realdonaldtrump
7	7	nigga
8	6	dem
9	5	joe
10	5	stupid
11	5	biden
12	4	hell
13	4	living
14	4	short

Fig. 4. Top Targets of Hate Speech in our Dataset

profanity are categorized as ”Hatespeech”. Lastly, the tweets which had both hatespeech and profanity are categorized as ”Both”.

Both (Hatespeech + Profanity), Hate Speech and Profanity are further divided and labelled according to their types of discrimination: Bully, Racism, Belief, Politics, Sexual, Self Hatred and Criticism. None (tweets without any hatespeech or profanity) is divided into Thoughts, Grateful, Inspiring, Praising and Wishes. This is shown in the TABLE I.

Fig. 5 illustrates the different Word Clouds generated from Tweets of different categories. They were coded with the help of python libraries in kaggle.

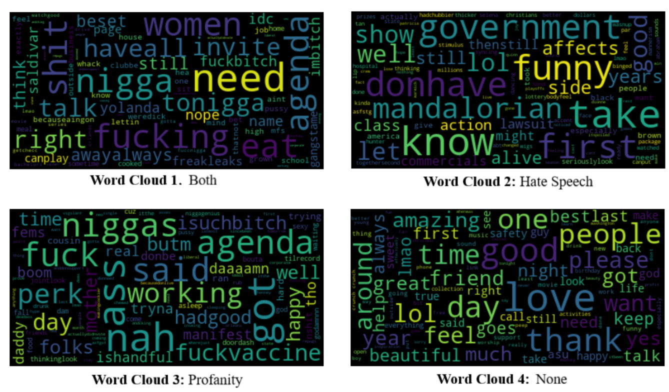

Fig. 5. Word Clouds of different Categories

In Fig. 6, the bar chart depicts the proportion of tweets that contain hate speech depending on the labels manually assorted to determine if a tweet depicts “self hatred”, “beliefs”, “sexual”, “racist”, “politics”, “bully” or “criticism”. Almost half of the total tweets i.e. 42.20% of them portrays criticism. Negativity towards politics and bullying also consists of 49.54% of the total tweets. A few proportion of around 8% of the tweets depicts self-hatred, hatred towards beliefs, sexual and racism.

In Fig. 7, the bar chart shows the proportion of tweets that contain profanity words based on the labels that were

TABLE I
MAIN CATEGORIZATIONS AND THEIR TYPES

Categories	**Labels**
Both (Hatespeech + Profanity)	Bully, Racism, Belief, Politics, Sexual, Self Hatred and Criticism
Hate Speech	Bully, Racism, Belief, Politics, Sexual, Self Hatred and Criticism
Profanity	Bully, Racism, Belief, Politics, Sexual, Self Hatred and Criticism
None	Thoughts, Grateful, Inspiring, Praising and Wishes

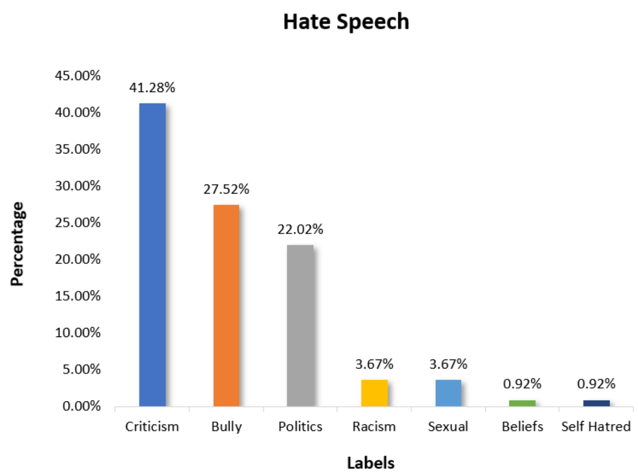

Fig. 6. Percentage of different Labels containing Hate Speech

manually sorted to decide whether a tweet depicted “self hatred”, “beliefs”, “sexual”, “racist”, “politics”, “bully” or “criticism”. More than half of the total tweets i.e. 57.58% of them portrays criticism towards people. Sexual and bully tweets covered 22.10% with just 1.9% difference between the two. Racist tweets are just 1% more than that of self-hatred consuming tweets. And minorities of the tweets attack the beliefs and politics by 2.02% and 1.01% respectively.

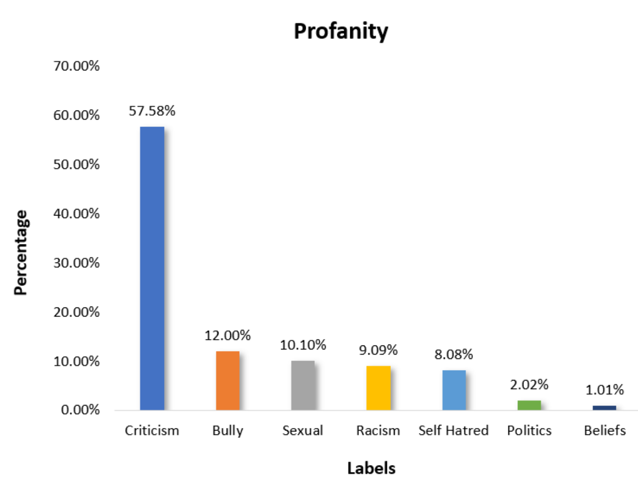

Fig. 7. Percentage of different Labels containing Profanity

In Fig. 8, the bar chart depicts the proportion of tweets that contain both hate speech and profanity based on the labels manually assorted to determine if a tweet depicts “self hatred”, “beliefs”, “sexual”, “racist”, “politics”, “bully” or “criticism”. Majority of the tweets containing slang and hatred are bullies i.e. 35.62% of the total. After that, most of the tweets are sexual covering 27.04% of the total tweets. One quarter of the tweets are criticism and attacking towards politics with 12.33% and 13.70% respectively. The percentage of racist tweets comes just after which is 8.22%. A small proportion of the tweets are portraying self-hatred and negativity towards beliefs in 1.37% each.

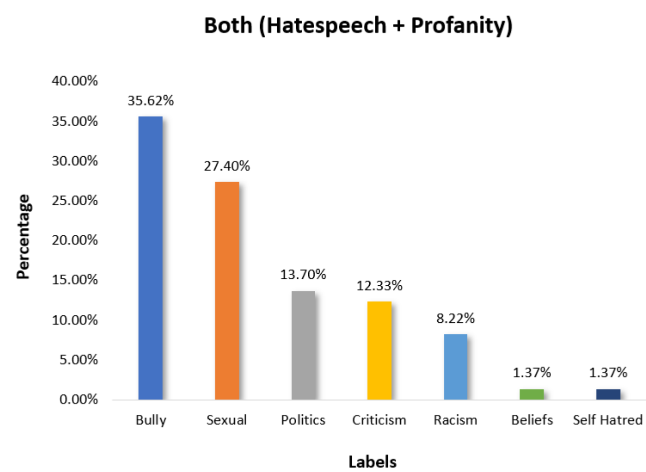

Fig. 8. Percentage of different Labels containing both Hate Speech and Profanity

In Fig. 9, the bar chart illustrates the proportion of tweets which do not contain any hate or profane speeches based on the labels manually assorted to determine if a tweet depicts “wishes”, “grateful”, “inspiring”, “praising”, and “thoughts”. More than 50% of the tweets i.e. 56.17% are the genuine thoughts of people. After that, gradually comes the percentage of praising, wishes, gratefulness and inspiration containing tweets. Almost 30% of the tweets contains praises and wishes and the rest of the 15% tweets consist of inspiration and gratefulness.

VI. FEATURE ENGINEERING AND VECTORIZATION

Vectorizing the input data is a preliminary step for feature engineering in Artificial Intelligence tasks. Presence of some words are frequently used in texts; like ‘nobody’, ’everybody’ , ‘never’, ‘always’, articles, and pronouns etc which are not relevant to literal meanings or original subject. To identify the target and meaning of those general statements before

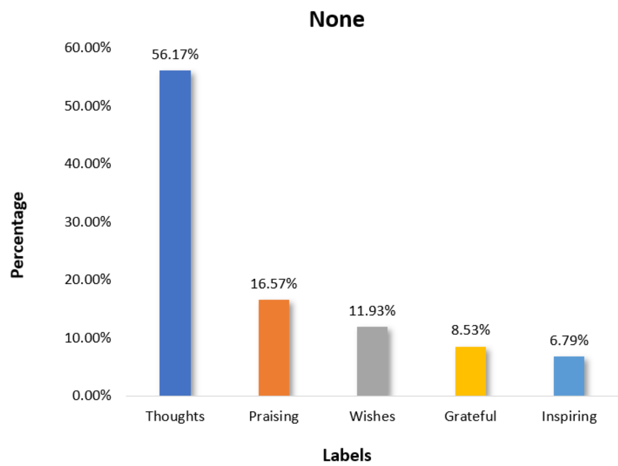

Fig. 9. Percentage of different Labels containing None

training any language model we need to convert the raw texts into numerical values e.g. vectors via different vectorization techniques. As every unique word is distinct in both syntactic and semantic concepts in the dataset, we need them to be unique vector representations before feeding into the models. We used some BOW(Bag of Words), TF-IDF, Sentiment Analysis Scoring and some state of the art word embeddings for this feature extraction task.

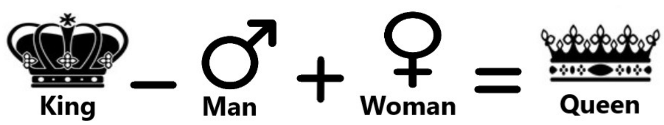

Fig. 10. Word Vector Representation of King - Man + Woman = Queen

A. Vectorizers

- **Bag of words (BoW) -** Bag of words is the simplest vectorization method to convert texts into fixed length vectors depending on word frequency in a given corpus. A text is represented as a bag of the word containing itself, overlooking the grammar or sentence structure. This feature generation method is highly considered for its simplicity, though it lacks a bit for determining contexts of a sentence.

- **TF-IDF -** The term is an abbreviation for Term Frequency (TF) Inverse Document Frequency (IDF); another simple vectorization method. This is based on the previous method where word frequency is considered and also focusing on the relevance of words. TF (term frequency) is the ratio of a word's occurrence to the total number of words in a document (shown in equation 1).

$$\mathsf{TF(w,d)} = \frac{\text{Occurrences of w in document d}}{\text{Total number of words in document d}} \quad (1)$$

And IDF (inverse document frequency) measures the significance of a word. The prepositions or pronouns have little importance in a sentence but are used most frequently for grammatical purposes. IDF provides a solution of this particular problem using the following equation, where a word is w in N documents (shown in equation 2). The algorithm reduces the weight of often used terms while increasing the weight of uncommon words found only in current documents.

$$\mathsf{IDF(w,D)} = \frac{\text{Total Number of documents N in corpus D}}{\text{Total number of documents containing w}} \quad (2)$$

- **Sentiment Analysis -** For our research purpose, gauging the sentiment of the users in twitter is the prime task and for this sentiment analysis score has been regarded to identify the emotional tone expressed in a tweet. A polarity analysis is needed for this score where words have been assigned to some values; +1, -1 and 0 as positive, negative and neutral respectively. Then the sum of these scores of a sentence is the sentiment score. Example:
  1) "I didn't (-1) study for the course and got poor(-1) marks" : **score = -2**
  2) "The day was so bright (+1) in the morning, I went for a walk and felt great (+1)" : **score= +2**
  3) "I didn't (-1) get an A in the course but passed (+1) anyway" : **score = 0**

B. Word Embeddings

Word embedding is the state-of-the-art approach to express text documents into vector representations. It is used to recognise a word's context in a document, their semantic and grammatical similarity, and their relationships with other words. Word embeddings, in a broad sense, are vector representations of a single word.

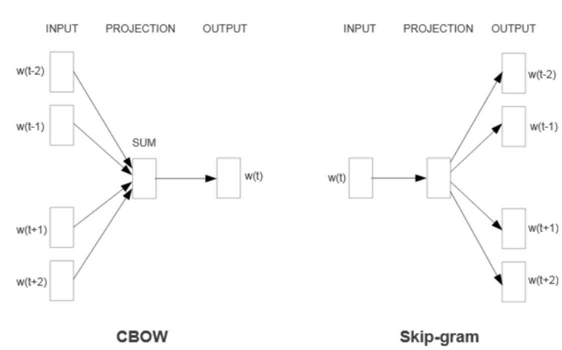

Fig. 11. Word2Vec Embedding

- **Word2Vec-** Word2Vec is one of the most widely used shallow neural network techniques for learning word embeddings. This conventionally uses two approaches to produce word embeddings (both involving Neural Networks): Skip Gram and Common Bag Of Words (CBOW). As the both underlying approaches of Word2Vec learn the intrinsic word representations for each word, providing a significant amount of data for training the model helps to guess words' meaning quite precisely.

- **Doc2Vec -** Doc2Vec is a generalized form of Word2Vec where there is no limitation of length of the document. This is another unsupervised word-embedding algorithm similar to word2Vec. The length of the vectors are always fixed for paragraphs, documents or texts.

- **Glove -** Glove is an unsupervised embedding algorithm that is derived from a large text corpus, enabling us to employ Transfer learning and train on our data in more depth. Moreover, this algorithm is based on matrix factorization techniques on context matrices of words which reduces the dimensionality of the data creating a lower stuffed dimensional matrix. These low dimensional matrices of words calculate faster and more accurately the meanings of each word. Though our dataset is quite small enough and does not need much dimensionality reduction.

VII. Models

After vectorization of the corpus, the pre-processed tweets are ready to train different models for the detection. These models take inputs of embedding vectors and compress them into a lower dimensional representation. This representation effectively captures the information in the sequence of words from the numerical forms. We used two types of models, Machine Learning based and Neural Network-based classifiers.

A. Machine Learning

After vectorization of the corpus, the pre-processed tweets are ready to train different machine learning models for the detection. These models take inputs of embedding vectors and compress them into a lower dimensional representation. This representation effectively captures the information in the sequence of words from the numerical forms. We used some state of the art models: Logistic Regression, Random Forest, Naive Bayes and SVM.

- **Naive Bayes -** Just like the name suggests, Naive Bayes classifiers are based on the principle of Naive Bayes' theorem. The theorem in the classifier states that the presence of one feature in a class is totally independent from the presence of any other feature. This approach is convenient for a large corpus and has always been outperformed by other sophisticated classification models.

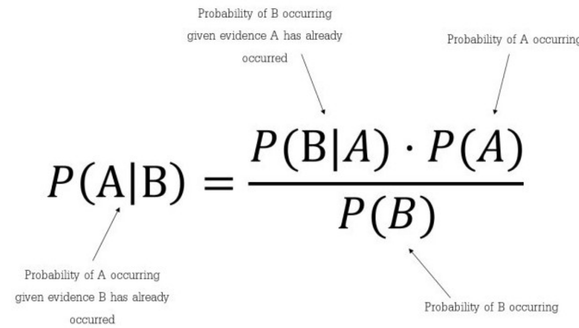

Fig. 12. Naive Bayes formula

- **Logistic Regression -** Logistic Regression is a machine learning algorithm that is a predictive model, mostly used for classification problems. The observations are assigned to a discrete set of classes based on the concept of probability. A logistic or linear regression model uses the sigmoid function as loss function and solves binary and multi-classification problems. The sigmoid functions map the vector inputs which are real values in between 0 and 1. Thus, this classifier provided us with the expected output based on possible probability scores.

- **Random Forest -** Random Forest classifier is easier and flexible for text classification problems with less adjustments of hyper-parameters. The supervised learning algorithm is an ensemble of decision trees that can measure the importance of different features. Later these trees decide the salient features that should be taken into consideration for a better result according to tasks.

- **SVM -** A SVM (Support Vector Machine) is a supervised learning method, popular for classification problems. This approach is best suited for a limited number of data. The principle idea of this method is that a hyper-plane has to fit at best between two categories of data from the input corpus. Support vectors are those points of the dataset that are nearest to that hyper-plane, and removal of either of them will change the positions of all.

B. Neural Network

We implemented an **attention based BiLSTM architecture**, along with Adam optimizer as a neural network model.

- **Attention-Layer -** Attention is a machine learning mechanism that tends to concentrate on the important portions of the dataset developing cognitive attention. This technique is a better approach than the encoder-decoder architecture in neural networks for encoding words into context vectors and later on decoding them

TABLE II
COMPARISON OF ML MODELS WITH DIFFERENT FEATURES TO DETECT HATESPEECH

		Logistic Regression	Random Forest	Naive Bayes	SVM
Bag of Words	Precision	0.85	0.75	0.79	0.96
	Recall	0.88	0.86	0.65	0.96
	F1 score	0.85	0.80	0.70	0.95
TF-IDF	Precision	0.75	0.88	0.79	0.83
	Recall	0.89	0.91	0.59	0.87
	F1 score	0.80	0.87	0.66	0.83
Sentiment Analysis	Precision	0.79	0.84	0.80	0.75
	Recall	0.86	0.87	0.78	0.86
	F1 score	0.80	0.84	0.79	0.80
Doc2Vec	Precision	0.75	0.78	0.82	0.75
	Recall	0.86	0.86	0.86	0.86
	F1 score	0.80	0.80	0.81	0.80
Combined Features	Precision	0.84	0.87	0.78	0.87
	Recall	0.87	0.89	0.66	0.88
	F1 score	0.84	0.86	0.70	0.87

for ML tasks. By assigning a score to each word, attention exerts varied emphasis on different terms. The context vector is then generated by aggregating the encoder hidden states using the softmax scores and a weighted sum of the encoder hidden states.

- **BiLSTM -** BiLSTM model is based on an RNN architecture, a bidirectional variat of LSTM. This model is fed with the dataset once from starting to end; and then again from end to start for which it learns better than one LSTM architecture. We used an attention-layer the BiLSTM architecture is implemented in our work, as this is supposed to perform better for the dataset we collected from twitter and it learns faster than a single LSTM for sequential learning problems. We have also included an adaptive optimization algorithm, Adam optimizer which intuitively adjusts the hyper-parameters for the task of this model.

- **Adam Optimizer -** Optimization algorithms help to improve the training of any model. We used Adam optimizer, which is an augmentation of SGD (Stochastic Gradient Descent) optimizer for deep learning approaches. As this optimizer intuitively tunes hyper parameters of the models and itself requires very minimal tuning, the computational efficiency is better and cost is moderate to use in our work.

VIII. EXPERIMENTAL SETUP

For training the dataset in Machine Learning based models (Logistic Regression, Random Forest, Naive Bayes and SVM), the dataset is first vectorized using different features (Bag of Words, TF-IDF, Sentiment Analysis and Doc2Vec). We have also vectorized our dataset using the combined features. This portion is the combination of the prior models (Bag of words + TF-IDF + Sentiment Analysis + Doc2Vec) to transform the words into numerical form. We, at first, trained our 80% dataset and then used 20% of our dataset for prediction.

For training the dataset in BiLSTM model with Word2Vec and Glove, the dataset is divided between training and test sets. Firstly, the function Sequence() is used which takes as input the length of the sequence. The first hidden layer of BiLSTM has 100 memory units, and its output layer is a fully linked layer that outputs one value each timestep. To predict the binary value, we utilize a sigmoid activation function on the output. A TimeDistributed wrapper layer is utilized around the output layer to forecast one value per timestep given the whole sequence as input. As a result, the BiLSTM hidden layer returns a sequence of values rather than a single value for the whole input sequence. Finally, because this is a binary classification problem, the binary log loss, also known in Keras as binary cross entropy, is employed. The weights are also determined using the efficient ADAM optimization algorithm, and the accuracy measure is calculated and published at each epoch. The BiLSTM is trained for a total of 10 epochs. Each epoch generates a fresh random input sequence for the network to be fit on which assures that the model does not recall any particular sequence but may instead generalize a solution to solve all conceivable random input sequences for this problem.

IX. RESULTS

A. ML Experimental Results of Hatespeech

TABLE II displays the experimental results of four machine learning-based models applied to our hate speech dataset: ”Logistic Regression,” ”Bag of Words,” ”Naive Bayes,” and ”SVM.” Using the four models, we ran “Bag of Words,” “TF-IDF,” “Sentiment Analysis,” “Doc2Vec,” and “Combined Features.” The dataset is divided into training and testing sets, with the test size set to 20%. The SVM model performs Bag of Words with the highest accuracy of 96%, the Random Forest model does TF-IDF and Sentiment Analysis with the highest accuracy of 91% and 87% , respectively, and all four models perform the same with Doc2Vec with an accuracy of 86% . Except for Doc2Vec, the Naive Bayes model performed every feature with the lowest accuracy. Finally, when all

TABLE III
COMPARISON OF ML MODELS WITH DIFFERENT FEATURES TO DETECT PROFANITY

		Logistic Regression	Random Forest	Naive Bayes	SVM
Bag of Words	Precision	0.93	0.96	0.81	0.96
	Recall	0.93	0.96	0.69	0.96
	F1 score	0.92	0.96	0.73	0.95
TF-IDF	Precision	0.92	0.96	0.80	0.92
	Recall	0.91	0.96	0.63	0.92
	F1 score	0.89	0.95	0.69	0.91
Sentiment Analysis	Precision	0.85	0.88	0.86	0.83
	Recall	0.88	0.89	0.82	0.87
	F1 score	0.85	0.88	0.83	0.83
Doc2Vec	Precision	0.77	0.83	0.77	0.77
	Recall	0.88	0.88	0.88	0.88
	F1 score	0.82	0.80	0.82	0.82
Combined Features	Precision	0.93	0.94	0.81	0.95
	Recall	0.93	0.94	0.70	0.95
	F1 score	0.92	0.94	0.74	0.95

of the features are integrated, Random Forest surpasses all other models with the best accuracy, however Naive Bayes performs poorly with an accuracy of less than 70% .

B. ML Experimental Results of Profanity

Similarly, TABLE III displays the results of the above-mentioned models when used to identify profanity. The dataset is partitioned into training and test sets in the same way. In this case, the Random Forest and SVM models outperform all other models in Bag of Words. Except for Naive Bayes, all of the other models have TF-IDF accuracy of greater than 90%. Like hate speech detection, all the models show similar results when run with Doc2Vec while detecting profanity. Based on the result, we may conclude that Random Forest outperforms all other models in detecting profanity. However, when all of the features are combined, the SVM model outperforms the other models with an accuracy of 95%.

C. NN Experimental Results of Hatespeech

TABLE IV
ACCURACY TABLE OF BILSTM TO DETECT HATESPEECH

		Attention based BiLSTM with Adam Optimizer
Word2Vec	Precision	0.85
	Recall	0.85
	F1 score	0.85
Glove	Precision	0.81
	Recall	0.47
	F1 score	0.55

TABLE IV illustrates the result when BiLSTM is run on Word2Vec and Glove model to detect hate speech. It is clearly evident that Word2Vec performs almost 2 times better than the Glove model.

D. NN Experimental Results of Profanity

Similarly, TABLE V shows the performance of BiLSTM when run on the two models to detect profanity. In this case, we noticed a considerable improvement in the performance of the Glove model. The accuracy of the Glove model in this case is 95%, which is the highest accuracy obtained from all of the other models and features.

TABLE V
ACCURACY TABLE OF BILSTM TO DETECT PROFANITY

		Attention based BiLSTM with Adam Optimizer
Word2Vec	Precision	0.90
	Recall	0.91
	F1 score	0.90
Glove	Precision	0.95
	Recall	0.95
	F1 score	0.95

X. CONCLUSION

As the influence of social media in daily life is deep-seated, moderating online contents should be taken into serious consideration. It has become an absolute necessity to establish accurate, efficient and automated methods to tag abusive language in these platforms. As automation of human language is quite complex, many researchers have been working in this field for years coming forward with different approaches from NLP, deep learning, neural networking, classic ML methods and features. In this paper, we tried to compare different models to detect hate speech and profane speech, and understand the conflict that may arise while differentiating them through artificial intelligence.

It is known to all, polarity analysis that is classifying negative, positive and neutral tweets is so much easier to do. But emotional analysis is difficult, as we can see some ambiguity in expression. People always do not intend to express the literal meaning of a particular word. However, we got some good accuracy scores training the models.

The models that we proposed, both ML-based and Neural Network architecture based, have performed very similarly. We have a notable limitation on the dataset being very small for training the models and predicting the outcome. Thus for future work the foremost concern will be to scrape a decent amount of data large enough to train the state-of-the art models. Secondly, we should work more on tuning the hyper-parameters of the models without any use of an optimizer if possible for a better result. Thirdly, while using neural network architectures we could try using dense layers or CNN-based models for a larger dataset. Moreover, this research idea can be carried on other social media platforms like Facebook, Youtube, Stack Overflow, Qoura etc, as these are some of the most extensively used and indispensable platforms for all of us.

References

[1] Martins, Ricardo, et al. ”Hate speech classification in social media using emotional analysis.” *2018 7th Brazilian Conference on Intelligent Systems (BRACIS)*. IEEE, 2018.

[2] Guterres, A. ”United Nations Strategy and Plan of Action on Hate Speech.” *Taken from: https://www.un.org/en/genocideprevention/hate-speech-strategy.shtml.* 2019.

[3] Davidson, Thomas, et al. ”Automated hate speech detection and the problem of offensive language.” *Proceedings of the International AAAI Conference on Web and Social Media*. Vol. 11. No. 1. 2017.

[4] M. Castillo. ”Facebook’s artificial intelligence still has trouble finding hate speech.” *Taken from: https://www.cnbc.com/2018/05/15/facebook-artificial-intelligence-still-finds-it-hard-to-identify-hate-speech.html.* 2018.

[5] ”Zuckerberg on refugee crisis: ’Hate speech has no place on Facebook’.” *Taken from: https://www.theguardian.com/technology/2016/feb/26/mark-zuckerberg-hate-speech-germany-facebook-refugee-crisis.* 2016.

[6] Yin, Dawei, et al. ”Detection of harassment on web 2.0.” *Proceedings of the Content Analysis in the WEB* 2 (2009): 1-7.

[7] Nobata, Chikashi, et al. ”Abusive language detection in online user content.” *Proceedings of the 25th international conference on world wide web.* 2016.

[8] Waseem, Zeerak, and Dirk Hovy. ”Hateful symbols or hateful people? predictive features for hate speech detection on twitter.” *Proceedings of the NAACL student research workshop.* 2016.

[9] Teh, Phoey Lee, Chi-Bin Cheng, and Weng Mun Chee. ”Identifying and categorising profane words in hate speech.” *Proceedings of the 2nd International Conference on Compute and Data Analysis.* 2018.

[10] Schmidt, Anna, and Michael Wiegand. ”A survey on hate speech detection using natural language processing.” *Proceedings of the Fifth International workshop on natural language processing for social media.* 2017.

[11] Ahluwalia, Resham, et al. ”Detecting hate speech against women in english tweets.” *EVALITA Evaluation of NLP and Speech Tools for Italian* 12 (2018): 194.

[12] Badjatiya, Pinkesh, et al. ”Deep learning for hate speech detection in tweets.” *Proceedings of the 26th International Conference on World Wide Web Companion.* 2017.

[13] Kiela, Douwe, et al. ”The Hateful Memes Challenge: Detecting Hate Speech in Multimodal Memes.” *arXiv preprint arXiv:2005.04790* (2020).

[14] Chen, Ying, et al. ”Detecting offensive language in social media to protect adolescent online safety.” *2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Conference on Social Computing.* IEEE, 2012.

[15] Deng, Lingjia, and Janyce Wiebe. ”How can NLP tasks mutually benefit sentiment analysis? A holistic approach to sentiment analysis.” *Proceedings of the 7th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis.* 2016.

[16] Zhu, Xu, Shiqun Yin, and Zongliang Chen. ”Attention based BiLSTM-MCNN for sentiment analysis.” *2020 IEEE 5th International Conference on Cloud Computing and Big Data Analytics (ICCCBDA)*. IEEE, 2020.

[17] Bhuta, S., Doshi, A., Doshi, U., Narvekar, M. (2014, February). ”A review of techniques for sentiment analysis of twitter data”. *2014 International conference on issues and challenges in intelligent computing techniques (ICICT) (pp. 583-591). IEEE.*

[18] Mehta, P., Pandya, S. (2020). ”A review on sentiment analysis methodologies, practices and applications”. *International Journal of Scientific and Technology Research, 9(2), 601-609.*

[19] Kharde, V., Sonawane, P. (2016). ”Sentiment analysis of twitter data: a survey of techniques”. *arXiv preprint arXiv:1601.06971.*

[20] Gautam, G., Yadav, D. (2014, August). ”Sentiment analysis of twitter data using machine learning approaches and semantic analysis”. *2014 Seventh international conference on contemporary computing (IC3) (pp. 437-442). IEEE.*

[21] Ahuja, S., Dubey, G. (2017, August). ”Clustering and sentiment analysis on Twitter data”. *2017 2nd International Conference on Telecommunication and Networks (TEL-NET) (pp. 1-5). IEEE.*

[22] Huq, M. R., Ahmad, A., Rahman, A. ”Sentiment analysis on Twitter data using KNN and SVM”. *International Journal of Advanced Computer Science and Applications, 8(6)*, 2017.

[23] Agarwal, A., Xie, B., Vovsha, I., Rambow, O., Passonneau, R. J. (2011, June). ”Sentiment analysis of twitter data”. *In Proceedings of the workshop on language in social media (LSM 2011) (pp. 30-38).*

[24] Wang, J., Yu, L. C., Lai, K. R., Zhang, X. (2016, August). ”Dimensional sentiment analysis using a regional CNN-LSTM model”. *In Proceedings of the 54th annual meeting of the association for computational linguistics (volume 2: Short papers) (pp. 225-230).*

[25] Murthy, G. S. N., Allu, S. R., Andhavarapu, B., Bagadi, M., Belusonti, M. (2020). ”Text based sentiment analysis using LSTM”. *Int. J. Eng. Res. Tech. Res, 9(05).*

[26] Rehman, A. U., Malik, A. K., Raza, B., Ali, W. (2019). ”A hybrid CNN-LSTM model for improving accuracy of movie reviews sentiment analysis”. *Multimedia Tools and Applications, 78, 26597-26613.*

[27] Minaee, S., Azimi, E., Abdolrashidi, A. (2019). ”Deep-sentiment: Sentiment analysis using ensemble of cnn and bi-lstm models”. *arXiv preprint arXiv:1904.04206.*

[28] Sosa, P. M. (2017). ”Twitter sentiment analysis using combined LSTM-CNN models”. *Eprint Arxiv, 2017, 1-9.*

[29] Chen, Y., Yuan, J., You, Q., Luo, J. (2018, October). ”Twitter sentiment analysis via bi-sense emoji embedding and attention-based LSTM”. *In Proceedings of the 26th ACM international conference on Multimedia (pp. 117-125).*