# Comparative studies of cyberbullying and sarcasm detection on social media

John Hani, Mohamed Nashaat, Mostafa Ahmed, Zeyad Emad

January 16, 2019

**Abstract**

In this comparative paper, we are going to discuss the different approaches and methods implemented by other papers to detect cyberbullying and sarcastic comments over social media. The reason we did so, because cyberbullying is an emerging problem that faces a lot of people on using the social media. Moreover, we focused on comparing between different classifiers that have been used in these papers like: (SVM)Support Vector Machine, Naive Bayes and Random Forest on advantages and drawbacks of each one of them related to the cyberbullying and sarcasm detection. We also mentioned the different preprocessing and the datasets that have been used in these papers. Finally, we present our proposed model for this problem.

## 1 Introduction

**As the increasing of social media nowadays there is an increasing in the cybercrimes and we all know that everybody now is using social media in his daily life. Cyberbullying now is one of the bad effects of the social media according to bullying statistics.org over half of the youth have been cyberbullied so the main objective of this study is to carry out comparative studies on the various Cyberbullying and sarcasm methods and we want to address the problems and the drawbacks that was in this methods hence we want to make our cyberbullying and sarcasm detection system the agenda of this paper consists of 5 sections. Section 2 describes related work regarding cyberbullying and sarcasm detection system. Section 3 discusses the comparison of the methods deployed in cyberbullying and sarcasm detection addressing its problems, Section 4 contains the proposed model and Section 5 contains the conclusion.**

## 2 Related Work

**The authors Nandhini and Sheeba have talked in [17] about how social networking is increasing in recent years accompanied with cyberbully. They said that detection of online harassment and provision of social media are the way to stand against cyberbullying. They used Levenshtein algorithm and Naive Bayes Classifier to detect cyberbully cases. They reached 91% accuracy on MySpace.com dataset with 500 posts. They finished saying that their system can help authorities to force laws on internet and fight cyberbullying.**

**The authors Nobata, et al have showed us in [19] that using abusive language has increased recently and can't be detected with the current approaches available. They used a framework called Vowpal Wabbit for classification. They developed a supervised classification methodology with NLP features that outperform a deep learning approaches. Finally, the F-score has reached 0.817 using dataset collected from comments posted on yahoo! News and Finance.**

**The authors Buchanan, et al discussed in [16] the serious nature of cyberbullying and how to detect it. They used War Of Tanks game chat messages to help in the detection of cyberbullying. They manually classified 5,000 messages into multiple categories for each message to determine it's level. Then they compared it with the simple naive automatic classification that uses sentiment analysis as features. Finally, the results were poor when compared with the manually classified results.**

**The authors Nandhini and Sheeba have discussed in [18] the importance of having systems that detects cyberbullying due to the increase of social networking activities. They purposed a system that help governments to detect cyberbullying and take actions to prevent it. Their system uses genetic operations(FuzGen) like crossover and mutation for optimizing the parameters before sending it to the Naive Bayes classifier. Finally, they reached 0.87 accuracy.**

**The authors Dadvar and de Jong have talked in [5] about how serious cyberbullying is, they hypothesized that the use of user's information is essential for more precise detection like: age and gender. They used SVM model in WEKA(data mining software) for classification and they used Mypes service that allows the aggregation of users' profiles so they can link users on multi social media platforms. They calculated the ratio of profane words and the ratio of pronouns in each post then used as features along with TFIDF. Finally, they reached 0.44 precision.**

**Here in [27] they propose a framework specific to cyberbullying detection using word embeddings that makes a list of pre-defined insulting words and assign different weights to obtain bullying features, which are then concatenated with Bag-of-Words and latent semantic features to form the final representation before feeding them into a linear SVM classifier.Thier problem that cyberbullying is a binary classification .they collected a dataset from twitter then they preprocess the dataset by removing the special characters including user mentions and urls and replace them by predefined characters. They used tfidf, latent semantic features and bullying features that comes from insulting seeds as feature extraction. The results of these framework was 79.4 as a recall.**

**In paper [20] they get their dataset from Myspace and manually marked them manually and these cause weakness in their dataset and they used stemming and stop word removal from as preprocesses. Then they used sentiment analysis and SVM classifier.**

**In this paper [4] they propose the Lexical Syntactic Feature (LSF) architecture to detect offensive words in social media they have results 98.24% as precision and 94.34% as recall in sentence offensive detection they have the dataset from YouTube comments and they have bag of words as feature extraction but the fact that Bag of words leads to high false positive.**

**Here in this paper [25] they made watch dog application that detect offensive words and images they have used sentiment analysis and adult image algorithm .They collected their dataset from user messages. Sentiment analysis is not enough because the computer cannot detect sarcasm.**

**[24] They propose a technique based on SNM. They collected the dataset from social media then the used SNA measurements and Sentiments as features. In this work they made 7 experiments .they have results precision accuracy is around 0.79 and the recall around 0.71.the accuracy of this work is too weak.**

**In This paper [23] enhanced the Naïve Bayes classifier for extracting the words and examining loaded pattern clustering. They proposed a method that resulted increasing accuracy and reliability of an experiment. Texts are fed into cluster and discriminant analysis stage which is able to identify abusive texts. The abusive texts are then clustered by using K-Mean. Naïve Bayes is used as classification algorithms to build a classifier from their training datasets that builds a predictive model. They used naïve Bayes to classify the abusive**

**text from 1 to 8. The categories contains communicative, activities approach, compliment, desensitization, isolation, reframing, personal information and relationship. Naïve Bayes performs the classification accuracy = 95.79%.**

**Here in [9] the primary goal of this research paper is to understand different methods for Automatic sarcasm detection. this paper describes techniques used for studying the various approaches available in Automatic Sarcasm Detection Field. They have used SVM classifier and it results F1 Score (0.41) with n gram and it results (0.56) with sentiment. The problem here that SVM need to be more precise in their analysis.**

**According to Vikas S Chavan and Shylaja S S in their paper [3]. Their pre-processing consists of 2 main parts the "removing unwanted strings" and "correcting words". Second part the standard feature extraction. They used they feature extraction methods for this part. First, N-gram method, they used the n-gram method to extract the tokens they are going to work on. Second, counting, in this part they are going to tokenize the comments and count the occurrence of every token. Third, TF-IDF, they used it to calculate the importance of the words according to their number of occurrences in the document. Third part is the additional features. They used 2 methods in this part which are capturing pronouns and skip-gram. Fourth part is feature selection. For this part they used the chi- square which is a method that is used to select the best features to work on. Last part, classification. They used 2 classifiers for this which are the logistic regression and support vector machine. The logistic regression achieved 73.76 accuracy and 0.6 recall and 0.644 Precision. But for the support vector machine they achieved 77.65 accuracy and 0.58 recall and 0.7 precision's Got their Datasets from Kaggle. They have multiple drawbacks like in the pre-processing part they used to correct the words that is written to bypass their classifier according to their paper they are going to use the dictionary to make this work but if the model encountered a new word that isn't available in the dictionary it is going to bypass it without alerting the user that there is a cyberbully in this comment. Also, they haven't used any sentiment or contextual analysis to make sense of the words which is going to make a big problem of false positive because there could be 2 friends talking and use any word that is in the dictionary of the bullying words so this is going to raise alert flag without any actual bullying.**

**[14] This paper introduces a new technique to detect cyberbullying on the social media through using 2 famous classifiers which are Naive**

**Bayes And SVM. They constructed a system that consists of 4 layers which are , Data Collection , Pre-Processing , Extraction and Classification. Data Collection. They Used a data set from Kaggle which is a group of messages fromspring and it had Questions , Answers and Severity. Pre-Processing. This stages they Made the data cleaning which exported it in 10 folders according to the severity class and removed the records that has characters under 15 and removed meaningless Words. And then they Tokenized the text into words . Then they transformed all the text to lower case, moving on they removed stop words and then Filter token that chooses words between 3-25 characters as less than 3 is stop word. Then stemming to make the word go back to it's original form. Last they generated N-Grams they used the N-Grams from 2-5. Feature Extraction. In this stage they used TF-IDF as a feature to calculate the weight of the word in the sentence. For the Classification They used Naive Bayes and SVM with linear Poly RBF and Sigmoid Kernels. Naive Bayes yields an average accuracy of 92.81%, SVM with a poly kernel yields an average accuracy of 97.11%. They Collected their Datasets from Fromspring Dataset From Kaggle. The Drawbacks ,They Removed the Stop words from the text which means they removed the pronouns and there could be text with any strong word but it isn't referring to actual person and will be classified as cyberbullying.**

**[1] This Paper Introduces a new way for Sarcasm Detection and their Proposed Model Consists of 4 Layers which are: Tweets Collection, Part of Speech Tagging, Feature Extraction and Classifiers. First Tweets Collection and Feature Extraction, they used RESTful Twitter Developer API to collect tweets from twitter server and they used the hashtags sarcasm and sarcastic to collect about 1000000 Tweets. Second Part of Speech tagging, this stage they divide the input text into atomic words and assign the words with appropriate Part of speech tag. Third part is the Feature Extraction, They used interjection and intensifier to build the feature set. An intensifier is either adjective or adverb or combination of both. Words like wow is considered as interjections. Fourth part Classifiers, they used classifiers Naive bayes, Decision tree, Ada boost, SVM, Random forest. Results, the system attains an accuracy (%) of 75.12, 80.27, 80.67, 80.79, and 80.07 using NB, DT, SVM, RF, and AdaBoost respectively. They Collected Their Data sets from Twitter manually. Drawbacks, They Relied on Hashtags on collecting datasets which could mislead them while collecting. The hashtag could be sarcastic while not containing any sarcasm content.**

**[22] The Proposed Model is made to detect sarcasm. The proposed system consists of three main levels which are, Data Pre-**

**Processing, Data Preparation, Sarcasm Detection. First the Pre-Processing Stage, the Pre-Processing stage consists of 3 Main parts which are , Hashtag identification, Emoji Replacement, Slang Dictionary Mapping. Starting with Hashtag identification in this stage they are identifying whether the words after the hash symbol is important and handy or not like #Sarcasm And #entertainment the first one helps the system that there could be sarcasm in this part by the second one is not helping too much. Moving on to the Emoji Replacement in this part they search for the emoji in the text and they replace it with weight Whether it is positive Weight of negative one. Moving on to Slang Dictionary in this phase they are searching for the slang and match it with slang from their dictionary and then they try replace the slang in the document with the phrase found in the dictionary. Second the Data Preparation Stage, the Data preparation stage consists of 6 layers which are, Word Tokenization, Part of Speech, Stemming and lemmitization, Feature identification, New Representation Creation. Starting with the Word Tokenization Stage in this part they are tokenizing the phrase to words, moving on to the part of speech they are searching in the adjectives if they found a lot of adjectives in the text then there will be most probable a sarcasm in the document as a lot of adjectives means that they are used to describe something. Stemming and lemmitization is used to make the word go back to its original form. Feature identification consists of multiple layers which are Blob polarity , Blob subjectivity, Capitalization, Positive sentiment, Negative Sentiment. After The Feature Selection Layer is done a new Representation is given by using all the features and their respective weights. Third Sarcasm Detection stage, they used classifiers : Decision Tree , Random Forest , Gradient Boosting , Adaptive Boosting , Logistic Regression, Guassien Naive Bayes. They Got their Datasets Manually From Twitter and labeled it. Results, With emoji and Slang Dictionary: Random Forest: 79.44 , 80.93 , 77.49 , Naive Bayes: 74.56 , 73.9, 75.1. The main Drawback I have seen in this document is that they remove the stop words which could lead to wrong classification because taking a message like it is silly, when they remove the stop words the only word is going to remain is silly which is going to be classified as sarcasm.**

**Sarcasm is another problem that faces the users of social media often. The sarcasm is the usage of the words in a different way such as insulting someone with positive words. The Proposed Method is according to Edwin Lunando and Ayu Purwarianti in their paper [15] they proposed a model that consists of three main parts as follows, Pre-Processing, Feature Extraction, Classification. First we are going to talk about the PreProcessing Part, mainly in their model the pre-processing part consist of the word correction from informal word**

**to formal word or to correct misspelled word and to convert the numeric characters used in the text into alphabetic character. Second the Feature Extraction, they used the Uni-gram, Negativity, Number of interjection words and Question Words. They used the Uni-gram because they found that it is more suitable for the Indonesian language because the Indonesian language mainly consists of informal words, they used the negation to calculate the weight of the word as when the word like "not" is places in the text it is going to change the weight of the following text. Then they calculated the negativity of the words according to the public negativity of a certain topic. Then they used the number of interjection words, they find these words because according to their calculations they found from every 100 text that have interjection word there is 20 text that is classified as sarcasm. Last feature extraction method is the Question word, when they find that the text has question word so this sentence is neutral has no weight in the text. Third part the classification, their classification model consists of two parts the first part is the part that they classify the text in to three classes that it is positive or negative or neutral. Then the second part that they classify the positive text to find whether it is an opinion or neutral and if it is an opinion it is positive or negative. They Got their Datasets Manually from Twitter. Their Results As Follows, they used 3 models Naive Bayes: 76.5% Maximum entropy: 76.7% Support Vector Machine: 77.3%.**

**[21] This paper unlike previous approach that consider a fixed window of a cyber-predator's questions within a dialogue, we exploit the whole question set and model it as a signal, whose magnitude depends on the degree of bullying content, their pre-processing: Stop-words removal, tokenization, stemming and Part-Of-Speech tagging. Applying SVM achieved a 49.8% result on accuracy.**

**[13] None till now covered cyberbullying in Arabic language and this paper wants to tackle this issue. Features used: Unigrams, TF-IDF, Lexicon, Bigrams. Machine Learning: Naive Bayes Scored 90.8514%, Support Vector Machine (SVM) Scored 94.1%.**

**[8] This paper shows that taking user context into account improves the detection of cyberbullying. 4626 comments from 3858 distinct users. The comments were manually labelled as bullying (9.7%) and non-bullying (inter-annotator agreement 93%). They applied SVM classifier and were able to reach results of up to 78% on precision and 55% on recall.**

**[26] This paper uses novel pronunciation based convolutional neural network (PCNN) thereby alleviating the problem of noise and**

**bullying data sparsity to overcome class imbalance. 1313 messages from twitter, 13,000 messages from formspring.me. Accuracy of the twitter dataset wasn't calculated due to it being imbalanced. While Achieving 56% on precision, 78% recall and 96% accuracy.**

**[7] This paper applied 3 models for their dataset gathered YouTube comment section, Multi-Criteria Evaluation Systems (MCES), machine learning: (Naive Bayes classifier, decision tree, SVM), Hybrid approach. The MCES score 72% on accuracy, while Naive Bayes scored the highest out of the three with 66%.**

**[2] This paper put most of its effort in Natural Language Processing (NLP) and SVM, their dataset is collected from twitter. While reaching 83.1% in accuracy, and 91.1% in precision.**

**[10] This paper proposed to adopt an unsupervised approach to detect cyber bully traces over social networks. They used the classifiers inconsistently over their dataset, applying SVM on FormSpring and achieving 67% on recall, applying GHSOM on YouTube and achieving 60% precision, 69% accuracy and 94% recall, applying Naive Bayes on Twitter and achieving 67% accuracy.**

**[12] This paper showed promising results due to relying on heavy feature extraction information gain using tfidf, LIWC and unusual capitalization. Their data is collected from Ask.fm. All being passed to the SVM classifier and scoring 99.4% in accuracy, 69% in precision, 84.9% in recall.**

**[11] This paper their aim was to detect explicit bullying language pertaining to (1) sexuality, (2) race & culture and (3) intelligence, acquiring their dataset from the YouTube comment section. They applied SVM and reached 66% in accuracy and Naive Bayes reached 63% in accuracy.**

**[6] This paper used a supervised learning approach to detect cyberbullying. They constructed a Support Vector Machine classifier using WEKA. Their dataset was collected from Myspace. They achieved 43% on precision, 16% in recall and their accuracy wasn't mentioned.**

# 3 Comparative Table

| Authors | Approach | Datasets | Performance |
|---|---|---|---|
| Vikas S Chavan and Shylaja S S | ML | They Collected thier datasets from Kaggle Website | Logistic Regression Accuracy = 73.76 And Support Vector Machine Accuracy = 77.65 |
| to Edwin Lunando and Ayu Purwarianti | ML | They Collected datasets manually from Twitter | Naive Bayes Accuracy = 76.5 Maximum entropy Accuracy = 76.7 Support Vector Machine Accuracy = 77.3 |
| S.K. Bharti n And B. Vachha And R.K. Pradhan And K.S. Babu And S.K. Jena | ML | They Got their dataset from Twitter | 6 algorithms precision = 97 |
| Harsh Dani And Jundong Li(B) And Huan Liu | ML | They collected their Datasets from Twitter and MySpace | KNN F1 = 0.6105 AUC = 0.75 using SICD |
| Maral Dadvar And Dolf Trieschnig And Franciska de Jong | ML | Collected their datasets from Youtube | MCES discrimination capacity = 0.72 Naive Bayes with discrimination capacity Accuracy = 0.66 |
| MONDHER BOUAZIZI AND TOMOAKI OTSUKI | ML | Collected their datasets from Twitter | SVM Accuracy = 83.1 |
| Walisa Romsaiyud And Kodchakorn na Nakornphanom And ETAL | ML | They Collected datasets from Perverted Justice and Twitter | Naive Bayes Accuracy = 95.79 |
| Noviantho And Sani Muhamad Isa And Livia Ashianti | ML | They Collected their dataset from Kaggle | Naive Bayes Accuracy = 92.81 SVM Accuracy = 97.11 |
| Paras Dharwal And Tanupriya Choudhury And Rajat Mittal And Praveen Kumar | ML | They Collected their Dataset From Twitter | |

| | | | |
|---|---|---|---|
| Anukarsh G Prasad; Sanjana S Skanda M Bhat, B S Harish | ML | They got Twitter data set with manually labeling it. | Random Forest: 79.44 , 80.93 , 77.49 Naive Bayes: 74.56 , 73.9, 75.1 |
| Santosh Kumar Bharti Reddy Naidu Korra Sathya Babu | ML | Collected Sarcastic Tweets from twitter manually | The system attains an accuracy of 75.12, 80.27, 80.67, 80.79, and 80.07 using NB, DT, SVM, RF, and AdaBoost respectively |
| Noviantho, Sani Muhamad Isa, Livia Ashianti | ML | Fromspring Dataset from Kaggle | Naive Bayes yields an average accuracy of 92.81, SVM with a poly kernel yields an average accuracy of 97.11. |
| Nobata, Chikashi and Tetreault | ML | Data is sampled from comments posted on Yahoo! Finance and News | All Features:F-Score of news dataset: 0.817 F-Score of Finance dataset: 0.795 |
| Dadvar, Maral and De Jong, Franciska | ML | The data-set is provided by Fundacion Barcelona Media collected from MySpace it's size is 381,000 posts, the ground truth data-set is 2,200 labeled by three students. | basline precision was 0.31 and gender-specific precision was 0.43 |
| Nandhini, B Sri and Sheeba, JI | ML | it contains data from Myspace and Formspring.me. 500 post from Formspring.me and 600 posts from Myspace. | MySpace Acc. is 94.50 percent Formspring.me Acc.is 94.01 percent |
| Nandhini, B Sri and Sheeba, JI | ML | They are available from the workshop on Content Analysis for the Web 2.0, it contains data from Myspace and Formspring.me | MySpace Acc. is 87 percent Formspring.me Acc.is 86 percent |
| Murnion, Shane and Buchanan and William J | ML | They collected the data using World Of Tanks game api which reached 26,000 messages which 5,000 messages were manually classified to compare it to the automatic classifier | The simple naive automatic classification has reached 91.6 percent accuracy |

| | | | |
|---|---|---|---|
| Li, Homa Hosseinmardi Shaosong Yang, etal | ML | From instagram and Ask.fm | Accuracy N/A |
| Sarna, Geetika and Bhatia, MPS | ML | Twitter Dataset | The system attains an precision of 0.7955, 0.7627, 0.5696, 0.8667, using NB, Decision tree, SVM and KNN respectively |
| Zhang,Jonathan Tong etal | ML | Fromspring Dataset and twitter | CNN an average accuracy of 0.946, CNN Random with accuracy 0.966 an PCNN accuracy of 0.968. |
| ordelman, davdar | ML | youtube | SVM 0.77 on user based |
| Vinita Nahar, Sanad Al-Maskari,etal 11 | ML | The data-set is provided by Fundacion Barcelona Media collected from MySpace it's size is 381,000 posts, the ground truth data-set is 2,200 labeled by three students. | basline precision was 0.31 and gender-specific precision was 0.43 |
| Rui Zhao, Anna Zhou | ML | it contains data from Twitter | Precision Enhanced Bag of words = 76.8 Percent |
| Vaibhav Suri, Sourabh Parime | ML | Contains data from Myspace | Accuracy N/A |
| Ying Chen, Yilu Zhou et al | ML | YouTube comments | Precision of SVM = 77.9 Percent , Recall of SVM = 77.8 Percent |
| Aishwarya Upadhyay, Arunesh et al | ML | They created their own dataset from Perverted-justice.com | Accuracy N/A |
| I-Hsien, Kaohisung et al | ML | They collected datasets from four websites: Facebook, Twitter, Ptt (https://www.ptt.cc) and CK101 https://ck101.com/). | Precision = 79 Percent ,Recall = 71 Percent |

# 4 Proposed Model

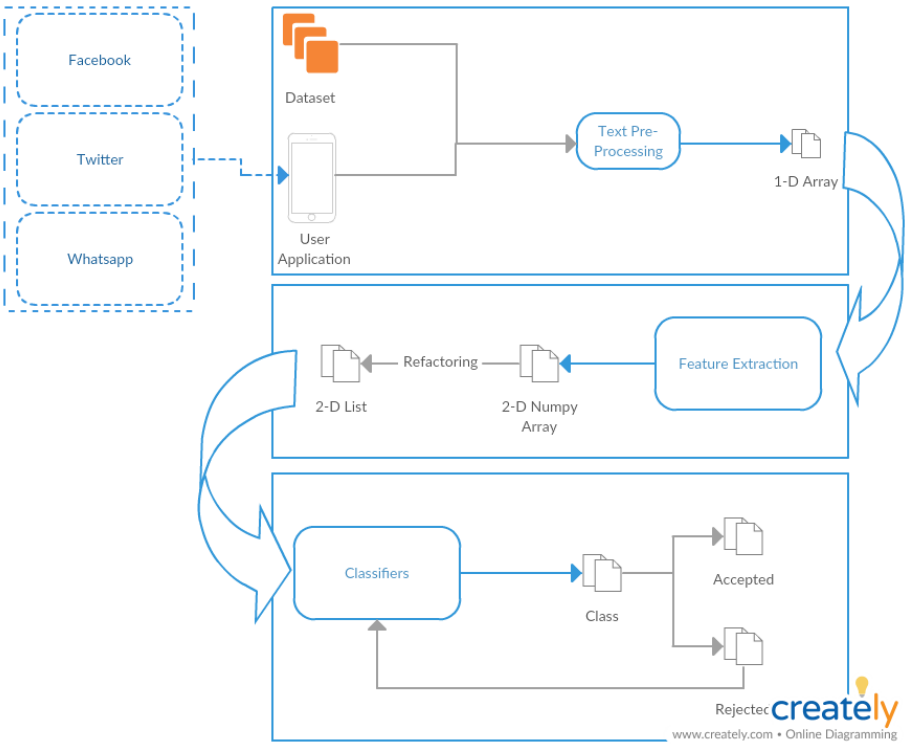


Figure 1: System Overview

**Our Proposed system consists of three main stages which are Pre-Pr Feature Extraction and Classification these three stages arocessing of words ,e ttext he maentersi in order nclassified as bullying text or not besides or proposed model has a self learning feature which enables the trained model to train on the error it has made in the classification based on the user rating and the approval of the system administrator**

## 4.1 Pre-Processing

**In this stage of the system we are making the text ready for the feature extraction stage by tokenizing the text,applying stemming and lemmitization , also we used Bing API for word correction, and we removed the encoding parts in the text.**

### 4.2 Feature Extraction

**In this stage we extract the Features from the text that we are going to push it in the model for the training and predicition for the feature extraction we are going to use three method which are TF-IDF , Linguistic Inquiry and finally the sentiment analysis. we are going to use the sentiment analysis to eliminate the false positive problem which faced many previous systems, the sentiment analysis depend on the context of the chat it get the polarity of the text overall not every sentence. and for the TF-IDF it gets the wights of the words in a certain text**

### 4.3 Classification

**This stage is the actual stage of the classification to either bullying or not and for this stage we are going to use three different models that we are going to train with different data. we are going to use logistic regression and Random Forrest and Naive Bayes. These three classifiers are going to be trained with different and for every classification process we are going to make the features enters in the three classifiers and make every classifier make its own prediction and then we are going to make a voting between the result and the voting isn't going to make all the classifiers has equal votes we are going to make the classifier with better accuracy has more voting points than the classifier that has less accuracy.**

### 4.4 Self Learning

**For this stage it is only stage to make the model self maintained which means that we receive rating from every user about every classification process and according to this rating the administrator asses this rating and either approve it so the classifier is going to train on the same data with the correct class of reject this rating and in this case the rating is dropped and never reach the classifier for training.**

## 5 Conclusion

**In conclusion we discussed in this paper the previous papers that tried to solve the online harassment problem, by going briefly through there proposed model and their results. Then we constructed a comparative paper to compare between different approaches in each paper. Finally, we presented our proposed model to solve the problem.**

# 6 References

# References


[1] S. K. Bharti, R. Naidu, and K. S. Babu, "Hyperbolic feature-based sarcasm detection in tweets: A machine learning approach," in *2017 14th IEEE India Council International Conference (INDICON)*. IEEE, 2017, pp. 1–6.

[2] M. Bouazizi and T. O. Ohtsuki, "A pattern-based approach for sarcasm detection on twitter," *IEEE Access*, vol. 4, pp. 5477–5488, 2016.

[3] V. S. Chavan and S. Shylaja, "Machine learning approach for detection of cyber-aggressive comments by peers on social media network," in *Advances in computing, communications and informatics (ICACCI), 2015 International Conference on*. IEEE, 2015, pp. 2354–2358.

[4] Y. Chen, Y. Zhou, S. Zhu, and H. Xu, "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)*. IEEE, 2012, pp. 71–80.

[5] M. Dadvar and F. De Jong, "Cyberbullying detection: a step toward a safer internet yard," in *Proceedings of the 21st International Conference on World Wide Web*. ACM, 2012, pp. 121–126.

[6] M. Dadvar, d. F. Jong, R. Ordelman, and D. Trieschnigg, "Improved cyberbullying detection using gender information," in *Proceedings of the Twelfth Dutch-Belgian Information Retrieval Workshop (DIR 2012)*. University of Ghent, 2012.

[7] M. Dadvar, D. Trieschnigg, and F. de Jong, "Experts and machines against bullies: A hybrid approach to detect cyberbullies," in *Canadian Conference on Artificial Intelligence*. Springer, 2014, pp. 275–281.

[8] M. Dadvar, D. Trieschnigg, R. Ordelman, and F. de Jong, "Improving cyberbullying detection with user context," in *European Conference on Information Retrieval*. Springer, 2013, pp. 693–696.

[9] P. Dharwal, T. Choudhury, R. Mittal, and P. Kumar, "Automatic sarcasm detection using feature selection," in *2017 3rd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT)*. IEEE, 2017, pp. 29–34.

[10] M. Di Capua, E. Di Nardo, and A. Petrosino, "Unsupervised cyber bullying detection in social networks," in *Pattern Recognition (ICPR), 2016 23rd International Conference on*. IEEE, 2016, pp. 432–437.

[11] K. Dinakar, B. Jones, C. Havasi, H. Lieberman, and R. Picard, "Common sense reasoning for detection, prevention, and mitigation of cyberbullying," *ACM Transactions on Interactive Intelligent Systems (TiiS)*, vol. 2, no. 3, p. 18, 2012.

[12] Y. J. Foong and M. Oussalah, "Cyberbullying system detection and analysis," in *2017 European Intelligence and Security Informatics Conference (EISIC)*. IEEE, 2017, pp. 40–46.

[13] B. Haidar, M. Chamoun, and A. Serhrouchni, "A multilingual system for cyberbullying detection: Arabic content detection using machine learning," *Advances in Science, Technology and Engineering Systems Journal*, vol. 2, no. 6, pp. 275–284, 2017.

[14] S. M. Isa, L. Ashianti *et al.*, "Cyberbullying classification using text mining," in *Informatics and Computational Sciences (ICICoS), 2017 1st International Conference on*. IEEE, 2017, pp. 241–246.

[15] E. Lunando and A. Purwarianti, "Indonesian social media sentiment analysis with sarcasm detection," in *Advanced Computer Science and Information Systems (ICACSIS), 2013 International Conference on*. IEEE, 2013, pp. 195–198.

[16] S. Murnion, W. J. Buchanan, A. Smales, and G. Russell, "Machine learning and semantic analysis of in-game chat for cyberbullying," *Computers & Security*, vol. 76, pp. 197–213, 2018.

[17] B. Nandhini and J. Sheeba, "Cyberbullying detection and classification using information retrieval algorithm," in *Proceedings of the 2015 International Conference on Advanced Research in Computer Science Engineering & Technology (ICARCSET 2015)*. ACM, 2015, p. 20.

[18] B. S. Nandhini and J. Sheeba, "Online social network bullying detection using intelligence techniques," *Procedia Computer Science*, vol. 45, pp. 485–492, 2015.

[19] C. Nobata, J. Tetreault, A. Thomas, Y. Mehdad, and Y. Chang, "Abusive language detection in online user content," in *Proceedings of the 25th international conference on world wide web*. International World Wide Web Conferences Steering Committee, 2016, pp. 145–153.

[20] S. Parime and V. Suri, "Cyberbullying detection and prevention: Data mining and psychological perspective," in *Circuit, Power and Computing Technologies (ICCPCT), 2014 International Conference on*. IEEE, 2014, pp. 1541–1547.

[21] N. Potha and M. Maragoudakis, "Cyberbullying detection using time series modeling," in *Data Mining Workshop (ICDMW), 2014 IEEE International Conference on*. IEEE, 2014, pp. 373–382.

[22] A. G. Prasad, S. Sanjana, S. M. Bhat, and B. Harish, "Sentiment analysis for sarcasm detection on streaming short text data," in *Knowledge Engineering and Applications (ICKEA), 2017 2nd International Conference on.* IEEE, 2017, pp. 1–5.

[23] W. Romsaiyud, K. na Nakornphanom, P. Prasertsilp, P. Nurarak, and P. Konglerd, "Automated cyberbullying detection using clustering appearance patterns," in *Knowledge and Smart Technology (KST), 2017 9th International Conference on.* IEEE, 2017, pp. 242–247.

[24] I.-H. Ting, W. S. Liou, D. Liberona, S.-L. Wang, and G. M. T. Bermudez, "Towards the detection of cyberbullying based on social network mining techniques," in *Behavioral, Economic, Socio-cultural Computing (BESC), 2017 International Conference on.* IEEE, 2017, pp. 1–2.

[25] A. Upadhyay, A. Chaudhari, S. Ghale, S. Pawar *et al.*, "Detection and prevention measures for cyberbullying and online grooming," in *Inventive Systems and Control (ICISC), 2017 International Conference on.* IEEE, 2017, pp. 1–4.

[26] X. Zhang, J. Tong, N. Vishwamitra, E. Whittaker, J. P. Mazer, R. Kowalski, H. Hu, F. Luo, J. Macbeth, and E. Dillon, "Cyberbullying detection with a pronunciation based convolutional neural network," in *2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA).* IEEE, 2016, pp. 740–745.

[27] R. Zhao, A. Zhou, and K. Mao, "Automatic detection of cyberbullying on social networks based on bullying features," in *Proceedings of the 17th international conference on distributed computing and networking.* ACM, 2016, p. 43.