Cyberbullying Detection System with Multiple Server Configurations

Rohit Pawar, Yash Agrawal, Akshay Joshi, Ranadheer Gorrepati, Rajeev R. Raje

Department of Computer and Information Science
Indiana University-Purdue University Indianapolis, Indianapolis, Indiana - 46202
{rspawar, agrawaly, aksjoshi, ranagorr, rraje}@iupui.edu

***Abstract*—Due to the proliferation of online networking, friendships and relationships - social communications have reached a whole new level. As a result of this scenario, there is an increasing evidence that social applications are frequently used for bullying. State-of-the-art studies in cyberbullying detection have mainly focused on the content of the conversations while largely ignoring the users involved in cyberbullying. To encounter this problem, we have designed a distributed cyberbullying detection system that will detect bullying messages and drop them before they are sent to the intended receiver. A prototype has been created using the principles of NLP, Machine Learning and Distributed Systems. Preliminary studies conducted with it, indicate a strong promise of our approach.**

I. Introduction

A. What is Cyber Bullying?

Cyberbullying is bullying that takes place via digital devices. It can occur via the Short Message Service (SMS), Text, and apps, or online in social media, forums, or gaming where people can view, participate in, or share content. Cyberbullying includes sending, posting, or sharing negative, harmful, false, or mean content about someone else. It can include sharing personal or private information about someone else, causing embarrassment or humiliation. There have been cyberbullying instances which have turned into unlawful or criminal behavior. Some of the most cited definitions of cyberbullying are:

- “An aggressive, intentional act carried out by a group or individual, using electronic forms of contact, repeatedly and over time against a victim who cannot easily defend him or herself.” [1]
- “Cyberbullying is when someone repeatedly makes fun of another person online or repeatedly picks on another person through e-mail or text messages or when someone posts something online about another person that they dont like.” [2]

The most common places where cyberbullying occurs are:

- Social Media forums such as Facebook®, Instagram®, Snapchat®, and Twitter®.
- SMS sent through devices.
- Instant Messages via devices, email provider services, apps, and social media messaging features.

B. Effects of Bullying

Bullying affects everyone involved: those who are bullied, those who bully, and those who witness bullying. Bullying is linked to many negative outcomes including impacts on mental health [3], substance use [4], and suicide [5]. Bullying is most common among kids. Kids who are bullied can experience negative physical, school and mental health issues. Kids who are bullied are more likely to experience:

- Depression and anxiety, increased feelings of sadness and loneliness, changes in sleep and eating patterns, and loss of interest in activities they used to enjoy. These issues may persist into adulthood [6].
- Health complaints - both physical and mental [7].
- Decreased academic achievement and standardized test scores and school participation [8].
- They are more likely to miss, skip, or drop out of school.
- All these issues might even lead to suicidal tendencies [5].

Hence, cyberbullying may result in social harm and needs to be curbed if not fully eliminated.

C. Countermeasures By Social Media

Social networks provide some degree of support for a safe web experience. Current market tools, that act as a safeguard, perform in the following fashion. These tools :

- remove the bullying content and disabling the account of anyone who bullies or attacks another, but only after the content is received by the intended receiver and is reported to the authorities.
- provide settings to block the person bothering the user.
- allows privacy settings to enable specific people to view the posts.

When observed closely, all these methods use filtering after the post or message is read by the user or has been posted on the user’s wall. There is a delay between when the message is posted and when it might be taken down by the authorities. In this time, many people may read the post or message, causing further harm to the intended receiver. This can have a lasting effect on the user, as mentioned before.

Hence, we need is a system which can detect cyberbulling before it ever reaches its target and can cause any sort of harm, physically or mentally, to the user - such a system is the focus of this paper.

D. Objectives of our System

Our specific objectives for this research are :

This is the author's manuscript of the article published in final edited form as:
Pawar, R., Agrawal, Y., Joshi, A., Gorrepati, R., & Raje, R. R. (2018). Cyberbullying Detection System with Multiple Server Configurations. 2018 IEEE International Conference on Electro/Information Technology (EIT), 0090–0095. https://doi.org/10.1109/EIT.2018.8500110

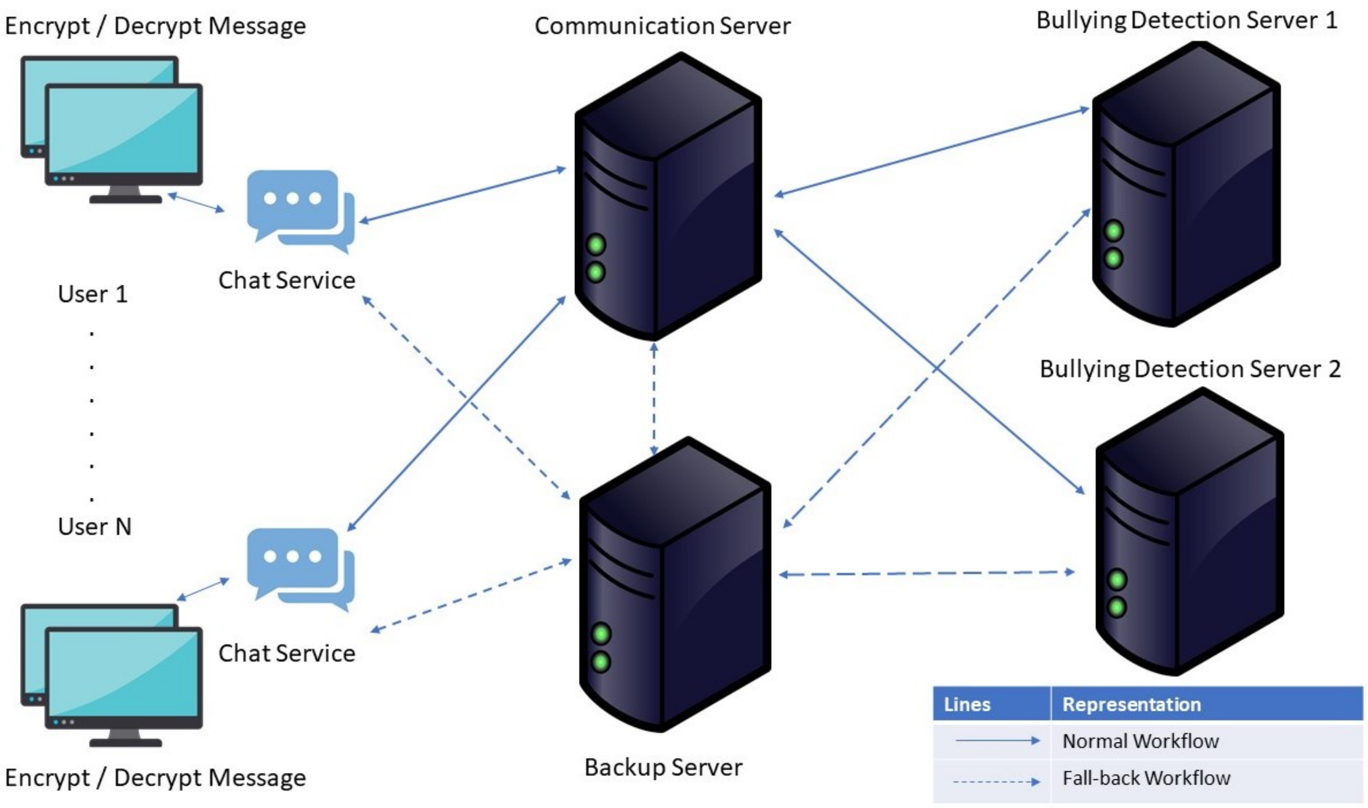

Fig. 1. System Design

- To implement an distributed cyberbulling detection system which uses machine learning algorithms to detect bullying messages.
- To examine various distributed techniques such as load balancing and their effects on the time and accuracy of detection of cyberbullying messages by empirical evaluations.

II. Related Work

In a recent study on cyberbullying detection, gender specific information [9] was used along with support vector machine model to train a gender specific text classifier. In other study, NUM and NORM features were devised by assigning a severity level to the badwords list [10]. NUM is a count and NORM is a normalization of the badwords respectively. The dataset consisted of 3,915 posted messages crawled from the Web Site, Formspring.me . They employed replication of positive examples upto ten times. Their findings showed that the C4.5 decision tree and an instance-based learner were able to identify the true positives with 78.5% accuracy. Zhao considered semantic-enhanced marginalized denoising auto-encoder (smSDA) a method developed by popular deep learning model to exploit the hidden feature structure of bullying information and learn a robust and discriminative representation of text [11]. Research has also been carried out based on text mining paradigms such as vandalism detection [12], spam detection [13] and detection of Internet abuse and cyber-terrorism [14]. Other interesting work is distributed-collaborative approach for cyberbullying detection [15]. The idea of this is was to design a scalable system using distribution along with machine learning and taking a second opinion in a “may-be” situation where one algorithm is not sure about the detection [16]. Our focus is to provide a scalable, fault tolerant, and secure system. Most of the bullying detection systems are focused on detection of cyber-bullying on offline dataset [17] but our proposed system works on real-time messages.

III. System Design

A. General Overview

We have designed a Cyber-Bullying Detection system that enhances a chat application using socket programming in Python, which allows multiple users to communicate with each other through the Communication Server. The system architecture is shown in Fig 1. System has two work-flows as below:

- Normal Work-flow: This work-flow is executed when there are no failures/errors in the system and its represented using solid lines in Fig 1. Whenever a user wants to send a message to another user then he uses a Chat Service to communicate to the Communication Server. The Communication Server forwards the received message to the Bullying Server and waits for response from the Bullying Server. Once the Communication Server receives a response from both the Bullying Server, it performs a logical OR of results and then decides whether to forward a message to the receiver or drop the message. The Communication Server forwards the message to the receiver only when if its non-bullying otherwise it drops the message.

- Fall-back Work-flow: This work-flow is executed when there are failures/errors in the system and is represented using dotted lines in Fig 1. If the Chat Service detects the failure of the Communication Server, then it re-connects to the Backup Server and continue its execution. The Backup Server retrieve all the users state information and history from the database. In case of Bullying Detection Server failure, the Communication Server/Backup Server takes care of bullying detection activity which enables a system to continue operating properly in the event of the failure.

  Communication between all the entities in the system is encrypted using AES algorithm to protect messages from an attackers as shown in Fig 1.

Our system, as shown in Fig 1, is divided into three components:

- Communication Server
- Bullying Detection Server
- Chat Service

We discuss each of them in detail below.

1) Communication Server: The Communication Server does the following things:

- Accepts multiple incoming connections from the user.
- Reads incoming messages from a particular user and delivers it to intended user or broadcasts them to all other connected users, in case of group communication.
- Forwards a message to the Bullying Server and takes decision about forward/drop message based on the Bullying Servers response.
- Takes over the bullying detection activity in case of a crash of the Bullying Server.

2) Bullying Detection Server: The Bullying Detection Server does the following things:

- Listens for incoming messages from the Communication Server.
- Runs the bullying detection algorithm and sends a response back to the Communication Server.

3) Chat Service: The chat service does the following things:

- Checks user input. If the user types in a message then sends it to the Communication Server.
- Listens for incoming messages from the Communication Server.

A user first sets up a connection with the Communication Server and authenticate using proper credentials. An user can send and receive messages with the help of the Communication Server and all the messages are encrypted/decrypted using a private key as shown in Fig 1.

B. Functionality Offered

We have implemented the following features in our system:

1) Unicast and Broadcast Messages Modes: In unicast message mode, any user can communicate with any other user using the Communication Server. One user can also broadcast the message to all other users using the Communication Server.

2) Fault Tolerance: Fault tolerance is the property that enables a system to continue operating properly in the event of the failure of (or one or more faults within) some of its components. We have implemented the following fault tolerance features in the system:

- If the Bullying Detection Server fails, the Communication Server takes over the functionality of the detection server and performs bullying detection.
- If one user is talking to the other user and the other user goes offline, an appropriate message is displayed to the sender.
- If the Communication Server fails, then Backup Server takes over the communication activity.

3) Modularity: We have divided servers functionality into two parts based on responsibility and to make system loosely coupled. The Communication Server takes care of client connections and handles incoming messages and the Bullying Server takes care of the bullying detection activity. This adds an extra level of redirection for every incoming messages but this division can easily allow multiple bullying detection algorithms on different servers to be utilized as needed. Such multiple algorithms will provide a collaborative environment for fetching and filtering inappropriate messages similar to described in [16].

4) Security: To protect messages from an attackers, we have implemented security at end points. Every user encrypts messages using the AES security algorithm and sends it to Communication Server. The Communication Server then forwards that message to the Bullying Server. The Bullying Server decrypts received message and checks the degree of bullying and sends the encrypted response to the Communication Server. The Communication Server decrypts the response from the Bullying Server and decides whether to forward or drop the message. The Communication Server forwards the encrypted message to intended user in case the message is non-bullying. Once a receiver receives the message, it decrypts the message and displays it on chat window. The key is shared among all the communication entities in an off-line manner to simplify design.

We have implemented authentication using simple username and password. Each user is required to login to the system using credentials before proceeding to the chat application.

5) Heterogeneity of Bullying Detection Algorithms: We have implemented two different bullying detection algorithms, i.e., Multinomial Naive Bayes (MNB) and Stochastic Gradient Descent (SGD) to detect cyber bullying. These algorithms were chosen since they will form a good base line for more complex experiments such as topic modeling,

TABLE I
Data and Label Instance

Data	Label
any makeup tips? i suck at doing my makeup lol ; Like tell me wht u wnna know?! Like wht do you use?!	0

that we would like to perform. Also, both algorithms scale very well [18][19] and an increase in the size of the data set will not be an issue. As indicated earlier, our system is designed in such a way that in future any additional algorithm can be implemented on a different server and it can be used for bullying detection. The module that incorporates these techniques acts as follows:

If a message sent is a cyber-bulling message, then the Bullying Detection Server returns 1 to the Communication Server. Based on the Bullying Servers response, the Communication Server drops the message and the message is never sent to the intended receiver. If the Bullying Detection Server returns 0 to the Communication Server, then the Communication Server just forwards the message to the receiver since a 0 signifies that the message is not a cyber-bulling message. For our system we have used Python's sci-kit learn [20].

IV. Process

A. Data

We are using the Formspring Dataset [21] for training our model. The original parameters of the dataset include the following fields:

- User Id
- Post
- Question
- Answer
- Asker
- Answer 1-3
- Severity 1-3
- Bullying 1-3

Following is an exmaple of an actual data instance provided in the data set:
"aprilpooh15 Q: any makeup tips? i suck at doing my makeup lol<br>A: Sure! Like tell me wht u wnna know?! Like wht do you use?! any makeup tips? i suck at doing my makeup lol Sure! Like tell me wht u wnna know?! Like wht do you use?! None No 0 n/a No 0 n/a"

This format is not suitable for training the model since there exist "n/a" values in the data instance and the instance labels for the data are embedded somewhere in the middle of the text. We need to clean up the data instances and convert them into a tabular format with the following fields:

- Data - the actual text conversation
- Label - 0 or 1 to classify the data as not bullying or bullying respectively

Table I shows the above mentioned example, after removing the unnecessary fields. We then perform a few preprocessing steps on the data. These steps are:

1) Case Conversion: Conversion of all the messages to lower case so that "How" and "how" are both counted as the same word and not separate words. This is done so that we do not have duplicated features, which is redundant.

2) Removal of Stop Words: We have used the predefined stop words provided by the nltk [22] package. There are a total of 183 stop words that the package provides, which we have removed from the dataset.

3) Removal of Punctuation Marks: Punctuation marks are of no significance to the model as well, since a question mark or an exclamation mark will not make the message a normal message or a cyber bullying one. Also, people are in the habit of using multiple punctuations in a chat message such as "........" or "????". This is irrelevant to the model and such repetitions are eliminated.

Another important aspect of text communication, is the use of smileys such as ":)" or ":(". By removing the punctuations, we are eliminating these as well.

After these preprocessing steps are performed, the message, *"Any makeup tips? i suck at doing my makeup lol ; Like tell me wht u wnna know?! Like wht do you use?!"* will transform into, *"makeup tips suck doing makeup lol like wht wnna use"*.

4) Synthetic Data Generation: Our overall data contains approximately 40,900 messages out of which only 3000 messages are cyber bullying messages. To tackle this class imbalance problem and improve the performance of the model, we artificially generated some new instances. For this, we performed the following steps:

- Found all 3000 cyber bullying messages after preprocessing and stored them in a list.
- Decide the additional number of data instances that we want to add to the dataset. We have chosen 20,000 so that atleast 1/3rd of the resulting dataset consists of bullying messages. The resulting dataset has approximately 60,900 messages which contain 23,000 cyber bullying messages.

B. Bag Of Words

Once all the pre-processing is complete, we now convert the string data, into Bag Of Words format. A bag of words representation, is a simple and basic frequency-based text representational format. This representation is used for our experiments. We have performed 10-fold Cross Validation for all our experiments. This basically means that each data point is in exactly 1 test set and in the other k-1, in this case 9 training sets. This is done so that, no matter how the data is divided, we always compute the average error across the folds to get a generalized score[23].

V. Results

We have carried out performance analyses and calculated average performance time of different operations with three

TABLE II
Performance Analysis in milliseconds

Operation	Single Server	Distributed Server	Load Balancer
Encryption	6.257	6.243	6.264
Decryption	0.9237	0.9123	0.9108
Communication	4.243	5.507	5.207
Bullying Detection	10.364	4.698	4.372
Total	21.7877	17.0343	17.0798

different setups as mentioned in Table II.

1) Single Server Configuration: In this configuration, a single server is taking care of communication as well as the bullying detection activity. This configuration does not provide the fault tolerance capability and hence, failure of the Communication Server will lead to entire system to fail. This acts the base case in our experiments.

2) Distributed Server Configuration: In this configuration, we divided the communication and bullying functionality. The bullying activity is handled by the Bullying Server. We have implemented two different Bullying Servers for two different algorithms. We have performed bullying detection in parallel for faster performance. For example, 2 different bullying detection algorithms on single server required 10.364 ms, but in case of distributed bullying the detection time is 4.372 ms. For better accuracy, we are performing the logical OR operation of results obtained from both the Bullying Servers.

3) Load Balancer Configuration: In this configuration also, we divided communication and bullying functionality. In distributed bullying detection, every incoming message is sent to both the Bullying Servers. While in the case of the load balancer approach, we assign the incoming messages in round robin fashion to balance the system workload.

We have carried out experiments using all 3 configurations and have summarized results for each setup in Table II. These results indicate that Load Balancer and Distributed Server Configurations are taking less time than Single Server Configuration. The reason for this behavior is because, the Single Server Configuration has to execute N bullying detection algorithms on same machine whereas the other configurations perform the execution of algorithms on seperate machines. The worst case time complexity to run N bullying detection algorithm on different server is maximum time (t) taken by any one of the algorithm and communication delay (c). Our experiment, for bullying detection operation in Table II, shows the total time required to run N algorithms in parallel on different machines is (t+c) whereas a single machine takes (N*t) to determine bullying.

We have also carried out performance analysis by increasing number of users as shown in Fig 2. In Fig 2, the X-axis represents the number of users and the Y-axis represents time in milliseconds. As seen from Fig 2, the end-to-end response time almost linearly increases as the number of users. Some points in Fig 2 are deviated little bit from the linear increment because the number of messages sent by the each users at a particular time is not constant.

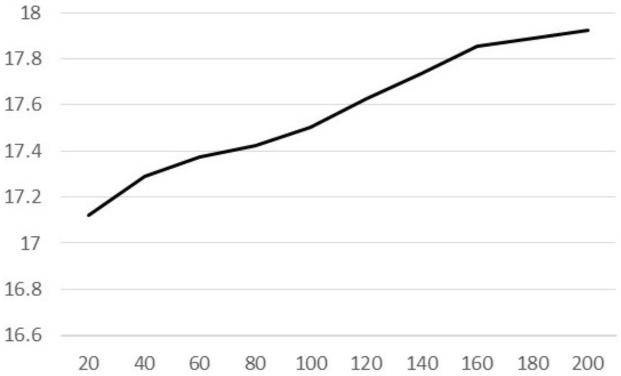

Fig. 2. Scalability Testing for Load Balancer Configuration

TABLE III
Comparison of F1 scores of the different models

Model	Message Class	Without Synthetic Data	With Synthetic Data
MNB	Normal	0.93	0.94
	Bullying	0.31	0.90
SGD	Normal	0.85	0.95
	Bullying	0.30	0.90

4) Machine Learning Models: As indicated earlier, for the machine learning models, we have implemented the following techniques [20]:

- Multinomial Naive Bayes (MNB)
- Stochastic Gradient Descent (SGD)

The results of the experiments are shown in Table III. We have used F-Scores as a performance measure since it gives an unbiased class-wise result which is important in our system. As we can see from the results, that class imbalance is a major challenge. It significantly drops the performance of the machine learning model which is not a desirable outcome. Hence, when synthetic data is included, it created a generalized corpus of data instances, making the model more robust and achieving a higher F score.

Considering the approach we have taken, and the techniques implemented, F scores provides a good base line, which can now be used for more advanced techniques for future experiments such as Topic Modeling. This data has not been used for training by others, so we cannot compare our results with other results.

VI. Conclusion and Future Work

Our proposed model not only focuses on accuracy but also on the performance. Hence, it can be integrated in popular social media systems such as Facebook and Twitter to prevent cyberbullying. Distributed cyberbullying detection system can be used to detect bullying in real-time without performance bottleneck which will help prevent cybebullying and its effect.

We plan to implement the following changes and additions in future:

- Get more data and perform more complex experiments on the data which include Non-Negative Matrix Factorization and Latent Dirichlet Allocation.
- Combine multiple classifiers to conduct polling in "could-be" scenarios in machine learning model to improve accuracy.
- Perform a large scalability study.
- Provide some APIs or services to third party users which can help our system to be integrated with other present and future approaches.

References

[1] P. K. Smith, J. Mahdavi, M. Carvalho, S. Fisher, S. Russell, and N. Tippett, "Cyberbullying: Its nature and impact in secondary school pupils," *Journal of child psychology and psychiatry*, vol. 49, no. 4, pp. 376–385, 2008.

[2] S. Hinduja and J. W. Patchin, "Cyberbullying: Neither an epidemic nor a rarity," *European Journal of Developmental Psychology*, vol. 9, no. 5, pp. 539–543, 2012.

[3] K. Kumpulainen, E. Räsänen, and K. Puura, "Psychiatric disorders and the use of mental health services among children involved in bullying," *Aggressive behavior*, vol. 27, no. 2, pp. 102–110, 2001.

[4] A.-H. Luukkonen, K. Riala, H. Hakko, and P. Räsänen, "Bullying behaviour and substance abuse among underage psychiatric inpatient adolescents," *European psychiatry*, vol. 25, no. 7, pp. 382–389, 2010.

[5] A. B. Klomek, A. Sourander, and M. Gould, "The association of suicide and bullying in childhood to young adulthood: a review of cross-sectional and longitudinal research findings," *The Canadian Journal of Psychiatry*, vol. 55, no. 5, pp. 282–288, 2010.

[6] W. M. Craig, "The relationship among bullying, victimization, depression, anxiety, and aggression in elementary school children," *Personality and individual differences*, vol. 24, no. 1, pp. 123–130, 1998.

[7] K. Rigby, "Health consequences of bullying and its prevention," *Peer harassment in school: The plight of the vulnerable and victimized*, vol. 310, 2001.

[8] J. Juvonen, Y. Wang, and G. Espinoza, "Bullying experiences and compromised academic performance across middle school grades," *The Journal of Early Adolescence*, vol. 31, no. 1, pp. 152–173, 2011.

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

[10] K. Reynolds, A. Kontostathis, and L. Edwards, "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.

[11] R. Zhao and K. Mao, "Cyberbullying detection based on semantic-enhanced marginalized denoising auto-encoder," *IEEE Transactions on Affective Computing*, vol. 8, no. 3, pp. 328–339, 2017.

[12] P.-N. Tan, F. Chen, and A. Jain, "Information assurance: Detection of web spam attacks in social media," in *Proceedings of Army Science Conference, Orland, Florida*, vol. 20, 2010.

[13] K. Smets, B. Goethals, and B. Verdonk, "Automatic vandalism detection in wikipedia: Towards a machine learning approach," in *AAAI workshop on Wikipedia and artificial intelligence: An Evolving Synergy*, pp. 43–48, 2008.

[14] D. A. Simanjuntak, H. P. Ipung, A. S. Nugroho, *et al.*, "Text classification techniques used to faciliate cyber terrorism investigation," in *Advances in Computing, Control and Telecommunication Technologies (ACT), 2010 Second International Conference on*, pp. 198–200, IEEE, 2010.

[15] A. Mangaonkar, A. Hayrapetian, and R. Raje, "Collaborative detection of cyberbullying behavior in twitter data," in *Electro/Information Technology (EIT), 2015 IEEE International Conference on*, pp. 611–616, IEEE, 2015.

[16] A. Mangaonkar, "Collaborative detection of cyberbullying behavior in twitter data," in *A Thesis Submitted to the Faculty of Purdue University in Partial Fulfillment of the Requirements for the degree of Master of Science Department*, 2017.

[17] S. Salawu, Y. He, and J. Lumsden, "Approaches to automated detection of cyberbullying: A survey," *IEEE Transactions on Affective Computing*, vol. PP, no. 99, pp. 1–1, 2017.

[18] J. Su, J. S. Shirab, and S. Matwin, "Large scale text classification using semi-supervised multinomial naive bayes," in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pp. 97–104, Citeseer, 2011.

[19] B. Recht, C. Re, S. Wright, and F. Niu, "Hogwild: A lock-free approach to parallelizing stochastic gradient descent," in *Advances in neural information processing systems*, pp. 693–701, 2011.

[20] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[21] "Retrieved from https://www.kaggle.com/swetaagrawal/formspring-data-for-cyberbullying-detection/data."

[22] S. Bird and E. Loper, "Nltk: the natural language toolkit," in *Proceedings of the ACL 2004 on Interactive poster and demonstration sessions*, p. 31, Association for Computational Linguistics, 2004.

[23] R. Kohavi *et al.*, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *Ijcai*, vol. 14, pp. 1137–1145, Montreal, Canada, 1995.