STM JOURNALS

Journal of Artificial Intelligence Research & Advances
ISSN: 2395-6720
Volume 6, Issue 3
www.stmjournals.com

Detecting Hateful Content on Social Media

Aaniya Gouse**, Afaq Alam Khan
Department of Information Technology, Central University of Kashmir, Jammu and Kashmir, India

Abstract

As humans have identified social media as an indispensable mode of interaction, their evil side seems to have spilled all over the social media platforms. With a humongous number of victims to hatred on the social media, we realized it must be rooted out. We proposed creating a classifier that learns from many a feature including lexical, semantic, syntactic, morphological and contextual features. This paper aims to revolve around employing the aforementioned features to learn the classifier and consequently, observe the performance of the classifier. As employing lexical these features in other classification tasks of opinion mining is observed to perform better, it's expected to work well with hate detection as well.

Keywords: *Social media, Natural Language Processing, Hate Speech, Semantic Features, Supervised Learning*

****Author for Correspondence*** E-mail: aaniya.gouse@gmail.com

INTRODUCTION

Since time immemorial, hate has been engrained in the mankind as a part and parcel and is vented out in the form of threat, abuse, violence, harassment etc. The longstanding issue of hatred seems to have crept onto the social media as we have already jumped into the midst of the Fourth Industrial Revolution, where the demarcation between the physical and the cyber world is thin [1]. Every single individual on the social media has been a victim to hatred or has at least witnessed it. While anonymity/fake identity of users on the web is one reason that rather makes it easier for the miscreants to target the vulnerable, misogyny, trolling, defaming and narcissism are the outcome. Spreading hatred online has resulted in blocking of other users or staying off the social media altogether [2]. Facebook removed 1.9 million comments which were found to promote terrorist propaganda during January–March 2018, the number of comments being exceeded by 800,000 comments as compared to previous three months [3]. The term "hate speech" continues to remain ambiguous as it lacks clearly defined boundaries. Nevertheless, hate speech is an expression that victimizes one or more persons on grounds of race, religion, ethnicity, etc. [4]. It is distinguished from normal/peaceful speech by an extremely thin line. For instance, under certain criteria ill-treating a person/entity verbally might be considered to be "free speech" but the moment it turns into violence of any form, it takes the shape of "hate speech". Many popular social media sites bank upon self-regulatory methods which include reporting a post/profile which a user may find offensive or a danger to peace. All the same, incidents involving hatred are on a hike more than ever before and as such, social media sites remain to be the greatest platform that contributes to hate crime. Therefore, a hate detector that destroys the cause of hatred while still budding must be in place. A hate classifier that keeps a check on appearance of hatred on social media and prevents its occurrence ahead of appearance is almost a necessity of the current times. The goal of this research is to come up with a hate detection system that is capable of detecting prospective hatred all over the social media. Therefore, an efficient hate detector can be immensely fruitful for the prevention of hate crime arising from various social media platforms. This paper revolves around methods for genocide prevention, cyber bullying prevention, troll detection, suicide prevention and detection of terrorist propaganda. Be it any façade that the hateful content may put on-whether sarcasm, misspellings or pure offense, hate must be distinguished and removed. The present

systems make use of statistical features more than the semantic features. Statistical features do not lead to much accurate results and the systems are usually prone to error and less scalable. As employing lexical, morphological, contextual and semantic features in other classification tasks of opinion mining is observed to perform better, we decided to make use of the same. Measuring the efficiency of the proposed hate detection system against three pre-labeled datasets, we calculated the results. As deducted from the state-of-the-art research, we realized a hate speech classifier which takes semantics of the text into account was not in place. Furthermore, the efficiency exhibited by the proposed technique of hate detection affirmed good results.

Contribution of Author

Our research has the following to offer:

1. An efficient method for pre-processing of text which includes correcting misspellings and eradicating slangs.
2. A classifier that is capable of classifying hatred out of a given corpus.
3. Linguistic analysis of text to reveal syntactic and semantic details of language.
4. Validation and evaluation tests that attest efficient performance of our system.

RELATED WORK

Kumar et al. [2] performed annotation of a corpus, grouping aggression as overt, i.e., apparent aggression and covert, i.e., nonapparent aggression. They further divided the categories on level two into physical aggression, sexual aggression, identity aggression and nonthreatening aggression. A challenge that they came across during their experiment was that abuse can put on a façade to appear like hatred which may eventually lead to misclassification. Gitari et al. [4] look at the shortage of labeled corpus as a challenge. As for this challenge, they proposed a rule-based approach to develop a hate speech classifier. Using Wilson et al. [5, 6] and WordNet [7] as the lexicon resources, they expanded the lexicon with hate related verbs and noun patterns, religion, race and ethnicity being the target groups. Taking the seed lexicon from the affect database, Neviarouskya et al. [8] augmented the lexicon using new terms from synonymy and antonymy relation on WordNet and existing base words to construct new lexemes. As observed, one of the greatest challenges to hate speech detection is the segregation of hate speech and offensive speech. Davidson et al. [9] remarks that while classifying the corpus into "hatred" and "offense", intermingling of classes in the results is vividly observed. Taking the context of speech into account is found to be of utmost importance. Hate Speech detection needs a huge manual effort. Beginning with manual look-up of usual hate words, Waseem et al. [10] used Twitter to collect the corpus which is then humanly annotated. The main contribution in hate speech is found to be that of men; therefore, gender is not used a feature. Also, use of location is found to be not fruitful to the scores. Moreover, they remarked character n-grams outmaneuver word n-grams. Waseem [11] states that knowledge of annotator is of utmost importance as it is reflected in the results of training. He remarks that expert annotators perform better than amateur annotators. As the dataset employed undergoes personal bias, there is a due amount of error owing to the false positives. Using semantic similarity between couple of words, Jan et al. [12] proposed an automated approach of sentiment analysis. They used a shortened form of Emolex as a resource, i.e., only six out of eight classes. Into these six classes, they classified the corpus and with the use of deep learning and unsupervised learning semantic features were detected. Vectorization of lexicon was then performed so as to classify data based on the score. Using Oxford Thesaurus and WordNet the lexicon was augmented and Word2Vec was used to generate vectors for each word thereby reducing each word to its semantic details.

METHODOLOGY

An innovative approach towards detection of hate speech is presented. Innovative in a way that the classifier is aimed to be a general-purpose hate speech classifier and is built using morphological, contextual, semantic as well as lexical features. Based upon the content of the post, the generalized multiclass hate speech classifier classifies the post into any number of classes that are provided to it.

Table 1: *Sample Posts.*

Serial number	Facebook post	Class
facebook_corpus_msr_431947	People need intellectual brain to understand what Mr. Modi is doing... people abusing him are Bewkoof.	OAG
facebook_corpus_msr_2018535	That's what happens when you skip your education to marry a rich guy 10 years elder to you. Being a homemaker is your obvious choice but what about independent working mothers and single mothers? Make your own choices but do not run down and belittle women who do not have your luxuries. And I'm sorry but what makes you think that it's okay to spend just an hour with a puppy? Like how can one even compare? She's ridiculous. Aishwarya Joshi Sreoshi Bagchi.	OAG
facebook_corpus_msr_461552	Go far Maruti only... steady long-term bet Sintex and Aarti volatile. Trading prospects Heritage food...no	NAG
facebook_corpus_msr_480836	Sonia, EVERY MOMENT MIND it, EVERY MAN's mind-operate TWO MOST WANTED CRIMINAL STAYING IN ASSAM-INDIA. SO, BE CAREFULL ANY MOMENT. GLOBALISATION'S SCIENCE SATELITTE PHONE THEM'S WINGS. OK? DO NOT WORRY, ANY MOMENT I AM STAYING WITH EVERY MEDIA HOUSE. OK?	NAG
facebook_corpus_msr_1561824	Yogi Adityanath should have come in her support. His support would have salutary effect on the loose cannons in his party.	CAG
facebook_corpus_msr_1724019	Wondering why educated ambassador is struggling to pay through Credit/Debit at a decent restaurant! Cannot imagine that diplomat of a developed nation is not having a card and he needs cash only for dinner.	CAG

Datasets

We performed the classification task with three datasets. First dataset [2] consists of 11999 Facebook posts in English; 2708 instances of "Overtly Aggressive" class; 4240 instances of "Covertly Aggressive" class; 5051 instances labeled as neither of the classes. Another dataset consisting of 16907 instances, out of which 1970, 3378 and 11559 labeled as "Racism", "Sexism" and "None", respectively is used [11]. Third dataset comprising of 20001 tweets is also used [13]. The tweets are humanly annotated; those which are "Cyber-aggressive" are labeled as 1 and those which are "Noncyber-aggressive" are labeled as 0. Tables 1, 2 and 3 present a sample of each dataset, respectively.

Automatic Hate Speech Classification

Based on supervised learning, we learned a hate speech classifier. The proposed classification algorithm is carried out with the aid of multiclass support vector machine (SVM). It is composed of the following parts:

a. Pre-processing
b. Lexicon generation
c. Feature selection
d. Available hate datasets [2, 11, 13]
e. Automatic hate classifier

Each of these is discussed in detail as follows. The following Figure 1 depicts the functioning of our system.

Table 2: *Sample Tweets.*

Tweet ID	Annotation
5.72343E+17	Racism
5.72341E+17	Racism
5.63593E+17	Sexism
5.63604E+17	Sexism
5.69619E+17	None

Table 3: *Sample Tweets.*

Content	Label
Get fucking real dude.	1
She is as dirty as they come and that crook Rengel the Dems are so fucking corrupt it's a joke. Make Republicans look like...	1
Why did you fuck it up? I could do it all day too. Let's do it when you have an hour. Ping me later to sched writing a book here.	1
Thanks for your kind words about sincerity sucks. That's always been one of my favorites.	0

Pre-processing Module

Crucially, pre-processing involves cleaning of data, analysis of features, annotation of data, and normalization of data. It involves removing upper case letters and stop words, tokenization, part of speech tagging, stemming, lemmatization, etc.

a. In order that the tweets/posts are tokenized, Stanford Core NLP is used.
b. Stemming is performed using Porter Stemmer.
c. Stop words are removed.
d. Usernames and URLs are removed.
e. Uppercase letters are removed.

f. Using dictionary module WordNet and replacer module NetLingo, slangs and abbreviations are removed from the post/tweet. The former retrieves proper meanings of words passing onto the latter which replaces words that have no proper meanings.

Figure 1 illustrates pre-processing of tweets and/or posts in detail.

Lexicon Generation

To annotate each sentence of the corpus a seed lexicon from www.hatebase.org was used. In order to keep check of hate speech, hatebase makes use of a wide vocabulary in varied languages pertaining to disability, sexual discrimination, ethnicity, caste etc. throughout nations which amount exceed 200 in number [14]. The initial seed set consisting of 1034 words as obtained from hatebase was then extended with the help of distributional semantic models resulting in 1559 words.

Feature Selection

So that the prediction performed turns out to be optimal features are extracted using Java. The features that we selected for our task are:

a. *Unigrams:* Unit tokens out of a corpus in the form of nouns, verbs, adjectives etc. regardless of order.
b. *Bigrams:* Two random unigrams taken together following a particular order. The word following the seed word is the most probable word that would follow that word in the corpus.
c. *Trigrams:* The kind of n-gram, where, n is taken as 3. The word preceding and succeeding the seed word is the most probable word that would follow or be followed by that word in the corpus.
d. *POS:* Takes into account verbs, adjectives, adverbs and nouns contained within the corpus as these are of utmost semantic significance.
e. *POS-bigrams:* Adjectives and proper nouns were drawn out of the corpus and thus, POS-bigrams were generated.
f. *Adjectives:* Using NLTK, adjectives were drawn out of the corpus.

Automatic Hate Classifier

We employed multiclass SVM kernel to implement our hate classifier. SVM makes use of the seed words as acquired during data normalization after these are preprocessed, i.e., brought into number form. The SVM multiclass formulation as conducted by Coursera [15] is used but along with an algorithm that improves the case of linear SVM as its faster. For instance, taking training examples (x_1,y_1) ... (x_n, y_n) are labeled as y_i in [1..k], the following problem of optimization while the process of training is carried out, is solved.

$\min 1/2\ \Sigma_{i=1..k} w_i * w_i + C/n \Sigma_{i=1..n} \xi i$

$\forall$ y in [1..k]: $[x_1 \bullet wy_i] >= [x1 \bullet wy] + 100*\Delta(y1,y)-\xi 1$

$\forall$ y in [1..k]: $[x_n \bullet wy_n] >= [x_n \bullet wy] + 100*\Delta(yn,y)-\xi n$

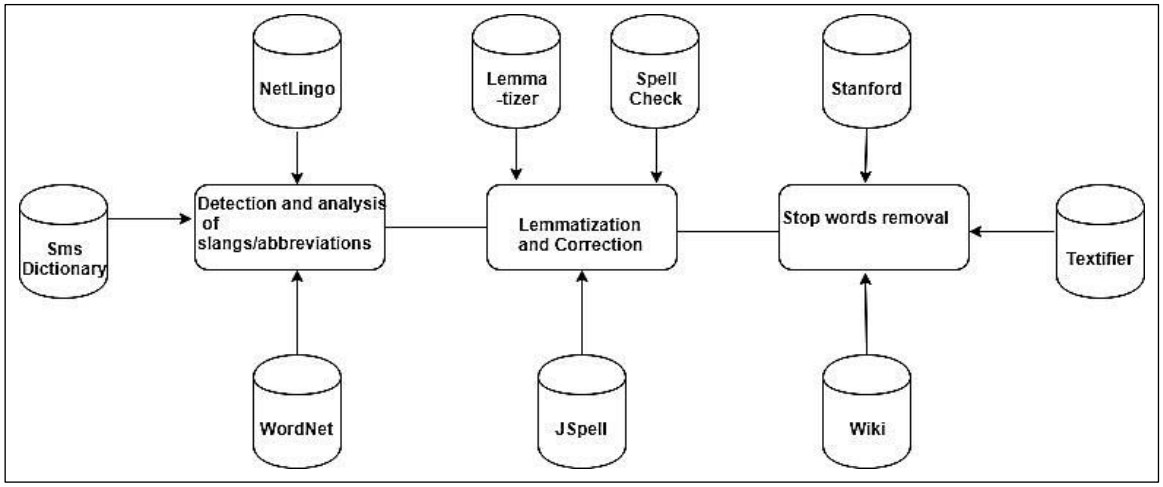

Fig. 1: *Pre-processing of Tweets/Posts.*

C is the regularization parameter that trades off margin size and training error. $\Delta(y_n, y)$ is the loss function that returns 0 if yn equals y, and 1 otherwise [15].

The hate classifier is given the tweets/posts as an input along with the labels. It outputs the tweets/posts each labeled against the provided label. Based on the features the hate classifier learns a model with the aid of SVM which then predicts labels pertaining hatred against ambiguous data.

Algorithm

The propounded algorithm of hate classification as suggested by us is presented in the following steps:

a. Choosing a set of seed words vis-à-vis hate speech namely, hatebase.org.
b. Using the datasets [2, 11, 13] training the Word2Vec model.
c. Augmentation of the seed lexicon in (a). This is done using a trained Word2Vec model which results in an expanded hate lexicon.
d. Carry out pre-processing of training dataset.
e. Extraction and selection of features for hate detection from the preprocessed dataset.
f. Vectorization of features into numbers so as to be trained using SVM.
g. Sorting of the feature set.
h. Assigning weights to each feature using TF-IDF such that each feature is represented as "Feature(Number):Weight".
i. Pre-processing, feature generation, vectorization is then performed upon the test dataset.

Evaluation, Validation, and Results

The hate classifier is given any of the chosen three datasets as an input. Given any of the chosen classes the classifier produces the tweet/post along with the label associated with it. We tested our classifier upon three datasets with the F1-Scores 73.9, 76.6 and 75.3 in each case, respectively. The measures that we used to evaluate our hate classifier include confusion matrices, accuracy, F1-score, recall and precision. In case of multiclass classification, confusion matrices prove to be a good measure of efficiency in particular. A general appearance of a confusion matrix is depicted by Table 4. While misclassified data are represented by false negatives and false positives, the correctly classified data in the table stands either as true positive or true negative (Table 4).

Table 4: *Confusion Matrix.*

	Class A prediction	**Class B prediction**
Class A real	True positive	False negative
Class B real	False positive	True negative

Precision is the measure of exactness of the model. From among the total number of records that exist in the dataset, it is calculated as the relevant records that we find. Mathematically,

$$Precision = \frac{tp}{tp + fp}$$

Recall is the measure of completeness of the model. From among the total number of relevant records that exist in the dataset, it is calculated as the relevant records that we find. There exists an inverse relation between precision and recall (Table 5). Mathematically,

$$Recall = \frac{tp}{tp + fn}$$

F-score is given by the harmonic mean of recall and precision.

$$F1 - Score = \frac{2(Precision * Recall)}{Precision + Recall}$$

The following Table 5 shows our obtained results.

Table 5: *Summed Up Results.*

	Precision	**Recall**	**F-Score**
Dataset [2]	76.5	71.5	73.9
Dataset [11]	80.2	73.7	76.6
Dataset [13]	72.1	78.8	75.3

Considering its benefits and efficiency the validation technique that was used is leave-one-out cross-validation (LOOCV) where training is performed on the entire dataset, but one small part is set aside over which it iterates (Figure 2). It is less biased but leads to higher variance and therefore, over fitting. Figure 2 provides an insight into LOOCV technique.

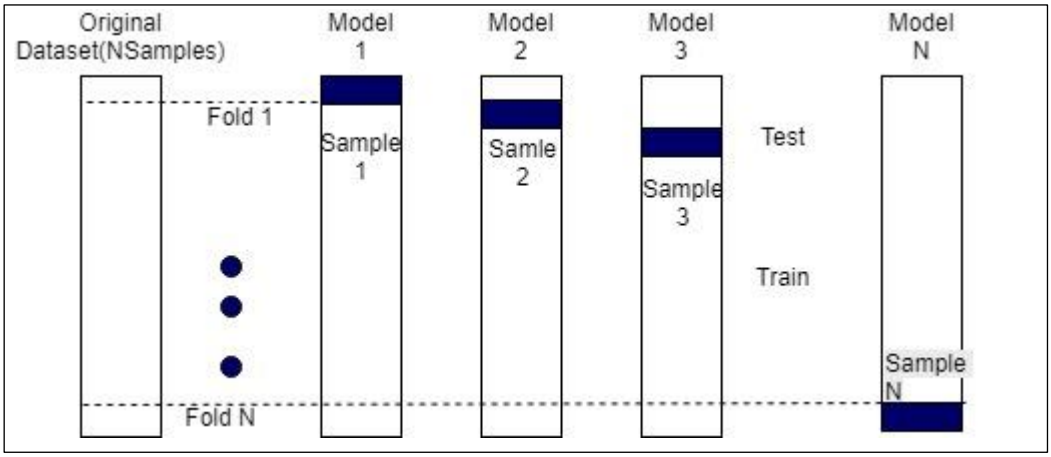

Fig. 2: *Leave-One-Out Cross-Validation.*

Comparative Analysis

We present the outcome that our algorithm produces and contrast it against the like techniques that have been used for hate detection (Table 6). We achieved an F1-Score of 76.6.

Table 6: *Comparison of Our Hate Classifier with Similar Techniques.*

Technique	F1-score
Proposed algorithm	76.6
Waseem Z, Hovy D. Hateful symbols or hateful people? predictive features for hate speech detection on Twitter	73.89
Waseem Z. Are you a racist or am i seeing things? annotator influence on hate speech detection on Twitter	53.43

CONCLUSION

Our research can be summed up in three main phases.

1. Expansion of the seed set as obtained from hatebase.org.
2. Capturing semantic similarities from amongst various linguistic items.
3. Classifying text into classes as provided to the classifier.

The evaluation carried out on our approach endorses its accuracy. Therefore, this experiment provides a demonstration on a general-purpose hate classifier that automatically classifies text into classes pertaining to hatred. Also, the results as obtained towards the end of the experiment prove that considering semantics of the text is beneficial to the results and thus employing distributional semantic models to capture semantics of the text is fruitful. Moreover, we have come up with an extension to the seed lexicon and intend to make it publicly available in near future.

REFERENCES

1. Schwab K. *The Fourth Industrial Revolution*. Currency. 2017 Jan 3.
2. Kumar R, Reganti AN, Bhatia A, Maheshwari T. Aggression-annotated Corpus of Hindi-English Code-mixed Data. *Computer Science*. arXiv preprint arXiv:1803.09402. 2018 Mar 26.
3. Rodriguez S. (2018). *Facebook says it's gotten a lot better at removing material about ISIS, al-Qaeda and similar groups.* [online] CNBC. Available from https://www.cnbc.com/2018/11/08/facebook-details-progress-vs-isis-al-qaeda-affiliates.html [Accessed 13 Mar. 2019].
4. Gitari ND, Zuping Z, Damien H, Long J. A lexicon-based approach for hate speech detection. *Int J Multimedia Ubiquitous Eng.* 2015 Apr; 10(4): 215–30p.
5. Riloff E, Wiebe J. Learning extraction patterns for subjective expressions. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, 2003.
6. Wiebe J, Riloff E. Creating subjective and objective sentence classifiers from unannotated texts. In *International Conference on Intelligent Text Processing and Computational Linguistics*, 2005 Feb 13, 486–497p. Springer, Berlin, Heidelberg.

7. Esuli A, Sebastiani F. Sentiwordnet: A publicly available lexical resource for opinion mining. *In LREC*, 2006 May 22; 6; 417–422p.
8. Neviarouskaya A, Prendinger H, Ishizuka M. SentiFul: A lexicon for sentiment analysis. *IEEE T Affective Comput*. 2011 Jan; 2(1): 22–36p.
9. Davidson T, Warmsley D, Macy M, Weber I. Automated hate speech detection and the problem of offensive language. AAAI Conference on Web and Social Media, arXiv preprint arXiv:1703.04009. 2017 Mar 11.
10. Waseem Z, Hovy D. Hateful symbols or hateful people? predictive features for hate speech detection on twitter. *In Proceedings of the NAACL Student Research Workshop*, 2016, 88–93p.
11. Waseem Z. Are you a racist or am i seeing things? annotator influence on hate speech detection on twitter. In *Proceedings of the First Workshop on NLP and Computational Social Science*, 2016, 138–142p.
12. Jan R, Khan AA. Emotion Mining Using Semantic Similarity. *Int J Synthetic Emotions (IJSE)*. 2018 Jul 1; 9(2): 1–22p.
13. Gupta M. (n.d.). [Blog] *Dataturks*. Available from https://dataturks.com/ [Accessed 8 Mar. 2019].
14. Hatebaseorg. (2019). Hatebaseorg. Retrieved 18 September, 2019, from https://hatebase.org/how_it_works
15. Pak A, Paroubek P. Twitter as a corpus for sentiment analysis and opinion mining. In *LREc*, 2010 May 19; 10: 1320–1326p.
16. Coursera. Applied Machine Learning in Python. [online] Available from https://www.coursera.org/lecture/python-machine-learning/cross-validation-Vm0Ie [Accessed March 2019].

Cite this Article
Aaniya Gouse, Afaq Alam Khan, Detecting Hateful Content on Social Media. *Journal of Artificial Intelligence Research & Advances.* 2019; 6(3): 25–31p.