An MLP based Approach of Hate Speech Detection on Twitter

Mohd Amjad[1], Mohd Zeeshan Ansari[1] and Nasim Alam[1]
[1]*Department of Computer Engineering*
Jamia Millia Islamia
New Delhi, India
(mamjad@jmi.ac.in, mzansari@jmi.ac.in, nasim163000@st.jmi.ac.in)

***Abstract*—** A considerable amount of growth is witnessed in recent years pertaining to hate speech, offensive language, sexism, racism, cyberbullying and other types of exploitation on popular social media platforms. Such abusive and offensive activities have gained exponential increase due to freedom or openness of people on social media platform to express their emotions without any fear and sensitivity towards the sentiment of readers. The social media platforms are unable to tackle the problem of this persistent online abuses, hate speech and offensive language on their platform in an efficient manner. Currently, a great deal of cost and time is involved to tackle this problem because the task is carried out manually to detect and remove such kind of posts. The key challenge for automatic detection of hate speech and on social media is to distinguish it from other kinds of similar text such offensive language, cyberbullying and another form of abuses. In this work, we performed feature extraction over twitter dataset which helps in identifying the hidden characteristics of hate speech on twitter data. We applied the MLP classifier to classify the text into hate and non-hate. We applied SVM and CNN as baseline models to compare the performance and achieved high accuracy when evaluated on four publicly available datasets.

***Keywords*— Hate speech, Offensive language, MLP, CNN, SVM, TF-IDF, POS.**

I. INTRODUCTION

Social media platforms like Facebook and Twitter have raised concerns about emerging dubious activity such as the intensity of hate, abusive and offensive behavior among its users. However, they are designated as a public space that provides greater opportunities to re-broadcast messages to large audience and even strangers can reply or put their views, opinion and can engage in public debates. Now a days, hate speech over social media platform is a major bone of contention for societies around all over the world. Many of the countries have their own separate cyber laws to tackle hate crime over social media. On 31st May 2016, Microsoft, Google, Facebook and Twitter, jointly agreed to a European Union code of conduct obligating them to review "the majority of valid notifications for removal of illegal hate speech" posted on their services within 24 hours[1]. In September 2017, EU top regulator found that these companies were unable to remove hate speech within 24 hours and they took more than a week to tackle 28% of the cases, so EU has threatened these companies on imposing heavy sanctions [2]. In October 2017, Germany has passed a bill named as NetzDG to regulate social media platforms to ensure they must remove hate speech within the stipulated time of 24 hrs. Many of the other countries are also interested in bringing new bills or amending the existing one to regulate social media platforms to curb hatred. Now a day many political parties use Twitter and other social media to promote their propaganda to influence voters. Sometimes it is used as a tool to tarnish someone's image, spread lies, hate speech and much more [3]. Since hate speech and offensive language used on social media platform affect our society or sometimes an individual as well, Twitter, Facebook, and many companies are doing lots of research work and spent a lot of money to curb this problem. However, after doing a lot of effort, they are still criticized for not doing enough work because it needs lots of manual effort to review the online posts, detect it as hate/offensive and delete these materials.

In our approach, we employed a multi-layer perceptron (MLP) as classifiers and used behavioral tendency of users towards racism and sexism to improve performance. Our main contributions are: (i) a deep learning based model for hate and offensive text classification which uses the user's behavioral characteristics as a feature. (ii) Efficient feature extraction related to positive, negative, neutral and compound sentiments which improve the performance of the MLP network for classification.

II. RELATED WORK

Hate speech, offensive language, cyberbullying and online abuse have impacted our society on a large scale in the recent time. So, there is a need for a scalable, automated approach to hate speech and offensive language detection. There are various methods of supervised learning like SVM, Naïve Bayes and Logistic Regression [5-15] for Hate speech detection on twitter. Furthermore, various techniques are applied for detection of offensive language [16-18]. [19] and [20] implement supervised learning methods for detecting racism and sexism. There is various research work has been done so far in the field of deep learning like CNN [21], CNN+GRU [22] to detect hate and offensive language on social media like Twitter. Greevy and Smeaton [5] proposed a

supervised method SVM to classify racist texts from different web pages. They crawled 3 million words formed a corpus. They applied bag of words and Bi-grams to extract features from each of the four datasets and used SVM for classification of racist text. They found that BoW gave the high precision of about 92.55% and recall of 87.00% on set-3. Burnap and Williams [7] proposed distributed lower-dimension representation of comments by using neural language model like bag of words, TF, TF-IDF, and paragraph2vec to detect Hate speech. They solved high dimensional data representation problem with classification but did not get the very good results of detecting hate speech. They selected the most important features by searching across character n-gram (one-gram, two-gram, tri-gram and four-gram) and performed 10-fold cross-validation to evaluate the model. They also considered meta information of users like Gender of the user, the average length of 1-4 words per tweet, Gender + Location and Gender+Location+Length. Davidson et al [8], proposed a supervised method of automatic hate speech and offensive language detection, in which, they used logistic regression with L2 regularization to overcome the overfitting and dimensionality reduction of data. They tested their baseline against Naïve Bayes, Decision Tree, Random-forest and Linear SVM. This was the first lexicon based multi-class hate and offensive language detection method that was given very good results while automatic detection, but sometimes it miss-classify offensive language as hate speech.

Lozano et al [19], proposed a unsupervised method of hate speech like racism and sexism detection, in this, they tried to find racist user as well as the user who pass sexist comment on Twitter during the election of a large country in 2016. They used clustering to classify the racist and sexist tweet. They also clustered users who favor one leader and spread racism and sexism, and other leader's supporter who spread racism and sexism on Twitter during the campaign. Jha and Mamidi [20], they used FastText [23] classifier to focus on the different form of Sexism named as Benevolent, which is very common on social media platforms. They first analyzed tweeter dataset posing sexism and classified it into three classes 'Hostile', 'Benevolent' and 'None' depending on the sexism type that represented by using SVM. They also used the sequence to sequence model by using tf-seq2seq framework given by [25] for Tensorflow [26]. Park and Fung [21] proposed a two-step method of doing classification on offensive language and a one-step method of performing one multi-class classification of detecting racism and sexism. To perform this, they used HybridCNN in one-step and Logistic regression in two-steps method. The HybridCNN made up of a combination of CharCNN and WordCNN Abidi et al [29] employed a deep learning based model on seven different datasets related to hate speech on Twitter. They employed CNN+GRU network architecture on these datasets to classify the hate speech as Racism and Sexism against a refugee community of a country. They performed a comparative evaluation on the largest publicly available dataset and found that the proposed method outperformed on all the baselines and is a state of the art among all. Chatzakou et al [24] used tweets as well as metadata like user information, time of retweet etc. They applied RNN on text to do feature extraction but final classification is postponed till meta-data passed through an MLP network. The feature matrix of tweets and metadata are concatenated and then final classification was performed.

III. PROPOSED MODEL

Preprocessing

We used two publicly available Twitter dataset of Hate speech for the evaluation on our model. Since tweets are raw texts containing lots of symbols, re-tweets, spelling mistakes etc, we do lots of preprocessing to make it clean. We do following preprocessing steps to clean the raw text

1. Convert texts into lower case and remove all the stop words.
2. Remove special symbols such as: & ! / \ ? & $; etc. using regular expressions.
3. Stemming and lemmatization.
4. Remove tokens having document frequency less than 5, which further removed sparse features which is less informative.
5. Further, we normalized the words like 'goooooood' to 'good' etc.

Feature Extraction and Selection

We used different types of features from the dataset such as TF-IDF, POS TF, and many others feature like VaderSentiment features, different syllables number of hashtags, mentions, Is retweet etc.

TF-IDF

The *tf-idf* is a weighting technique to assign a weight to a word or term in a document or tweet. The *tf-idf* can be given as follows:

$$\text{tf-idf}_{w,t} = \text{tf}_{w,t} \times \text{idf}_{w}$$

where $\text{tf-idf}_{w,t}$ represents assigning of weight to word (term) w in a tweet (document) t.

The value of $\text{tf-idf}_{w,t}$ is highest when word w present many times within very fewer tweets, lower when word w occur less number of time in a tweet and lowest when word w present in almost all the tweets. We used *tf-idf* to wight unigram, bigram, and tri-gram features.

POS tag

The POS tag is assigning to any of the parts of speech from noun, verb, adjective, and so on to each word in a tweet. This process is performed for each row of tweets in the dataset. We then represent bigram and trigram to POS tag and kept each tweet having frequency more than five to remove spatial features from the dataset. We used the transformation based POS tagging method for automatic tagging of parts of speech to a word in a tweet. To tag each word, the transformation-based approach uses transformation rules and transforms it from one state to another. It mines linguistic information in a readable form automatically.

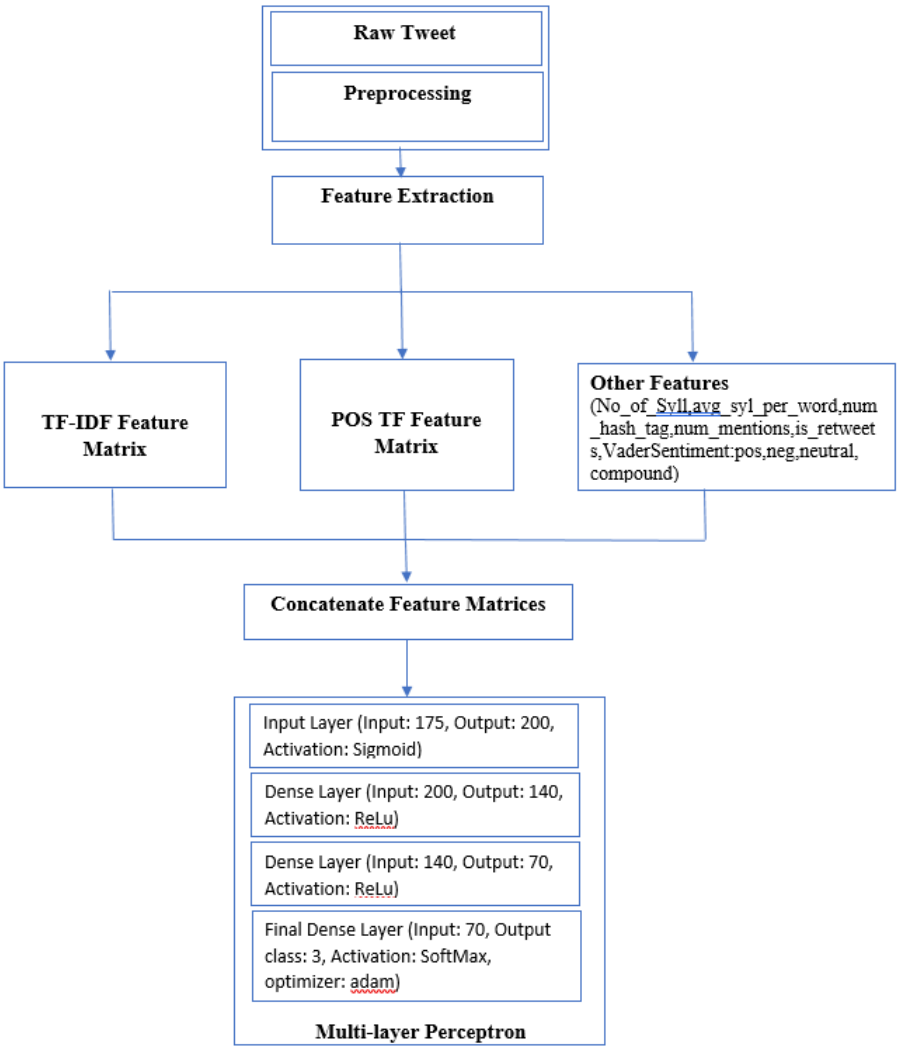


Fig.1 Multi-layer perceptron network flow graph for hate speech and offensive language detection on twitter.

Other Features

We extracted sentiment based features by using the Vader-Sentiment analyzer. We assigned four sentiments positive, negative, neutral and compound polarities to each word. We extracted features like number of mentions, number of hashtags, average number of syllables per tweet, number of unique words, number of retweets etc. in a tweet. We convert these other features into a feature matrix. We further concatenated these feature matrices (TF-IDF, POS Tag, and Other Features) to make a single feature matrix.

We used logistic regression to select important features that are important to our problem and stored it in a 2D matrix. This feature matrix has the same number of rows as the number of tweets (documents) in the dataset and number of the column represents the features corresponding to each tweet. This feature matrix is passed to the MLP network for final classification and learning task.

Multi-layer perceptron (MLP) based model

We used multi-layer perceptron as a deep learning model for hate speech and offensive language detection as shown in Fig.3. MLP network model consists of an input layer, three hidden layers and a soft-max layer as the output layer. The number of nodes in the input layer is the same as a number of column in the feature matrix and sigmoid as an activation function. The three hidden layers contain 200, 140 and 70 number of nodes and Rectified linear unit (Relu) as an

Dataset	No of Tweets	Classes (%Tweets)	Target Class
DT	24,783	Hate(11.6%), offensive(76.6%), Neither (11.8%)	Hate, Offensive
WZ-L	16,093	Racism(12.01%), Sexism(19.56%), None (68.41%)	Racism, Sexism

Table 1: Dataset description

activation function in each layer. Finally, we used a soft-max as an output layer having a loss as categorical class entropy, Adam as an optimizer and softmax as an activation function. We passed the data in the batch size of 16 and run the model for 50 epochs while training. Back-propagation algorithm has been used for training and weight updation.

IV. DATASETS

We crawled tweets specific to our problem using publicly available twitter data set in the form of tweet-id and label. We crawled tweets from twitter using tweepy API corresponding to each tweet-id and saved it as CSV file. Davidson [15] classified only hate speech, not its type they used the dataset having tweets annotated as hate or not hate only two classes, we refer this dataset as DT. The dataset created in Waseem et al [19] is named as WZ which has three classes: sexism, racism, and non-hate. The WZ-L dataset is created by [15].

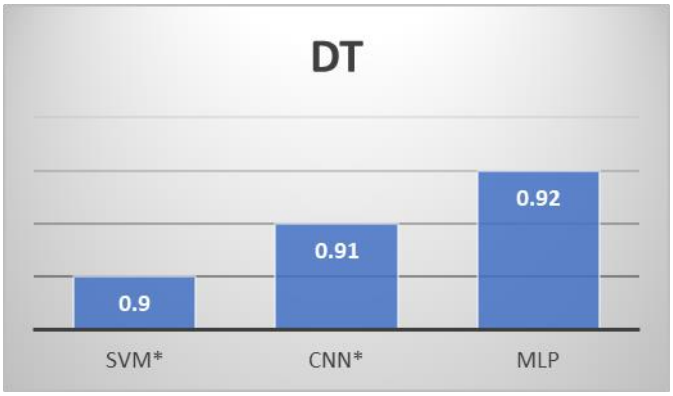
V. EXPERIMENT & RESULTS

For comparative evaluation setup, we also used Convolutional neural network (CNN) as well as Support vector machine (SVM). Just before feeding the sample data to CNN it is required to represent each word sample (tweets) to have the same number of words, so if any of tweet is having variable length then zero has been padded. In CNN, the first data sample is converted into 200-dimensional vector form by using GloVe. The GloVe is made by Google which is trained over 4 billion words, and each word is represented in a vector space depending on how it is related to its closed word. A convolutional neural network is having three 2D convolution layer each correspond to a max-pool layer, followed by a dropout of 0.5, then a dense layer as the output layer. Each Conv2D layer is having 128 filters of size 200*200 and at a time 2,3 and 4 vectors are selected to convert into a feature map and ReLu (Rectified Linear unit) as an activation function. Max-pooling layer is used to downsample the feature map by selecting the maximum of the square filter, and this operation is done all over the feature vector by striding the filter by 1. Finally, a dense layer is used for classification.

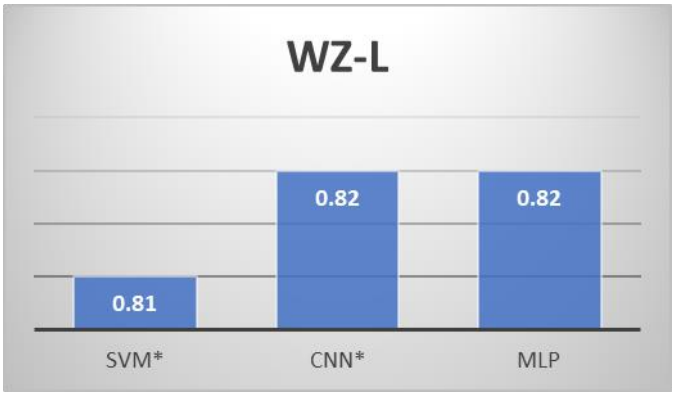
Datasets	SVM*	CNN*	MLP
WZ-L	0.81	0.82	**0.82**
DT	0.90	0.91	**0.92**

Table 2: Accuracy of MLP against SVM* and CNN*

We also used support vector machines (SVM) as a baseline model to check the performance of our proposed model. We passed features (TF-IDF, POS TF, and Other features) to the SVM model The Table.2. demonstrates the accuracy of our proposed MLP model against baselines SVM*, CNN*. The SVM* means using SVM as classifier after efficient feature extraction and selection for our proposed work. The CNN* means using CNN with customized layers and hyper-parameter as per work and datasets. Our proposed MLP based model gives an accuracy of 82% on WZ-L dataset which is greater than using SVM* on same dataset and same as CNN* on this dataset. We got 92% accuracy on dataset DT (hate) which is better than previous state of art result by Davidson et al as 87%. Fig.2 (a) shows the performance of three different models on the DT dataset. Further it illustrates that MLP gives an accuracy of 92% which is more than Davidson et al of 87% and SVM*. We used our same features extracted and selected for our model and passed it to SVM (termed as SVM*) and found that SVM* gives 90% accuracy while previously Davidson et al got 87% accuracy on SVM. Fig.2 (b) shows the performance of three different models on the WZ-L dataset. MLP gives the same accuracy of 82% as compared to CNN* and got 1% more accuracy than SVM*. In MLP based model we got more percentage of features selected relevant to the problem. SVM* gives the least accuracy of 81%.



(a) DT Dataset



(b) WZ-L Dataset

Fig.2 (a) Accuracy on DT Dataset (b). Accuracy on WZ-L Dataset

Table 3(a) illustrates the performance of our proposed model DT dataset. We got the highest precision of 0.95 for class offensive and least precision of 0.82 for class Hate. We got the highest recall of 0.96 for class offensive and least recall of 0.28 for class Hate. We got a highest F1 score of 0.96 for class offensive and a least F1 score of 0.42 for class hate. We got an average precision of 0.92, average recall of 0.92 and average F1 measure as 0.91 for dataset DT. Table 3(b) illustrates the performance of our proposed model on WZ-L dataset. We got the highest precision of 0.85 for class Sexism and least precision of 0.81 for class Racist. We got the highest recall of 0.93 for class None and least recall of 0.61 for class Sexism. We got a highest F1 score of 0.88 for class None and a least F1 score of 0.71 for class Sexism. We got an average precision of 0.83, average recall of 0.83 and average F1 measure as 0.82 for dataset WZ-LS.

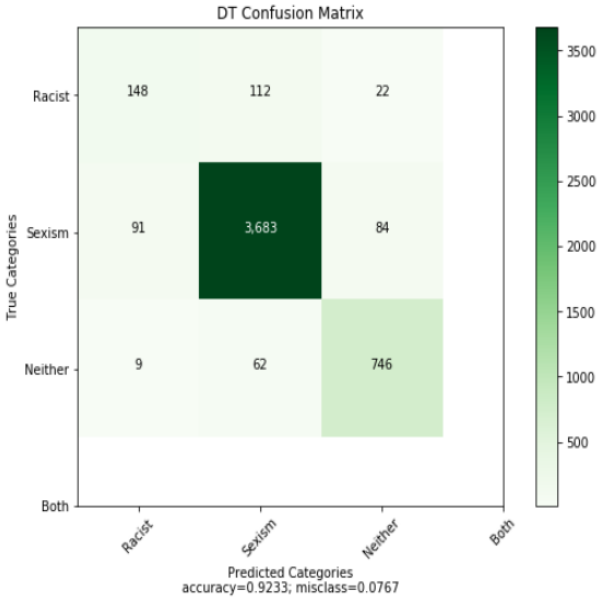
DT			
class	Precision	Recall	F1
Hate	0.60	0.52	0.56
Offensive	0.95	0.80	0.87
Neither	0.87	0.91	0.89
Overall	0.92	0.91	0.92

(a) DT Dataset

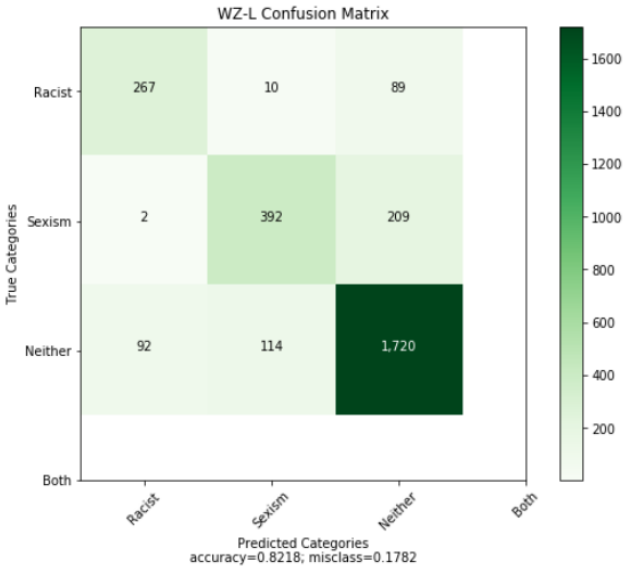
WZ-L			
class	Precision	Recall	F1
Racist	0.81	0.68	0.74
Sexism	0.85	0.61	0.71
None	0.83	0.93	0.88
Overall	0.83	0.83	0.82

(b) WZ-L Dataset

Table 3. (a). Performance of MLP on DT Dataset (b). Performance of MLP on WZ-L Dataset



(a) Confusion Matrix of DT Dataset



(b) Confusion Matrix of WZ-L Dataset

Fig.3 (a) Confusion Matrix of MLP on DT and (b) WZ-L Datasets

VI. CONCLUSION & FUTURE WORK

We investigated the deep learning based approach for detecting Hate and offensive language on Twitter. We used two publicly available hate and offensive language datasets for evaluation of our model. We performed feature extraction and selection from twitter dataset. We used tf, tf-idf, POS, and other features like sentiment (pos, neg, neutral, and compound) polarity, number of hashtags, retweets, and syllables. MLP network helps in classification of hate, offensive tweets. We also applied two other baseline models SVM with extracted features used in our model and CNN with 200D GLoVe word embedding, which has three convolutional filters of size 2,3,4. We got the highest accuracy of 93% on WZ-S.exp dataset, 83% on WZ-LS, 82% on WZ-L dataset and 92% on DT dataset. The effective feature extraction and selection helps us in getting almost 1% improved accuracy on WZ-S.exp and WZ-LS datasets and same accuracy on WZ-L dataset as compared to the previous state of art work. As we used tweets only two datasets for our work evaluation, we can further use metadata of tweets. We can use metadata based on networks and users like #followers and #friends, strength and effect of friends, the effect of mentions on a user, #posts, favorite tweets etc. along with tweets.

REFERENCES

[1] "Facebook, YouTube, Twitter and Microsoft sign EU hate speech code". *The Guardian*. Retrieved 7 June 2016." *https://www.theguardian.com/technology/2016/may/31/facebook-youtube-twitter-microsoft-eu-hate-speech-code"*

[2] "EU says it'll pass online hate speech laws if Facebook, Google, and others don't crack down". The Verge. Retrieved Sep 28th 2017. "*https://www.theverge.com/2017/9/28/16380526/eu-hate-speech-laws-google-facebook-twitter"*.

[3] Iginio Gagliardone, Danit Gal, Thiago Alves, Gabriela MartinezJ," countering online hate speech", in United Nations Educational, Scientific and Cultural Organization 7, place de Fontenoy, 75352 Paris 07 SP, France.

[4] "EU threatens to crack down on Facebook over hate speech" by Daniel Boffey in The Guardian, Brussels. Retrieved on April 11th 2018." https://www.theguardian.com/technology/2018/apr/11/eu-heavy-sanctions-online-hate-speech-facebook-scandal.

[5] Greevy E and Smeaton A F. "Classifying racist texts using a support vector machine"; In Proceedings of the 27th Annual International ACM SIGIR Conference on Research andDevelopment in Information Retrieval SIGIR '04, pages 468–469, New York, NY, USA, 2004. ACM.

[6] Burnap P and Williams M. "Cyber hate speech on twitter: An application of machine classification and statistical modeling for policy and decision making. Policy and Internet", 7(2):223–242, 2015

[7] Djuric N, Zhou J, Morris R, Grbovic M, Radosavljevic V, and Bhamidipati N. "Hate speech detection with comment embeddings" In Proceedings of the 24th International Conference on World Wide Web, pages 29–30. ACM, 2015.

[8] Davidson T, Warmsley D, Macy M, and Weber I. "Automated hate speech detection and the problem of offensive language"; In Proceedings of the 11th Conference on Web and Social Media. AAAI, 2017.

[9] Kwok I and Wang Y. "Locate the hate: Detecting tweets against blacks"; In Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence, AAAI'13, pages 1621–1622. AAAI Press, 2013.

[10] Mehdad Y and Tetreault J. "Do characters abuse morethan words?" In Proceedings of the SIGDIAL 2016 Conference,pages 299–303, Los Angeles, USA, 2016. Association for Computational Linguistics

[11] Warner W and Hirschberg J. "Detecting hate speech on the world wide web"; In Proceedings of the Second Workshop on Language in Social Media, LSM '12, pages 19–26. Association for Computational Linguistics, 2012.

[12] Waseem Z and Hovy D. "Hateful symbols or hateful people? predictive features for hate speech detection on twitter";In Proceedings of the NAACL Student Research Workshop, pages 88–93. Association for Computational Linguistics, 2016.

[13] Waseem Z. "Are you a racist or am i seeing things? annotator influence on hate speech detection on twitter"; In Proc. of the Workshop on NLP and Computational Social Science, pages 138–142. Association for Computational Linguistics, 2016.

[14] Xiang G, Fan B, Wang L, Hong J, and Rose C; "Detecting offensive tweets via topical feature discovery over a large scale twitter corpus"; In3` Conference on Information and Knowledge Management, pages 1980–1984. ACM, 2012.

[15] Yuan S, Wu X, and Xiang Y; "A two phase deep learning model for identifying discrimination from tweets"; In Proceedings of 19th International Conference on Extending Database Technology, pages 696–697, 2016.

[16] Xiang G, Fan B, Wang L, Hong J and Rose C. 2012. "Detecting offensive tweets via topical feature discovery over a large-scale twitter corpus"; In 21st ACM CIKM,1980–1984

[17] Clarke I, and Grieve J, 2017. "Dimensions of abusive language on twitter"; In Proceedings of the First Workshop on Abusive Language Online, 1–10.

[18] Mehdad Y, and Tetreault J R. 2016. "Do characters abuse more than words?" In SIGDIAL, 299–303.

[19] Lozano E, Cede˜no J, Castillo G, Layedra F, Lasso H, and Vaca C. 2017 "Requiem for online harassers: Identifying racism from political tweets"; In 4th IEEE Conference on eDemocracy & eGovernment (ICEDEG), 154–160.

[20] Jha A, and Mamidi R. 2017. "When does acompliment become sexist? analysis and classification of ambivalent sexism using twitter data"; In 2nd Workshop on NLP and Computational Social Science, 7–16.

[21] Park H. J. and Fung P. "One-step and two-step classcation for abusive language detection on twitter"; In ALW1: 1st Workshop on Abusive Language Online, Vancouver, Canada, 2017. Association for Computational Linguistics.

[22] Zhang Z, Robinson D and Tepper J, "Detection Hate Speech on Twitter Using a Convolution-GRU based DNN" In 15th ESWC 2018 conference on Semantic web.

[23] Joulin A, Grave E, Bojanowski P, and Mikolov T. 2016. "Bag of tricks for efficient text classification"; arXiv preprint arXiv:1607.01759.

[24] Founta M. A., Chatzakou D, Kourtellis N, Blacknurn J, Vakali A, Leontiadis I, "A Unified Deep Learning Architecture for Abuse Detection"; In 32nd AAAI conference on Artificial Intelligence Hilton New Orleans Riverside, New Orleans, Louisiana, USA 2018, vol = abs/1802.00385.

[25] D. Britz, A. Goldie, T. Luong, and Q. Le. 2017. Massive Exploration of Neural Machine Translation Architectures. ArXiv e-prints .

[26] Mart´ın Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. 2016. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. arXiv preprint arXiv:1603.04467.1