Using Cognitive Learning Method to Analyze Aggression in Social Media Text

Sayef Iqbal and Fazel Keshtkar[(✉)]

Science and Mathematics Department, St. John's University College of Professional Studies Computer Science, 8000 Utopia Parkway, Jamaica, NY 11439, USA
{sayef.iqbal16,keshtkaf}@stjohns.edu

Abstract. Aggression and hate speech is a rising concern in social media platforms. It is drawing significant attention in the research community who are investigating different methods to detect such content. Aggression, which can be expressed in many forms, is able to leave victims devastated and often scar them for life. Families and social media users prefer a safer platform to interact with each other. Which is why detection and prevention of aggression and hatred over internet is a must. In this paper we extract different features from our social media data and perform supervised learning methods to understand which model produces the best results. We also analyze the features to understand if there is any pattern involved in the features that associates to aggression in social media data. We used state-of-the-art cognitive feature to gain better insight in our dataset. We also employed ngrams sentiment and Part of speech features as a standard model to identify other hate speech and aggression in text. Our model was able to identify texts that contain aggression with an f-score of 0.67.

Keywords: Hate speech · Aggression · Sentiment · Social media · Classification

1 Introduction

According to Wikipedia[1], aggression is defined as the action or response of an individual who expresses something unpleasant to another person [4]. Needless to say, aggression in social media platforms has become a major factor in polarizing the community with hatred. Aggression can take the form of harassment, cyberbullying, hate speech and even taking jabs at one another. It is growing as more and more users are joining the social network. Around 80% of the teenagers use social media nowadays and one in three young people have been found victims of cyberbullying [25].

The rise of smartphones and smart devices and ease of use of social media platforms have led to the spread of aggression over the internet [8]. Recently,

---

[1] https://en.wikipedia.org/wiki/Aggression Date: 11/22/2018.

social media giants like Facebook and Twitter took some action and have been investigating this issue (i.e. deleting suspicious accounts). However, there is still a lack of sophisticated algorithms which can automatically detect these problems. Hence, more investigation needs to be done in order to address this issue at a larger scale. On the other hand, due to the subjectivity of the aggression and hate associated with aggression, this problem has been challenging as well. Therefore, an automatic detection system for front line defense against such aggression texts will be useful to minimize spread of hatred across social media platforms and it can help to maintain a healthy online environment.

This paper focuses on generating a binary classification model for analyzing any pattern from 2018 shared task TRAC (Trolling, Aggression and Cyberbullying) dataset [12]. The data initially was annotated into three categories as follows:

– Non Aggression (NAG)- there is no aggression in the text
– Overtly Aggressive (OAG)- text contains open aggressive lexical features
– Covertly Aggression (CAG)- text contains aggression without open acknowledgement of aggressive lexical features

Examples of each category are shown in Table 1.

Table 1. Some examples with original labels and our modified labels.

Text examples	Original label	New label
Cows are definitely gonna vote for Modi ji in 2019 ;)	CAG	AG
Don't u think u r going too far u Son of a B****........#Nigam	OAG	AG
Happy Diwali.!!let's wish the next one year health, wealth n growth to our Indian economy.	NAG	NAG

To analyze the aggression patterns, in this paper, we focus on building a classification model using Non Aggression (NAG) and Aggression (AG) classes. We combine the overlapping OAG and CAG categories into the AG category from the initial dataset. In this research, we investigate a combination of features such as word n-gram, LIWC, part of speech and sentimental polarity. We also applied different supervised learning algorithms to evaluate our model. While most of the supervised learning methods produced promising results, Random Forest classifier produced the best accuracy (68.3%) and f-score (0.67) while also producing state of the art true-positive rate of (83%). However, all the classifiers produced results with greater accuracy and precision for our proposed binary class (AG) and (NAG) than the initial three classes of (NAG), (CAG) and (OAG).

We also analyzed n-gram and LIWC features that were used for model building and found that it mostly affirms the presence of non-aggressive content in texts. This paper serves to lay the ground for our future work which is to identify what differentiates OAG from CAG.

The rest of the paper is organized as follows: Related Work section gives a brief overview of the research already done in this area. The Methodology section describes our methodology and the details about the dataset, pre-processing steps, feature extraction and the algorithms that were used. Experiments and Result section presents the experiments and results from the proposed model and finally, the conclusion and future works are discussed in Conclusion and Future work section.

2   Related Work

Several studies have been done in order to detect aggression level in social media texts [5]. Some research focuses on labelling the texts as either expressing positive or negative opinion about a certain topic. Raja and Swamynathan in his research analyzed sentiment from tweet posts using sentiment corpus to score sentimental words in the tweets [15]. They proposed a system which tags any sentimental word in a tweet and then scores the word using SentiWordnet's word list and sentimental relevance scoring using an estimator method. The system produced promising sentimental values of words. However, the research did not focus on the analysis of the lexical features or sentimental words, such as how often they appear in a text and what kinfd of part of speech does the word(s) belong to.

Samghabadi et al. analyzed data for both Hindi and English language by using a combination of lexical and semantic features [19]. They used Twitter dataset for training and testing purposes and applied supervised learning algorithms to identify texts as being Non Aggressive, Covertly Aggressive and Overtly Aggressive. They used lexical feature such as word n-gram, char-n-gram, k-skip n-grams and tf-idf word transformation. For word embedding, they employed Word2vec [24] and also used Stanford's sentimental [21] tool to measure the sentiment scores of words. They also used LWIC to analyze the texts from tweets and Facebook comments [22]. Finally, they used binary calculation to identify the gender probability to produce an effective linguistic model. They were able to retrieve an f-score of 0.5875 after applying classifiers on their linguistic featured model. In contrast to this research, our system produced results with higher f-score (0.67) even though we used different feature set and employed different approach for supervised learning and analysis.

On the other hand, Roy et al. [17] used Convolution Neural Networks (CNN) [11] and Support Vector Machine (SVM) classifiers on the pre-processed dataset of tweets and Facebook comments to classify the data. They employed the pre-processing technique of removing regular expressions, urls and usernames from the text. They used an ensemble approach using both CNN and SVM for classifying their data. In contrast to our research which produced better results when using Random Forest classifier, the performance of their system improved when SVM was used on unigrams and tf-idf features along with CNN with a kernel size of 2 x embedding size. The system was able to classify the social media posts with an f-score of 0.5099.

On a different note, Sharma et al. proposed a degree based classification of harmful speeches that are often manifested in posts and comments in social media platforms [20]. They extracted bag of word and tf-idf features from pre-processed Facebook posts and comments that was annotated by three different annotators subjectively. They performed Naive Bayes, Support vector Machine and Random Forest classifiers on their model. Random Forest worked the best on their model and gave results with an accuracy of 76.42%.

Van Hee et al. explored the territory of cyberbullying, a branch of aggression, content in social media platform [23]. Cyberbullying can really affect the confidence, self-esteem emotional values of a victim especially in the youth. They propose a linear SVM supervised machine learning method for detecting cyberbullying content in social media by exploring wide range of features in English and Dutch corpus. The detection system provides a quantitative analysis on texts for a way to signal cyberbullying events in social media platform. They performed a binary classification experiment for the automatic detection for cyberbullying texts with an f-score of 64.32% and for English corpus. Unlike in our research, [23] did not employ sentiment or use any psycholinguistic feature for supervised learning methods. However, our system was able to produce slightly better f-score result even though we use different dataset.

Sahay et al. in their research address the negative aspect of online social interaction. Their work is based on the detection of cyberbullying and harassment in social media platforms [18]. They perform classification analysis of labelled textual content from posts and comments that could possible contain cyberbullying, trolling, sarcastic and harassment content. They build their classification model based on n-gram feature, opposed to our system in which we consider other features like part of speech along side n-gram features. They perform different machine learning algorithms to evaluate their feature-engineering process generating a score between 70–90% for the training dataset.

Similarly, Reynolds et al. perform a lexical feature extraction on their labelled dataset that was collected from web crawling that contained posts mainly from teenagers and college students [16]. They proposed a binary classification of 'yes' or 'no' for posts from 18,554 users in Formspring.me website that may or may not contain cyberbullying content. They perform different supervised learning method on their extracted features and found J48 produced the best true positive accuracy of 61.6% and an average accuracy 0f 81.7%. While the researchers were able to retrieve results with better accuracy they do not analyze the texts within which is a strong focal point of our research.

Dinkar et al. propose a topic-sensitive classifier to detect cyberbullying content using 4,500 comments from youtube to train and test their sub-topic classification models [7]. The sub-topics included, sexuality, race and culture, and intelligence. They used tf-idf, Ortony lexicons, list of profane words, part of speech tagging and topic-specific unigrams and binary grams as their features. Although they applied multiple classifiers on their feature model, SVM produced

the most reliable with kappa value of above 0.7 for all topic-sensitive classes and JRip producing most accurate results for all the classes. They found that building label-specific classifiers were more effective than multiclass classifiers at detecting cyberbullying sensitive messages.

Chen et al. also propose a Lexical Syntactic Feature (LSF) architecture to detect use of offensive language in social media platforms [6]. They included a user's writing style, structure, lexical and semantic of content in the texts among many others to identify the likeliness of a user putting up an offensive content online. They achieved a precision of 98.24% and recall of 94.34% in sentence offensive detection using LSF as a feature in their modelling. They performed both Naive Bayes and SVM classification algorithm with SVM producing the best accuracy result in classifying the offensive content.

However, in this paper, we propose a new model, which to the best of our knowledge, has not been used in previous researches. We build a model with a combination of psycholinguistic, semantic, word n-gram, part of speech and other lexical features to analyze aggression patterns in the dataset. Methodology section explains the details of our model.

3   Methodology

In this section we discuss the details of our methodology, dataset, pre-processing, feature extraction, and algorithms that have been used in this model. The data was collected from shared TRAC [12].

3.1   Dataset

The dataset was collected from the TRAC [12] workshop (Trolling, Aggression and Cyberbullying) 2018 workshop held in August 2018, NM, USA. TRAC focuses on investigating online aggression, trolling, cyberbullying and other related phenomena. The workshop aimed to create a platform for academic discussions on this problem, based on previous joint work that they have done as part of a project funded by the British Council. Our dataset was part of the workshop's English data that comprised of 11,999 Facebook posts and comments with 6,941 comments labelled as Aggessive and 5,030 as Non-aggressive. The comments were annotated subjectively into three categories NAG, CAG and OAG by research scientists and reviewers who organized the workshop. We decided to use a binary class of AG and NAG for these texts. Figure 1 illustrates the distribution of the categories of aggression in texts. We considered complete dataset for analyzing and model building. The corpus is code-mixed, i.e., it contains texts in English and Hindi (written in both Roman and Devanagari script). However, for our research, we only considered using English text written in Roman script. Our final dataset, excluding Devanagari script, contained 11,999 Facebook comments.
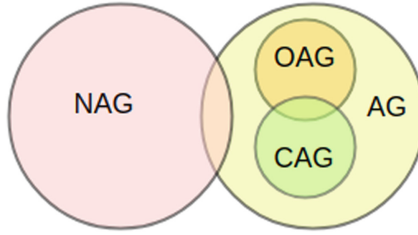
Fig. 1. Distribution of dataset OAG(22.6%) CAG(35.3%) & NAG(42.1%)

3.2   Pre-processing

Pre-processing is the technique of cleaning and normalization of data which may consist in removing less important tokens, words, or characters in a text such as 'a', 'and', '@' etc. and also lowering capitalized words like 'APPLE'.

The texts contained several unimportant tokens, for instance, urls, numbers, html tags, and special characters which caused noise in the text for analysis. We cleaned the data first by employing NLTK (Natural language and Text Processing Toolkit) [2] stemmer and stopwords package. Table 2 illustrates the transformation of text before and after pre-processing.

Table 2. Text before and after pre-processing

Before	Respect all religion sir, after all we all have to die, and after death there Will be no disturbance and will be complete silence
After	Respect religion sir die death disturbance complete silence

3.3   Feature Extraction

In this section we describe the features that we extracted from the dataset. We extracted various features, however, for the sake of this specific research, we only consider the following features due to their better performance in our final model. We adapted the following features- part of speech, n-grams (unigrams, bigrams, trigrams), tf-idf, sentiment polarity and LIWC's psycholinguistic feature. Figure 2 illustrates the procedure that was adapted in the process of feature extraction to build a model for supervised learning.

Part-of-Speech Features. Part-of-Speech (PoS) are classes or lexical categories assigned to words due to the word's similar grammatical and syntactic properties. Typical examples of part of speech include adjective, noun, adverb, etc. PoS helps us to identify or tag words into certain categories and find any pattern they create with regards to aggression and non-aggression texts. For the purposes of this research, we applied NLTK's [2] part of speech tagging package
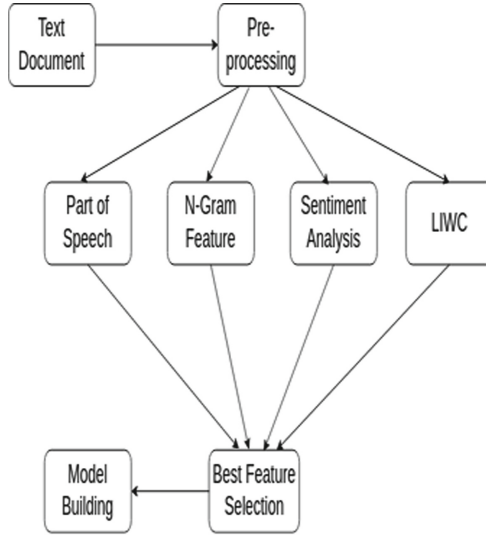
Fig. 2. Feature extraction architecture of system.

on our dataset to count the occurrences of PoS tags in each text. This led to the extraction of 24 categories of words. For instance extracting PoS tags from the text *respect religion sir die death disturbance complete silence* leaves us with 'respect': NN, 'religion': NN, 'sir': NN, 'die': VBP, 'death': NN, 'disturbance': NN, 'complete': JJ, 'silence': NN where NN represents for tagging a noun word and JJ and VBP for adjective and verb of non-3rd person singular present form, respectively.

N-grams Features. Language model or n-grams in natural language processing (NLP) refers to the sequence of n items (word, token) from texts. Typically, n refers to the numbers of words in a sequence gathered from a text after applying text processing techniques. N-gram is commonly used in NLP for developing supervised machine learning models [3]. It helps us identify which words tend to appear together more then often. We implemented Weka [9] tool to extract unigram, bigram and trigram word feature from these texts. We utilized Weka's Snowball stemmer to stem the words for standard cases and Rainbow stopwords to further remove any potential stop words. We considered tf-idf score as values of word n-gram instead of their frequencies. Over 270,000 tokens were extracted after n-gram feature extraction. We also employed Weka's built-in ranker algorithm to identify which features contribute most towards the correct classification of the texts. This helped us understand which words were most useful and related to our annotated classes. We considered only top 437 items for further analysis. We dropped features ranked below 437 as they were barely of any relevance as per ranker algorithm. Table 3 illustrates some examples of unigram,

bigram and trigram after applying n-gram feature extraction on the text '*respect religion sir die death disturbance complete silence*'.

Table 3. Examples of n-gram features

n-gram	Example of n-gram tokens
unigram	Respect, religion, disturbance
bigram	Respect religion, disturbance complete
h trigram	Respect religion sir, die death disturbance

Sentiment Features. Sentiment features are used to analyze any opinion expressed from texts as having a positive, negative or neutral emotion [10]. Sentiment analysis, especially in social media texts, is an important technique to monitor public opinion over a topic. It helps to understand the opinion expressed in a text by performing and evaluating the sentiment value of each word and overall text.

We used TextBlob [1] to evaluate the score of sentiment polarity of each pre-processed word and the text as a whole. TextBlob provides easy access to common text-processing operations. The package converts sentences to a list of words and performs word-level sentiment analysis to give a sentiment polarity score for each text. Sentiment polarity is a floating number ranging from –1.0 to 1.0. A number closer to –1.0 is an expression of negative opinion and a number closer to 1.0 is an expression of positive opinion. We keep track of the document id and the corresponding sentiment polarity score as a feature. For instance, the text '*respect religion sir die death disturbance complete silence*' produced a sentiment polarity score of –0.10 with subjectivity of –0.40. However, we only consider polarity score in the feature.

Linguistic Inquiry and Word Count Features. LIWC (Linguistic Inquiry and Word Count) performs computerized text analysis to extract psycholinguistic features from texts [14]. We utilized LIWC 2015 psychometric text analyzer [13] in order to gain insight on our data. The features provide understanding to textual content by scoring and labelling the text segments into many of its categories. In this research we applied Weka's ranking algorithm on LIWC features to rank the most significant and useful LIWC feature that contributed most towards classifying the texts as AG or NAG. We found 12 such LIWC features which were crucial in our analysis and which produced the best accuracy and f-score for our classifiers. 3 illustrates the distribution of the psycholinguisting features among 11,999 facebook comments. Each document may contain one or more of these cognitive features.
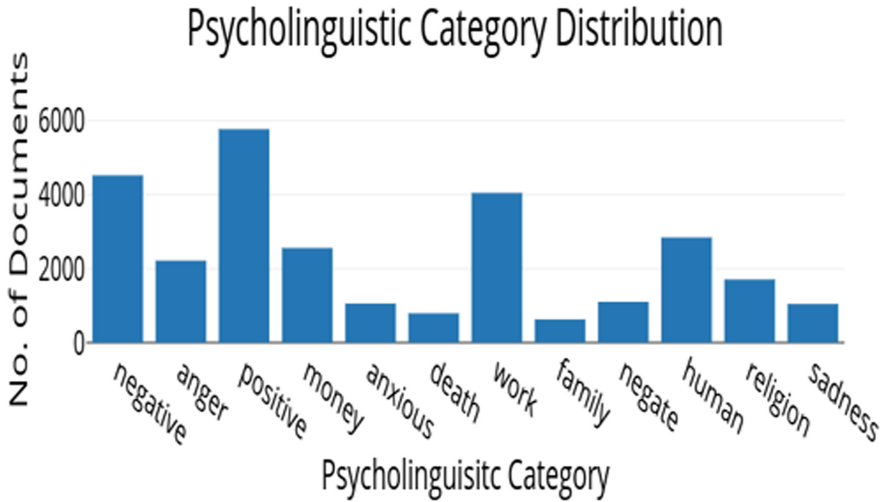
Fig. 3. Psycholinguist category distribution using LIWC

4   Experiments and Results

4.1   Experimental Setup

In this section we evaluate the performance of our model using supervised learning algorithms. We report accuracy and f-score of different supervised learning methods on the models that was created using the features explained in previous sections. We also evaluated the validity of our models and identify vital features and patterns that caused high and low performances in our system.

4.2   Result

We considered different combinations of features to build the best possible model that could eventually lead to higher performance. We ran various algorithms such as Support Vector Machine and Random Forest on different combination of features. Some combination of features performed better than others and we picked the one that produced best result. We noted from our results that Random Forest produced better results. Table 4 shows the results obtained by applying these classifiers on different combination of features.

We kept n-gram words as our gold standard feature in model building and then applied different combinations of other features. The features that were used in model building were Unigram (U), Bigram (B), Trigram (T), Sentiment polarity (SP), Part of Speech (PoS) and LIWC (LIWC). We applied different classifier using both 10-fold cross validation and 66% data for training using multiple combination of these features. The best results were obtained when

Table 4. F-score of classifiers using 66% data for training

F score		
Feature	SVM	Random forest
U+B+T	0.6100	0.6340
U+B+T+SP	0.6360	0.6500
U+B+T+SP+LIWC	0.6410	0.6450
U+B+T+SP+LIWC+PoS	0.6450	**0.6700**

considering U, BU, SP, GI and SW as features and using 66% of the data for training and 33% for testing.

Figure 4 shows the confusion matrix of our model for both classes, AG and NAG. The confusion matrix was generated by applying Random Forest classifier on 66% of the data (testing set) using U+B+T+SP+PoS+LIWC features in the model. Interestingly, according to the confusion matrix, upon applying Random Forest classifier on the model, 1,930 out of 2,351 of the AG class texts in the test set were identified correctly. This leads us to understand that the true positive for aggression in texts was 83% which is extremely promising.
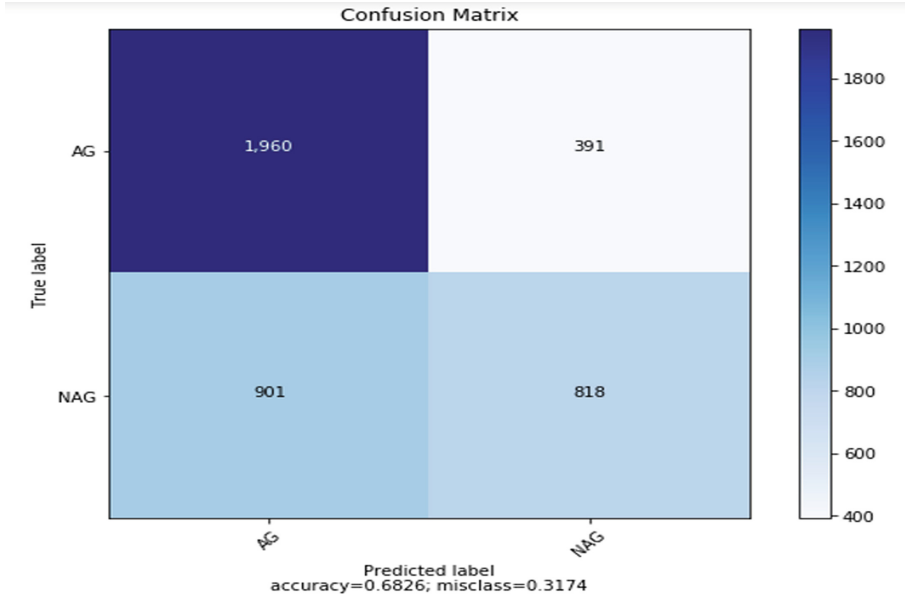
Fig. 4. Confusion matrix of random forest classifier.

We also found that the sentiment polarity score for texts were evenly distributed among both the classes, even though it was evaluated as a vital feature

by ranking algorithm. And it was among the top feature ranked by ranker algorithm which contributed towards higher accuracy and f-score.

We also found that some of the words happened to exist in both unigram and bigram, for instance, 'loud' in 'loudspeaker' and 'loud noise'. This leaves us to understand that those words are key in classifying the texts. When considering word n-gram feature, there were very few bigrams in the model as it mostly comprised of unigrams and contained words that related to religion and politics. Also, most words in texts that were annotated as aggressive comprised of adjectives and nouns.

4.3   Discussion and Analysis

A common issue with the dataset was that it often contained either abbreviated or unknown words and phrases which could not be extracted by using any of the lexicons. Hence, these words and phrases were left out of our analysis. Also, some texts contained either stop words or a mixture of stop words and emoticons which led to the removal of all of the content upon pre-possessing. Performing pre-processing on the text *hare pm she q ni such or the hare panic mr the h unto ab such ran chalice* led to the removal of whole text. This also prevented us from further analyzing the text even though it may potentially have had some aggressive lexical or emoticons. But because emoticons can be placed sarcastically in texts, we did not consider it as a feature in our model.

There were some texts which comprised of non-English words. The words in these texts switched between English and other languages which made our analysis difficult as it was solely intended for English corpus. Some words like 'dadagiri', which means 'bossy' in Hindi context, were not transliterated, which is why the semantic of the text could not be captured. The sentiment polarity score for the text '*chutiya rio hittel best mobile network india*' was 0.0 where clearly it should have been scored below 0.0 as it contained a strong negative word in another language (Hindi in Roman script).

Analyzing Result Data. Adjective (JJ), Verb non-3rd person singular present form (VBP) and NN (noun) were among the prominent part of speech for n-gram words that we extracted. Figure 5 illustrates the distribution of part of speech in our n-gram words feature.

Since Sentiment Polarity (SP) was among the top features ranked by Weka, (SP) as a feature identified 6,094 texts correctly and 5,905 texts incorrectly. Out of the 6,094 that were correctly identified, only 1,861 texts were labelled as AG and 4,233 as NAG. Figure 6 illustrates the distribution of AG and NAG labels after performing sentiment polarity analysis on the texts. Also, of the top 434 n-gram words ranked by Weka, (SP) identified 396 n-gram words as NAG and only 37 as AG. This clearly indicates, that sentiment polarity is a good feature in identifying NAG texts.
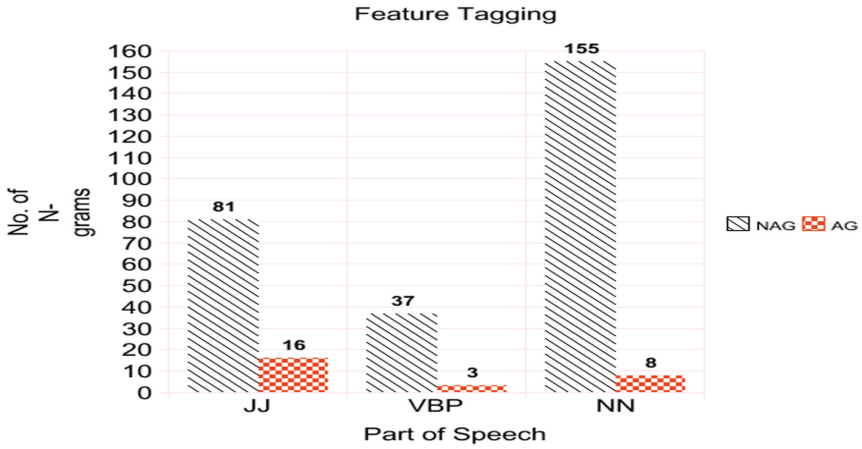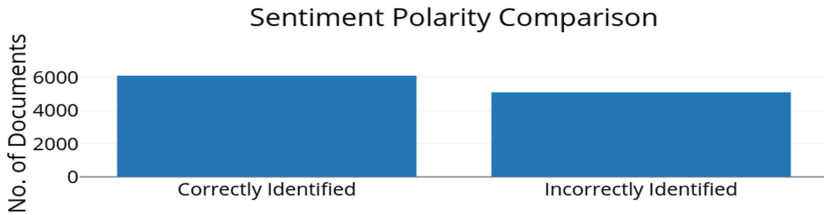
Fig. 5. Part of speech tagging of n-gram features

Fig. 6. Comparison of sentimental polarity feature

5   Conclusion and Future Work

In this paper, we propose an approach to detect aggression in texts. We tried to understand patterns in both AG and NAG class texts based on the part of speech and sentiment. The model produced promising results as it helps us to make clear distinction between texts that contain aggression and those that do not. Our System architecture also adapted well to the feature extraction process for aggression detection.

For future work, we plan to use more lexicon features for sentiment analysis in order to further improve the accuracy and f-score value for correct classification of our model. We also plan to use hashtags and emoticons which we think will be promising features. These features will help us to identify more important words and contents from texts that were not detected. We would also like to investigate on the sub domains of Aggression- Covertly Aggressive and Overtly Aggressive contents and identify distinguishing factors between them.

References

1. Bagheri, H., Islam, M.J.: Sentiment analysis of twitter data. arXiv preprint arXiv:1711.10377 (2017)
2. Bird, S., Loper, E.: Nltk: the natural language toolkit. In: Proceedings of the ACL 2004 on Interactive Poster and Demonstration Sessions, p. 31. Association for Computational Linguistics (2004)
3. Brown, P.F., Desouza, P.V., Mercer, R.L., Pietra, V.J.D., Lai, J.C.: Class-based n-gram models of natural language. Comput. Linguist. **18**(4), 467–479 (1992)
4. Buss, A.H.: The psychology of aggression (1961)
5. Chatzakou, D., Kourtellis, N., Blackburn, J., De Cristofaro, E., Stringhini, G., Vakali, A.: Mean birds: detecting aggression and bullying on twitter. In: Proceedings of the 2017 ACM on Web Science Conference, pp. 13–22. ACM (2017)
6. Chen, Y., Zhou, Y., Zhu, S., Xu, H.: Detecting offensive language in social media to protect adolescent online safety. In: Privacy, Security, Risk and Trust (PASSAT), 2012 International Conference on and 2012 International Confernece on Social Computing (SocialCom), pp. 71–80. IEEE (2012)
7. Dinakar, K., Reichart, R., Lieberman, H.: Modeling the detection of textual cyberbullying. Soc. Mob. Web **11**(02), 11–17 (2011)
8. Görzig, A., Frumkin, L.A.: Cyberbullying experiences on-the-go: when social media can become distressing. Cyberpsychology **7**(1), 4 (2013)
9. Holmes, G., Donkin, A., Witten, I.H.: Weka: a machine learning workbench. In: Intelligent Information Systems, 1994. Proceedings of the 1994 Second Australian and New Zealand Conference on, pp. 357–361. IEEE (1994)
10. Keshtkar, F., Inkpen, D.: Using sentiment orientation features for mood classification in blogs. In: 2009 International Conference on Natural Language Processing and Knowledge Engineering, pp. 1–6 (2009). https://doi.org/10.1109/NLPKE.2009.5313734
11. Kim, Y.: Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882 (2014)
12. Kumar, R., Reganti, A.N., Bhatia, A., Maheshwari, T.: Aggression-annotated corpus of Hindi-English code-mixed data. arXiv preprint arXiv:1803.09402 (2018)
13. Pennebaker, J.W., Boyd, R.L., Jordan, K., Blackburn, K.: The development and psychometric properties of liwc2015. Technical Report (2015)
14. Pennebaker, J.W., Francis, M.E., Booth, R.J.: Linguistic inquiry and word count: Liwc 2001. Mahway: Lawrence Erlbaum Assoc. **71**(2001), 2001 (2001)
15. Raja, M., Swamynathan, S.: Tweet sentiment analyzer: Sentiment score estimation method for assessing the value of opinions in tweets. In: Proceedings of the International Conference on Advances in Information Communication Technology & Computing, p. 83. ACM (2016)
16. Reynolds, K., Kontostathis, A., Edwards, L.: Using machine learning to detect cyberbullying. In: Machine Learning and Applications and Workshops (ICMLA), 2011 10th International Conference on, vol. 2, pp. 241–244. IEEE (2011)
17. Roy, A., Kapil, P., Basak, K., Ekbal, A.: An ensemble approach for aggression identification in English and Hindi text. In: Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018), pp. 66–73 (2018)
18. Sahay, K., Khaira, H.S., Kukreja, P., Shukla, N.: Detecting cyberbullying and aggression in social commentary using NLP and machine learning. people (2018)
19. Samghabadi, N.S., Mave, D., Kar, S., Solorio, T.: Ritual-uh at trac 2018 shared task: aggression identification. arXiv preprint arXiv:1807.11712 (2018)

20. Sharma, S., Agrawal, S., Shrivastava, M.: Degree based classification of harmful speech using twitter data. arXiv preprint arXiv:1806.04197 (2018)
21. Socher, R., et al.: Recursive deep models for semantic compositionality over a sentiment treebank. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pp. 1631–1642 (2013)
22. Tausczik, Y., Pennebaker, J.: The psychological meaning of words: Liwc and computerized text analysis methods. J. Lang. Soc. Psychol. **29**, 24–54 (2010)
23. Van Hee, C., et al.: Automatic detection of cyberbullying in social media text. arXiv preprint arXiv:1801.05617 (2018)
24. Wang, H.: Introduction to word2vec and its application to find predominant word senses. http://compling.hss.ntu.edu.sg/courses/hg7017/pdf/word2vec %20and%20its%20application%20to%20wsd.pdf (2014)
25. Zainol, Z., Wani, S., Nohuddin, P., Noormanshah, W., Marzukhi, S.: Association analysis of cyberbullying on social media using apriori algorithm. Int. J. Eng. Technol. **7**, 72–75 (2018). https://doi.org/10.14419/ijet.v7i4.29.21847