**Cyberbullying Detection Using Support Vector Machine (SVM) Algorithm**

A Thesis

Presented to

the School of Computing and Information Technology Faculty

Asia Pacific College

In Partial Fulfillment

of the Requirements for the Degree of

Bachelor of Science in Computer Science Major in Systems Software

By

Ballesteros, Faith I.

Capuz, Paulo Miguel S.

Juarez, Jason E.

Mallari, Samantha G.

Samillano, Eva R.

Ms. Lorena Rabago

Subject Professor

March 30, 2017

**Table of Contents**

[**1. Introduction 4**](#_Toc478536402)

[**1.1 Background of the Problem 4**](#_Toc478536403)

[**1.2 Statement of the Problem 5**](#_Toc478536404)

[**1.3 Objectives 6**](#_Toc478536405)

[**1.3.1 Main Objective 6**](#_Toc478536406)

[**1.3.2 Specific Objectives 6**](#_Toc478536407)

[**1.4 Scope and Limitations 6**](#_Toc478536408)

[**2. Review of Related Literature 10**](#_Toc478536409)

[**2.1 Cyberbullying Literatures 10**](#_Toc478536410)

[**2.1.1Social Media as its Channel and its Implications on Cyberbullying 10**](#_Toc478536411)

[**2.2.2 Offline Consequences of Online Victimization: School Violence and Delinquency 10**](#_Toc478536412)

[**2.2 Text Classification 11**](#_Toc478536413)

[**2.2.1 Comparative Assessment of the Performance of Three WEKA Text Classifiers Applied to Arabic Text 12**](#_Toc478536414)

[**2.2.2 Machine Learning Methods for Spam Email Classification 13**](#_Toc478536415)

[**2.2.3 Predicting Age and Gender in Online Social Networks 14**](#_Toc478536416)

[**2.2.4 Classifying Typhoon Related Tweets 16**](#_Toc478536417)

[**2.3 Cyberbullying Detection 17**](#_Toc478536418)

[**2.3.1 Locate the Hate: Detecting Tweets against Blacks 17**](#_Toc478536419)

[**2.3.2 Comparison of Machine Learning Algorithms for Offensive Language Filtering 17**](#_Toc478536420)

[**2.3.3 Modeling the Detection of Textual Cyberbullying 19**](#_Toc478536421)

[**2.3.4 Automatic Detection and Prevention of Cyberbullying 20**](#_Toc478536422)

[**2.3.5 Improved Cyberbullying Detection using Gender Information 22**](#_Toc478536423)

[**Dadvar, Jong, Ordeiman, and Trieschnigg (2012) believed that the incorporation of gender specific language features will improve the accuracy of a cyberbullying detection system. To test this idea, they conducted an experiment on improving cyberbullying detection with the aid of gender specific features. 22**](#_Toc478536424)

[**2.3.6 Automated Role Detection in Cyberbullying Incidents 22**](#_Toc478536425)

[**3. Theoretical Framework 25**](#_Toc478536426)

[**3.1 Audience Segregation by Ervin Goffman 25**](#_Toc478536427)

[**3.2 Text Classification 27**](#_Toc478536428)

[**3.2.1 Support Vector Machine 28**](#_Toc478536429)

[**3.3 Bag of Words 29**](#_Toc478536430)

[**3.4 Performance Measures 30**](#_Toc478536431)

[**3.4.1 Precision 31**](#_Toc478536432)

[**3.4.2 Recall 31**](#_Toc478536433)

[**3.4.3 Accuracy 32**](#_Toc478536434)

[**3.4.4 F Measure 32**](#_Toc478536435)

[**3.4.5 Kappa Statistics 32**](#_Toc478536436)

[**4. System Design 34**](#_Toc478536437)

[**4.1 System Overview 34**](#_Toc478536438)

[**4.2 System Objectives 34**](#_Toc478536439)

[**4.2.1 Main Objective 34**](#_Toc478536440)

[**4.2.2 Specific Objectives 34**](#_Toc478536441)

[**4.3 Scope and Limitation 34**](#_Toc478536442)

[**4.4 System Architecture 35**](#_Toc478536443)

[**4.4.1 Experimentation Phase 35**](#_Toc478536444)

[**4.4.1.1 Data Corpus 35**](#_Toc478536445)

[**4.4.1.2 Cleaning of the dataset 36**](#_Toc478536446)

[**4.4.2 Training Phase 38**](#_Toc478536447)

[**4.4.2.1 Data Collection 38**](#_Toc478536448)

[**4.4.2.2. Data Cleaning 38**](#_Toc478536449)

[**4.4.2.3 Bag of Words 38**](#_Toc478536450)

[**4.4.2.3.1 Tokenization 38**](#_Toc478536451)

[**4.4.2.3.2 Term-Frequency Value 39**](#_Toc478536452)

[**4.4.2.3.3 WEKA 39**](#_Toc478536453)

[**4.4.2.3.4 Support Vector Machine 39**](#_Toc478536454)

[**4.4.3 Reporting of Cyberbullying Incident** 39](#_Toc478536455)

[**4.4.3.2.1 Instantiating a report** 39](#_Toc478536456)

[**4.4.3.2.2 Setting the report attributes** 40](#_Toc478536457)

[**4.4.3.2.3 Setting the message content** 40](#_Toc478536458)

[**4.4.3.2.4 Sending the report** 40](#_Toc478536459)

[**5. Results and Discussion** 40](#_Toc478536460)

[**6. Conclusion and Recommendations** 42](#_Toc478536461)

# **Introduction**

# **1.1 Background of the Problem**

Bullying came into existence even before men evolved into intellectual beings. [1] Boehm stated that primates execute bullying-like deportment against their own kind to establish dominion over them. Bullying was further redefined from dominance to a mere destructive act, rendered by their ability to communicate through language. Recent technological advancements have extended the way people communicate. With the proliferation of the Internet, people can easily connect with one another simultaneously through chat rooms, email, instant messaging, forums, and social networking sites.[2] However, alongside with the modern advancements in communication, an old pervasive issue arises with a new form in new environment – cyberbullying. [3] Cyberbullying is defined as a form of harassment that occurs via the Internet which includes vicious forum posts, name calling in chat rooms, creating fake profiles on social networking sites, and sending cruel messages. [4]

Philippines is also known as the “social media capital of the world”. [5] A recent study conducted by We Are Social states that Filipinos spent an average of 4 hours and 17 minutes per day on social networking sites. [6] However, as the number of Filipino social media users continuously increases, it consequently intensifies the problem of cyberbullying. [7] A survey administered by Stairway Foundation Inc. revealed that 80% of Filipinos have been cyberbullied through social media. [8] Popular cyberbullying incidents in the Philippines are Paula Jaime Salvosa’s, a.k.a. “Amalayer” incident [9], Raymond Malinay’s prank involvement [10] and DJ Karen Bordador’s cyberbullying experience, following her arrest with her boyfriend in a drug-related buy bust operation. [11] Cyberbullying has become prevalent in academic institutions as well. [12] To illustrate, Asia Pacific College (APC), an educational institution specializing in I.T. related courses, has had its own share of cyberbullying incidents, with the most recent case which involves an edited photo of a student posted on Facebook which aims to mock his physical appearance. These cases only represent a portion of cyberbullying instances that have been formally reported.

There are still numerous unreported cases of online bullying that led to the introduction of Republic Act 10627 (Anti Bullying Act of 2013), which requires all elementary and secondary school to adopt policies that will prevent and address cyberbullying in their educational institutions. [13] In 2015, House Bill 5718 was also proposed to provide consequences for cyberbullying act wherein perpetrators shall face a penalty six years. [14] Social networking sites have adopted various strategies to protect their users by preventing and intervening in cyberbullying situations. To illustrate, most social media sites offer privacy settings which allows users to limit the amount of information that can be viewed publicly. They also have a reporting tool page wherein user can report instances of online bullying to social media administrators. YouTube offers Safety Mode, an opt-in setting where users can filter search results. Facebook has moderation and profanity blocklist that can be used to filter a set of harmful words on a page. Twitter offers Mute Feature that allows a user to remove a person’s tweets from his timeline without them knowing. Despite these efforts, their methods seemed to be inefficient because it is impossible to monitor all activities in the cyberspace given the vast amount of information available online. In addition to this, their methods rely heavily on the users to submit a report before taking an action. However, since Philippines remains to be on a conservative level, Filipinos are reluctant to admit that they have been cyberbullied. [8] To address these limitations, several studies were conducted to facilitate the process of monitoring the vast amount of online information and to trace cyberbullying automatically and accurately. [15] These studies utilized various statistical machine learning approaches such as Naïve Bayes [16] [17] and Support Vector Machine [18] in automating the detection of cyberbullying occurrences.Current studies on automatic cyberbullying detection are focusing mainly on optimizing the accuracy of detection. [20] However, these studies were limited to the detection of online bullying instances and not on follow up procedures once a cyberbullying event has been identified which should be given an utmost importance. [15]

The current research extends the technological feasibility of automating the process of cyberbullying detection in social media posts by integrating an automatic report generating tool alongside with the detection model to combat online bullying. Thus, once a cyberbullying instance has been flagged, the system will automatically send a report to the administrator via email which includes the content of the harmful post. With this, the moderators can easily monitor each and every activity occurring in the Web.

## **1.2 Statement of the Problem**

How can the current process of identifying and reporting cyberbullying occurrences online be improved?

## **1.3 Objectives**

### **1.3.1 Main Objective**

This project aims to improve the current, manual processes of identifying and reporting cyberbullying occurrences online by developing a cyberbullying detection system

### **1.3.2 Specific Objectives**

* To research concepts regarding the development of the cyberbullying detection system
* To gather textual data for the corpus
* To perform text preprocessing methods on the dataset
* To perform text annotation on the dataset
* To implement machine learning algorithm using WEKA
* To generate a cyberbullying detection model
* To develop a cyberbullying detection system
* To test the accuracy of the system in terms of identifying and reporting cyberbullying statements
* To document all processes performed

## **1.4 Scope and Limitations**

The manual processes referred to in the main objective comprised of the enabling of privacy settings, accessibility of a reporting tool page, introduction of “Safety Mode” (YouTube) and “Mute” (Twitter), and the creation of a moderation and profanity blocklist (Facebook), for these are the procedures being done by renowned social networking sites in the Philippines to combat cyberbullying and will therefore require certain improvements based on the teams’ researches.

The following are the concepts that were researched on: text classification, Support Vector Machine (SVM) algorithm, and requirements needed for the system.

The corpus (dataset) currently consists of 2000 statements which were obtained from either public Facebook and Twitter posts or Youtube comments. The totality of these statements pertained to the major controversial issues in the Philippines (e.g. those involving the LGBT community, drugs, scandals and other major issues of famous people - celebrities, political entities, and the like).

Text preprocessing methods that were done on the dataset include cleaning, tokenization – the process of breaking down a statement into smaller pieces, and conversion of the dataset in Bag-of-Words form. The cleaning of the dataset involved the removal of all special characters (excluding apostrophes and hyphens), non-readable text (e.g. asdfghjkl), emoticons, links, and foreign language characters. The conversion of the statements within the corpus to create a unigram Bag-of-Words referred to the replacement of string values with numerical values that can be understood by WEKA.

Three schemes were used for text annotation namely cyberbullying, not cyberbullying, and ambiguous cyberbullying. Among the 2000 statements used, 1000 were annotated by the researchers while the other 1000 were distributed among APC students as well.

The machine learning algorithm that was utilized is the Support Vector Machine algorithm. The decision to do so was greatly influenced by the related literatures the proponents of this project have included in this document.

The concept of text classification aided in the development of a cyberbullying detection model. Cyberbullying occurrences in public social media posts expressed using the Filipino language will be detected, based on the context as to how they are typically comprehended with and/or stated by Filipinos residing within Metro Manila. This will be outputted by WEKA toolkit.

The program for the system, which allowed the automated identification and reporting of cyberbullying occurrences online to take place, was hard-coded using the Java programming language.

900 statements were utilized in WEKA in order to form the cyberbullying detection model. 10-fold Cross Validation was used for determining the accuracy, precision, recall, F-measure, and Kappa statistics of the constructed cyberbullying detection model. Overall, it yielded an accuracy rate of 57.89%. It was initially experimented on the corpus data before integrating it with the system. The basis of determining the model’s accuracy will now depend on the following: as long as it was able to classify cyberbullying from non-cyberbullying statements and output the following results accordingly (in real-time, limited to the administrators, and can be sent as part of a user-generated email), the system will be deemed successful.

The whole project was documented according to the official format used by Asia Pacific College.

**1.5 Significance**

The main significance of this research project is aimed towards the improvement of identifying and reporting cyberbullying occurrences most especially in Metro Manila, Philippines. As mentioned earlier, the model was designed according to cyberbullying in the Philippine setup, indicating that it will only be able to classify statements expressed in the Filipino language. Doing so would greatly benefit the majority of the people expressing themselves using this particular language, which a great number of Metro Manillans do. They will be able to entitle themselves to a more efficient way of dealing with cyberbullying which would then guarantee them a fun and safe experience in social media.

The findings of this study will redound to the benefit of researchers who want to explore the field of both Cyberbullying and Natural Language Processing (NLP) – a field combining the areas of computer science, artificial intelligence, and computational linguistics to comprehend human languages. [19] The study provides detail on the processes of text classification and Linear Support Vector Machine Algorithm. As for the researchers who would want to explore the field of cyberbullying, this study can further enhance their knowledge on what cyberbullying is, the classification of cyberbullying and non-cyberbullying events, and the different categories of cyberbullying, based on sensitive issues in the Philippines.

Cyberbullying advocates, specifically those willing to help Filipinos, will be assisted by the system in the fulfillment of their advocacies as it will create a significant leap in terms of resolving such incidents through real-time identification and reporting processes to be conducted on each statement.

According to the Students’ Handbook, a book which contains all the policies, rules, and regulations being implemented in Asia Pacific College, it was the students of APC with whom the cyberbullying provision was intended for. However, the head of the Office of the Student Affairs himself noted the ineffectivity of the current identifying and reporting procedures being adapted in the school, as fewer reports where turned-in by the students despite annually receiving confessions during student interviews. This project can then benefit the students in terms of promoting a higher level of cyberbullying monitoring that can assure their safety within the cyberspace.

Manually moderating a number of social media sites appear to be a tedious job. In addition to that, the population of APC students come in great numbers as well. To address this problem, the researchers opted to integrate the cyberbullying detection model with a software development in order to automate the moderation process, which would, in turn, greatly lessen the difficulty of monitoring such amounts of data.

# **2. Review of Related Literature**

## **2.1 Cyberbullying Literatures**

Several studies in the social sciences has been devoted to understanding the nature of cyberbullying and the extent of its prevalence among children and young adults. [20] This section focuses on the findings of the studies conducted with regards to cyberbullying.

### **2.1.1Social Media as its Channel and its Implications on Cyberbullying**

Gonzales (2014) conducted a qualitative study to explore the relationship between social media and cyberbullying. [7] Through the use of focus interview analysis, he was able to gather information from eight experts from various field of specialization. From his study, he came up with the following conclusions:

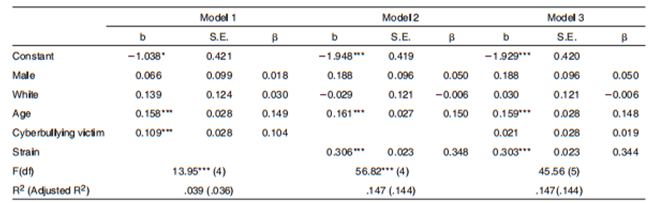
* Social media is the root cause of cyberbullying.
* There is no specific law in the Philippines that clearly defines punishable acts for cyberbullying.
* Self-discipline must be imposed by all social media users.
* Cyberbullying can be avoided, if people have a better understanding of social media.
* Social media users should be wary of sharing personal information in the cyberspace.
* The victim should report to the authority once the bully pose a serious threat to his life or liberty.

### **2.2.2 Offline Consequences of Online Victimization: School Violence and Delinquency**

Hinduja and Patchin (2012) conducted a study to determine the relationship between victimization, strain, and deviant behavioral choices of the cyberbullying victims. [21] Moreover, they used the general strain theory (GST) to identify both the emotional and behavioral effects of cyberbullying.

The proponents conducted an online survey methodology to obtain data from 1,388 adolescents. They used two primary independent measures (cyberbullying victimization and strain), a dependent variable (offline problem behaviors) and three demographic control variables such as age, race, and gender. Cyberbullying victimization is a scale that is composed of eight types of online victimization ranging from relatively minor forms of bullying to a more serious forms of harassment. The strain scale, on the other hand, refers to the common coping mechanism of a victim and is composed of nine items. The dependent variable is composed of an eleven-item index which represents the respondent’s behavior for the past six months. It ranges from a minor form of deviance to a more serious forms of delinquency.

For their experiment, a series of stepwise ordinary least squares (OLS) were estimated to explore the relationship between cyberbullying victimization, strain, and offline problem behaviors. In total, three models were created. The first model shows the relationship between cyberbullying victimization and offline problem behaviors, the second model illustrates the relationship between strain and offline problem behaviors, and the third model illustrates the relationship between cyberbullying victimization and strain and offline problem behaviors. As shown in Table 2.0, the first model proves that cyberbullying victimization is significantly related to offline problem behaviors which means youth who experience cyberbullying are more likely to participate in problem behaviors offline. The second model shows that strain is positively related to offline problem behaviors. Thus, youth who experience more strain are more likely to engage in offline problem behaviors. The third model illustrates that strain has a significant relationship with delinquency. The result of the third model demonstrates that strain serves as a mediator for the relationship between cyberbullying victimization and offline problem behaviors mainly because strain can be attributed on the effect of cyberbullying victimization on offline problem behaviors.



*Table 2.0: Ordinary Least Squares Regression - Delinquency Regressed on Strain and Cyberbullying Victimization*

## **2.2 Text Classification**

Recently, various machine learning approaches for automated text classification has witnessed a surge in terms of application. [22] This section presents the different applications of text classification including the methods that were employed by the researchers. It also presents the comparison of each approach when applied to different classification problems.

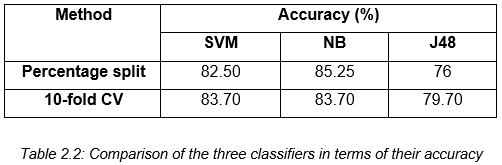
### **2.2.1 Comparative Assessment of the Performance of Three WEKA Text Classifiers Applied to Arabic Text**

Wahbeh and Al-Khabi (2012) conducted an experiment to illustrate the performance of three different text classification techniques: SVM, Naïve Bayes, and C4.5 in classifying Arabic text documents. [23]

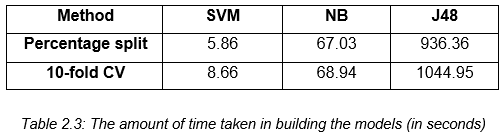
The first phase of their experiment begins with the creation of the corpus by gathering Arabic text documents from different websites: Kooora, news-all, and from Saheeh Al-Bukhari book and other websites. These data were already classified into number of categories such as Sport, Economic, Religion, Politics, and Mohammed sayings. They gathered a total number of 1000 documents (250 documents for each category) for their corpus.

As for the preprocessing step, any occurrences of digits and punctuation marks were removed.  Next, the set of characters were normalized into a canonical form. Third, non-Arabic text, special characters, and stop words were also removed. The last step involved in pre-processing includes the tokenization of the documents. All of the preprocessing steps were done using a tool created in C#. These documents were converted into ARFF format by utilizing WEKA TextDirectoryToArrf converter and StringToWordVector.

For their preliminary experiment, they utilized the percentage split which involves the process of dividing the data into two partitions: 60% was used for training phase while the remaining 40% was used for testing phase. Furthermore, they used 10-fold cross-validation technique for both dataset. These experiments were done to know if there will be improvements in the accuracy when the 10-fold cross-validation method is applied instead of the percentage split alone. Table 2.2 shows the comparison of the performance of three classifiers with respect to the percentage split method and 10-fold cross-validation. As shown in Table 2.2, the 10-fold cross-validation has significantly improved the accuracy for each classifier.

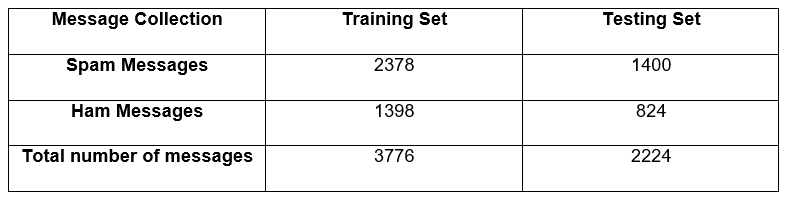
****

Aside from the accuracy of each classifier, they also measured the time taken for constructing each model. As shown in table 2.3, SVM requires the shortest amount of time to build the model. It was followed by the NB classifier. Lastly, J48 requires the largest amount of time in building the model.



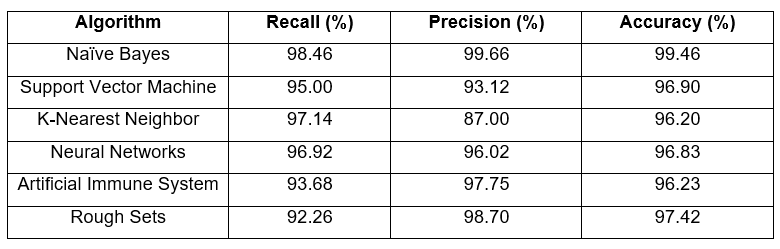
### **2.2.2 Machine Learning Methods for Spam Email Classification**

Awad and ELseuofi (2011) compared the performance of different machine learning algorithms in classifying spam emails. [24] Their experiment begins with the construction of a corpus by compiling both spam and legitimate emails from SpamAssassin, a collection of publicly available emails. This collection contains a total number of 6000 emails. Their dataset was divided the corpus into two sets: training and testing.

  
*Table 2.4: Corpora of Spam and Ham Messages*

Each email was further divided into three different parts: subject (the title of the email), from (the name of the sender) and body (the main part of the message). The preprocessing steps involve the removal of common words and case-change, wherein each word in the body is converted into small letters. Each message was converted to a feature vector which results into 21,700 attributes.

They selected a number of 100 features. These features were the most frequent words in spam mails. In addition to this, every email in the training dataset was denoted as a feature vector. Once the preprocessing steps were done, they applied different machine learning algorithms: Naïve Bayes, K-Nearest Neighbor, Artificial Neural Networks, Support Vector Machine, Artificial Immune System, and Rough Sets. To evaluate the performance of each classifier, they used precision, recall, and accuracy. As shown in Table 2.5, Naïve Bayes outperformed the other classifiers in terms of precision, recall, and accuracy.

  
*Table 2.5: The performance of different machine learning algorithms in spam email classification*

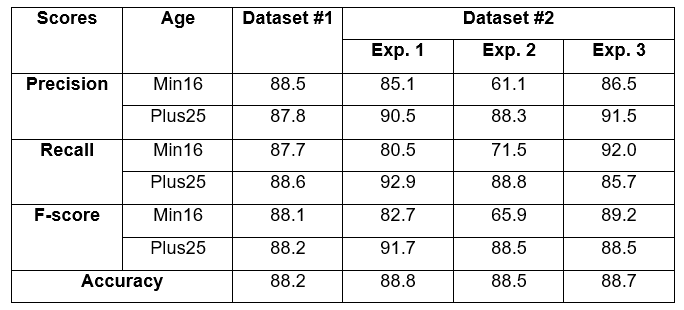
### **2.2.3 Predicting Age and Gender in Online Social Networks**

Peersman, Dalesman, and Vaerenbergh (2010) conducted a study to explore the feasibility of detecting age and gender using statistical text classification and the usefulness of this approach when applied to short texts. [25]

The experimentation phase begins by obtaining 1,537,283 Flemish Dutch posts from Netlog. Relevant information such as age and gender of the authors were also identified in the corpus. The first step in pre-processing involves extracting only the last post of each interaction. Tokenization was also applied to the dataset, which results into a total number of 18,713,627 tokens. Moreover, each token was converted to a lowercase and four or more consecutive identical characters were reduced to three. The third step in pre-processing involves grouping the data using the profile data. In this step, the corpus is divided into following subclasses: min16 (from 11 to 15 years old), plus16 (16 and older), plus18 (18 and older) and plus25 (25 and older). The metadata for both genders were also incorporated and the following classes were derived: min16\_male, min16\_female, plus25\_male and plus25\_female. For their experiment, they used 10,000 posts per class and then we first set up our experiments with 10,000 posts per class and then subsequently decreased it to 5000 and 1000 posts per class.

For the feature selection process, they applied the Chisquare (χ2) metric. The feature set was limited to token and character features: word unigrams, bigrams and trigrams, character bigrams and trigrams, and tetragrams. The feature sets were built by selecting the 1000, 5000, 10,000 and 50,000 features with the highest Chisquare values. Once the features have been selected, each document is represented as a binary vector for the SVM classifier. Moreover, the SVM classifier was trained using Liblinear package. The performance was evaluated using 10-fold cross validation as experimental regime.

In their first experiment, they reduced the number of classes in both train and test sets from the four complex classes to two in order to compare the result to those from the first dataset, which was balanced according to age only. In their second experiment, the classifier was trained into four complex classes then the number in the classifier’s output was reduced to two classes in order to determine whether the extra gender information the classifier had acquired would generate to a better age prediction on the test sets. The third experiment involves the reduction of the number of classes in both training and test sets to two age classes and gender was included as an extra feature in every instance. Table 2.6 illustrates the overview of the results of the three experiments in comparison with the first dataset.

  
*Table 2.6: The result of the three experiments in comparison with the first dataset*

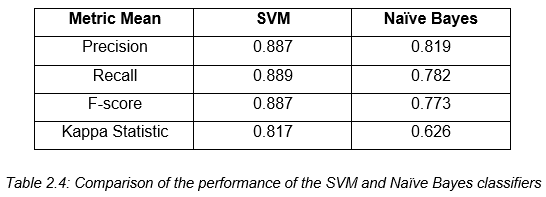
### **2.2.4 Classifying Typhoon Related Tweets**

In a study conducted by Lam, Paner, Macatangay, and Delos Santos (2014), [26] they illustrated the classification of typhoon related tweets into six categories:

• Resource coordination  
 • Urgent rescue needed  
 • Urgent rescue resolution  
 • Damage reporting  
 • Missing people  
 • Media storm coverage

The experimentation phase begins with the collection of 2,356 tweets using Tweet Miner. Furthermore, these data were stored in the SQLite database.  The preprocessing steps involve the filtering of tweets that do not contain an official hashtag. Moreover, each data in the set were converted into lowercase for two main purposes: to normalize the tweet by removing duplicate words from inconsistent casing and to remove official hashtags. All of these steps were done using Tweet Filter. Additionally, the filtered tweets are converted into BoW representation in ARFF format.

For their experiment, they trained both SVM and Naïve Bayes classifiers in WEKA. Furthermore, these classifiers were tested using ten-fold cross validation. For the evaluation metrics, they used precision, recall, f-score, and kappa statistics. As shown in Table 2.7, the SVM classifier outweighs the performance of Naïve Bayes classifier in both metrics.



*Table 2.7: The comparison of the performance of SVM and Naïve Bayes*

## **2.3 Cyberbullying Detection**

Several studies have been conducted in automating the detection of cyberbullying on social networking sites to flag harmful messages and prevent these messages from remaining in the cyberspace by providing timely responses. [15] This section focuses on the various methods used by different researchers in automating the process of detecting cyberbullying and its multiple forms (racial discrimination and offensive language). It also examines the effectivity of each approach.

### **2.3.1 Locate the Hate: Detecting Tweets against Blacks**

Kwok and Wang (2013) applied a supervised machine learning approach in detecting tweets which pertains to racial discrimination. [27] In their experiment, they designed a survey to gauge the complexity of identifying hate speech with the use of Fleiss’ Kappa to assess the reliability of agreement. They began by compiling a number of 100 tweets that contains keywords that are found in hate speech. Three annotators were assigned to classify whether a tweet was offensive or not, and the severity level of offensive tweets from a scale of 1 to 5 (with five being the most offensive). However, since the calculated percentage of overall agreement was only 33%, they assumed that it would be more difficult for machines to classify tweets accurately.

As for the classification of racist and nonracist tweets, they implemented Naïve Bayes classifier. The experimentation phase begins by constructing a dataset of racist and nonracist tweets. Moreover, a total number of 24,852 tweets were obtained. The preprocessing steps involve the removal of URLs, mentions, stopwords, and punctuation, transformation of each word to a lowercase, and normalization of words. By analyzing the tweets in the survey, they were able to derive labels such as offensive words, reference to painful historical contexts, stereotypes, threatening, and others. In addition to this, their feature set was limited to unigrams which results into 9437 unique words in the racist training dataset and 8401 unique words in the nonracist training dataset.

A 10-fold cross-validation method was utilized to evaluate the performance of their classifier. Furthermore, they were able to achieve an accuracy of 76% and an error rate of 24%.

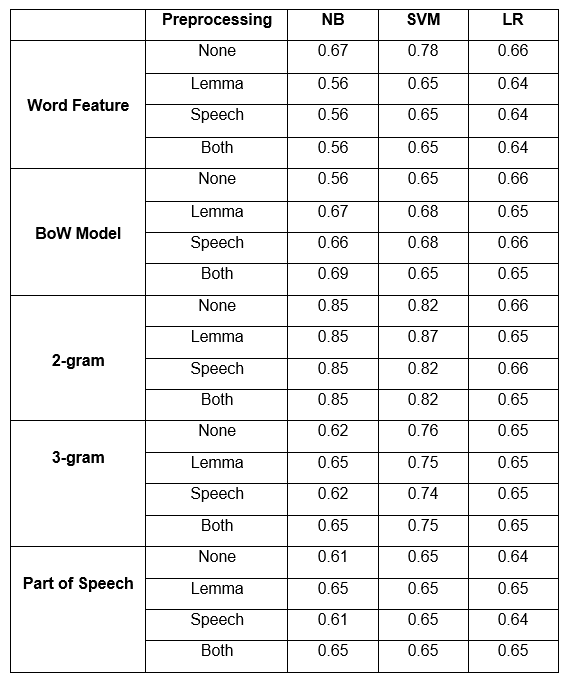
### **2.3.2 Comparison of Machine Learning Algorithms for Offensive Language Filtering**

Buckman (2012) employed different machine learning techniques to identify and filter players of the videogame League of Legends, who are responsible for authoring offensive posts. [28] The experimentation phase begins with the construction a corpus by crawling a Python script which utilizes the BeautifulSoup library in the Riot Games Website. A total number of 30011 cases were downloaded and stored in JavaScript Object Notation. The first 11 cases were used to aid in the development of the software and the remaining 30000 cases were allocated for the test corpus. The dataset was divided into two partitions: training and testing. Each dataset contains a number of 100 cases.

The first step in preprocessing the data involves the use of a control method wherein each text was left unaltered. A spell checker was also applied to fix the common errors and replace misspellings with the proper words. This process was done using Enchant software package and the aspell dictionary. In addition to this, the words were lemmatized using Python’s Natural Language Toolkit.

The process of feature extraction involves the use of n-gram models, grammatical parsing, and word features. The n-gram models were programmed using Python. In grammatical parsing, each word was tagged with one of the Penn Treebank POS tags such as coordinating conjunction, comparative adjective, and personal pronouns. The word features that were extracted are word length, words per line, number of capital letters, and letters per word. A total number of 10 features were obtained.

For their experiment, they employed three machine learning algorithms: Naïve Bayes, Support Vector Machine, and Logistic Regression. As shown in Table 2.8, SVM outweighs the performance of Naïve Bayes and Linear Regression with an accuracy of 87%, achieved by extracting 2-grams from data which had been lemmatized.

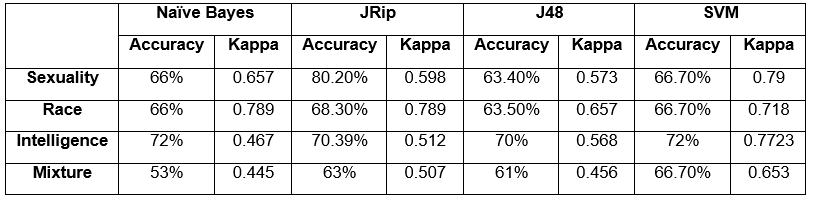


*Table 2.8: Accuracies of Trials*

### **2.3.3 Modeling the Detection of Textual Cyberbullying**

Dinakar, Reichart, and Lieberman (2011) proposed a method in creating a cyberbullying detection model. [20] Their experiment begins with the creation of a corpus composed of YouTube comments by using YouTube PHP API. They were able to obtain a number of comments that exceeds 50,000. The comments were partitioned into clusters of physical appearance, sexuality, race and culture, and intelligence. In addition to this, 1500 comments from each clusters were annotated to three categories: sexuality, race and culture, and intelligence. As for those comments that were not related to the cluster, they were given a label “none”. Each dataset was subjected to four operations: the removal of stop-words, stemming, removal of unnecessary sequence of characters, and cleaning. The dataset for each cluster were further divided into three partitions: 50% training, 30% validation and 20% test data. Moreover, they used four supervised learning methods: Naïve Bayes, SVM, JRip, and J48.

They extracted two kinds of feature from each dataset: general features and specific features. The general features were common across all datasets for both experiments and they are composed of: TF-IDF, Ortony lexicon for negative, list of profane words, and POS bigrams (JJ\_DT, PRP\_VBP, and VB\_PRP). The label specific-features are composed of topic specific unigrams and bigrams. To measure the effectivity of each classifier, they used accuracy and kappa statistics.

  
*Table 2.9: The comparison of the performance of Naïve Bayes, JRip, J48 and SVM*

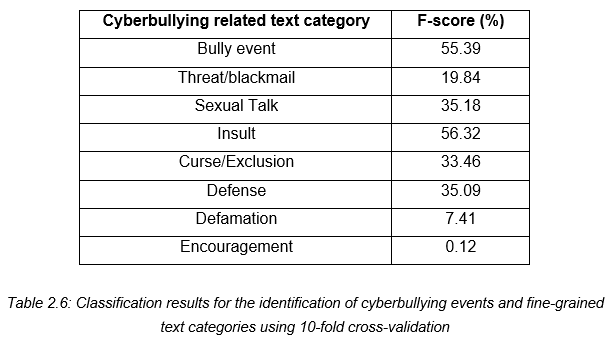
As shown in Table 2.9, JRip yields the best performance in terms of accuracy while SVM is the most reliable as measured by kappa statistics. In addition to this, the binary classifiers trained for each individual labels performed better than multi-class classifiers trained for all the labels.

### **2.3.4 Automatic Detection and Prevention of Cyberbullying**

In the experiment of Van Hee et. al (2015), they proposed a method for automating the identification of cyberbullying events and their classification into cyberbullying categories.  [18] The experimentation phase begins with the creation of corpus by collecting 91, 370 Dutch posts from Ask.fm. Moreover, they illustrated two levels of annotation: First, the assignment of harmfulness score to the post on a three-point scale wherein 0 indicates non-cyberbullying event, 1 indicates mild cyberbullying event, and 2 indicates severe cyberbullying event. Moreover, the roles in a cyberbullying event were also identified: victim, harasser, bystander-defenders (who discourage the harasser) and bystander-assistant (who take part in the actions of the harasser). At the second level of annotation, each data was classified into fine-grained text categories related to cyberbullying: insults, threats, sexual talk, defamation, defense and curse. In total, 85,462 Dutch posts were successfully annotated using brat rapid annotation tool. Moreover, the interannotator agreement scores were calculated using Kappa. They obtained a Kappa score of 0.69 in the identification of cyberbullying events. Additionally, the Kappa scores for the fine-grained cyberbullying categories such as Threat, Insult, Defense, Sexual Talk, and Threat vary from 0.52 to 0.66.

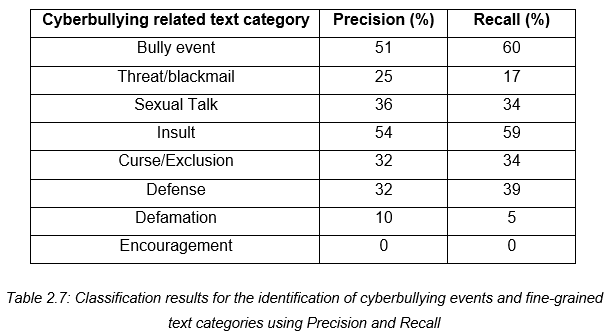
The preprocessing steps involved tokenization, PoS-tagging and lemmatization to the data by utilizing LeTs Preprocess Toolkit. They implemented two types of lexical features for their experiment: bag-of-word features and polarity features based on existing sentiment lexicons. Thus, it results into a set of 300,000 features. The proponents utilized a Support Vector Machine (SVM) as their classification algorithm. All of their experiments were carried out using Pattern.

For their preliminary experiment, the evaluation was done using 10-fold cross-validation. Moreover, they used F-score for their evaluation metric. Table 2.10 shows the result of their preliminary experiment by using F-score.



*Table 2.10: Classification results for the identification of cyberbullying events and fine-grained text categories in terms of F-score*

Table 2.11 illustrates the performance of both precision and recall with regards to the identification of cyberbullying event and their classification into fine-grained text categories.



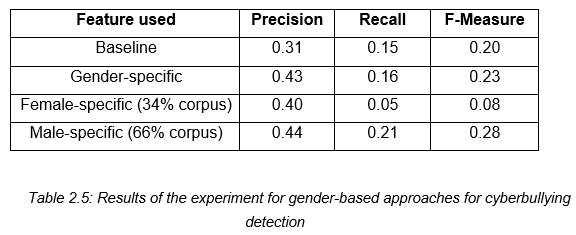
*Table 2.11: Classification results for the identification of cyberbullying events and fine-grained text categories in terms of precision and recall*

### **2.3.5 Improved Cyberbullying Detection using Gender Information**

Dadvar, Jong, Ordeiman, and Trieschnigg (2012) believed that the incorporation of gender specific language features will improve the accuracy of a cyberbullying detection system. [29]To test this idea, they conducted an experiment on impdadvarroving cyberbullying detection with the aid of gender specific features.

Their dataset was composed of MySpace posts provided by Fundacion Barcelona Media. In total, the corpus contains 381,000 posts wherein 34% was written by male and 67% were from female. However, they were only able to utilize 2,200 posts for their experiment. Furthermore, the dataset was annotated into two categories: harassing and non-harassing. They analyzed the use of foul words among the 100,000 posts and compared the most frequently used foul words by each gender. By utilizing Wilcoxon signed rank test, they were able to determine the different frequencies of foul words in each gender.

For harassment classification, they utilized four types of features: first, profane words (including their acronyms and abbreviations), personal pronouns, second person pronouns, and TFIDF. These features were employed to train the classifier. Moreover, they constructed a Support Vector Machine (SVM) classifier in WEKA. First, they utilized the posts written by both genders as their dataset, then they trained the classifier separately for each respective gender group. In evaluating the accuracy of the classifier, they used 10-fold cross validation and calculated its precision, recall and F-measure. As shown in Table 2.5, the incorporation of gender-specific features improved the overall accuracy measures.



*Table 2.12: The result of improving cyberbullying detection using gender-specific features*

### **2.3.6 Automated Role Detection in Cyberbullying Incidents**

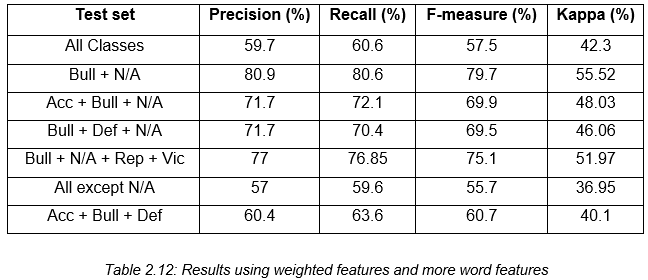
Cheng and Ng (2016) conducted an experiment on the detection of cyberbullying roles. [30] Their experiment begins with the creation of a corpus by gathering data from Facebook and Youtube. In total, 6000 posts/comments written in both English and Tagalog were collected (1500 for YouTube and 4500 for Facebook). The dataset was cleaned by removing unnecessary symbols. Furthermore, it underwent normalization through the use of NormAPI. Lastly, each data was manually annotated into six classes: Bully, Accuser, Defender, Reporter, Victim, and N/A (which pertains to the instances that do not belong to the any of the class).

They implemented four types of features for their experiment: bag-of-word, TF-IDF, profane words, and word shape or the instances written in all uppercase. The experiment was conducted 7 times, each with a different set of role classes. The combination of the roles is as follows:

* All classes
* Bully and N/A (Bull + N/A)
* Accuser, Bully, and N/A (Acc + Bull + N/A)
* Bully, Defender, and N/A (Bull + Def + N/A)
* Bully, N/A, Reporter, and Victim (Bull + N/A + Rep + Vic)
* Accuser, Bully, Defender, Reporter, and Victim (All except N/A)
* Accuser, Bully, and Defender (Acc + Bull + Def)

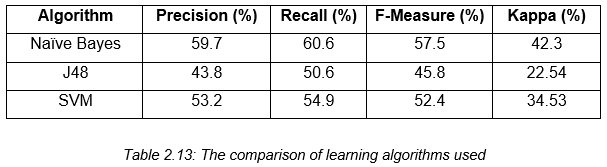
For their first experiment, they used an initial set of 25 word features in each class. They checked the presence of both words that are written in all capital letters and those which contains profanity. From a total number of 150 features, it was decreased into 93 unique word features. Their second experiment involves the removal of both intersecting words and other added features. Thus, if a word feature is found in more than 1 class it will be removed in the feature set. The total number of 150 features was decreased into 63. As shown in Table 2.9, there was a decrease in all measures. Their third experiment involves the removal of both profanity and all capital words as features. In this experiment, the model was able to predict more bully and defender roles by removing both profane and full capital words.

For their fourth experiment, they utilized a weighting system that will assign weights to word features. This experiment was done in order for the model to be able to distinguish the respective classes for each feature. There was a significant improvement in the results as compared to the previous experiments. Thus, the assignment of weights can further help the classifier in identifying the features for each of the classes. The next experiment involves adding more features to the current set. Some word features were replaced with more relevant ones such as nouns and proper nouns. More common words were also removed in this phase. The initial number of 25 word features per class was increased into 50. The last experiment obtained a highest accuracy compared to the previous ones. Thus, by adding more relevant features, the roles of the bully, accuser, and victim were able to have more correctly classified instances.



*Table 2.12: The result of the last experiment*

Lastly, the experiment that yield the highest accuracy was tested using different algorithms: Naïve Bayes, J48 and Support Vector Machine. As shown in Table 2.12, among the three algorithms that were utilized, SVM yield the highest accuracy.



*Table 2.13: The comparison of the performance of Naïve Bayes, J48 and SVM*

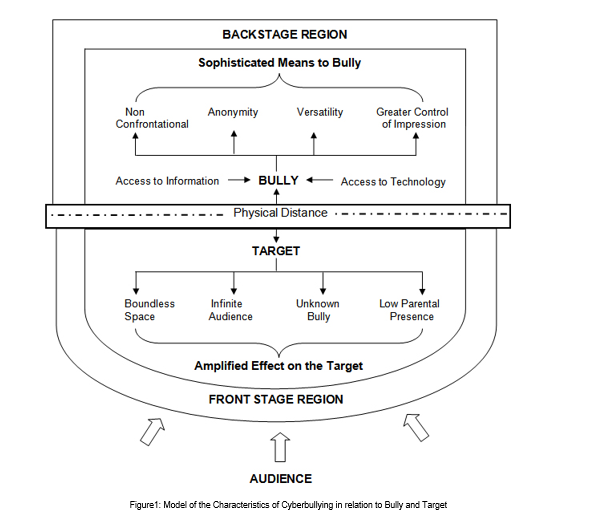
# **3. Theoretical Framework**

## **3.1 Audience Segregation by Ervin Goffman**

In his book “The Presentation of Self in Everyday Life”, Ervin Goffman introduced the mechanisms of audience segregation. He describes how people play different roles in different situations. It is a mechanism wherein an individual performs roles, in order to create a favorable image of themselves and leave a good impression to others that is linked to the role they perform. The role that the individual performs is based on who their audience is.

Nowadays, more and more people are getting inclined to social networking sites because it provides an easier way for social interactions and communications. These sites allow users to share personal information about themselves through text, pictures, and other forms of media which in turn, creates an image for each user; however, the representation of oneself in the cyberspace is on a global scale in front of an audience which is possibly unknown and infinite. In social networking sites, the user’s privacy is threatened because a large audience might have access to his personal information. In order to handle privacy issues, there were few social media sites that offer limited options for making one’s profile visible for a specific set of individuals. As for some cases, audience segregation is used as a solution to protect user’s privacy; however, Goffman’s segregation of audiences is a lot harder in the era of the Internet. Difficulties begin when the audience is used to a certain type of performance from an individual or team but observes another performance which does not create the same impression which results to cyberbullying. The impression created on a social networking profile may not resemble an individual’s real life identity.

The nature of communicating in the cyberspace facilitates the potential for anonymous interactions. It was discovered that bullies who choose to use electronic means can easily hide their real identity and make themselves anonymous. Anonymity can be created through the use of temporary email addresses, fictitious names or unknown mobile number. The perception of anonymity in social media serves as a disinhibitor so that people are more likely to do and say things online that they would not do or say in a face to face situation. Another key characteristic of cyberbullying is the potential to reach a limitless audience. Due to the boundless nature of cyberspace, the audience is not confined to a single setting (such as school or office) but has the potential to be viewed by a global audience.

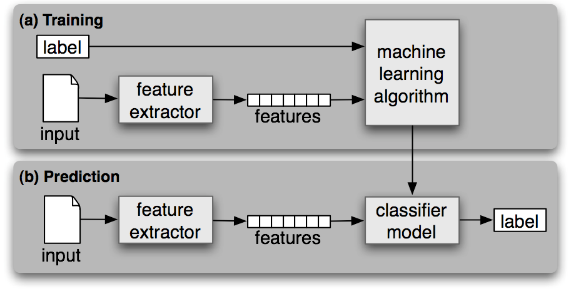


Goffman's framework offers not only a way of thinking about space in terms of performance but also a way of thinking about how people may act differently depending on the audience and setting which are relevant to an exploration of cyberbullying. Goffman defined three roles in this mechanism: performer, audience, and outsider. These roles can be paralleled to the roles of a target, bully, and bystander. By framing bullying as a performance, a framework is provided that enables us to consider the bystander group as an audience and how different settings may affect how young people act towards others. In order to set the scene for a performance, Goffman made a distinction between the two regions of social space where an individual interacts. The front region is defined as the public performance area. The backstage region is a place wherein the performer can privately prepare for the performance or where members of a group can openly construct the impression they are planning to give. By using Goffman’s framework of performance, cyberspace interactions can be executed by the bully in the backstage region which impacts on the target in the public front stage region. As the backstage region is a place that performers may privately prepare away from the audience, this provides time and space for the bully to plan the ways in which they wish to target others. The physical distance which cyberspace interactions facilitate may also result in the bully managing the impression ‘given off’, the ability for the bully to conceal their identity and the tone and meaning being open to wider interpretation.

## **3.2 Text Classification**

Machine Learning focuses on building systems that can learn from examples. It aims to automate the process of learning in order to make accurate predictions through the use of examples. In relation with NLP, Machine Learning is used to understand the meaning of natural language, therefore, machines have to learn how to do it. One of the examples of how Machine Learning and Natural Language Processing can be leveraged to enable machines to better understand human language is text classification. In text classification, each text document is classified into one or more categories. Since the manual process of categorizing documents can be a laborious task especially if there are several number of documents, machine learning automates the process of text classification.

With the aid of machine learning, the goal of text classification is to build classifiers by learning the characteristics of the categories from a set of pre-classified documents (Sebastiani, 2002). There are several kinds of classifiers that are suitable for different text classification problems. Therefore, choosing the right classifier is crucial for the performance of the program. The decision criterion of a classifier is learned automatically from the training data. Thus, once the classifier has been trained, it can predict the category of the new data. This approach is also called statistical text classification.  Figure 3.3 illustrates the process of statistical text classification.



*Figure 3.3: Statistical Text Classification*

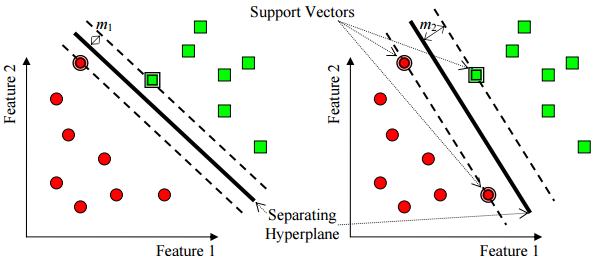
As shown in Figure 3.3, the process of statistical text classification begins with feature extraction wherein a feature extractor is used to convert each input value to a feature set, which captures the relevant information about each input that will be used in order to classify them. Both features and labels are fed into the machine learning algorithm in order to generate a model. During prediction, the same feature extractor will be used to convert new inputs into feature sets. These feature sets are fed into the model, which in turn, will produce predicted labels.

Some applications of text classification are spam filtering, email routing (Busemann, Schmeier & Arens, 2000), language identification, and genre classification (Marina & Mark, 2008).

### **3.2.1 Support Vector Machine**

In a machine learning approach to text classification, an algorithm will be used in learning how to classify documents by producing a model to map the input and output. One of the most popular models used in text classification is linear model, which uses the linear combination of feature-values. There are several linear models and one of the most commonly used model is Support Vector Machine (SVM).

Vapnik et al. developed Support Vector Machine, a supervised learning model that is used to analyze data in text classification or regression. It is based on Structural Risk Minimization principle from computational learning theory. SVM performs classification by creating a k-dimensional hyperplane that separates the data into two categories. The number of dimension is equivalent to the number of features an object can possess. In text classification, a feature can be a number of occurrence of particular word in the whole document.



*Figure 3.4: Support Vector Machine with two features*

In a set of training examples wherein each data has already been labeled, an SVM training algorithm produces a model that will assign new examples to one of the categories which makes it a non-probabilistic binary linear classifier. An SVM model represents the examples (or support vectors) as points in space. SVM seeks to find a line (or hyperplane) that separates the examples based on their labeled classes. The two dashed lines drawn in parallel to the hyperplane represents the distance between the hyperplane and the closest vectors to the line. Moreover, the distance between a dashed line and the hyperplane is called the margin. Thus, whenever a data is added, the side of the hyperplane where it lands will determine the class that will be assigned to it. Figure 3.4 illustrates how SVM works with two features wherein points are plotted on a 2-dimensional plane.

## **3.3 Bag of Words**

In statistical text classification, each input is treated as a feature vector. One of the most common methods used in transforming a text document into a feature vector is through the use of “bag-of-words” representation, in which a set of text documents is converted into a numeric feature vectors wherein the order of word occurrences and grammar are ignored. Moreover, it is defined as an order less document representation (Salton & McGill, 1983). In this model, the count of words is given the utmost importance. Each word is represented by a vector of the word counts that appear in the whole document. In this scheme, each individual token occurrence frequency is treated as a feature. Regardless of the simplicity of Bag-of-Words in data representation, it often achieves high performance. (Lewis, 1992).

Once the text has been converted into a BoW model, various measures can be computed to characterize the text. One of the most popular type of features from the BoW model is term-frequency, the number of times a certain term appears in the text. However, term frequency is not considered as the best representation for the text. Oftentimes, insignificant words (such as articles) always yield the highest frequency in the text. These limitations led to the introduction of Term Frequency – Inverse Document Frequency which seeks to diminish the weight of terms that occur very frequently in the document and increases the weight of terms that occur rarely (Jones, 1972).

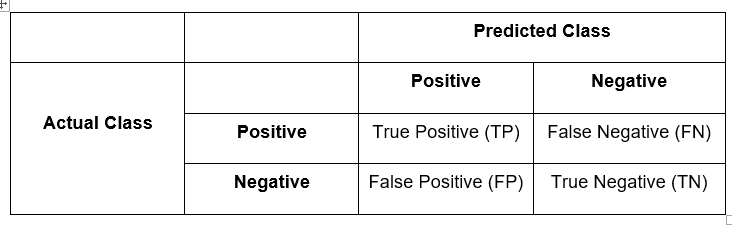
In the concept of TF-IDF, the high weight is conceived by a high frequency and a low term frequency in the whole document. Thus, the weights tend to filter out common terms. The ratio in the idf log function is always higher than or equal to 1, while the value of idf is always higher than or equal to 0. Moreover, when a term appears frequently in the documents, the ratio inside the logarithm approaches 1, bringing the idf and tf-idf closer to 0 (Josef, 2009).

## **3.4 Performance Measures**

Most evaluation for document classifier is conducted experimentally. Thus, it is used to measure its effectiveness or the quality of its predictions on the classification of data. Predictions made are either considered Positive or Negative and expected judgments are called True or False (Pinto, Olieveira & Alves).

As shown in Figure 3.5, a confusion matrix is a table that has two rows and two columns which shows the total number of false positives, false negatives, true positives, and true negatives. Moreover, it allows more detailed analysis than a mere proportion of correct guesses (or accuracy).

* True positive refers to the number of examples predicted positive that are actually positive
* False positive refers to the number of examples predicted positive that are actually negative
* True negative refers to the number of examples predicted negative that are actually negative
* False negative refers to the number of examples predicted negative that are actually positive



*Table 3.5: Confusion Matrix*

### **3.4.1 Precision**

Precision is used to measure the exactness of the classifier. Moreover, it refers to the fraction of predicted positive which are actually positive. It is also called positive predictive value (PPV). A high precision indicates less false positives, while a classifier with a low precision means there are more instances of false positives. Precision can be improved by decreasing the recall.

The formula for precision is the number of positive predictions divided by the total number of positive class values predicted.

### **3.4.2 Recall**

Recall refers to the fraction of those that are actually positive that were predicted as positive. It is used to measure the completeness of a classifier. Moreover, it is also called the true positive rate or sensitivity. Higher recall indicates less instances of false negatives, however, a classifier with lower recall means there are more instances of false negatives. Recall can be improved by decreasing the precision primarily because it is harder to be precise as the number of samples are increasing.

The formula for recall is the number of positive predictions divided by the number of positive class values in the test data.

### **3.4.3 Accuracy**

The accuracy is the percentage of instances that were correctly classified into their respective classes. It is also called sample accuracy.

One of the disadvantages of accuracy is it can yield to misleading result if the dataset is unbalanced or the number of samples in different classes vary. To illustrate, a model can predict the value of the class with the highest number of samples for all predictions and achieve a high classification accuracy.

### **3.4.4 F Measure**

The F-measure (or F-score) is used to measure the accuracy of the test by considering both precision and recall in computing the score. It conveys balance between precision and recall wherein it reaches its best value at 1 and its worst value at 0.

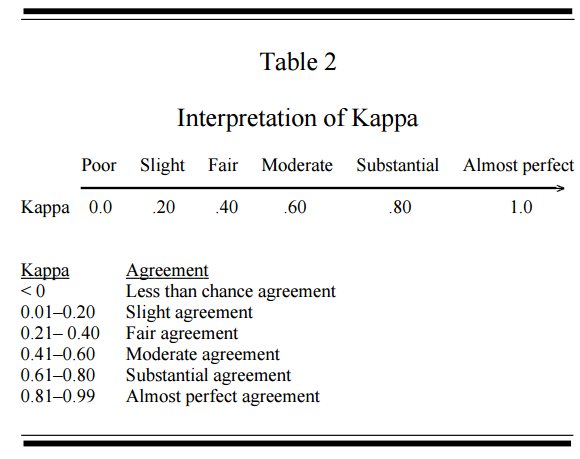
Two of the commonly used F measures are measure and measure. The measure puts more emphasis on the false negatives by weighing recall higher than precision. measure puts more emphasis on reducing false negatives by weighing recall lower than precision.

### **3.4.5 Kappa Statistics**

Interobserver agreement is a procedure to enhance the believability of data by comparing observations from two or more people who are evaluating the same thing. In evaluating, the observers would agree just by chance. Thus, kappa provides numerical rating of the degree to which this occurs. The calculation is based on the difference between the numbers of agreement that are actually present compared to the numbers of agreement that would be expected to be present by chance.



Figure 3.5 illustrates how Kappa measure the differences by standardizing into a -1 to 1 scale.



*Figure 3.5: Kappa Interpretation*

# **4. Design and Methodology**

## **4.1 System Overview**

Quickgarde is a prototype that can both detect and report cyberbullying occurrences. Moreover, it can detect harmful posts written in English and Tagalog.

## **4.2 System Objectives**

### **4.2.1 Main Objective**

The system’s main purpose is to be able to detect statements that contain cyberbullying elements and procure reports to the administrator via email.

### **4.2.2 Specific Objectives**

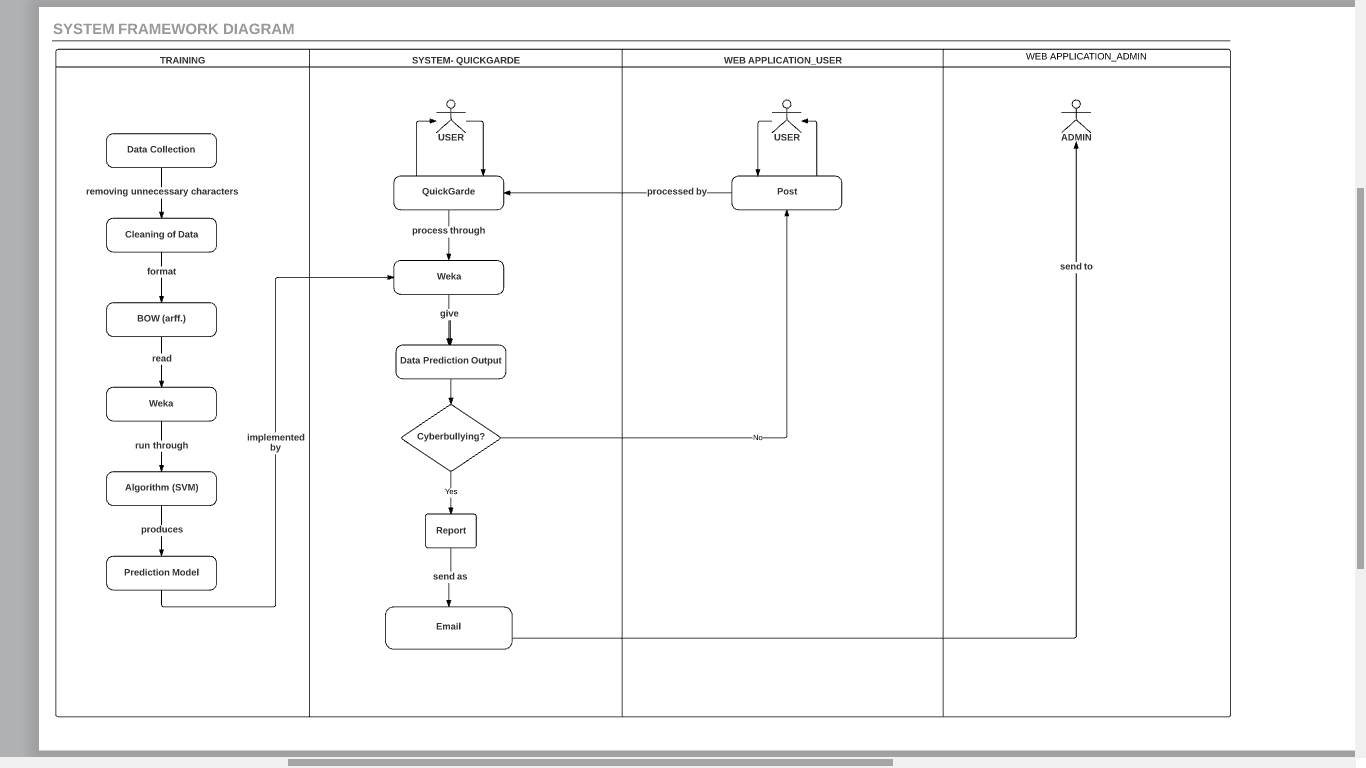
* Automate processes included in the research conducted
* Automate the process of monitoring social media sites
* Detect cyberbullying occurrences
* Flag detected cyberbullying occurrences
* Report cyberbullying incidents to the administrator via email

## **4.3 Scope and Limitation**

Quickgarde is a cyberbullying detection system that can detect and report cyberbullying occurrences. It covers only harmful posts written in English and Tagalog language. Moreover, its functionalities are limited only to textual data. The system was programmed in Java.

As for the process of detecting instances of online bullying, the system is incorporated with a cyberbullying detection model which utilizes a Support Vector Machine algorithm deployed through WEKA. Once a cyberbullying post has been detected, a report regarding the incident will be sent to the administrator. The report contains the content of the cyberbullying post.

## **4.4 System Architecture**

****

*Figure 4.1 System Architecture*

## **4.4.1 Experimentation Phase**

### **4.4.1.1 Data Corpus**

Social networking sites such as Youtube, Facebook and Twitter were used as sources of data for the corpus. The dataset from Youtube contains comments from videos focusing on controversial events in the Philippines such as cases of bashing against Filipino celebrities and video bloggers, and scandals wherein politicians and celebrities are involved because these topics are often a rich source for objectionable and rude comments (Dinakar, Reichart & Lieberman, 2011).

In Facebook, several posts from the different universities' confession pages were collected because these pages allow anyone to share personal secrets, rumors, gossips, and anything else they might want others to know about but are hesitant to post publicly or in a way that is tied to their identity. Thus, the anonymity of the person posting a confession makes these pages vulnerable to cyber bullying activities. In Twitter, various posts from random Filipino netizens were obtained. Twitter is also prone to cyber bullying attacks since users can easily create fake accounts to launch their bullying cyber-attacks against people they don’t like or disagree with. In 2011, a study conducted by the University of Wisconsin-Madison found that 15,000 abusive tweets per hour, which equals 100,000 abusive tweets a week. [17]

Import.io, a web scraping tool, was utilized to extract data from these social media sites. It is a tool which allows people to convert unstructured web data into a tabular format and store it in an Excel or CSV file. [18] The only field in the table that was used in collecting data for the corpus was the textual content of the post while the other features such as the user information, links, and others were disregarded. A total number of 2000 statements written in Filipino and English were obtained.

### **4.4.1.2 Cleaning of the dataset**

The cleaning procedure that was applied on the dataset involved the removal of all special characters (excluding apostrophes and hyphens), non-readable text (e.g. asdfghjkl), emoticons, links, and characters belonging to various foreign countries' writing systems. This was done in order to prevent complications from arising particularly during the experimental phase of the project. Such characters do not make any sense with regard to the detection of cyberbullying occurrences, therefore their appearance may contribute to a probable decrease in the accuracy rate of the model. Apostrophes and hyphens, on the other hand, were retained for they help join characters together in order to yield another word. Since the presence of distinct features were used as basis for the frequency of each word in every statement, it is important to include all words preserved in forms understandable by Filipinos within the dataset. This was procedure was done using regular expressions.

#### Data Annotation

Once the preprocessing steps were accomplished, the dataset was further subjected to annotation. For this step, each data was classified into three labels: Cyberbullying, Non-Cyberbullying, and Ambiguous Cyberbullying (a case wherein the annotator was unable to identify whether a certain post implies cyberbullying or not). For this process, 100 questionnaires (that contains 10 sentences (with a total number of 1000 statements) taken from the corpus were distributed among APC students. As for the remaining 1000 statements, three annotators were assigned to label each statement. The participants will manually label each data into three categories. Furthermore, the labeled data will be used in training the classifier.

**4.4.1.4 Tokenization**

In this phase, all of the statements that were cleaned will be divided per each word within a particular statement based on the whitespaces separating them. This function will help provide each distinct occurrence of all the words that were part of the statements stored within the corpus. Once this process had been accomplished, it will determine the number of occurrences (frequency) of each feature as they occur in every statement. The acquired numerical values will then be used in the implementation of the Bag-of-Words.

4.4.1.5 Bag-of-Words

The dataset was transformed into a Bag-of-Words model, in which a set of text documents is converted into a numeric feature vectors wherein the order of word occurrences and grammar are ignored. It is primarily used as a tool of feature generation. The process begins by creating a list of unique words from the text. Once a list has been created, the number of times a word appears in a document will be computed. From the bag-of-words we removed all words that contained digits.



*Figure 4.2: Bag of Words Model*

After cleaning the dataset, the csv (comma-separated values) file was converted into .arff (Attribute-Relation File Format) format since it is the one being used in WEKA. In this format, the distinct features will be represented by the attributes, and the relation as the whole corpus itself. At the bottom part of the file, the number of occurrences (of each word in every statement) along with the annotations placed by both the researchers and their correspondents (in every statement), will be placed. Such data initially came from the .csv file containing the cleaned, parsed, and evaluated words comprising each of the 2000 statements.

#### 4.4.1.6 Support Vector Machine

Classification is the task of identifying the label for a single entity from a set of data. in order to determine cyberbullying from not-cyberbullying data, an SVM classifier was trained on a set of labeled data. Thus, these words are essentially treated as features that the classifier will use to model the positive instances of cyberbullying as compared to non-cyberbullying and ambiguous cyberbullying.

The Support Vector Machine algorithm was the only text classification algorithm that was used in the research project. It was implemented in the WEKA toolkit, a data processing and machine learning tool.

#### 4.4.1.7 Cyberbullying Detection Model

Among the 2000 statements, a total number of 900 was used for this experiment. The sole experiment that was performed involved the use of the Support Vector Machine (SVM) algorithm on, supposedly, the 2000 statements.

In this phase, the algorithm will be implemented together with the processed data in WEKA. The flagging of cyberbullying statements takes place in this phase. There will be charts that the tool will present to indicate how it classified a particular statement.

## **4.4.2 Training Phase**

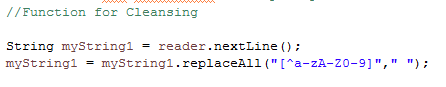
This section illustrates the processes involved in the automation of the processes in the experimentation phase.

### **4.4.2.1 Data Collection**

The training phase begins by collecting data from the user through the use of a console.

### **4.4.2.2. Data Cleaning**

The following lines of code perform the process of cleaning the user input:



### **4.4.2.3 Bag of Words**

### **4.4.2.3.1 Tokenization**

The input will be divided into a series of words.

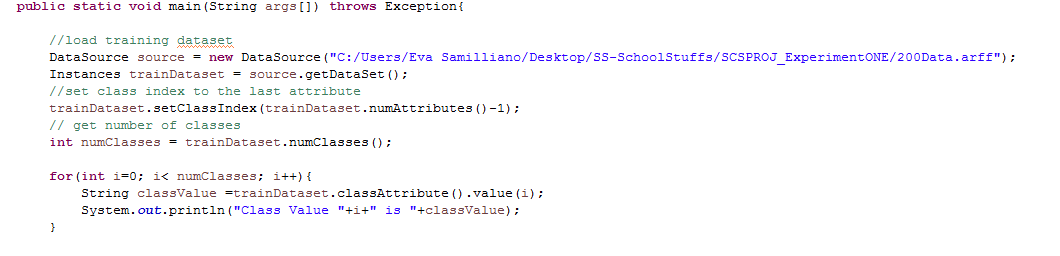


### **4.4.2.3.2 Term-Frequency Value**

Once a list was constructed, the term frequency will be computed to characterize the text. For this process, a method called ***getUniqueKeys***will determine the number of times a word appears in the text.

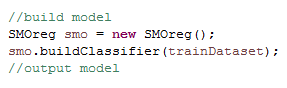
### **4.4.2.3.3 WEKA**

The following lines of code calls the function of WEKA:



### **4.4.2.3.4 Support Vector Machine**

The following lines of code build an SVM classifier which will perform the text classification task for cyberbullying and non-cyberbullying classes.



## **4.4.3 Reporting of Cyberbullying Incident**

This section contains the processes involved in the report generation procedure of the system.

#### 4.4.3.1 Prerequisites

The first requisite for the report generation process is a web server. For this project, we utilized xampp package alongside with the Mercury Mail Transport System as a mail server. In addition to this, Outlook 2013 was used as the mail client.

#### 4.4.3.2 Cyberbullying Report Generation

Once a cyberbullying incident has been detected, it sends an email to the administrator regarding the post.

### 

### **4.4.3.2.1 Instantiating a report**

The report generation begins with the instantiation of a report through the use of the following codes: ***Message msg = new MimeMessage(session);***

### **4.4.3.2.2 Setting the report attributes**

The attributes of the report was specified as well: ***msg.setSubject*** contains the title of the report, ***msg.setRecipients*** specifies the receiver of the report, and ***msg.setSentDate*** specifies the date the report was sent.

### **4.4.3.2.3 Setting the message content**

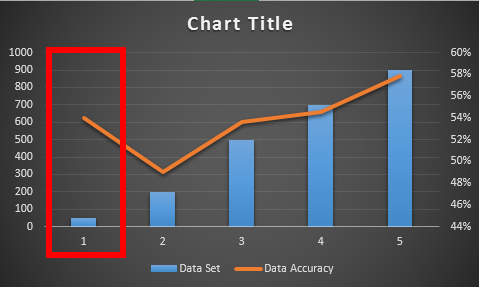
The function msg.setText displays the content of the report, which in this case is the content of the detected cyberbullying post.

### **4.4.3.2.4 Sending the report**

The function ***Transport.send*** sends the report to the administrator via Outlook.

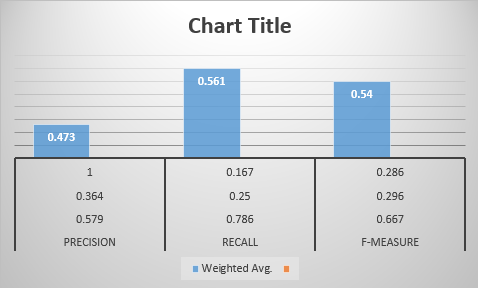
# **5. Results and Discussion**

For the first part of the experimentation phase, a total number of 50 statements were extracted from the corpus. These statements were represented through the use of Bag-of-Words (BoW) model which resulted into a number of 378 attributes. In addition to this, they were partitioned into two datasets: testing and training (10 statements were used for testing and 40 statements for training). These datasets were trained using an SVM classifier deployed in WEKA. The model yields an accuracy of 10% for the testing set and 54% for the training set. The second part of their experiment includes the extraction of 900 statements from the corpus. However, as for this part, they utilized 10-fold cross-validation rather than dividing the data into two sets. As the sample data became larger, the attributes in the BoW model has also increased. The BoW model yields a total number of 7062 attributes from 900 statements. The purpose of conducting small experiments before inserting the whole data from the corpus is to illustrate how the accuracy of the model can change depending on the number of data. The model yields an accuracy of 57.89%. As shown in Figure 5.1, the accuracy of the model has significantly increased when more data was added into it.



*Figure 5.1: The accuracy of the SVM classifier*

Figure 5.2 illustrates the results of other measures that were used in evaluating the model’s performance. The model generated a precision of 47%, a recall of 16% and an f-score of 54%.



*Figure 5.2: Other performance measures*

The training phase was composed of steps on automating the manual procedures on the experimentation phase. The SVM classifier and preprocessing steps were programmed using Java in Eclipse IDE. In order to test this phase, the result was printed on a designated text file.



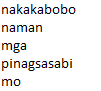
*Figure 5.3: An example of a user input in the console*

Once a text has been entered, the data was subjected to cleaning which involves the removal of unnecessary characters.



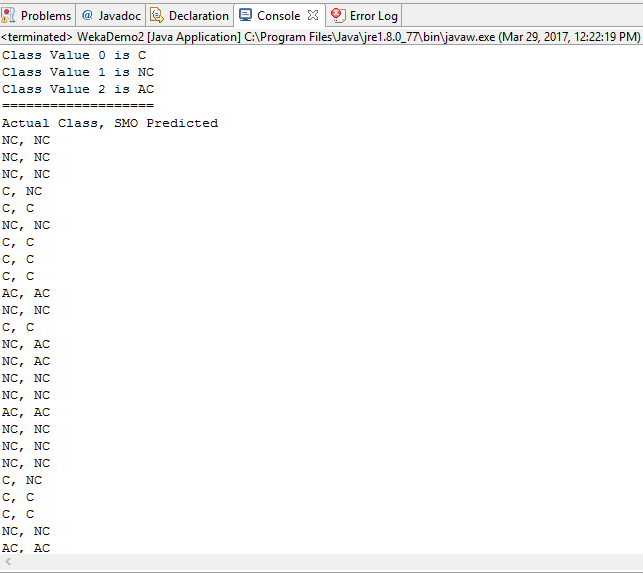
*Figure 5.4: An example of data cleaning procedure*

Moreover, the cleaned data was further partitioned into tokens or a list of words.



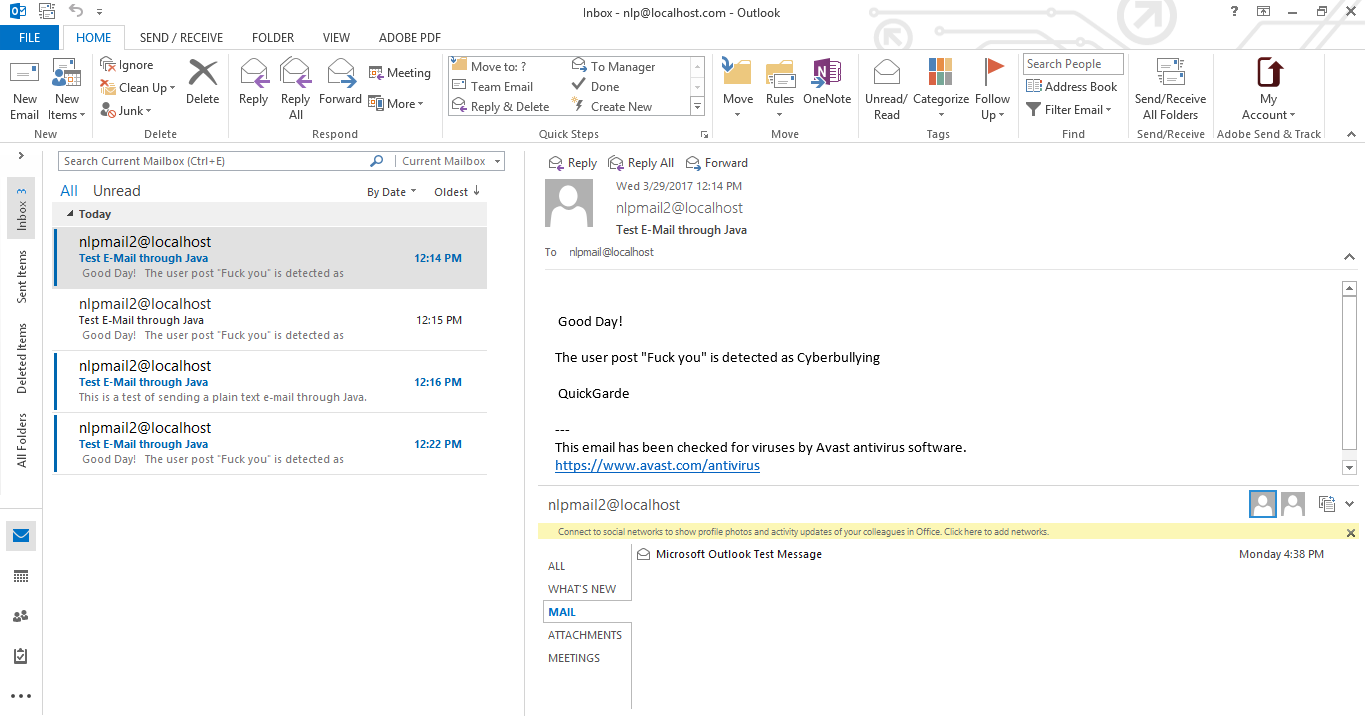
*Figure 5.5: Tokenization process*

Once the input was separated into a list of words, it was compared in an array of attributes derived from the Bag-of-Words model from the experimentation phase. This process was used in determining the frequencies of each word in the list. Once the number of word occurrences has been identified, the classifier determines whether the post implies cyberbullying or not. The output for this phase was printed on the console. Figure 5.6 shows the comparison between the actual class and the class that was predicted by the SVM classifier.



*Figure 5.6: SVM Classifier*

Thus, once a harmful post has been detected, the system establishes a connection with the mail server and automatically generates a report which includes the post of the user. Figure 5.7 illustrates the output of the system once a report has been generated.



*Figure 5.7: The report that was generated by the system*

## **6. Conclusion and Recommendations**

In this paper, we presented a supervised machine learning approach to combat online bullying through the development of Quickgarde, a cyberbullying detection system which can detect harmful posts and generate reports to the administrator via email regarding the cyberbullying incident. The project started by conducting an experiment in an attempt to create a cyberbullying detection model by training an SVM classifier in WEKA. The experimentation phase began with the creation of the corpus by obtaining data from social media sites such as Facebook, YouTube, and Twitter. The dataset was subjected to cleaning and was further annotated into three categories: cyberbullying, non-cyberbullying and ambiguous cyberbullying. Moreover, each statement was partitioned into tokens and represented in a Bag-of-Words (BoW) model which yields a total number of 7062 attributes. The SVM classifier was deployed in WEKA and generates an accuracy of 57.89%. For the training phase of the project, the manual procedures in the experimentation phase were automated using a Java program. The automation process was used to develop a cyberbullying detection system. Thus, once a cyberbullying post has been detected, it automatically sends a report to the administrator regarding the content of the post via email.

As of now, the system relies solely on the user input in the console. However, we are planning to extend the scope of the Quickgarde system by integrating it to a web application (or a social networking site prototype) as a website add-in. Moreover, the report generation will be further enhanced by creating a web application for the admin, wherein he can view the reports in an organized tabular format which includes the name of the person who authored the post, the content of the post, and the time and date it was posted.

We also aim to improve the classifier’s accuracy by adding more data to the corpus and performing additional preprocessing steps such as stemming and lemmatization.

**Sources**

1. Boehm, C. (2012). Moral Origins: The Evolution of Virtue, Altruism, and Shame.

2. Sheoran, J. (2012). Technological Advancement and Changing Paradigm of Organizational Communication. International Journal of Scientific and Research Publications, 2(12).

3. Dadvar, M. De Jong, F. (2012). Cyberbullying detection: A step toward a safer internet yard. WWW 2012 – PhD Symposium. WWW 2012 PHD Symposium

4. What is Cyberbullying? (2011). Retrieved from https://www.stopbullying.gov/cyberbullying/what-is-it/index.html

5. Ellwood-Clayton, B. (2006). All we need is love—and a mobile phone: texting in the Philippines. Cultural Space and Public Sphere in Asia 2006

6. Digital in 2017: Global Overview. (2017, January 24). Retrieved from http://wearesocial.com/sg/blog/2017/01/digital-in-2017-global-overview

7. Gonzales, R. (2014). Social Media as a channel and its Implications on Cyber Bullying. DLSU Research Congress 2014.

8. FROM HUMILIATION TO THREATS: Survey says 80% of young teens in PHL experience cyberbullying (2016, March 30). Retrieved from http://www.gmanetwork.com/news/story/560886/lifestyle/parenting/80-of-young-teens-in-phl-experience-cyberbullying-survey

9. Lacuata, R. (2014, June 27). How cyberbullying changed 'Amalayer' girl. Retrieved from http://news.abs-cbn.com/lifestyle/06/26/14/how-cyberbullying-changed-amalayer-girl

10. Tulad, V. (2012). Cyberbullying: A victim's tale of lies and the madness of crowds. Retrieved from http://www.gmanetwork.com/news/story/274156/hashtag/cyberbullying-a-victim-s-tale-of-lies-and-the-madness-of-crowds

11. Torres, M. (2016). Netizens react to Monster radio DJ Karen Bordador's arrest. Retrieved from https://kami.com.ph/40852-monster-radio-dj-karen-bordadors-arrest-stirs-netizens.html

12. Minor, M. Smith, G. Brashen, H. (2013) Cyberbullying in Higher Education. Journal of Educational Research and Practice 2013, 3(1), pp. 15–29

13. RA 10627: The Anti-Bullying Act. (2015, January 22). Retrieved from http://www.elegal.ph/republic-act-no-10627-the-anti-bullying-act/

14. Republic Act No. 10627. (2013, September 12). Retrieved from http://www.gov.ph/2013/09/12/republic-act-no-10627/

15. Van Royen, K. Poels, K. Daelemans, W. Vandebosch, H. Automatic monitoring of cyberbullying on social networking sites: From technological feasibility to desirability. Telematics and Informatics, 32(1).

16. Sintaha, M.   Satter, S.  Zawad, N.  Swarnaker, C.   Hassan, A. (2016). Cyberbullying Detection using Sentiment Analysis in Social Media (Unpublished doctoral dissertation). Department of Computer Science & Engineering, BRAC University

17. Marathe, S. Shirsat, K. (2015). Contextual Features Based Naïve Bayes Classifier for Cyberbullying Detection on YouTube. International Journal of Scientific & Engineering Research, 6(1).

18. Van Hee et. al (2015). Automatic Detection and Prevention of Cyberbullying. HUSO 2015: The First International Conference on Human and Social Analytics.

19. Madnani, N. (N.D). Getting Started on Natural Language Processing with Python. ACM Crossroads, 13 (4).

20. Dinakar, K. Reichart, R. Lieberman, H. (2011). Modeling the Detection of Textual Cyberbullying. International AAAI Conference on Web and Social Media.

21. Hinduja, S. Patchin, J. (2007). Offline Consequences of Online Victimization: School Violence and Delinquency. Journal of School Violence.

22. Sebastiani, F. (2002). Machine Learning in Automated Text Categorization. ACM Computing Surveys, 34(1), pp. 1–47.

23. Wahbeh, A. Al-Kabi, M. (2012). Comparative Assessment of the Performance of Three WEKA Text Classifiers Applied to Arabic Text. ABHATH AL-YARMOUK: "Basic Sci. & Eng.", 21(1), pp. 15- 28

24. Awad, W. Elseofi, S. (2011). Machine Learning Methods for Spam Email Classification. International Journal of Computer Science & Information Technology, 3(1).

25. Peersman, C. Daelemans, W. Vaerenbergh, L. (2011). Predicting Age and Gender in Online Social Network.

26. Lam, A. Paner, I. Macatangay, J. Delos Santos, D. (2014). Classifying Typhoon Related Tweets. 10th National Natural Language Processing Research Symposium.

27. Kwok, I. Wang, Y. (2013). Locate the Hate: Detecting Tweets against Blacks. Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence**.**

28. N.A (2012).Comparison of Machine Learning Algorithms for Oﬀensive Language Filtering. Jersey Shore Science Fair.

29. Dadvar, M. De Jong, F. Ordelman, R. Trieschnigg, D. (2012). Improved Cyberbullying Detection Using Gender Information.

30. Cheng, C. Ng, A. (2016). Automated Role Detection in Cyberbullying Incidents. Proceedings of the 16th Philippine Computing Science Congress.