

# **Detecting Cyberbullying in Social Media Text Data**

**by**

**QUINCY WAMBUI NJOROGÉ**

**(w1922671)**

Supervised by

NATALIA YERASHENIA

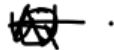
Submitted in partial fulfilment of the requirements of  
the School of Computer Science & Engineering  
of the University of Westminster  
for award of the Master of Science

SEPTEMBER 2023

## DECLARATION

I, *Quincy Wambui Njoroge*, declare that I am the sole author of this Project; that all references cited have been consulted; that I have conducted all work of which this is a record, and that the finished work lies within the prescribed word limits.

This has not previously been accepted as part of any other degree submission.


 .

Signed : .....

Date : 06-09-2023 .....

## FORM OF CONSENT

I, *Quincy Wambui Njoroge*, hereby consent that this Project, submitted in partial fulfilment of the requirements for the award of the MSc degree, if successful, may be made available in paper or electronic format for inter-library loan or photocopying (subject to the law of copyright), and that the title and abstract may be made available to outside organisations.

 .

Signed : .....

Date: 06-09-2023  
.....

## **Abstract**

The widespread adoption of social media, with over 3.96 billion users globally by July 2020, has led to a troubling surge in cyberbullying incidents, posing severe emotional and psychological harm. Our study aims to address this by enhancing cyberbullying detection through the combining of datasets from Instagram and Twitter. Using CRISP-DM methodology, we used Natural Language Processing to preprocess our datasets and applied the following machine learning algorithms: SVM, Naive Bayes, and Logistic Regression as well as ensemble methods (bagging). For Feature extraction, we utilised the Count vectorizer and TF-IDF techniques. The results of our study are as follows: for Twitter, Logistic Regression (LR) with count vectorizer showed the best results (precision 88%, F1, and recall 87%). Instagram's best model was LR with TF-IDF (precision, recall, and F1 at 81%). Upon combining datasets, SVM and LR with TF-IDF excelled on Twitter (precision 86%, F1 and recall 87%), while LR with count vectorizer performed well on Instagram (precision, recall, and F1 at 77%). Our study highlights the promising results achieved by combining datasets from various social media platforms, with comparable or improved performance compared to platform-specific data. Future research should focus on the growing need for cross-system user modelling to comprehensively understand cyberbullying in the interconnected world of multiple social media platforms. Additionally, they should consider expanding the range of social media platforms, employing advanced machine learning techniques, and proactively collecting new data to stay up-to-date with evolving online behaviours and language, ultimately enabling real-time prevention and intervention strategies.

## **Acknowledgements**

I am filled with profound gratitude towards Dr. Natalia Yerashenia, my supervisor, for her constant encouragement, invaluable guidance, constructive criticism, and the academic liberty she provided me with throughout my research journey. I would also like to express my heartfelt appreciation to my mother, Leah W. Kiura, whose financial support, unwavering encouragement, and the opportunities she provided made it possible for me to undertake this study. Furthermore, I extend my thanks to the committed members of the Computer Science Department, whose valuable assistance and collaboration played a pivotal role in ensuring the success of this thesis.

## Table of contents

|  |           |
|--|-----------|
| <b>Table of contents.....</b>                                    | <b>1</b>  |
| <b>Chapter 1: Introduction.....</b>                              | <b>3</b>  |
| 1.1 Background.....  | 3         |
| 1.2 Motivation.....  | 4         |
| 1.3 Problem Statement.....                                       | 4         |
| 1.4 Research Objectives.....                                     | 5         |
| <b>Chapter 2: Literature Review.....</b>                         | <b>6</b>  |
| 2.1 Defining Cyberbullying.....                                  | 6         |
| 2.2 Comparing Different Machine Learning Models.....             | 7         |
| 2.2.1 Support Vector Machines(SVMs).....                         | 7         |
| 2.2.2 Logistic Regression.....                                   | 9         |
| 2.2.3 Naive Bayes.....   | 10        |
| 2.2.4 Ensemble Models.....                                       | 11        |
| 2.2.4.1 Bagging Ensemble.....                                    | 11        |
| 2.2.4.2 Boosting Ensemble.....                                   | 12        |
| 2.3 Performance Measures.....                                    | 13        |
| 2.4 Machine Learning Techniques for Cyberbullying Detection..... | 15        |
| 2.5 Issues and Challenges.....                                   | 20        |
| <b>Chapter 3: Research Methodology.....</b>                      | <b>21</b> |
| 3.1 Research Resources.....                                      | 22        |
| 3.2 Data Collection.....   | 22        |
| Dataset 1: Twitter Dataset.....                                  | 23        |
| Dataset 2: Instagram Dataset.....                                | 23        |
| 3.3 Text Preprocessing.....                                      | 24        |
| 3.3.1 Expand Contractions.....                                   | 24        |
| 3.3.2 Spell check.....   | 25        |
| 3.3.3 Stop Word Removal.....                                     | 26        |
| 3.3.4 Tokenization.....  | 27        |
| 3.3.5 Lemmatization.....   | 28        |
| 3.3.6 Other Preprocessing Steps.....                             | 29        |
| 3.4 Feature Extraction.....                                      | 30        |
| 3.4.1 Bag of Words(BoW).....                                     | 30        |
| 3.4.2 N-grams.....   | 31        |
| 3.4.3 Term Frequency-Inverse Document Frequency(TF-IDF).....     | 31        |
| 3.5 Implementation.....  | 32        |
| 3.5.1 Selected Algorithms.....                                   | 32        |
| <b>Chapter 4: Research Findings.....</b>                         | <b>34</b> |
| 4.1 Methodology and Code Implementation.....                     | 34        |

|   |           |
|---|-----------|
| 4.2 Exploratory Data Analysis.....  | 36        |
| 4.3 Model Performance on Twitter Dataset.....                               | 43        |
| 4.4 Model Performance on Instagram Dataset.....                             | 44        |
| 4.5 Model Performance on Combined Dataset.....                              | 46        |
| 4.6 Ensemble Learning.....  | 48        |
| 4.6.1 Interpreting Ensemble Learning Performance for Twitter Dataset.....   | 50        |
| 4.6.2 Interpreting Ensemble Learning Performance for Instagram Dataset..... | 52        |
| 4.6.3 Interpreting Ensemble Learning Performance for Combined Dataset.....  | 54        |
| <b>Chapter 5: Conclusion and Future Work.....</b>                           | <b>55</b> |
| 5.1 Conclusion.....   | 55        |
| 5.2 Limitations and Challenges.....   | 56        |
| 5.3 Future Work.....  | 57        |
| <b>Chapter 6: References List.....</b>                                      | <b>59</b> |



## **Chapter 1: Introduction**

### **1.1 Background**

The impact of social media on our everyday routines has been nothing short of remarkable. What was once considered a simple means of communication has now evolved into a vital component of our daily existence. It is hard to imagine going a day without checking our feeds. Whether staying connected with loved ones, getting up-to-date news, sharing our experiences or promoting our businesses, social media has become an indispensable tool for modern-day living. Moreover, with the outbreak of the COVID-19 pandemic, the cyberspace has witnessed an unprecedented surge in traffic. According to Kemp's (2020) research, by the start of July 2020, the number of social media users had grown by 10% from the previous year, reaching a staggering total of 3.96 billion, which accounted for more than half of the world's population.

Despite the impressive growth and numerous benefits of social media, an alarming surge in cyberbullying has blemished the otherwise promising landscape, tarnishing the experience of its users. The COVID-19 pandemic has further exacerbated this, as highlighted by Light (2020), an organisation that monitors online harassment, hate speech between children and teenagers in online chats increased by 70% in just a few months into 2020. Interestingly, according to Hinduja and Patchin (2019), 95% of American teenagers are active online, with a significant majority accessing the internet through mobile devices. Consequently, this prevalence of online engagement has made it the primary medium for cyberbullying incidents. Patchin (2021) supports this observation, stating that 46% of students aged 13 to 17 have been victims of cyberbullying at some point in their lives. In a related study, Anderson (2018), as cited by the Pew Research Centre, discovered an even higher percentage of 59% among U.S. teens who reported instances of online bullying or harassment.

Cyberbullying is not limited to teenagers; adults are also significantly affected. A study conducted by Duggan (2014) revealed that 40% of adults have experienced online harassment. These statistics paint a distressing picture of both teenagers and adults being impacted by cyberbullying. One potential explanation for this phenomenon could be the widespread use of social media platforms, which often serve as the primary medium for cyberbullying incidents. The broad accessibility and popularity of these platforms create opportunities for bullies to target and harass vulnerable individuals, exposing them to emotional and psychological harm.

The detrimental effects of cyberbullying have similar effects to traditional bullying and can lead to depression, low self-esteem, and even suicide attempts, as shown by research conducted by Dehue, Bolman and Völlink (2008). Cyberbullying, unlike traditional bullying, occurs online and can persistently follow the victim around, happening at any time of day or night, every day of the week (NSPCC, 2016). Victims often find this kind of bullying distressing because it is hard to manage and many people can see it. In addition, online bullies often create fake accounts to remain anonymous, making it challenging to identify who is responsible for the harassment.

## **1.2 Motivation**

Bullying can have long-lasting effects that extend into adulthood. It can cause depression and in some cases, even lead to self-harm or suicide. Both mental and physical harm can result from bullying. This harmful behaviour can also contribute to an increase in violent behaviour and crime. Furthermore, it can cause victims to isolate themselves and develop physical or mental illnesses. There have been efforts to improve online child safety through initiatives like the "[Stop Bullying](#)" campaign and the [Cyberbullying Research Center](#). Despite these commendable efforts, undesirable content still exists online.

Although the internet has many benefits, it also has a dangerous side in the form of cyberbullying. This problem affects not only children but also adults, with 40% of adults in the U.S. experiencing online harassment, according to Duggan (2014). As previously mentioned, 46% of students aged 13 to 17 have also faced cyberbullying at some point. To prevent negative outcomes, we need to be able to identify harmful messages effectively. However, with the overwhelming amount of information available online, we require advanced algorithms to detect potential risks automatically. This is why we focus on detecting bullying across various social media platforms, empowering individuals to take proactive steps in combating this issue.

## **1.3 Problem Statement**

It is challenging to detect cyberbullying on social media networks and current research has gaps that need to be addressed. One such gap is the potential for combining datasets from multiple platforms to create a more comprehensive and reliable model. Many existing research studies have only focused on training data from one social media platform. However, by gathering data from various platforms, we can capture a broader range of cyberbullying behaviours and patterns, making the detection system more effective.

It is also essential to test the accuracy and performance of the model using platform-specific test sets. Each social media platform has unique dynamics, so testing the model on individual test sets specific to each platform can provide valuable insights into its effectiveness across diverse online environments. Addressing these research gaps will help develop more reliable and effective cyberbullying detection systems and we will be able to understand which models work well with which social media platforms as well as assess the generalisation of the models when trained on combined datasets compared to individual platform datasets.

## **1.4 Research Objectives**

To define our project's objectives, we will use Bloom's Taxonomy, a framework for educational objectives. Bloom's Taxonomy provides a guide for developing objectives that encompass different levels of thinking (Krathwohl, 2002). Guided by this framework, we have formulated the subsequent objectives aimed at enhancing our understanding of cyberbullying detection techniques and their practical application. These objectives include:

1. Conduct a comprehensive literature review to understand the various facets of cyberbullying, its forms, and the existing research within social networks.
2. Develop a reliable cyberbullying detection model using supervised machine learning algorithms on a combined dataset from two social media platforms.
3. Evaluate model performance on distinct test datasets from each platform, assessing generalisation and success across diverse platforms.
4. Explore ensemble learning techniques to develop an ensemble model for cyberbullying detection, leveraging the strengths of different models to enhance overall performance.

## **Chapter 2: Literature Review**

This chapter explores cyberbullying by analysing previous research studies. The chapter covers the various definitions of cyberbullying established over time and explores the research conducted on the topic within social networks. This includes the detection techniques and performance measures that have been utilised. Additionally, the chapter discusses the limitations researchers face when applying machine learning to identify instances of cyberbullying on online social networks.

### **2.1 Defining Cyberbullying**

Bullying is commonly defined as repeated, aggressive behaviour by individuals or groups against a vulnerable victim (Smith et al., 2008). In today's digital era, bullying has evolved into cyberbullying, which makes use of digital tools such as social media, mobile phones, gaming platforms and messaging apps to harm, humiliate or provoke those who are targeted (UNICEF, 2023). Contrary to a simplistic view that equates cyberbullying with a mere transition from traditional face-to-face bullying to online contexts (Fulantelli et al., 2022), it is crucial to acknowledge that cyberbullying entails intricate behaviours. The distinctive characteristics of social media platforms prompt a re-interpretation of aggression, repetition and power imbalance, necessitating a deeper understanding of the complexities associated with this form of harassment (Whittaker and Kowalski, 2014). When it comes to cyberbullying, the way aggression is expressed and how it plays out differs from traditional bullying in person. In-person bullying is often easy to recognise through aggressive body language, physical violence and changes in tone and facial expressions. In contrast, online aggression presents itself differently, with verbal abuse, humiliation, threats, intimidation, and even impersonation being prevalent. Online repetition can come in multiple forms. It may involve a perpetrator who repeatedly sends abusive messages or makes derogatory comments to target a victim for an extended period. Another type of repetition is the viral spread of abusive content, where other users like, share and circulate it across several platforms. These repetitive actions have a severe negative impact on the victim, causing prolonged suffering. In traditional bullying, power imbalances often manifest through physical strength or social status. However, in cyberbullying, power imbalances can be seen through differences in technological expertise (Whittaker and Kowalski, 2014). For instance, a bully with advanced technological skills can use their knowledge to engage in malicious activities like hacking into the victim's accounts or using sophisticated techniques to avoid detection. This exacerbates the power imbalance between the bully and the victim.

In their research on cyberbullying perpetration and victimisation, Lee, Abell and Holmes (2015) adopted a classification approach that categorises cyberbullying based on different modes. Through the development and validation of two scales, they identified three distinct subscales within each scale: verbal/written bullying, visual/sexual bullying and social exclusion. In contrast, Willard's (2007) book introduces a classification of cyberbullying that focuses on distinct types regardless of mode. This classification identifies eight types of cyberbullying, including flaming, harassment, denigration, impersonation, outing, trickery, exclusion and cyberstalking. Recent studies have identified new forms of cyberbullying, including the use of memes (Jaiswal, 2021; Nandi et al., 2022) and revenge porn (Jaiswal, 2021; Ehman and Gross, 2019; Kamal and Newman, 2016). Advances in technology, such as the emergence of deepfakes, have led to an increase in image manipulation and exploitation, allowing for the creation of deceptive and fabricated explicit material (Hinduja, 2021). Cybermobbing is also a recent form of cyberbullying that involves a group of individuals collectively engaging in aggressive and harmful behaviour towards a targeted individual (Seeker, 2015). Despite being insufficiently studied, it can be precarious as it can cause the victim to feel distressed and isolated, overwhelmed by the belief that everyone is against them.

## **2.2 Comparing Different Machine Learning Models**

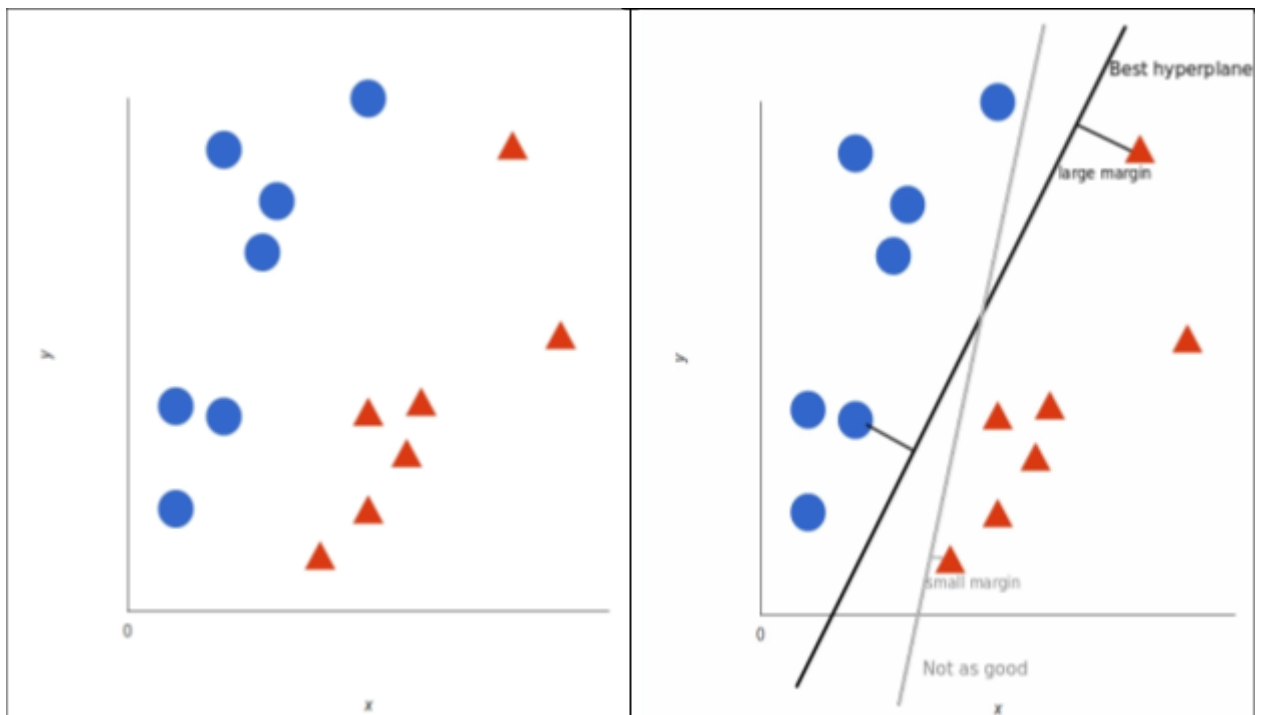
In this section, we focus on comparing several machine learning algorithms we intend to use for our study.

### **2.2.1 Support Vector Machines(SVMs)**

SVMs are supervised machine learning models that are widely used in text classification (Joachims, 1998). SVMs use nonlinear mapping to transform training data into a higher dimension. In the new dimension, the SVM searches for the optimal linear separating hyperplane, which is commonly known as the "decision boundary". This boundary separates the tuples of one class from another. The SVM identifies the hyperplane by analysing the support vectors and margins (Han, 2011). SVMs are highly accurate compared to other algorithms due to their capacity to model complex nonlinear decision boundaries (Han, 2011). This makes it suitable for text classification, as it is especially good when you only have a small collection of a few thousand labelled examples to learn from.

To gain a better understanding of SVMs, let us explore how they work with an example. Suppose we have a dataset with two classes that can be separated linearly: texts with cyberbullying

(represented by blue) and texts without cyberbullying (represented by red). Additionally, our dataset has two input features,  $x$  and  $y$ . With this input data, the SVM classifier will plot the data on a plane and then produce a hyperplane as its output. In this case, the hyperplane will be a line because it is 2-dimensional. The line serves as the decision boundary that optimally separates each class (i.e., blue and red), resulting in their classification. For SVM, the best hyperplane is the one that will have the minimum classification error (Han, 2011).



*Image 2.1: Mapped data points on a plane*

SVM searches for the best hyperplane to separate data into different classes. It does this by finding the hyperplane with the maximum margin, which is the distance between the hyperplane and the nearest data points of each class. These closest data points are called "support vectors" and they help SVM define the margin and effectively separate the data. The hyperplane with the largest margin is considered the best classifier because it can generalise well with new, unseen data points. A larger margin means a clearer separation between the classes, resulting in better classification (Han, 2011).

### 2.2.2 Logistic Regression

Logistic regression is a supervised machine learning algorithm that is used for binary classification tasks. The logistic regression model was first developed in the 20th century within the field of biological sciences(Berkson, 1944). Logistic regression always produces output values that fall between the range of 0 and 1 making it suitable for binary classification. It is used to analyse the relationship between a set of independent variables and the dependent binary variables. For example, if a text is cyberbullying or not.

Logistic regression involves identifying a linear relationship between the features of the data and the dependent variable. It differs from linear regression in that it does not predict the outcome itself, but rather the log odds of the outcome. Log odds represent the likelihood of an outcome occurring. The formula is displayed below;

$$\log(\text{odds}) = \beta_0 + \beta_1x_1 + \beta_2x_2 + \dots + \beta_nx_n$$

The constant term in the regression equation above is represented by  $\beta_0$ , while the coefficients of the independent variables are represented by  $\beta_1, \beta_2, \dots, \beta_n$ . The independent variables themselves are denoted as  $x_1, x_2, \dots, x_n$ . In order to determine the likelihood of a certain outcome, the log odds are converted using the logistic function. This is accomplished through the following equation:

$$p = 1 / (1 + e^{(-\log(\text{odds}))})$$

'p' represents the probability.

The logistic function is a sigmoid function that compresses the log odds to a range of 0 to 1. This enables the logistic regression model to estimate probabilities that fall within the 0 to 1 range, making it appropriate for binary outcomes. After training the logistic regression model, it becomes possible to predict the probability of the outcome for new data points. To do this the values of the independent variables are input into the model and used to calculate the predicted probability.

### 2.2.3 Naive Bayes

Naive Bayes is an algorithm that uses the Bayes theorem and probability theory to make predictions. The term "naive" is derived from the assumption that an attribute's effect on a class is independent of other attributes (Han, 2011). Despite its simplicity, Naive Bayes has demonstrated efficiency and good performance in many real-world applications making it an ideal model for text classification.

Naive Bayes method uses Bayes theorem, which calculates the probability of an event happening based on previous knowledge of another event (Ray, 2019). The formula for Bayes' theorem is as follows:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

*Image 2.2: Bayes Theorem Formula*

When classifying data, A represents the label for the class, while B represents the attributes that are associated with the input data point. The objective is to identify the most probable class label based on the observed attributes. For example, the formula for a binary classification problem (two classes, C1 and C2) is as follows:

$$P(C_1|B) = \frac{P(B|C_1) \cdot P(C_1)}{P(B)}$$
$$P(C_2|B) = \frac{P(B|C_2) \cdot P(C_2)}{P(B)}$$



*Image 2.3 Example of a binary classification formula using Naive Bayes*

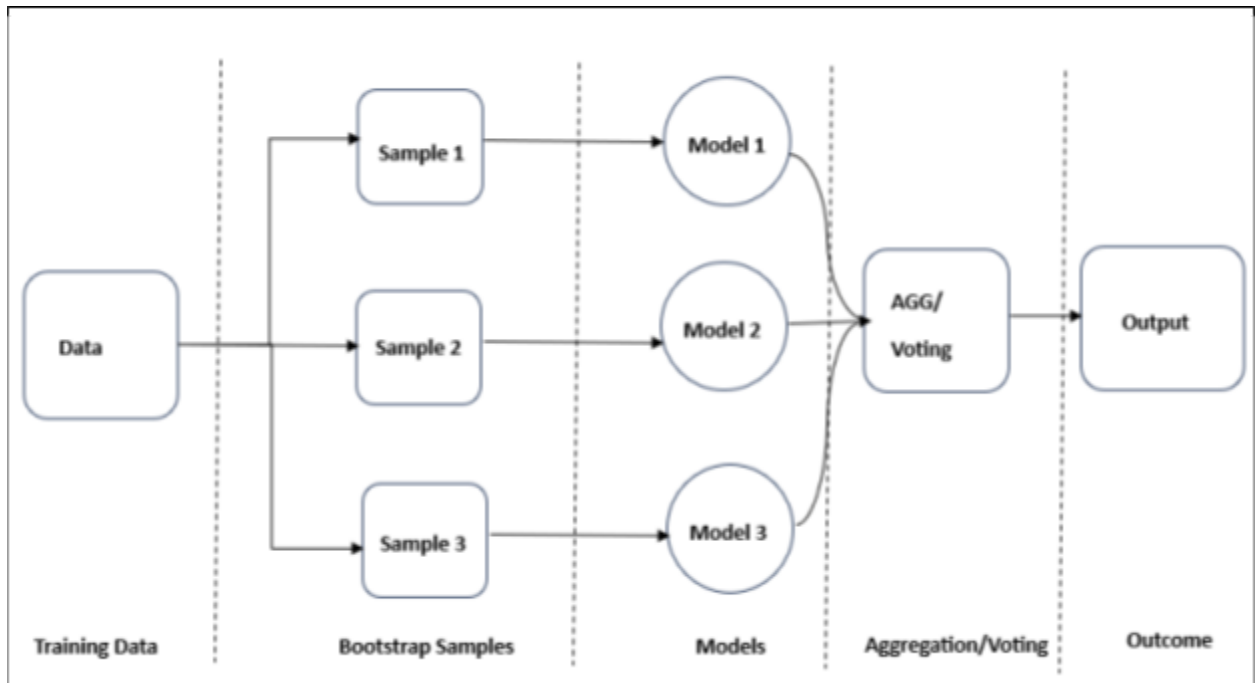
$P(C1)$  and  $P(C2)$  are the prior probabilities of classes  $C1$  and  $C2$ , respectively, while  $P(B|C1)$  and  $P(B|C2)$  are the likelihoods of detecting the features  $B$  given each class.

## **2.2.4 Ensemble Models**

Ensemble learning is an emerging field in machine learning. It involves the integration of several base learners as ensemble members to collaboratively generate predictions that are then consolidated into a singular output. This technique aims to improve their performance on average and enhance their generalisability (Yang and Yang, 2017; Scikit-learn.org, 2012). Ensemble learning can be traced back to 1965 when Nilsson introduced the concept as a means to address classification tasks in supervised learning. The idea is to create a robust classifier by having multiple algorithms work together to overcome individual weaknesses. A study by Hansen and Salamon (1990) has shown that using an ensemble of neural networks provides better performance than using a single one. As highlighted in (Opitz and Maclin, 1999), Bagging and Boosting are the two main types of ensemble learning methods.

### **2.2.4.1 Bagging Ensemble**

The Bootstrap aggregation also referred to as “Bagging” was first introduced by Breiman (1996). Bagging involves three main steps - bootstrapping, parallel training, and aggregation. During the bagging process, a bootstrapping sampling technique is used to extract “n” subsets from the training set by randomly selecting instances and allowing for replacement, which means the same instance can be selected multiple times. These subsets are then independently and parallelly trained on base learners. The output is then averaged for regression tasks or voted for classification tasks. It is worth mentioning that this ensemble adopts a parallel architecture for its ensemble system.



*Image 2.4: Bagging architecture*

#### **2.2.4.2 Boosting Ensemble**

Schapire (1990) conducted a study addressing the issue of enhancing the accuracy of a hypothesis produced by a learning algorithm in the distribution-free (PAC) learning model. The study proposes a method to convert a weak learning algorithm into one that can achieve high accuracy. The research demonstrates that a learning model, which requires the learner to perform slightly better than guessing, is as potent as a model where the learner's error can be reduced to any level. This is referred to as boosting. Similar to bagging, a random sample of data is chosen for training but in this case, the process is sequential. Each model attempts to address the weaknesses of the previous model. As the process iterates, the weak learners are combined to create a strong learner that can produce more accurate predictions. It is worth mentioning that this ensemble adopts a cascade architecture for its ensemble system.

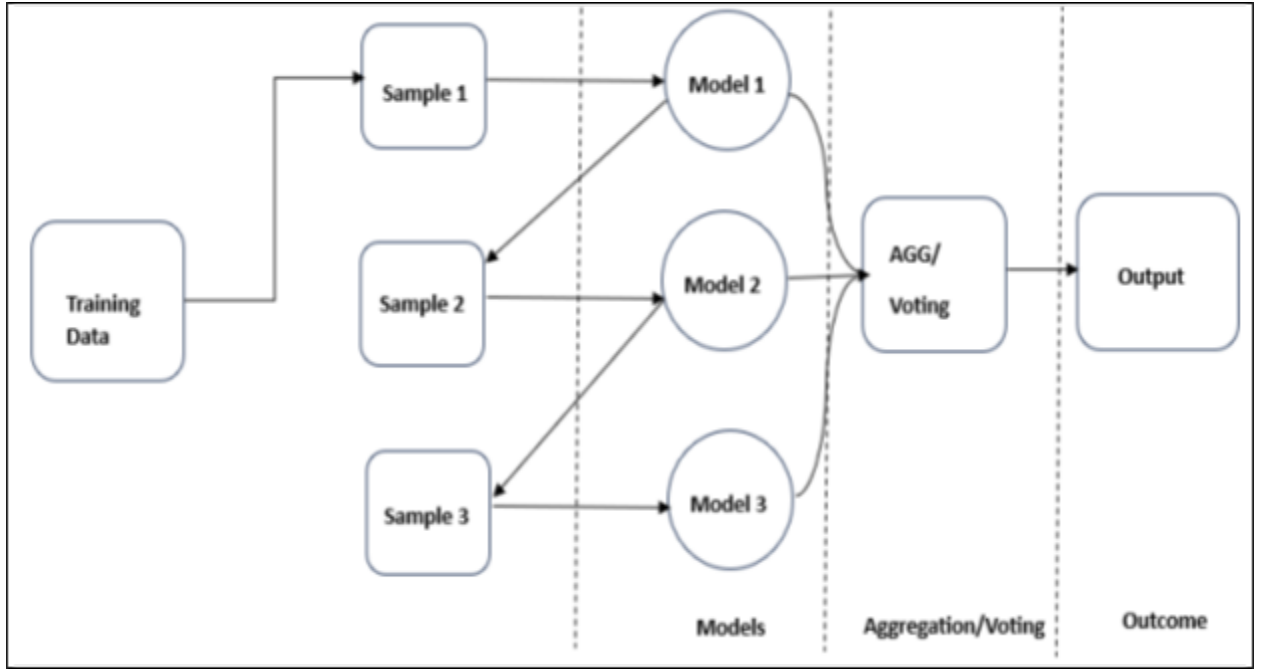


Image 2.5: Boosting Architecture

## 2.3 Performance Measures

After training models, it is important to estimate how accurate the classifier or regression model is by testing it on unseen data and evaluating using different measures. In our case of cyberbullying detection, which is a classification problem, various metrics are employed. In this section, we will present measures for assessing how good a classifier is at predicting class labels.

**Accuracy:** The accuracy of a model is calculated by dividing the number of correct predictions by the total number of predictions made (Puthenveedu, 2022). This is shown in the following equation;

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

where TP is True Positive, TN is True Negative, FP is False Positive, and FN is False Negative.

**Precision:** Precision measures the ratio of true positives to all predicted positives, helping to determine relevant data within a class (Muneer & Fati, 2020; Puthenveedu, 2022). The equation is as shown below;

$$Precision = \frac{TP}{TP+FP}$$

where TP is True Positive and FP is False Positive.

**Recall:** Recall is the proportion of correctly predicted Real Positive cases(Puthenveedu, 2022). It is calculated as follows:

$$Recall = \frac{TP}{TP+FN}$$

where TP is True Positive and FN is False Negative.

**F-score:** By combining precision and recall values, we can obtain a weighted average that captures the properties of both metrics (Muneer & Fati, 2020; Puthenveedu, 2022), resulting in an effective metric. The formula is shown below;

$$F1\ Score = \frac{2*Recall*Precision}{Recall+Precision}$$

The confusion matrix is also frequently used. The confusion matrix, as shown in *Image 2.6*, provides a visual representation that effectively displays the performance of a classification model.

|                          | Predicted cyberbullying | Predicted Non-cyberbullying |
|--------------------------|-------------------------|-----------------------------|
| Actual cyberbullying     | True Positive (TP)      | False Negative (FN)         |
| Actual non-cyberbullying | False Positive (FP)     | True Negative (TN)          |

True Positive (TP) refers to accurately identifying and classifying instances as "cyberbullying" when they indeed belong to the cyberbullying category.

True Negative (TN) refers to correctly identifying and classifying instances as "non-cyberbullying" when they truly belong to the non-cyberbullying category.

False Negative (FN) occurs when instances that are actually "cyberbullying" are mistakenly classified as "non-cyberbullying."

False Positive (FP) happens when instances that are genuinely "non-cyberbullying" are incorrectly classified as "cyberbullying."

*Image 2.6: Confusion matrix*

When working with balanced datasets, where the class distribution is relatively equal or slightly skewed, accuracy can be considered a reliable metric. However, in scenarios where the class distribution is heavily imbalanced, accuracy becomes an unreliable measure of performance. For instance, let us consider a dataset where only 10% of the posts exhibit cyberbullying behaviour. Despite misclassifying all posts as non-bullying, a classification model would still attain a 90% accuracy, which may appear impressive compared to some reported results in the field. In such cases, alternative metrics such as F1 score, precision and recall are commonly used to evaluate performance on unbalanced datasets. In some of the studies we explored, there is ambiguity surrounding the term 'accuracy,' and it is uncertain whether it is being used specifically to denote overall accuracy or if it encompasses precision, recall, or other related metrics.

## 2.4 Machine Learning Techniques for Cyberbullying Detection

In this section, we get into an in-depth exploration of various effective and efficient machine learning techniques employed for cyberbullying detection.

With the rapid expansion of online content, it is increasingly challenging for human reviewers to effectively identify and report instances of cyberbullying on social media platforms. However, machine learning presents a viable solution to this challenge. The identification of cyberbullying is usually framed as a classification problem (Salawu, He and Lumsden, 2020) and it requires the use of a dataset that can either be labelled or unlabeled. Supervised machine learning methods are commonly employed when working with labelled datasets, while unsupervised machine learning techniques are used when handling unlabeled datasets. According to Bonaccorso (2019), supervised learning uses a training set and a testing set to label a sample. The algorithm is based on the concept of a "hidden teacher" that provides immediate and precise feedback after each prediction. By analysing patterns and relationships in the training data, the algorithm can effectively classify data instances and assess its performance on the testing set. Conversely, in an unsupervised scenario, the absence of a "hidden teacher" necessitates the model to independently learn and identify patterns within the data, without any external guidance or labels (Bonaccorso, 2019). Given unlabeled data, unsupervised algorithms try to identify patterns and cluster similar data instances into cohesive groups (Lee, 2019).

Researchers have proposed numerous automated approaches to detect cyberbullying, showcasing their ability to analyse and accurately identify instances of online harassment across diverse social platforms like Twitter, Instagram and YouTube. An instance of this is in 2017, Noviantho et al. created a classification model using Support Vector Machine (SVM) and Naive Bayes. The model was based on a dataset of 1600 Formspring.me textual conversations obtained from Kaggle, labelled with Question, Answer and Severity fields. The authors eliminated conversations with less than 15 characters and meaningless words like "haha", "hehe", "emm" and "umm" from the dataset, leaving them with 12,729 observations. Out of these, 11,661 were labelled as non-cyberbullying and 1,068 as cyberbullying. The researchers employed a data-balancing approach to address the class imbalance, considering three different scenarios with two (cyberbully and non-cyberbully), four (non-cyberbully, cyberbully with low severity, cyberbully with middle severity, and cyberbully with high severity), and eleven (non-cyberbully and cyberbully with severity levels ranging from 1 to 10) classes, respectively. Naive Bayes and SVM with linear, poly, RBF and sigmoid kernels were used for classification. The study found that SVM with a polynomial kernel had the highest average accuracy of 97.11%, while Naive Bayes had an average accuracy of 92.81%. The researchers also noted that SVM with a polynomial kernel outperformed other SVM kernels and Naive Bayes.

Similarly, Miftah Andriansyah et al. (2017) conducted a study utilising text classification through Support Vector Machines (SVM) on a dataset extracted from Instagram. The researchers focused on collecting comments from two prominent Indonesian celebrity accounts known for their controversial nature. A total of 1053 comments were gathered, which served as the training data for their analysis. Given the Indonesian context, the collected comments exhibited noise in the form of emoticons and various languages. To address this, the researchers standardised the comments, including outlining abbreviations and omitting emoticons. They then manually classified the comments, determining whether they fell into the category of bullying or not. To classify the data, the researchers employed an SVM model. To evaluate the model's performance, they conducted tests using a separate set of 34 randomly selected comments. This test set was also obtained from the existing comments on Instagram. The SVM model was able to classify with an accuracy of 79.41%

To fight against aggressive behaviour on social media, Al-Garadi et al. (2019) implemented a model to reduce cyberbullying. They used data from multiple sources, including Wikipedia, YouTube, Twitter, and Instagram, to develop a model employing various machine learning algorithms. They conducted a comparative analysis of these algorithms, specifically SVM, K-Nearest Neighbour, Random Forest and Decision Trees. After a thorough evaluation, the researchers concluded that SVM outperformed the other three machine learning models.

Researchers have dedicated significant efforts to understanding the nuances between offensive and hateful content prevalent online. Two studies, conducted by Gaydhani et al. (2018) and Watanabe, Bouazizi, and Ohtsuki (2018), share a common goal of automatically classifying tweets into distinct classes based on their content. While both studies used similar datasets and tackled the issue of classification, they adopted different approaches and presented varying findings. Gaydhani et al. (2018) focused on classifying tweets into three categories: hateful, offensive, and clean. They combined datasets from Crowdfunder and GitHub, with the former encompassing three classes ("Hateful," "Offensive," and "Clean") and the latter consisting of "Sexism," "Racism," and "Neither" categories. Employing feature extraction techniques, such as n-grams and term frequency-inverse document frequency (TFIDF), they trained multiple machine learning models. Notably, logistic regression emerged as the most effective model, achieving an accuracy of 95.6%. On the other hand, Watanabe, Bouazizi, and Ohtsuki (2018) leveraged similar datasets to tackle the detection of hate speech on Twitter. They combined three datasets, including one from Crowdfunder and another previously used by Zeerak Talat and Hovy (2016).

Their focus was on differentiating between offensive and non-offensive tweets, as well as classifying tweets into hateful, offensive or clean classes. To achieve this, they employed a range of classification techniques and identified various features such as sentiment-based, semantic, unigram, and pattern-based features. Their model achieved an accuracy of 87.4% for binary classification and 78.4% for ternary classification. While both studies shared the goal of automatic classification, Gaydhani et al. (2018) primarily focused on three classes hateful, offensive, and clean and achieved higher accuracy using logistic regression, whereas Watanabe, Bouazizi, and Ohtsuki (2018) expanded their analysis to differentiate between offensive and non-offensive tweets and used a broader range of features.

Nurrahmi and Nurjanah (2018) applied SVM and KNN to detect cyberbullying on Twitter data. They followed a specific process to collect and analyse the data. Initially, they devised a labelling system to identify text containing explicit harassment. They referred to eight rules:

1. Counting the number of bad words in a tweet.
2. Counting the number of words showing negative emotion.
3. Counting the number of words showing positive emotion.
4. Looking for combinations of first-person pronouns, negative emotion, and second-person pronouns as indicators of cyberbullying.
5. Identifying combinations of second-person pronouns with bad words as indicators of cyberbullying.
6. Identifying combinations of first-person pronouns, words showing negative emotion, and third-person pronouns or proper nouns as indicators of cyberbullying.
7. Identifying combinations of third-person pronouns or proper nouns with bad words as indicators of cyberbullying.
8. Identifying combinations of links, bad words, and pronouns as indicators of cyberbullying

Once the rules were established, the researchers collected data from Twitter related to a controversial post. Since the data was unlabeled, they implemented a crowdsourcing approach involving Indonesian participants who volunteered to label each piece of data. To prepare the data for analysis, they performed preprocessing tasks such as removing numbers, symbols, and mentions and tokenizing the data. After the preprocessing stage, the researchers obtained a dataset consisting of 301 cyberbullying tweets, 399 non-cyberbullying tweets, 2,053 words expressing negative emotion, and 129 swear words. To detect cyberbullying, they applied two



algorithms: Support Vector Machine (SVM) and K-Nearest Neighbors (KNN). The evaluation of the results revealed that the SVM algorithm achieved the highest F1 score, which was 67%.

Sahay, Singh, Khaira, and Kukreja (2018) conducted a study focusing on the classification of cyberbullying instances in text. Their objective was to identify features that could effectively distinguish bullying comments from non-bullying comments. The researchers utilised datasets from Twitter and YouTube, resulting in a total of 6,594 data samples, with 3,947 samples from YouTube and 2,647 samples from Twitter. To prepare the data for analysis, the researchers performed preprocessing tasks and employed feature engineering techniques. They created two types of features: count vector features and TF-IDF (Term Frequency-Inverse Document Frequency) features. These features were designed to capture the relevant characteristics of the text data. For the classification, the researchers evaluated multiple machine learning algorithms, including Support Vector Machine (SVM), Logistic Regression, Gradient Boosting, and Random Forest. After conducting their experiments, they found that Support Vector Machine and Gradient Boosting performed better than the other methods in accurately classifying instances of cyberbullying.

Mangaonkar, Hayrapetian, and Raje (2015) adopted a unique approach to improve cyberbullying detection by applying collaborative computing principles. Their study demonstrated that this approach is more accurate and efficient compared to the standalone method. They collected datasets from Twitter and web pages and manually labelled them as either bullying or non-bullying. They then created two datasets. The first dataset was balanced, containing 170 bullying tweets and 170 non-bullying tweets. The second dataset was imbalanced, with 177 bullying tweets and 1,163 non-bullying tweets. The use of these different datasets aimed to evaluate the algorithm's performance on varied data sets, with the imbalanced dataset reflecting real-life tweet distributions encountered during training. The researchers applied Naive Bayes, Support Vector Machine, and Logistic Regression models to the data. For the balanced dataset, Logistic Regression performed better, with precision and recall greater than 60% using both bigram and word tokenizer. With the imbalanced dataset, Logistic Regression achieved more than 30% correct predictions. Additionally, they explored collaborative techniques such as AND parallelism, OR parallelism and Random 2 OR parallelism to examine improvements in precision, accuracy and recall. OR parallelism yielded the best recall results, while AND parallelism achieved the highest accuracy. This research highlights the positive impact of collaborative techniques in reducing time and enhancing detection accuracy. It is worth noting that the models

were not tuned, so there is potential for improved performance with further adjustments. SVM performed poorly in this context, although previous research has shown that it can perform well, suggesting that tuning the model could yield better results.

## **2.5 Issues and Challenges**

In this section, we will discuss some of the challenges associated with detecting cyberbullying on social media data using machine learning techniques.

One significant challenge that complicates the detection of cyberbullying is the influence of cultural factors and the constantly evolving nature of cyberbullying. What constituted cyberbullying in the past may not be relevant today, especially with the advent of social media (Al-Garadi et al., 2019). Machine learning algorithms need diverse and representative data examples to be effective, which requires collaboration across disciplines to develop culturally diverse examples (Al-Garadi et al., 2019).

Another issue in detecting cyberbullying on social media is the changing nature of language. Language is continually evolving, especially among younger generations, with new slang and abbreviations regularly being incorporated into the way we converse online (Al-Garadi et al., 2019). Researchers should aim to develop algorithms that effectively identify cyberbullying behaviour by detecting and understanding new slang. Furthermore, they should continuously update the training processes of machine learning algorithms to incorporate newly introduced words. This will maintain the relevance and accuracy of the models in identifying instances of cyberbullying on social media.

The subjective nature of human data analysis also poses a significant challenge in building effective prediction models for human behaviour, specifically in the context of cyberbullying detection using machine learning. Subjectivity emerges during the labelling and feature engineering processes, impacting the accuracy and objectivity of the models. Furthermore, the dynamic and non-generic nature of big data makes it difficult to understand and preserve context within machine learning models. Adapting to changes in human behaviour is critical, necessitating the constant updating of prediction models to incorporate changes in cyberbullying techniques used by internet users (Al-Garadi et al., 2019; Boyd and Crawford, 2012).

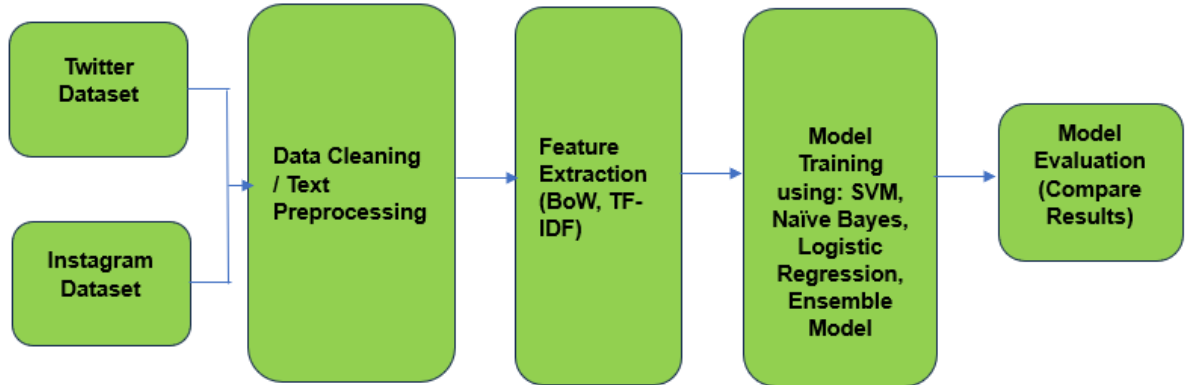
### **Chapter 3: Research Methodology**

In this chapter, we will explain the research approach we plan to use for identifying cyberbullying on social media through machine learning techniques. Drawing upon the insights from the literature review, we found that the most common method for addressing this problem involves utilising Natural Language Processing (NLP) and Machine Learning (ML). In the first step, the cyberbullying dataset is collected and preprocessed using NLP techniques. This is important to ensure the data is properly prepared for training machine learning algorithms. Next, the preprocessed dataset is used to train the machine learning algorithms, enabling them to detect bullying text on social media.

The research methodology has been structured to effectively accomplish the research goals by drawing inspiration from the CRISP-DM framework. CRISP-DM was initially introduced in the 1990s (Martinez-Plumed et al., 2020). This framework outlines a series of steps in the data mining process which we will follow as highlighted below:

1. Understanding the business problem which we have done above in Chapter 1 and Chapter 2 by identifying the specific business problem related to cyberbullying on social media and defining clear objectives.
2. Comprehending the data by gathering and assessing relevant social media text data and understanding the characteristics of cyberbullying content and related metadata.
3. Preparing the data through text data preprocessing and exploring ways to handle imbalanced data.
4. Developing machine learning models suitable for text classification and evaluating the models' performance using appropriate metrics.

The sections that follow will expound more into a comprehensive explanation of each stage of the CRISP-DM process that we will be using. This will offer a thorough insight into the methodology we are employing for this research.



*Image 3.1: Layers of Cyberbullying Detection Process*

### 3.1 Research Resources

In order to make our research more efficient, we will use Python as our primary programming language. Python is an open-source language with extensive libraries that will provide us with a strong foundation for our work. One of the libraries we will rely on the most is Sklearn (Scikit-Learn), which is a comprehensive machine-learning library for Python. Sklearn offers us a variety of tools and algorithms for supervised machine-learning tasks, which will make it easier for us to build and evaluate predictive models. Its user-friendly API and comprehensive documentation will help us to implement our project. Additionally, Python has a range of Natural Language Processing (NLP) libraries which will be very helpful for our text preprocessing tasks. NLP is crucial to our research as it allows us to extract meaningful insights from textual data. Python's NLP libraries, such as NLTK (Natural Language Toolkit) will give us much needed tools for tasks such as tokenization, stop word removal, lemmatization and more. These libraries will automate many of the complex tasks involved in working with textual data, saving us time and ensuring the accuracy of our analyses.

### 3.2 Data Collection

To implement the detection of cyberbullying, a large training corpus dataset is required to give to the machine learning algorithms. We will collect data from different sources. For this study, we will use two datasets. Both datasets will be collected from reliable sources that have been carefully chosen to facilitate binary classification. The first dataset will be obtained from [IEEEDataPort](#), and it consists of Twitter data with multiple classes while the second dataset was directly collected

from the authors of a research study focusing on Instagram comments and was manually annotated by experts.

## Dataset 1: Twitter Dataset

The Twitter data which will be sourced from IEEEDataPort was authored by Wang, Fu and Lu (2020). It consists of around 47,000 tweets gathered from Twitter, each categorised into several classes like age, ethnicity, gender, not cyberbullying, other, and religion. However, our focus is only on identifying cyberbullying cases. Hence, we will convert the multiple categories into a simple binary classification of cyberbullying and non-cyberbullying.

| tweet_text   | cyberbullying_type  |
|--|---------------------|
| sofrir Bullying na escola, sair e minha mãe passou 1 ano pra perceber e me criticou meu Pai vive viajando e me perguntou sobre isso pe       | not_cyberbullying   |
| @PaulBalbas it's so random!  | not_cyberbullying   |
| And so are you, obviously. Please tell me why not ALL Black lives matter, since #BLM is not doing anything about that?! Can you name an      | ethnicity           |
| Girls' bully hell An Elsie's River High School teen tried to kill herself by taking an undisclosed amount of tablets last Monday àæto esc    | age                 |
| @josiehall beaut, happy birthday too xx  | not_cyberbullying   |
| Gay Filipino Comedian Apologises to News Executive for Rape Joke Gone Bad ... - http://tinyurl.com/ovhzqeu http://plurk.com/p/iodaux         | gender              |
| gay jokes i feel i can speak for myself bc i'm very Out at this point but yeah i feel u with rape jokes                                      | gender              |
| @Skawtnyc @athenahollow @twoscooters are not exactly on speaking terms.  | other_cyberbullying |
| @Vandaliser @sajid_fairooz @IsraeliRegime There was no Muslim golden age. Those states were always slave states.                             | religion            |
| @discerningmumin But the western world gave up slavery on it's own. No one forced them from the outside.                                     | religion            |
| #Bahrain :A Smart Little Kid #Saudi & #UAE :The Protective Parents #Qatar :The Selfish Bullying Kid Nxt Door #Iran :A Sick Retarded Crim     | other_cyberbullying |
| We believe gratitude is as practice! So, from all of us at Q Christian, thank you for giving your heart, presence, and advocacy to the missi | religion            |
| Petty ass idiots. RT @XXL: Kendrick's stretch mark bars have some people unhappy https://t.co/sQdmq0hs46                                     | other_cyberbullying |
| it's wild that all the girls who bullied me in high school think i might actually buy their pyramid scheme makeup today.                     | age                 |
| gay' as a slur, or make rape jokes, & who want to talk abt the 'friendzone' but are afraid of living outside the 'manbox'                    | gender              |
| Disagree with Christians attempting to argue & defend him as a moral agent or worse try to portray him as Christian. Even if some            | religion            |
| #MKR really stretching it out this year  | not_cyberbullying   |
| LOL, Christianity is the one establishing binary concepts, and the DIALECTIC was first developed by Fichte .Also be CAREFUL ur argument      | ethnicity           |
| girls on snapchat from high school be so quick to try to sell me their products like lmao weren't u the one that bullied and talked shit     | age                 |
| @fmorgan2k9 stacks upon stacks of oreilly books. what you can barely see is robot parts hanging down. :)                                     | other_cyberbullying |
| @MaxRilumenthal @mehdirhasan @tnr Mohammed approved in the Quran of sex slavery. ISIS is a firm believer in sex slavery.                     | religion            |

Image 3.2: Sample tweets from the Twitter dataset, showcasing the different classes present in the dataset

## Dataset 2: Instagram Dataset

The second dataset will be collected specifically for cyberbullying detection from Instagram comments. This dataset consists of 10,000 comments obtained directly from the authors of the research papers Chelmis and Yao (2019) and Yao, Chelmis, and Zois (2019). Each comment has been manually reviewed by ten experts who have classified whether it is harassing or not. The focus on harassing comments was chosen because they are common in various types of unwanted behaviour, such as cyber harassment and cyberbullying.

| text  | label             |
|---|-------------------|
| @suspiciouscranberry haha lkr (created at:2012-08-31 23:57:50)  | non-cyberbullying |
| Bloody fuckin animal! u rock sir! (created at:2012-12-13 06:41:55)  | non-cyberbullying |
| I luv u Jesus (created at:2014-08-14 14:55:59)  | non-cyberbullying |
| Fucking pathetic... (created at:2012-12-24 19:30:27)  | non-cyberbullying |
| Yah I did actually and i'm mature enough to admit that. How's your pillow or electric toothbrush in bed by the way? @marissaaaa_ (created at:2012-11-02 21:34:03) | non-cyberbullying |
| and plus i think noone knows the real story... The ACTUAL thing that happened... Yes people are saying he was minding his own buisness, and other e               | non-cyberbullying |
| @redbone687 LMAO at instapolic (created at:2012-11-02 21:34:03)   | non-cyberbullying |
| @_missbritto Haha true! God Bless you too! (created at:2013-01-10 22:01:09)   | non-cyberbullying |
| great ass video (created at:2013-02-15 19:31:44)  | non-cyberbullying |
| u representing the @ fo show..yo shit is all hits (created at:2013-04-29 04:30:46)  | non-cyberbullying |
| Dude... Xbox isn't a lifestyle... It's a hobby... Life isn't about sex and drugs. You are the most oblivious stupid minded fool I have ever met. Just stop fo     | non-cyberbullying |
| Im Jewish u @\$Shole (created at:2013-06-18 04:26:00)   | non-cyberbullying |
| Lol i know the chick wit the red w hat!! (created at:2012-10-28 06:45:09)   | non-cyberbullying |
| yo mama is so fat she jumped in the air and got stuck (created at:2013-03-15 08:04:18)  | non-cyberbullying |
| !!!!!! (created at:2012-09-09 16:27:59)   | non-cyberbullying |
| KMichelle and Rasheeda is great (created at:2013-03-15 01:57:36)  | non-cyberbullying |
| No no no BAD (created at:2012-08-14 02:29:47)   | non-cyberbullying |
| I can't with these two lol (created at:2012-12-07 03:39:00)   | non-cyberbullying |
| Granma :**( (created at:2013-01-29 07:16:12)  | non-cyberbullying |
| Cuteee (created at:2012-12-09 09:49:30)   | non-cyberbullying |
| @redbone687 u mad bitch cuz u look like shit them big ass African lips on ur face we kno u suck all the dick u probably suck dick for change u look               | cyberbullying     |
| you're so incredibly good looking (created at:2012-07-10 03:28:45)  | non-cyberbullying |

*Image 3.3: Sample Instagram comments from the dataset, highlighting instances of cyberbullying and non-cyberbullying comments as annotated by the experts*

Both datasets serve the purpose of cyberbullying detection, however, they differ in terms of platform, annotation methodology and data size. The Twitter data has a larger sample size and different labels, which means we have to process the data to make it binary. Instagram data is already binary and specifically focuses on cyberbullying and non-cyberbullying comments.

### 3.3 Text Preprocessing

When dealing with raw text data, especially from social media it often includes irrelevant information that can lower the accuracy and efficiency of the model. To address this issue, text preprocessing is necessary. By utilising NLP techniques, we will perform the following steps on our datasets:

#### 3.3.1 Expand Contractions

When we communicate on social media, we often use contractions, which can cause issues for NLP models as they lack clarity and context. Contractions are shortened words with an apostrophe, such as "shouldn't" instead of "should not." Expanding these contractions can help NLP models comprehend the text and make sense of it. It also promotes consistency in the text data.

Code:

```
✓ On ▶ # Expanding Contractions
import contractions
# Define a function to expand contractions
def expand_contractions(text):
    return contractions.fix(text)

# Apply the function
sentence = "I wouldn't have thought we'd be able to find such a great place, but I'm glad we did!"
print(expand_contractions(sentence))
```

Output:

I would not have thought we would be able to find such a great place, but I am glad we did!

### 3.3.2 Spell check

It is quite common to come across typographical errors in real-world text data, but correcting them is crucial for better model performance. To achieve this, we will be utilising Python's Text Blob library, as illustrated below.

Code:

```
✓ On ▶ # Spellcheck
from textblob import TextBlob

# Function to perform spell check on a text
def spell_check(text):
    blob = TextBlob(text)
    corrected_text = blob.correct()
    return str(corrected_text)

# Apply the function
sentence = "I caan't belive howw quikly the weather can chhange in this beutiful countrey."
print(spell_check(sentence))
```

Output:

I can't believe how quickly the weather can change in this beautiful country.

### 3.3.3 Stop Word Removal

In order to enhance the effectiveness of our model, it is important that we remove stop words from our dataset. Stopwords consist of commonly used words in a given language and they mainly consist of prepositions, pronouns and conjunctions. Because these stopwords provide insignificant information to the text, it is necessary to remove them to prioritise more important content. Removing the stop words cuts down the number of tokens utilised for training which leads to a reduction in training time and a smaller dataset.

In our research, we will be using English datasets, so we will remove English stop words. We will be using the Natural Language Toolkit (NLTK) library. The English stop words we will be removing from our dataset are shown below.

Code:



```
✓ 0s ▶ !pip install nltk

✓ 0s ▶ import nltk
from nltk.corpus import stopwords
import nltk
nltk.download('stopwords')
stop_words = stopwords.words('english')
print(stop_words)
```



Output:

```
['i', 'me', 'my', 'myself', 'we', 'our', 'ours', 'ourselves', 'you', "you're", "you've",  
"you'll", "you'd", 'your', 'yours', 'yourself', 'yourselves', 'he', 'him', 'his',  
'himself', 'she', "she's", 'her', 'hers', 'herself', 'it', "it's", 'its', 'itself', 'they',  
'them', 'their', 'theirs', 'themselves', 'what', 'which', 'who', 'whom', 'this', 'that',  
"that'll", 'these', 'those', 'am', 'is', 'are', 'was', 'were', 'be', 'been', 'being',  
'have', 'has', 'had', 'having', 'do', 'does', 'did', 'doing', 'a', 'an', 'the', 'and', 'but',  
'if', 'or', 'because', 'as', 'until', 'while', 'of', 'at', 'by', 'for', 'with', 'about', 'against',  
'between', 'into', 'through', 'during', 'before', 'after', 'above', 'below', 'to', 'from',  
'up', 'down', 'in', 'out', 'on', 'off', 'over', 'under', 'again', 'further', 'then', 'once',  
'here', 'there', 'when', 'where', 'why', 'how', 'all', 'any', 'both', 'each', 'few',  
'more', 'most', 'other', 'some', 'such', 'no', 'nor', 'not', 'only', 'own', 'same', 'so',  
'than', 'too', 'very', 's', 't', 'can', 'will', 'just', 'don', "don't", 'should', "should've",  
'now', 'd', 'll', 'm', 'o', 're', 've', 'y', 'ain', 'aren', "aren't", 'couldn', "couldn't",  
'didn', "didn't", 'doesn', "doesn't", 'hadn', "hadn't", 'hasn', "hasn't", 'haven',  
"haven't", 'isn', "isn't", 'ma', 'mightn', "mightn't", 'mustn', "mustn't", 'needn',  
"needn't", 'shan', "shan't", 'shouldn', "shouldn't", 'wasn', "wasn't", 'weren',  
"weren't", 'won', "won't", 'wouldn', "wouldn't"]
```

### 3.3.4 Tokenization

To help machine learning models understand sentences, we will use a process called tokenization on our data. This process breaks down text into individual units called tokens. Tokenization enables the models to see the structure of the text, so they can understand each word by itself and also how it fits into the larger text. This makes it easier for the model to grasp the meaning of sentences and improve its understanding of language. We will utilise the Natural Language Toolkit (NLTK) library. As shown below the sentences will be broken down by words.

Code:

```
✓ [14] # Apply Tokenisation
      nltk.download('punkt')
      from nltk.tokenize import sent_tokenize, word_tokenize

✓ [15] def text_to_tokens(text):
      tokens = []
      for sentence in sent_tokenize(text):
          for word in word_tokenize(sentence):
              tokens.append(word)
      return tokens

✓ [16] sentence = "Hope, is the only thing stronger than fear! #Hope"
      print(text_to_tokens(sentence))
```

Output:

```
['Hope', ',', 'is', 'the', 'only', 'thing', 'stronger', 'than', 'fear', '!', '#', 'Hope']
```

### 3.3.5 Lemmatization

Lemmatization transforms derived words into their root form and groups them together. For instance, both "Dances" and "Dancing" would be converted to "Dance," where "Dance" becomes the lemma. Although stemming and lemmatization share a common aim, they take different approaches. Stemming removes the last characters of a word, which can sometimes lead to misinterpretation, like "Caring" becoming "Car." In contrast, lemmatization takes into account the context and changes the word to its basic form, so "Caring" becomes "Care". Since lemmatization maintains context and produces more meaningful results, it is the chosen method for preprocessing our text data. Python's WordNetLemmatizer package will be used to apply this to our dataset. Below we can see an illustration of what will happen to our text.

Code:

```

nltk.download("wordnet")

def apply_lemmatization(tokens):
    keep = []
    lemmatizer = WordNetLemmatizer()
    pos_tags = nltk.pos_tag(tokens)
    for token, pos in pos_tags:
        pos_un = get_wordnet_pos(pos)
        keep.append(lemmatizer.lemmatize(token.lower(), pos=pos_un))
    return keep

def get_wordnet_pos(tag):
    if tag.startswith('J'):
        return 'a' # Adjective
    elif tag.startswith('V'):
        return 'v' # Verb
    elif tag.startswith('N'):
        return 'n' # Noun
    elif tag.startswith('R'):
        return 'r' # Adverb
    else:
        return 'n' # Default to noun if the POS tag is not one of the above

sentence = "Running, runners, and ran are different verb forms, but after lemmatization, they all become 'run'."

```

Sentence before lemmatization:

Running, runners, and ran are different verb forms, but after lemmatization, they all become 'run'.

After lemmatization:

run , runner , and ran be different verb form , but after lemmatization , they all become 'run ' .

### 3.3.6 Other Preprocessing Steps

When working with text data from social media, it is crucial to consider the additional messiness that comes with it. To address this, we intend to apply different text preprocessing methods. This will involve removing unwanted elements such as ASCII characters, numbers, hyperlinks, "@" mentions, punctuation, extra spaces, and emojis. Additionally, we will ensure consistency throughout the data by converting all text to lowercase. By taking these steps, we can obtain cleaner and more manageable text data that can be used for further analysis and modelling.

### 3.4 Feature Extraction

Machine learning algorithms can only take in and understand data in numerical format. In order to effectively model our text data, it is important to convert the unstructured text into a structured format, specifically numerical. This process is called feature extraction and it involves converting text data into numerical features, which serve as inputs for machine learning algorithms. TF-IDF, bag of words(Count Vectorization) and n-grams are three various methods of feature extraction that are mainly used when data is in text format.

#### 3.4.1 Bag of Words(BoW)

The Bag of Words (BoW) approach is a widely used and simple technique for extracting text features. This method involves creating a bag that includes all the unique words found in the text corpus, ignoring word order and only taking into account their frequency of use. Each document is represented as a vector, with individual elements corresponding to each word in the vocabulary. The value of each element represents the frequency of that word in the document. BoW is an effective way to transform unstructured text data into a structured format, which can be used for various machine learning algorithms to detect patterns and relationships in the data. This approach is useful for cyberbullying detection using social media text.

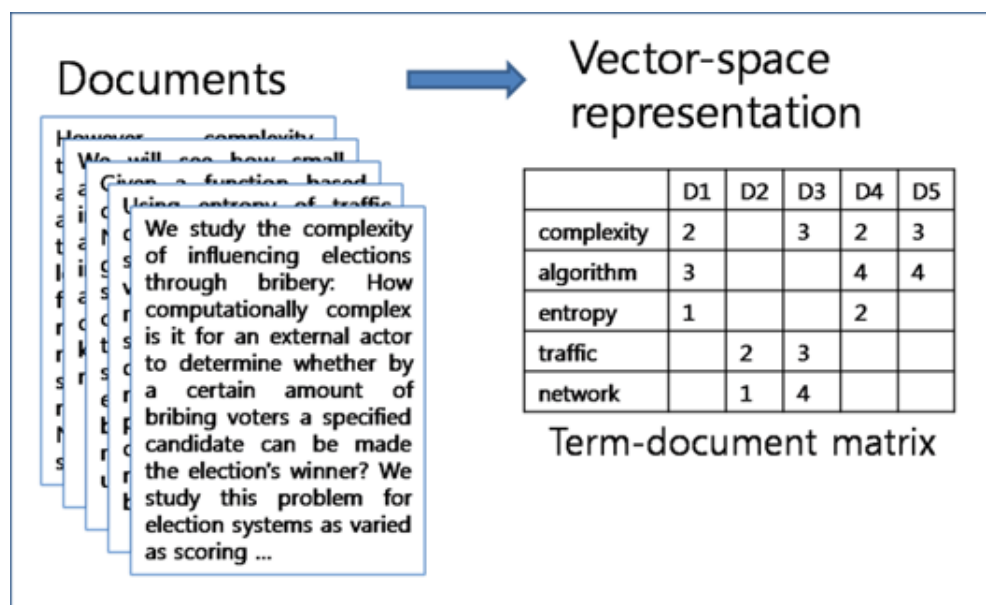


Image 3.4: Bag of words representation

### 3.4.2 N-grams

While the bag of words approach is useful, it overlooks the significance of word order in the text. To overcome this limitation, the n-grams technique is employed to extract n consecutive word sequences from the text. This technique captures more patterns and enables us to retain the context and semantic meaning present in the text. N-grams are available in various types such as bigrams (2-grams), which divide the text into pairs of adjacent words, and trigrams (3-grams), and beyond for higher N-values. N-grams are particularly advantageous in tasks where word order and context are critical.



Image 3.5: N-grams representation

### 3.4.3 Term Frequency-Inverse Document Frequency(TF-IDF)

TF-IDF is significant in understanding the importance of words in a document. It takes into consideration both the frequency of a word's appearance in a specific document (Term Frequency) and how rare it is across all documents (Inverse Document Frequency). By identifying words that are frequently used in one document but not often in all documents, TF-IDF assigns more weight to these words in their representation. This technique enables us to recognise significant and unique words in each document while downplaying the importance of common words. With TF-IDF, we can easily convert text data into numerical values that accurately indicate the importance of words in their respective contexts.

$$TF = \frac{\text{Number of times a word "X" appears in a Document}}{\text{Number of words present in a Document}}$$
$$IDF = \log \left( \frac{\text{Number of Documents present in a Corpus}}{\text{Number of Documents where word "X" has appeared}} \right)$$
$$TF\ IDF = TF * IDF$$

*Image 3.6: TF-IDF representation*

### **3.5 Implementation**

This section outlines the practical implementation of the cyberbullying detection algorithms that have been chosen for this study. The algorithms have been carefully selected based on their effectiveness and relevance, as established through the comprehensive review of relevant literature in Chapter 2.2.

#### **3.5.1 Selected Algorithms**

We will employ the following algorithms for the purpose of detecting cyberbullying instances within the analysed text data:

1. Support Vector Machine (SVM): As we have discussed in our literature review, using SVM in our approach to detect cyberbullying would be advantageous. SVM is capable of handling both linear and non-linear data, making it effective in distinguishing between cyberbullying and non-cyberbullying content within text data. This is achieved by finding the optimal hyperplane that maximises the margin between different classes, ultimately improving the strength of our detection model.
2. Logistic Regression: Logistic Regression will help us classify instances by estimating the probability of them belonging to a specific category. This information is useful in determining the likelihood of a text containing cyberbullying content. The simplicity and interpretability of this method allow us to understand how individual features contribute to

the classification result. With effective feature engineering and regularisation techniques, we can improve the accuracy of our detection mechanism.

3. **Naive Bayes:** Despite Naive Bayes assumption of feature independence, it remains extremely effective in text classification tasks, as it can capture the probabilistic relationships between words in a document. This is particularly advantageous for identifying specific word patterns and contextual cues that are indicative of cyberbullying behaviour. With the proper preparation and consideration of prior probabilities, Naive Bayes allows us to precisely flag cases of cyberbullying, enhancing our ability to take action against this harmful behaviour.

To perform our machine learning experiments, we will follow these steps:

1. **Twitter Dataset Testing:** First, we will train the machine learning algorithms using the Twitter dataset and evaluate their performance. This step allows us to assess how well the algorithms work with Twitter data.
2. **Instagram Dataset Testing:** Next, we will repeat the same process with the Instagram dataset. We will train the algorithms with Instagram data and evaluate their performance. This step will help us understand how the algorithms perform specifically with Instagram data.
3. **Combining Datasets:** After evaluating the algorithms separately on Twitter and Instagram data, we will combine the two datasets. This consolidation creates a single dataset that incorporates data from both social media platforms.
4. **Combined Dataset Testing:** With the combined dataset, we will train the algorithms once more, but this time we will test them on individual platform-specific test sets. This approach will allow us to determine how well the algorithms generalise across different platforms, ensuring accurate classifications for both Twitter and Instagram.

## **Ensemble Model Development**

While the algorithms mentioned above are useful on their own, we aim to explore enhancing their effectiveness by creating an ensemble model. By combining the results from SVM, logistic regression and Naive Bayes, we can develop a more reliable prediction for identifying cyberbullying. We will explore the application of a parallel ensemble for this task.

## **Chapter 4: Research Findings**

In this section, we discuss our experiment on binary classification, which involves training models to predict whether a text falls under the class of cyberbullying. We will analyse the results of various experiments and their significance. The first experiment focuses on implementing different algorithms on individual datasets and comparing the results. The second experiment involves using a combination (parallel ensemble) of the algorithms to determine if it would yield better results. It is also important to note that based on our study objective, we explore the use of combined datasets for training and testing the algorithms on platform-specific test sets to see how well the algorithms generalise.

### **4.1 Methodology and Code Implementation**

In the data collection phase, we carefully prepared Twitter and Instagram raw datasets from Wang, Fu and Lu (2020) and Chelmiss and Yao (2019) and Yao, Chelmiss, and Zois (2019). The data underwent preprocessing steps highlighted in Chapter 3.2 to ensure quality and suitability for analysis. Some of these steps included tokenization and stop-word removal. The code snippet with clear examples of each process is in Chapter 3.2.

To implement the algorithms, our approach involved the use of SVM, Naive Bayes and Logistic Regression models. These algorithms were chosen due to their suitability for handling the specific characteristics of our dataset and addressing the research questions at hand as highlighted in Chapter 2.2. We chose an 80-20 training-test split due to our sizable datasets. This split ratio ensures that most of the data is used for training, helping the model learn effectively. We also have test data to evaluate how well the model performs on new data, confirming its ability to generalise. The following code snippet provides a simplified representation of the core steps involved:



```

import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import CountVectorizer
from imblearn.over_sampling import SMOTE
from sklearn.svm import SVC
from sklearn.linear_model import LogisticRegression
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import classification_report

# Load your dataset
twitter_df = pd.read_csv('/content/drive/MyDrive/MSC Data science/Thesis/Final data/clean_twitter.csv')
#rename column
twitter_df.rename(columns={"joined_text": "text"}, inplace=True)

X = twitter_df['text']
#encode for train
y, class_names = pd.factorize(twitter_df['label'])

# Splitting the data into 80-20 train-test split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

```

Figure 4.1: Code snippet of Train-Test split before Training Algorithm

```

[2] # Apply CountVectorizer to text data
count_vectorizer = CountVectorizer(max_features=5000) # max_features
X_train_counts = count_vectorizer.fit_transform(X_train)
X_test_counts = count_vectorizer.transform(X_test)

# Apply SMOTE to handle class imbalance
smote = SMOTE(random_state=42)
X_train_resampled, y_train_resampled = smote.fit_resample(X_train_counts, y_train)

# Support Vector Machine (SVM)
svm_model = SVC(kernel='linear', C=1)
svm_model.fit(X_train_resampled, y_train_resampled)
svm_predictions = svm_model.predict(X_test_counts)

print("Support Vector Machine (SVM) Classification Report:")
print(classification_report(y_test, svm_predictions))

```

Figure 4.2: Code snippet of Training Algorithm and parameters

For a more detailed view of the code implementation and to access the full code, I encourage visiting the [GitHub repository](#) associated with this research.

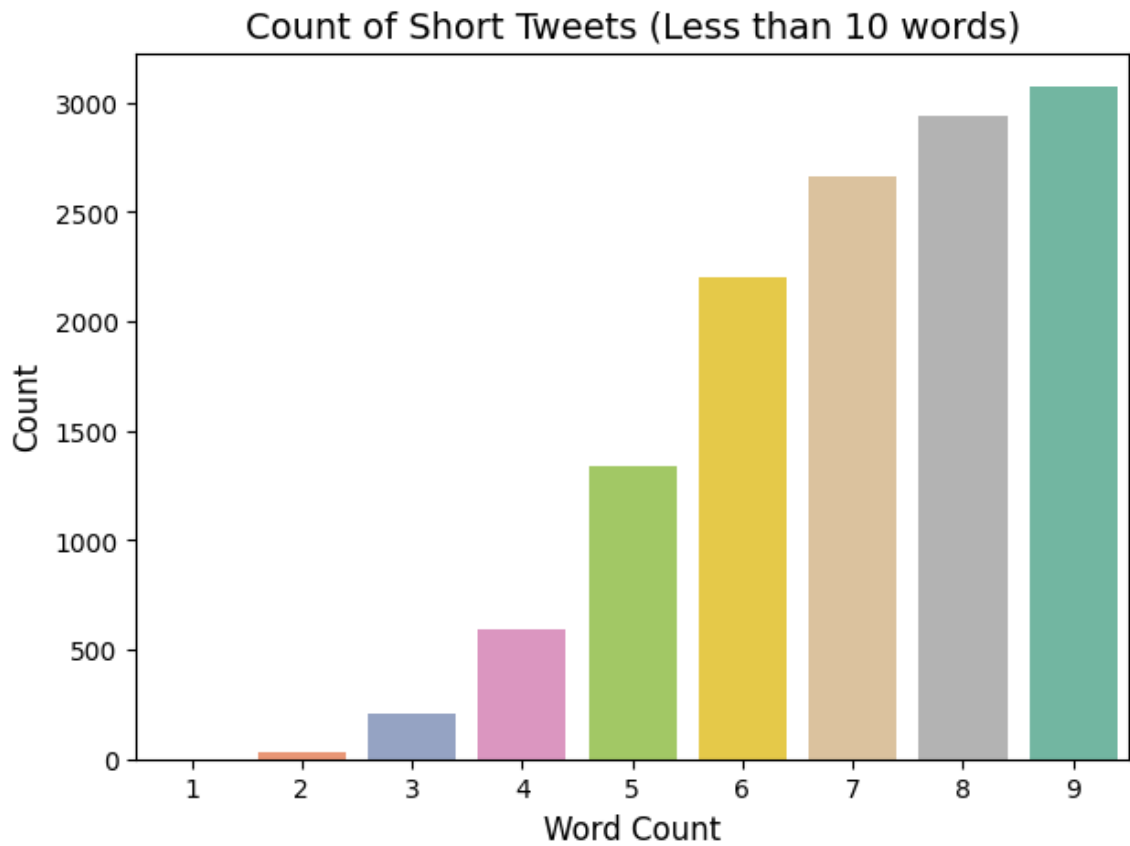
## 4.2 Exploratory Data Analysis

In order to gain valuable insights from our datasets, we first had to prepare the text using natural language processing (NLP) techniques as described in Chapter 3.2 of our methodology. These techniques involved several steps such as tokenization, removal of stopwords, expansion of contractions, spellchecking and removal of regular expressions and hyperlinks from both our Instagram and Twitter datasets.

After completing the preprocessing step, we focused on analysing the comments and tweets that had less than 10 words. Our main objective was to determine whether these short texts were relevant or if they should be disregarded. We conducted an in-depth exploration and found that a considerable number of tweets in the Twitter dataset were linked to cyberbullying, particularly those that had less than 10 words. These tweets often contained explicit cyberbullying language as seen in Fig 4.3. We then visualised the data by creating a bar chart, as shown in Figure 4.4. This chart displays the count of tweets with various word counts among those with fewer than 10 words. The results of this visualisation led us to exclude tweets with less than 4 words from our dataset only.

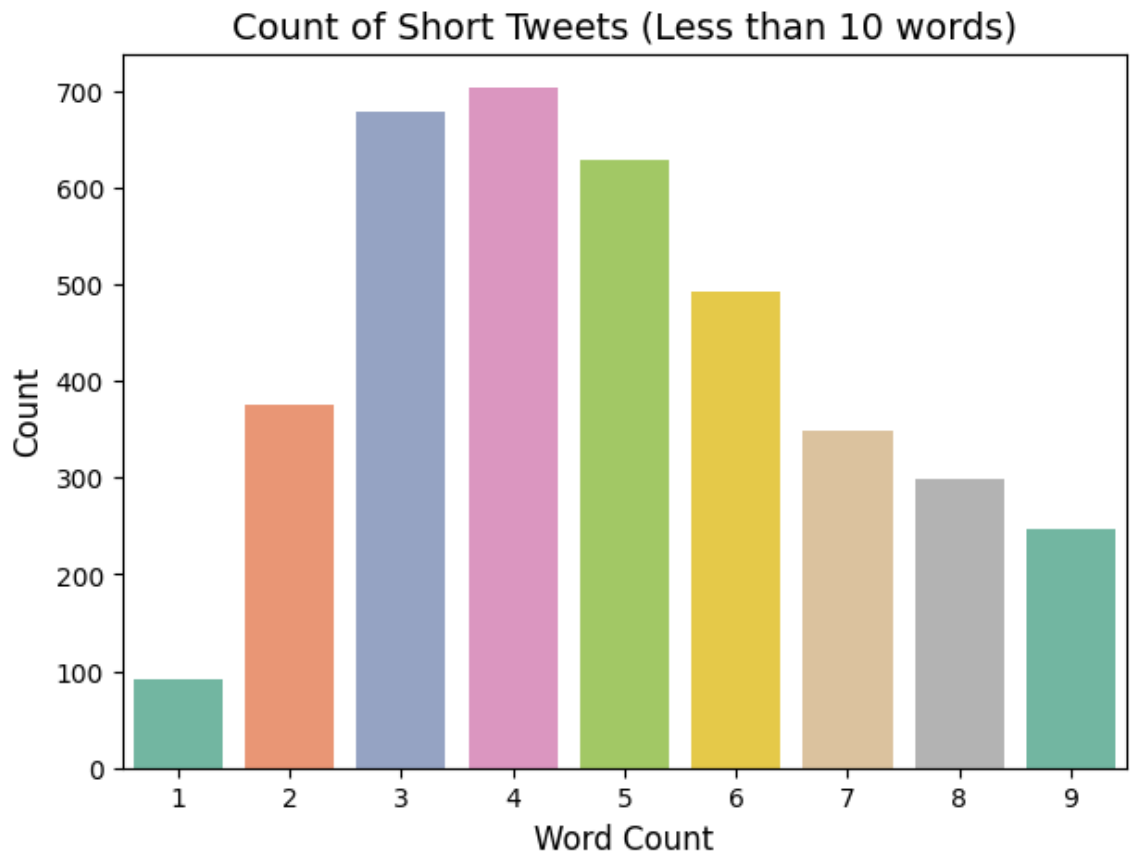
|       | joined_text                                       | label         | word_count |
|-------|---|---------------|------------|
| 8269  | let u weary well due season shall reap faint      | cyberbullying | 9          |
| 10109 | may believe year ago spoken fingerprint sarah ... | cyberbullying | 9          |
| 10673 | ogg want box girl high pulled boxing funny apo... | cyberbullying | 9          |
| 11379 | need showing humanity country people develop h... | cyberbullying | 9          |
| 11597 | imagine signing living wrong bully school pulled  | cyberbullying | 7          |
| ...   | ...   | ...           | ...        |
| 39038 | still co high school bully                        | cyberbullying | 5          |
| 39039 | maybe people like stepping leg judge              | cyberbullying | 6          |
| 39040 | amber high school bully high school fucking bully | cyberbullying | 8          |
| 39041 | old due cafe grandmother blue hair                | cyberbullying | 6          |
| 39042 | school bully leaving join government              | cyberbullying | 5          |

Fig 4.3: Twitter data frame showing tweets with less than 10 words



*Fig 4.4: Twitter word count plot*

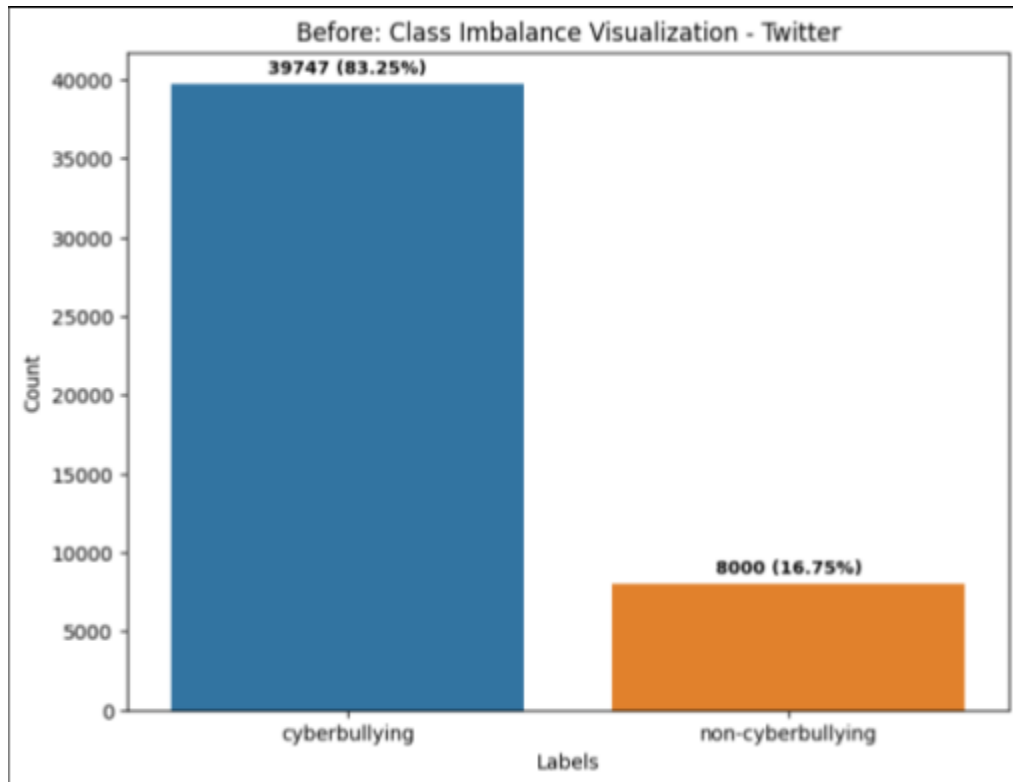
During our analysis of the Instagram dataset, we discovered an interesting trend in comments consisting of less than 10 words, as illustrated in Figure 4.5. The majority of these comments were found to have between 3 to 6 words, with a total count of 3,864. Despite their length, these comments may hold significant value for our research and hence, we have chosen to retain them rather than discard them altogether.



*Fig 4.5: Instagram word count plot*

To proceed with our analysis, we needed to check if our data had a class imbalance. This is crucial because standard classifiers tend to favour the majority class when there is an imbalance (Padurariu and Breaban, Mihaela Elena, 2019). As per our research methodology, the Twitter dataset had 47,589 records, out of which 39,589 (83%) were cyberbullying and 8,000 (17%) were non-cyberbullying records. The Instagram dataset had 10,000 records, with 2,000 (20%) cyberbullying and 8,000 (80%) non-cyberbullying records.

After applying the NLP techniques discussed in Chapter 3.2 to preprocess our datasets and removing some tweets and comments where necessary, it resulted in a new number of records. The Twitter dataset now contains 39,043 records, with 34,155 (87%) classified as cyberbullying and 4,888 (13%) as non-cyberbullying. The Instagram dataset has 5,618 records, with 1,384 (25%) classified as cyberbullying and 4,234 (75%) as non-cyberbullying. The differences are illustrated in the bar plots below.



*Figure 4.6: Visualising Class Imbalance in the Twitter Dataset: Before Preprocessing*

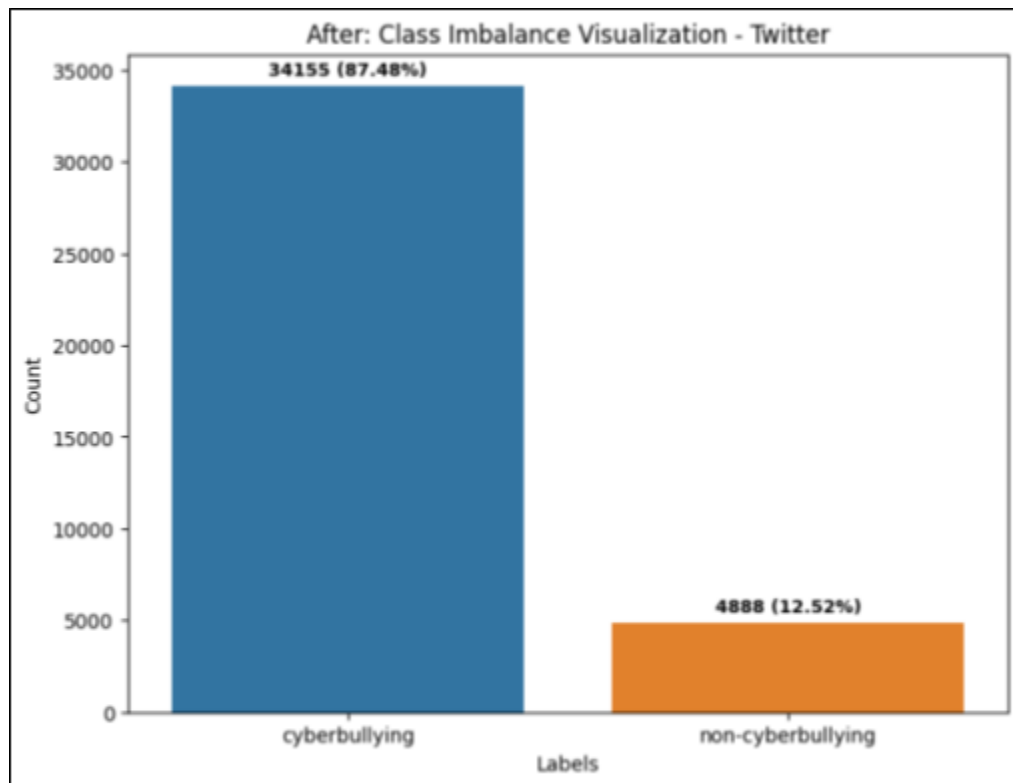


Figure 4.7: Visualising Class Imbalance in the Twitter Dataset: After Preprocessing

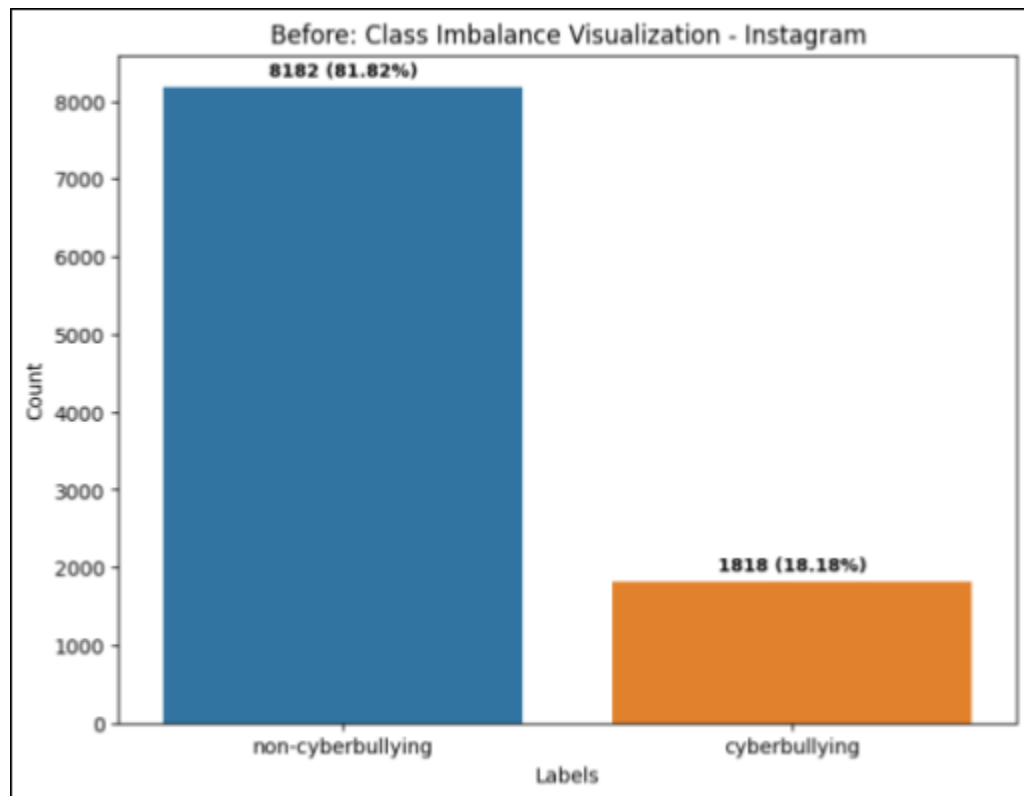
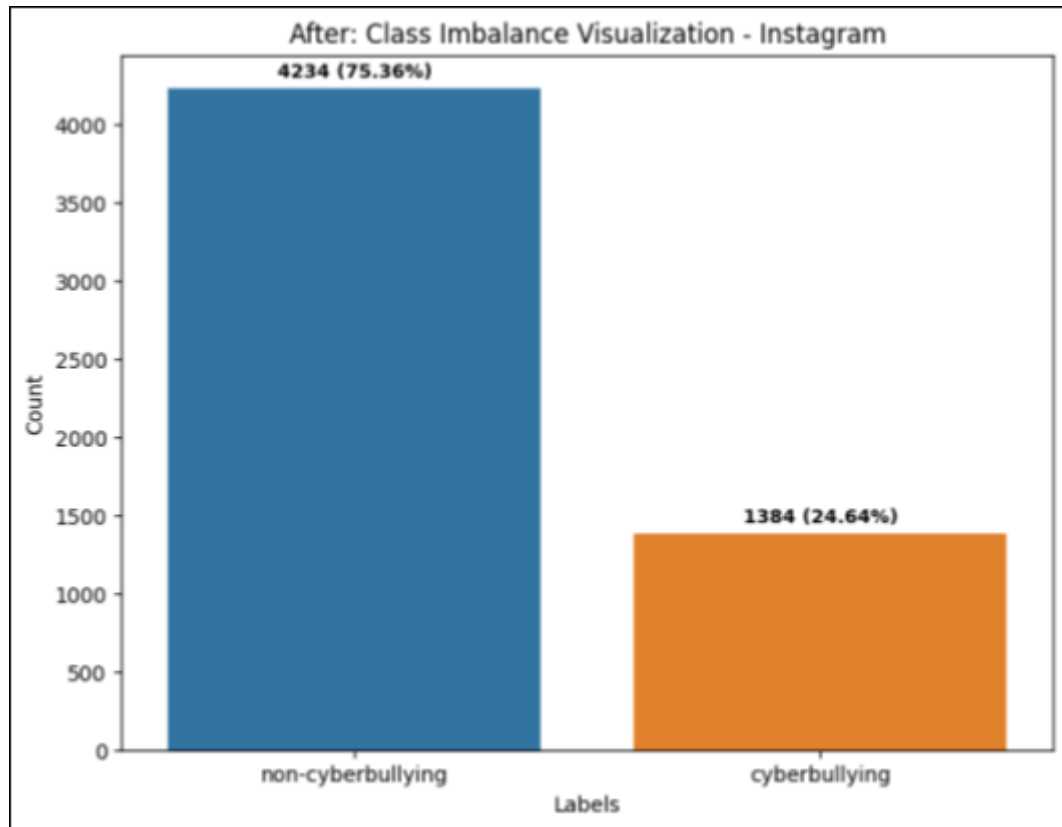


Figure 4.8: Visualising Class Imbalance in the Instagram Dataset: Before Preprocessing



*Figure 4.9: Visualising Class Imbalance in the Instagram Dataset: After Preprocessing*

Based on the plot, it is evident that the non-cyberbullying category is underrepresented in the Twitter dataset, whereas the cyberbullying category is underrepresented in the Instagram dataset. This disparity may lead to biased model performance, resulting in reduced identification of the minority category and skewed evaluation metrics. To address this issue, we used the Synthetic Minority Over-sampling Technique (SMOTE) throughout the modelling process.

SMOTE addresses this by generating synthetic samples for the minority class. To generate a synthetic sample for a minority class, this method selects a sample from the original dataset and finds its  $k$  nearest minority class neighbours in the feature space. It then randomly selects one of the  $k$  nearest neighbours and generates a new synthetic sample by interpolating between the selected minority class sample and the randomly selected neighbour(sole, 2023).

Additionally, we also used the F1 measure as our primary metric to avoid bias due to the imbalance of distributions. The datasets were divided randomly into two parts using a 20:80 ratio.



80% of the data was used for training and the remaining 20% was used for testing. The experiments involved using three algorithms: Naïve Bayes, Logistic Regression and SVM-Linear.

### **4.3 Model Performance on Twitter Dataset**

Tables 4.1 and 4.2 summarise the results of applying two feature extraction methods, Countvectorizer and TF-IDF, to assess models' capabilities in identifying instances of cyberbullying. The metrics assessed include F1 score, precision and recall. After using the count vectorizer, the Support Vector Machine (SVM) classifier achieved an 87% precision, recall and F1 score. This means that the SVM model can accurately detect substantial cases of cyberbullying and efficiently capture a significant portion of positive cases in the dataset. The F1 score indicates that the model effectively balances the risks of making incorrect positive predictions and missing actual positive instances.

The Logistic Regression(LR) like SVM had a balanced trade-off between precision(88%) and recall(87%). This balance highlights that the LR model is good at getting both its positive predictions right and capturing a good portion of actual positive cases. The F1 score, also at 87%, supports this, showing that the LR model can handle the trade-off between accuracy and inclusiveness well. The Naive Bayes(NB) had a slightly lower recall(85%) and F1 score(86%) compared to the other models. Despite this, it did showcase a considerable precision rate of 88%. Which means it is better at minimising false positive predictions.

When using TF-IDF, the SVM classifier showed a higher precision score of 89% when compared to using the count vectorizer. However, there was a decrease in the recall score of 84% and F1 score of 86%. Similarly, the LR model had an increase in precision to 90%, but a drop in recall (84%) and F1 score (85%). Naive Bayes had a decrease in all metrics, with a slight drop in precision to 87%, recall to 83% and F1 score to 85%.

After assessing the F1 scores and analysing precision and recall scores, we can see that the count vectorizer feature extraction method produced better results overall. Although the TF-IDF models had decent scores, they mainly would prioritise reducing false positive predictions. In contrast, the count vectorizer models had better measures that would focus on minimising false positive predictions and accurately identifying all relevant instances in the dataset. When using count vectorizer models, Logistic Regression is the most effective at detecting instances of

bullying while keeping false positives and false negatives to a minimum. However, the SVM model is also a suitable option despite having a slightly lower precision of 1%.

| Model | Feature Extraction | Accuracy | Precision | Recall | F1-score |
|-------|--------------------|----------|-----------|--------|----------|
| SVM   | CountVectorizer    | 87       | 87        | 87     | 87       |
| LR    | CountVectorizer    | 87       | 88        | 87     | 87       |
| NB    | CountVectorizer    | 85       | 88        | 85     | 86       |

*Table 4.1: Results achieved by each classifier employing the Count Vectorizer feature extraction method*

| Model | Feature Extraction | Accuracy | Precision | Recall | F1-score |
|-------|--------------------|----------|-----------|--------|----------|
| SVM   | TF-IDF             | 84       | 89        | 84     | 86       |
| LR    | TF-IDF             | 84       | 90        | 84     | 85       |
| NB    | TF-IDF             | 83       | 87        | 83     | 85       |

*Table 4.2: Results achieved by each classifier employing the TF-IDF feature extraction method*

#### 4.4 Model Performance on Instagram Dataset

Tables 4.3 and 4.4 summarise the results of applying each algorithm to the Instagram dataset. When using the count vectorizer, the metrics for the SVM classifier show a precision score of 74%, a recall of 69%, and an F1 score of 71%. However, in comparison to the performance measures of the Twitter models, these metrics appear to be relatively low. Additionally, the recall metric is particularly concerning as it indicates that this model may miss several positive instances of cyberbullying. This could result in false negatives, where instances that are actually positive are predicted as negative, and a significant portion of actual cyberbullying instances may not be effectively captured.

In terms of performance, the LR model outperformed SVM. It achieved a precision of 76%, a recall of 73%, and an F1 score of 74%. Although there is room for improvement, these metrics are still decent. Moreover, the LR model strikes a better balance between reducing false positives and identifying a considerable number of true positive instances, compared to SVM. In comparison to the SVM and LR models, the Naive Bayes model archives better metrics, with an F1 score of 78%. Additionally, it has a higher recall rate (77%), meaning that it returns the most relevant results, although some irrelevant ones may also be included. It also has a higher precision score (77%).

When using the TF-IDF extraction method, SVM performs better than the count vectorizer model and maintains a balance between its precision (79%) and recall (78%), reflected in its F1 score of 79%. However, when compared to the count vectorizer NB model, the TF-IDF Naive Bayes model experiences a decrease in recall and F1 score. It does, however, see an increase in precision. This trade-off results in a precision of 80% and a lower recall of 74%, ultimately leading to a 75% F1 score. The Logistic Regression model achieves the best performance, with an 81% F1 score, precision and recall. TF-IDF models have better performance metrics than count vectorizer models, suggesting their potential for better overall performance.

Based on our analysis of F1 scores, we identified that Logistic Regression with TF-IDF was the most effective in detecting cases of bullying with minimal false positives and false negatives. However, the F1 scores ranged from 71% to 81%, indicating that there is still potential for improving the models' performance.

| Model      | Feature Extraction | Accuracy | Precision | Recall | F1-score |
|------------|--------------------|----------|-----------|--------|----------|
| <b>SVM</b> | CountVectorizer    | 69       | 74        | 69     | 71       |
| <b>LR</b>  | CountVectorizer    | 73       | 76        | 73     | 74       |
| <b>NB</b>  | CountVectorizer    | 77       | 79        | 77     | 78       |

*Table 4.3: Results achieved by each classifier employing the Count Vectorizer feature extraction method*

| Model      | Feature Extraction | Accuracy  | Precision | Recall    | F1-score  |
|------------|--------------------|-----------|-----------|-----------|-----------|
| <b>SVM</b> | TF-IDF             | 78        | 79        | 78        | 79        |
| <b>LR</b>  | TF-IDF             | <b>81</b> | <b>81</b> | <b>81</b> | <b>81</b> |
| <b>NB</b>  | TF-IDF             | 74        | 80        | 74        | 75        |

*Table 4.4: Results achieved by each classifier employing the TF-IDF feature extraction method*

#### 4.5 Model Performance on Combined Dataset

We combined datasets from Instagram and Twitter to create a single dataset. We evaluated the algorithms on platform-specific test sets to see how well they perform on each. Tables 4.5 and 4.6 below show the classification reports for each platform's results, including precision, recall and F1-score metrics. This analysis demonstrates the model's ability to generalise across different data sources when trained with data from multiple social media sites and provides valuable insights into its behaviour on how it performs on specific social media platforms.

After using the count vectorizer, the SVM classifier showed favourable results for the Twitter dataset with a precision of 87% indicating a significant accuracy in predicting cyberbullying instances, a recall of 82% and an F1 score of 84%. However, the Instagram dataset yielded lower metrics with a precision of 76%, a recall of 75% and an F1 score of 75%. The low recall suggests that the SVM classifier might struggle to identify all instances of the cyberbullying class on Instagram as effectively as it does on Twitter. When comparing the performance of the Naive Bayes model, it shows a decrease in both precision and recall when analysing one platform versus another. On Twitter, the NB model maintains a precision of 87%, while also having a higher recall of 84% and F1 score of 85%. However, on Instagram, the NB model has a lower performance compared to the SVM model in terms of precision, recall and F1 score.

LR on Twitter achieves the highest precision of 88% followed by a recall of 83% and an F1 score of 85%. These metrics are similar to the NB model only that with LR, the precision is 1% higher and with NB recall is 1% higher otherwise they both have similar F1 scores. This indicates that these models are effective at capturing a high proportion of relevant instances while maintaining a reasonable level of accuracy on Twitter data. On Instagram, the LR precision, recall, and F1 scores all are 77% which is the highest performance compared to SVM AND NB. This

performance suggests that the LR model's effectiveness in identifying relevant instances remains consistent, even in the context of the different platforms.

When TF-IDF is employed, the SVM model on the Twitter dataset shows improved results. Its precision stands at 86%, recall at 87% and F1 score at 86%. These numbers are higher compared to the SVM model using the count vectorizer. This suggests that the SVM model with TF-IDF performs better when dealing with Twitter data. However, its performance on Instagram experiences a decline in recall(68%) and F1 score(70%). This reduction in accuracy on Instagram suggests that the model might encounter challenges when applied to the characteristics specific to the Instagram dataset. The NB model's performance on Twitter data demonstrates improved metrics in comparison to the NB model using the count vectorizer. While there is a slight 1% decrease in precision to 86%, both recall and F1 scores show improvement, reaching 85% and 86% respectively. Applying NB with TF-IDF to the Instagram dataset does not result in substantial changes compared to the count vectorizer model, the precision rises to 76%, recall decreases to 72% and the F1 score remains constant at 74%.

Logistic Regression, leveraging TF-IDF on Twitter demonstrates a precision of 86% and a high recall of 87%, indicating its effectiveness in capturing a significant proportion of relevant instances. On Instagram, the model maintains consistent precision (78%) but quite low recall (67%) and F1(69%). This suggests that using LR with the count vectorizer would likely yield better performance compared to using LR with the TF-IDF feature extraction method.

Drawing from our analysis of the test sets across both platforms, we find that on the Twitter dataset, employing TF-IDF feature extraction with both the SVM and LR models leads to better performance compared to the other models. On the other hand, when looking at the Instagram dataset, the LR model using count vectorizer feature extraction emerges as the most effective performer. These divergent outcomes across datasets underscore the sensitivity of the models to data nuances. The disparities in precision, recall and F1 scores emphasise the necessity of modifying the models to suit the characteristics of each dataset individually, in order to achieve optimal performance across diverse contexts.

| Model      | Social Media Platform | Feature Extraction | Accuracy | Precision | Recall | F1-score |
|------------|-----------------------|--------------------|----------|-----------|--------|----------|
| <b>SVM</b> | Twitter               | CountVectorizer    | 82       | 87        | 82     | 84       |
|            | Instagram             |                    | 75       | 76        | 75     | 75       |
| <b>LR</b>  | Twitter               | CountVectorizer    | 83       | 88        | 83     | 85       |
|            | Instagram             |                    | 77       | 77        | 77     | 77       |
| <b>NB</b>  | Twitter               | CountVectorizer    | 84       | 87        | 84     | 85       |
|            | Instagram             |                    | 74       | 74        | 74     | 74       |

*Table 4.5: Results achieved by each classifier employing the Count Vectorizer feature extraction method*

| Model      | Social Media Platform | Feature Extraction | Accuracy | Precision | Recall | F1-score |
|------------|-----------------------|--------------------|----------|-----------|--------|----------|
| <b>SVM</b> | Twitter               | TF-IDF             | 87       | 86        | 87     | 86       |
|            | Instagram             |                    | 68       | 77        | 68     | 70       |
| <b>LR</b>  | Twitter               | TF-IDF             | 87       | 86        | 87     | 86       |
|            | Instagram             |                    | 67       | 78        | 67     | 69       |
| <b>NB</b>  | Twitter               | TF-IDF             | 85       | 86        | 85     | 86       |
|            | Instagram             |                    | 72       | 76        | 72     | 74       |

*Table 4.6: Results achieved by each classifier employing the TF-IDF feature extraction method*

## 4.6 Ensemble Learning

Above, we compared the performance of individual classifiers for detecting cyberbullying. Our goal in this section is to create a cyberbullying detection classifier using parallel ensemble techniques. Although advanced deep learning classifiers have high accuracy, they require a lot of computing power. We aim to strike a balance between processing speed and computational resources while maintaining good classification performance by adopting ensemble methods.

For this purpose, we selected Naive Bayes, Logistic Regression and SVM as our base classifiers. We then combined these classifiers using voting in four different variations to measure their performance.

For this dissertation's experiments, we used a parallel ensemble approach based on recommendations by previous researchers Hansen and Salamon (1990). Our approach explored diverse combinations of machine learning classifiers, illustrated in Tables 4.7 and 4.8. Each ensemble configuration included a selection of Naive Bayes, Logistic Regression and SVM classifiers. The framework established four distinct classifier combinations using a voting mechanism to evaluate their performance.

In this study, the concept of "soft voting" combined the predictions of individual classifiers. Unlike "hard voting," where each model gets to vote for a prediction and the final decision is made based on the majority vote, soft voting is more flexible in that, instead of just counting votes, it incorporates the class probabilities provided by each classifier. This approach considers the confidence levels associated with each classifier's predictions, resulting in a more sophisticated ensemble decision-making process.

A snippet of the code used for implementing the ensemble with soft voting:

```

# Creating the classifiers
svm_classifier = SVC(kernel='linear', C=1, probability=True)
logreg_classifier = LogisticRegression(max_iter=1000)
naive_bayes_classifier = MultinomialNB(alpha=1.0)

# Creating ensemble using voting classifier
ensemble_classifier = VotingClassifier(estimators=[
    ('svm', svm_classifier),
    ('logreg', logreg_classifier),
    ('nb', naive_bayes_classifier)
], voting='soft') # 'soft' voting since it is a probability-based ensemble

# Training the ensemble on the training data
ensemble_classifier.fit(X_train_resampled, y_train_resampled)

```

```

graph TD
    subgraph VotingClassifier
        svm[SVC]
        logreg[LogisticRegression]
        nb[MultinomialNB]
    end
    svm --> output
    logreg --> output
    nb --> output

```

Fig 4.10: Code snippet of SVM+LR+NB implemented to train dataset

During the experiments, important parameters were considered. The probability parameter for the Support Vector Machine classifier was set to True, enabling the calculation of class probabilities for soft voting. The max\_iter parameter for Logistic Regression was set to ensure convergence during the training process.

#### 4.6.1 Interpreting Ensemble Learning Performance for Twitter Dataset

We assessed the ensemble performance of Twitter and Instagram datasets across metrics including Recall, Precision, F1-Score and Accuracy. Results are summarised in Tables 4.7 and 4.8. For the Twitter dataset, SVM+NB+LR, SVM+LR and LR+NB achieved the highest F1 score(87%). SVM+LR, SVM+NB+LR and LR+NB had the highest Precision(88%). SVM+LR, SVM+NB+LR and LR+NB had the highest recall(87%), followed by SVM+NB had the lowest recall(86%).

|                  | Accuracy | Precision | Recall | F1 |
|------------------|----------|-----------|--------|----|
| <b>SVM+NB+LR</b> | 87       | 88        | 87     | 87 |
| <b>SVM+LR</b>    | 87       | 88        | 87     | 87 |



|               |           |           |           |           |
|---------------|-----------|-----------|-----------|-----------|
| <b>SVM+NB</b> | <b>86</b> | <b>88</b> | <b>86</b> | <b>87</b> |
| <b>LR+NB</b>  | <b>87</b> | <b>88</b> | <b>87</b> | <b>87</b> |

Table 4.7: Results achieved by the parallel ensemble on the Twitter dataset

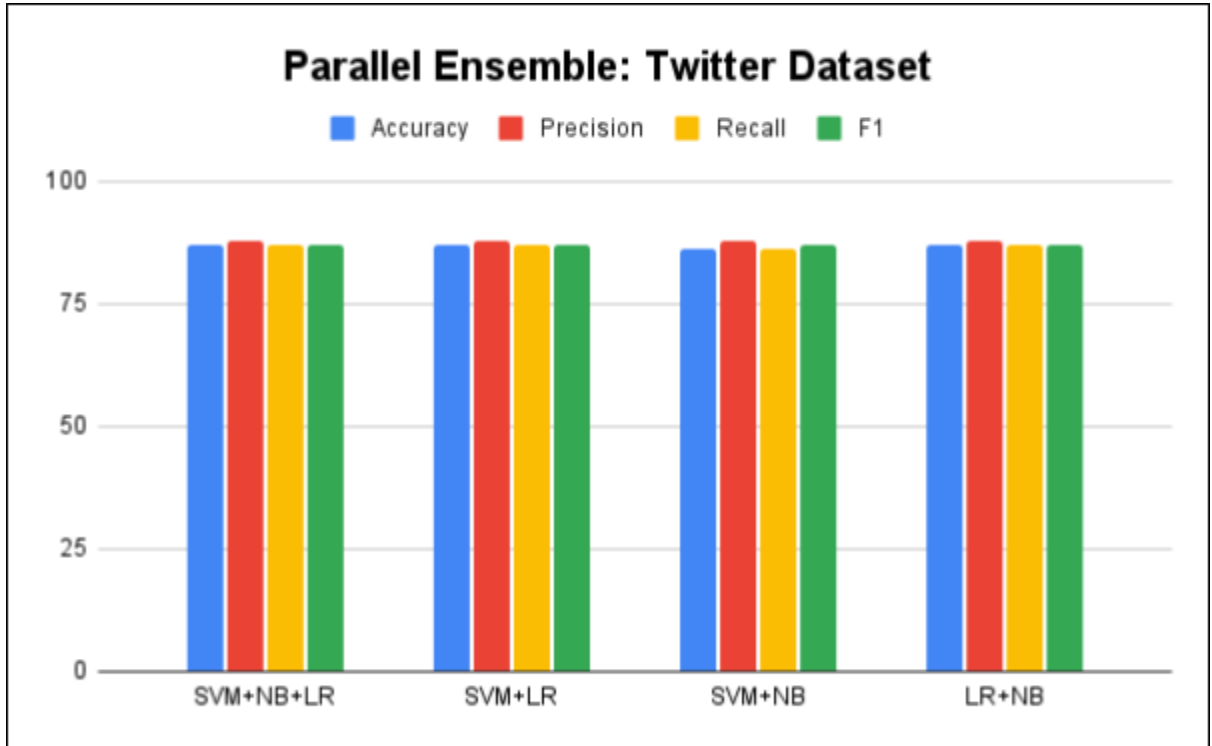


Figure 4.11: Twitter parallel ensemble results graph

When we look at the Twitter dataset results, we can see that based on the F1 score, there is not much improvement from the individual classifier performance we evaluated before. However, compared with previous experiments, the model maintains a trade-off between recall and precision when it comes to detecting instances of cyberbullying. Therefore, this balance ensures the classifier effectively identifies cyberbullying instances while minimising the number of incorrect predictions. However, since the Twitter dataset is unbalanced and has more bullying instances than non-bullying, our primary focus is on the F1 scores.

The ensemble learning results show that the models we used are better at identifying instances of cyberbullying correctly (recall) than precisely classifying them (precision). This means the models are designed to be sensitive in catching real cases of cyberbullying. This approach aims to make

sure we do not overlook any actual instances of cyberbullying, even if it means occasionally labelling some non-cyberbullying cases as positive. This strategy is in line with the goal of prioritising user safety and well-being in cyberbullying detection applications.

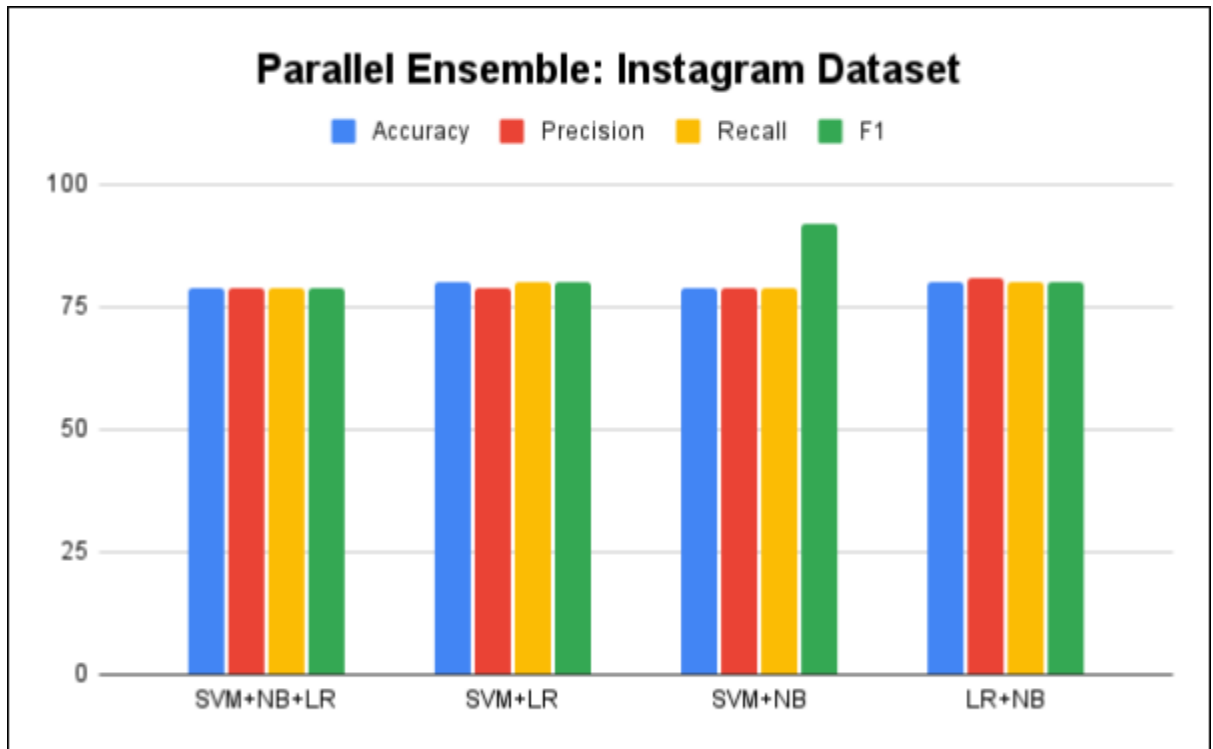
Overall, the ensembles SVM+NB+LR, SVM+LR AND LR+NB seem to have a good balance between precision, recall and F1. They achieved the highest F1 and recall, indicating that they can effectively identify cyberbullying instances while maintaining overall correctness. SVM+NB also demonstrated strong performance in both precision, recall and F1, suggesting its ability to accurately classify instances of cyberbullying.

#### 4.6.2 Interpreting Ensemble Learning Performance for Instagram Dataset

The SVM+NB(92%) F1 score was the highest followed by SVM+LR and LR+NB which all had 80% score. SVM+LR and LR+NB had the highest precision score (81%), followed by SVM+NB+LR, SVM+NB and SVM+LR which all had 79%. LR+NB and SVM+LR had the highest recall (80%), followed by SVM+NB+LR and SVM+NB having 79%. SVM+LR and LR+NB had the highest Accuracy scores (80%), followed by SVM+NB (79%) and SVM+NB+LR (79%).

|                  | Accuracy | Precision | Recall | F1 |
|------------------|----------|-----------|--------|----|
| <b>SVM+NB+LR</b> | 79       | 79        | 79     | 79 |
| <b>SVM+LR</b>    | 80       | 79        | 80     | 80 |
| <b>SVM+NB</b>    | 79       | 79        | 79     | 92 |
| <b>LR+NB</b>     | 80       | 81        | 80     | 80 |

*Table 4.8: Results achieved by the parallel ensemble on the Instagram dataset*



*Figure 4.12: Instagram parallel ensemble results graph*

Looking at the different ways we combined the models, the LR+NB combination catches our attention. It does a good job of finding cyberbullying instances. This combination has a good precision of 81%, a recall of 80% and an F1 score of 80%. Another combination of SVM+NB also does well. It stands out as it has a high F1 score of 92% indicating it performs well in achieving a balance between precision and recall. This implies that the ensemble is effective at correctly identifying positive cases while minimising false positives.

After analysing the Instagram dataset, we found that the results were mixed. The LR model was the most effective when using individual classifiers, with a precision, recall and F1 score of 81%. Depending on the desired outcome, we can choose between LR, LR+NB or SVM+NB. If the priority is higher precision, even if it means missing some bullying messages, then LR is the best choice. However, for more consistency, LR+NB or SVM+NB would be more suitable.

#### 4.6.3 Interpreting Ensemble Learning Performance for Combined Dataset

In the case of the Twitter test set, the LR+NB combination attained a high F1 score of 86%. The combinations of SVM+LR+NB, SVM+LR and LR+NB had the highest precision scores of 88%. When it comes to recall, the combinations of SVM+LR+NB, SVM+NB and LR+NB had the highest scores at 84%.

For the Instagram test set, the SVM+LR+NB combination achieved the highest F1 score of 79%. When considering precision, both the SVM+LR+NB and LR+NB combinations attained the highest value at 78%. Similarly, for recall, the SVM+LR+NB and LR+NB combinations achieved the highest value at 79%. After evaluating the metrics of all ensembles, it is clear that their values are very close to each other, which indicates a consistent performance level.

| Model            | Social Media Platform | Accuracy | Precision | Recall    | F1-score  |
|------------------|-----------------------|----------|-----------|-----------|-----------|
| <b>SVM+LR+NB</b> | Twitter               | 84       | 88        | 84        | 85        |
|                  | Instagram             | 79       | <b>78</b> | <b>79</b> | <b>79</b> |
| <b>SVM+LR</b>    | Twitter               | 83       | 88        | 83        | 85        |
|                  | Instagram             | 77       | 77        | 77        | 77        |
| <b>SVM+NB</b>    | Twitter               | 84       | 87        | 84        | 85        |
|                  | Instagram             | 78       | 77        | 78        | 77        |
| <b>LR+NB</b>     | Twitter               | 84       | <b>88</b> | <b>84</b> | <b>86</b> |
|                  | Instagram             | 79       | 78        | 79        | 78        |

*Table 4.9: Results achieved by the parallel ensemble on the combined dataset*

After analysing the results and comparing them to those obtained from individual classifiers, it is clear that the Twitter dataset shows better performance with SVM and LR models. However, even though it does not surpass the individual classifiers, the LR+NB model is the most effective one in the ensemble. Moving on to the Instagram dataset, the ensemble SVM+LR+NB outperforms the individual classifier LR, proving its strength in this context.

We have analysed our Twitter and Instagram data extensively using machine learning. As a result, we have identified the best models for training and application on each platform. We have also combined the data from both platforms to create a cohesive dataset, showing their adaptability on each platform. In the next chapter, we will explain our findings and their broader implications, the limitations and challenges faced as well as suggest areas for future research.

## **Chapter 5: Conclusion and Future Work**

This section presents our research conclusion, limitations and challenges and suggests potential areas for future research that expand on our findings and contribute to the overall dissertation goal.

### **5.1 Conclusion**

The focus of this thesis is to explore the issue of cyberbullying on social media. To gain a deeper understanding of this issue, we started by exploring the existing literature on cyberbullying. This helped us identify how cyberbullying can negatively impact individuals and provided insight into why it occurs. By doing so, we were able to develop strategies to prevent cyberbullying and minimise its harmful effects. One important finding was the need for a clear definition of bullying. Not all negative statements made on social media platforms qualify as bullying. For example, friends may use slang and casual language when communicating with each other due to their close relationship. In addition, we developed a simple framework where we discussed about cyberbullying. Our objective is to create a model that can detect instances of cyberbullying by analysing data from two different social media sites.

In order to conduct our study, we gathered the necessary data from two social media platforms: Instagram and Twitter. We obtained the Twitter dataset from IEEE and the Instagram dataset directly from the authors Chelmiss and Yao (2019). Using NLP techniques, we preprocessed and explored the datasets. We then applied three machine learning algorithms (SVM, Naive Bayes and Logistic Regression) to each dataset. Additionally, we created a new dataset by combining both the Instagram and Twitter datasets and applied the same algorithms to this new dataset with the purpose of finding the best model for our machine-learning problem. We utilised both TFIDF and CountVectorizer techniques to extract features.

Among the models used on the Twitter dataset, the Logistic Regression (LR) using count vectorizer emerged as the best performing model with a precision of 88% and an F1 and recall of 87%. The ensemble SVM+LR+NB also performed similarly to the LR with comparable metrics. The Logistic Regression with TF-IDF was identified as the best performing model for the Instagram dataset, with precision, recall and F1 metrics of 81%. Additionally, the ensemble of the LR+NB(precision:81%, recall and F1:80%) and SVM+NB(F1 92%) models also displayed good metrics. Depending on the desired outcome, we can choose between LR, LR+NB or SVM+NB. If the priority is to have higher precision, even if it means missing some bullying messages, then LR is the best choice. However, for more consistency, LR+NB or SVM+NB would be more suitable.

When tested on individual platform test sets, the results of the combined dataset showed that the SVM and LR models with TF-IDF performed the best, with a precision and F1 score of 86% and a recall of 87%, specifically on the Twitter test set. These models outperformed the ensemble model and were the most effective. During the evaluation of classifiers for the Instagram test set, the LR model that employed the count vectorizer turned out to be the best individual model based on its precision, recall and F1 scores of 77%. Nonetheless, the SVM+LR+NB ensemble outperformed the LR model with a precision score of 78% and F1 and recall scores of 79%.

When assessing the models' performance, it is worth noting that combining datasets from various social media platforms has produced promising results. If we compare the performance to the models that used platform-specific data for training, they performed equally or better. This suggests the potential for achieving strong performance by merging data from different platforms and training algorithms, followed by evaluating them on platform-specific test sets to identify instances of cyberbullying.

In summary, our study has shed light on the pervasive concern of cyberbullying that occurs online and poses significant challenges to the well-being of individuals, especially in the context of virtual interactions. An effective solution to this problem lies in integrating technological capabilities with a deep understanding of human behaviour.

## **5.2 Limitations and Challenges**

Our research study has some limitations that are worth noting as they have impacted the findings. Firstly, we only focused on two social media platforms, Twitter and Instagram, which limited our ability to identify cyberbullying across a wider range of online platforms. Users often switch between various platforms, and thus our study could have missed insights from other sources.

Secondly, our ability to gather data from Twitter and Instagram was limited because we would have had to pay to retrieve tweets or comments. Instead, we chose to utilise pre-existing secondary data. If we had access to newly collected data, it would have allowed us to examine more recent content and capture more nuanced language, which could have greatly enhanced the quality of our study. Although our datasets were substantial for our study's scope, having larger datasets in the future could yield more comprehensive results. Lastly, our machine learning approach was limited to a parallel ensemble method and only three machine learning algorithms due to the project's scope. Expanding our methods and models may provide a more comprehensive understanding of cyberbullying. These limitations highlight the need for broader data collection, improved data access and diversified machine learning techniques in future research endeavours.

scraped

### **5.3 Future Work**

The field of cyberbullying detection requires continuous improvement to keep up with the ever-expanding social media and digital technology landscape. As highlighted in the literature review, it is evident that a universally accepted and comprehensive definition of cyberbullying is imperative on a global scale. Without a clear understanding of what constitutes cyberbullying, it becomes challenging to develop effective models or countermeasures. Therefore, there is an urgent need to establish guidelines that standardise and unify the concept, enabling accurate research and efficient prevention and intervention efforts.

Additionally, the existing studies on cyberbullying detection largely focus on individual platforms and their effects on that environment only. While these studies provide valuable insights into specific instances of cyberbullying, they often overlook the interconnected nature of online interactions. In today's digital landscape, individuals frequently maintain active profiles across numerous social media platforms. It is not uncommon for someone to have accounts on several platforms and actively engage with each of them simultaneously. For instance, a victim of cyberbullying may encounter harassment on Twitter and subsequently express their emotions and experiences through a post on Instagram. These examples underscore the growing necessity of exploring the concept of cross-system user modelling, where we study and analyse users' behaviours and interactions across different online platforms to gain a comprehensive understanding of their digital experiences. This approach will not only enhance detection accuracy but also enable more effective prevention and intervention strategies.

While this study has made significant progress in cyberbullying detection by combining datasets from Instagram and Twitter and evaluating the models to individual platforms, there remains

exciting avenues for future research. First, expanding the scope to include additional social media platforms would provide a more comprehensive understanding of cyberbullying across a broader range of online environments. Second, exploring advanced machine learning techniques, such as deep learning and natural language processing, could further enhance model accuracy and sensitivity. Thirdly, it is important to prioritise gathering new data instead of solely relying on past collected social media data. This approach is crucial for capturing the latest trends in online conversations, including new slang words and linguistic nuances. Using only past data risks missing out on the constantly evolving nature of online discourse, where language and communication patterns change over time. Therefore, it is essential to proactively collect data to keep up with the ever-changing digital landscape. Additionally, investigating the temporal aspects of cyberbullying, including how it evolves over time, can provide valuable insights for the development of real-time prevention and intervention strategies.



## Chapter 6: References List

Al-Garadi, M.A., Hussain, M.R., Khan, N., Murtaza, G., Nweke, H.F., Ali, I., Mujtaba, G., Chiroma, H., Khattak, H.A. and Gani, A. (2019). Predicting cyberbullying on social media in the big data era using machine learning algorithms: Review of literature and open challenges. *IEEE Access*, 7, pp.70701–70718. doi:<https://doi.org/10.1109/ACCESS.2019.2918354>.

Anderson, M. (2018). *A Majority of Teens Have Experienced Some Form of Cyberbullying*. [online] Pew Research Center: Internet, Science & Tech. Available at: <https://www.pewresearch.org/internet/2018/09/27/a-majority-of-teens-have-experienced-some-form-of-cyberbullying/> [Accessed 20 Jun. 2023].

Berkson, J. (1944). Application of the Logistic Function to BioAssay. *Journal of the American Statistical Association*, [online] 39(227), pp.357–365. doi:<https://doi.org/10.2307/2280041>.

Bonaccorso, G. (2019). *Hands-on unsupervised learning with Python : implement machine learning and deep learning models using Scikit-Learn, TensorFlow, and more*. Birmingham, UK: Packt Publishing.

Boyd, D. and Crawford, K. (2012). Critical Questions for Big Data. *Information, Communication & Society*, [online] 15(5), pp.662–679. Available at: <https://www.dhi.ac.uk/san/waysofbeing/data/communication-zangana-boyd-2012.pdf> [Accessed 10 Jul. 2023].

Breiman, L. (1996). Bagging Predictors. *Machine Learning*, [online] 24(2), pp.123–140. doi:<https://doi.org/10.1023/A:1018054314350>.

Chelms, C. and Yao, M. (2019). Minority Report: Cyberbullying Prediction on Instagram. In: *The 11th ACM Conference on Web Science*. [online] doi:<https://doi.org/10.1145/3292522.3326024>.

Cyberbullying Research Center (2015). *Cyberbullying Research Center*. [online] Cyberbullying Research Center. Available at: <https://cyberbullying.org/> [Accessed 24 Jun. 2023].

Dehue, F., Bolman, C. and Völlink, T. (2008). Cyberbullying: Youngsters' Experiences and Parental Perception. *Cyberpsychology & behavior : the impact of the Internet, multimedia and virtual reality on behavior and society*, 11, pp.217–23. doi:<https://doi.org/10.1089/cpb.2007.0008>.

DUGGAN, M. (2014). *Part 1: Experiencing Online Harassment*. [online] Pew Research Center: Internet, Science & Tech. Available at: <https://www.pewresearch.org/internet/2014/10/22/part-1-experiencing-online-harassment/#fnref-1> 2087-2 [Accessed 24 Jun. 2023].

Ehman, A.C. and Gross, A.M. (2019). Sexual cyberbullying: Review, critique, & future directions. *Aggression and violent behavior*, 44, pp.80–87.

Fulantelli, G., Taibi, D., Scifo, L., Schwarze, V. and Eimler, S.C. (2022). Cyberbullying and Cyberhate as Two Interlinked Instances of Cyber-Aggression in Adolescence: A Systematic Review. *Frontiers in Psychology*, [online] 13. doi:<https://doi.org/10.3389/fpsyg.2022.909299>.

Gaydhani, A., Doma, V., Kendre, S. and Bhagwat, L. (2018). *Detecting hate speech and offensive language on twitter using machine learning: An n-gram and TFIDF based approach*. *arXiv [cs.CL]*.

Han, J. (2011). *Data mining: concepts and techniques*. Morgan Kaufmann.

Hansen, L.K. and Salamon, P. (1990). Neural network ensembles. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(10), pp.993–1001. doi:<https://doi.org/10.1109/34.58871>.

Hinduja, S. (2021). *Deepfakes and Cyberbullying*. [online] Cyberbullying Research Center. Available at: <https://cyberbullying.org/deepfakes> [Accessed 29 Jun. 2023].

Hinduja, S. and Patchin, J. (2019). *Cyberbullying: Identification, Prevention, & Response*. [online] Available at: <https://cyberbullying.org/Cyberbullying-Identification-Prevention-Response-2019.pdf> [Accessed 20 Jun. 2023].

Jaiswal, H. (2021). Memes, Confession Pages and Revenge Porn- The Novel Forms of Cyberbullying. *SSRN Electronic Journal*. [online] doi:<https://doi.org/10.2139/ssrn.3816609>.

Joachims, T. (1998). Text categorization with Support Vector Machines: Learning with many relevant features. *Machine Learning: ECML-98*, [online] pp.137–142. doi:<https://doi.org/10.1007/bfb0026683>.

Kamal, M. and Newman, W.J. (2016). Revenge pornography: Mental health implications and related legislation. *Journal of the American Academy of Psychiatry and the Law Online*, 44, pp.359–367.

Kemp, S. (2020). *Digital 2020: July Global Statshot*. [online] DataReportal – Global Digital Insights. Available at: <https://datareportal.com/reports/digital-2020-july-global-statshot> [Accessed 15 Jun. 2023].

Krathwohl, D.R. (2002). A revision of Bloom's taxonomy: An overview. *Theory into practice*, 41(4), p.212218.

L1ght (2020). *L1ght Releases Groundbreaking Report On Corona-Related Hate Speech and Online Toxicity*. [online] L1ght | Preventing Online Toxicity. Available at: <https://l1ght.com/l1ght-releases-groundbreaking-report-on-corona-related-hate-speech-and-online-toxicity/> [Accessed 20 Jun. 2023].

Lee, J., Abell, N. and Holmes, J. (2015). Validation of Measures of Cyberbullying Perpetration and Victimization in Emerging Adulthood. *Research on Social Work Practice*, 27. doi:<https://doi.org/10.1177/1049731515578535>.

Lee, W.-M. (2019). *Python machine learning*. [online] Indianapolis, In: Wiley. Available at: <https://ebookcentral.proquest.com/lib/westminster/detail.action?docID=5747364#> [Accessed 6 Jul. 2023].

López, F. (2021). *Ensemble Learning: Bagging & Boosting*. [online] Medium. Available at: <https://towardsdatascience.com/ensemble-learning-bagging-boosting-3098079e5422#:~:text=Bagging%20or%20Bootstrap%20Aggregation%20was> [Accessed 13 Aug. 2023].

Maher, D. (2008). Cyberbullying: An Ethnographic Case Study of One Australian Upper Primary School Class. *Youth Studies Australia*, 27.

Mangaonkar, A., Hayrapetian, A. and Raje, R. (2015). Collaborative detection of cyberbullying behavior in Twitter data. pp.611–616. doi:<https://doi.org/10.1109/EIT.2015.7293405>.

Martinez-Plumed, F., Contreras-Ochando, L., Ferri, C., Hernandez Orallo, J., Kull, M., Lachiche, N., Ramirez Quintana, M.J. and Flach, P.A. (2020). CRISP-DM Twenty Years Later: From Data

Mining Processes to Data Science Trajectories. *IEEE Transactions on Knowledge and Data Engineering*, [online] 33(8), pp.1–1. doi:<https://doi.org/10.1109/tkde.2019.2962680>.

Miftah Andriansyah, Akbar, A., Afina Ahwan, Nico Ariesto Gilani, Ardiono Roma Nugraha, Rizki Nofita Sari and Remi Senjaya (2017). *Cyberbullying comment classification on Indonesian Selebgram using support vector machine method*. 2017 Second International Conference on Informatics and Computing (ICIC), pp.1–5.

Muneer, A. and Fati, S.M. (2020). A Comparative Analysis of Machine Learning Techniques for Cyberbullying Detection on Twitter. *Future Internet*, 12(11), p.187. doi:<https://doi.org/10.3390/fi12110187>.

Noviantho, Sani Muhamad Isa and Ashianti, L. (2017). *Cyberbullying classification using text mining*. 2017 1st International Conference on Informatics and Computational Sciences (ICICoS), pp.241–246.

NSPCC (2016). *Bullying and cyberbullying*. [online] NSPCC. Available at: <https://www.nspcc.org.uk/what-is-child-abuse/types-of-abuse/bullying-and-cyberbullying/> [Accessed 20 Jun. 2023].

Nurrahmi, H. and Nurjanah, D. (2018). *Indonesian Twitter Cyberbullying Detection using Text Classification and User Credibility*. pp.543–548. doi:<https://doi.org/10.1109/ICOIACT.2018.8350758>.

Opitz, D. and Maclin, R. (1999). Popular Ensemble Methods: An Empirical Study. *Journal of Artificial Intelligence Research*, [online] 11, pp.169–198. doi:<https://doi.org/10.1613/jair.614>.

Padurariu, C. and Breaban, Mihaela Elena (2019). Dealing with Data Imbalance in Text Classification. *KnowledgeBased and Intelligent Information & Engineering Systems: Proceedings of the 23rd International Conference KES2019*, [online] 159, pp.736–745. doi:<https://doi.org/10.1016/j.procs.2019.09.229>.

Patchin, J.W. (2021). *2021 Cyberbullying Data*. [online] Cyberbullying Research Center. Available at: <https://cyberbullying.org/2021-cyberbullying-data> [Accessed 20 Jun. 2023].

Puthenveedu, S. (2022). *Cyberbullying Detection using Ensemble Method*. [online] Available at: <https://repository.library.carleton.ca/downloads/cn69m4951> [Accessed 22 Aug. 2023].

Rabindra Nath Nandi, Alam, F. and Preslav Nakov (2022). Detecting the role of an entity in harmful memes: Techniques and their limitations. *ArXiv*, abs/2205.04402.

Ray, S. (2019). *6 Easy Steps to Learn Naive Bayes Algorithm (with code in Python)*. [online] Analytics Vidhya. Available at: <https://www.analyticsvidhya.com/blog/2017/09/naive-bayes-explained/> [Accessed 11 Aug. 2023].

Reachout (2019). *Cyberbullying and teenagers - ReachOut Parents*. [online] Reachout.com. Available at: <https://parents.au.reachout.com/common-concerns/everyday-issues/cyberbullying-and-teenagers> [Accessed 20 Jun. 2023].

Sahay, K., Singh Khaira, H. and Kukreja, P. (2018). *Detecting Cyberbullying and Aggression in Social Commentary using NLP and Machine Learning*. [online] Nishchay Shukla International Journal of Engineering Technology Science and Research. Available at: [http://ijetsr.com/images/short\\_pdf/1517199597\\_1428-1435-oucip915\\_ijetsr.pdf](http://ijetsr.com/images/short_pdf/1517199597_1428-1435-oucip915_ijetsr.pdf) [Accessed 10 Jul. 2023].

Salawu, S., He, Y. and Lumsden, J. (2020). Approaches to automated detection of cyberbullying: A survey. *IEEE Transactions on Affective Computing*, 11, pp.3–24. doi:<https://doi.org/10.1109/TAFFC.2017.2761757>.

Schapire, R.E. (1990). The strength of weak learnability. *Machine Learning*, [online] 5(2), pp.197–227. doi:<https://doi.org/10.1007/BF00116037>.

Scikit-learn.org. (2012). *1.11. Ensemble methods — scikit-learn 0.22.1 documentation*. [online] Available at: <https://scikit-learn.org/stable/modules/ensemble.html> [Accessed 11 Aug. 2023].

Seeker (2015). *How Mob Mentality Gets Worse Online*. YouTube. Available at: <https://www.youtube.com/watch?v=bV5ngreR7Hk> [Accessed 29 Jun. 2023].

Smith, P.K., Mahdavi, J., Carvalho, M., Fisher, S., Russell, S. and Tippett, N. (2008). Cyberbullying: its nature and impact in secondary school pupils. *Journal of child psychology and psychiatry, and allied disciplines*, [online] 49(4), pp.376–85. doi:<https://doi.org/10.1111/j.1469-7610.2007.01846.x>.

sole (2023). *Overcoming Class Imbalance with SMOTE: How to Tackle Imbalanced Datasets in Machine Learning*. [online] Train in Data Blog. Available at: <https://www.blog.trainindata.com/overcoming-class-imbalance-with-smote/> [Accessed 27 Aug. 2023].

U.S. Department of Health and Human Services (2021). *StopBullying.gov*. [online] StopBullying.gov. Available at: <https://www.stopbullying.gov/> [Accessed 24 Jun. 2023].

UNICEF (2023). *Cyberbullying: What is it and how to stop it*. [online] [www.unicef.org](http://www.unicef.org). Available at: <https://www.unicef.org/end-violence/how-to-stop-cyberbullying> [Accessed 28 Jun. 2023].

Wang, J., Fu, K. and Lu, C.-T. (2020). SOSNet: A graph convolutional network approach to fine-grained cyberbullying detection. pp.1699–1708. doi:<https://doi.org/10.1109/BigData50022.2020.9378065>.

Watanabe, H., Bouazizi, M. and Ohtsuki, T. (2018). Hate speech on twitter: A pragmatic approach to collect hateful and offensive expressions and perform hate speech detection. *IEEE Access*, 6, pp.13825–13835. doi:<https://doi.org/10.1109/ACCESS.2018.2806394>.

Whittaker, E. and Kowalski, R. (2014). Cyberbullying Via Social Media. *Journal of School Violence*, 14, pp.11–29. doi:<https://doi.org/10.1080/15388220.2014.949377>.

Willard, N.E. (2007). *Cyberbullying and cyberthreats: Responding to the challenge of online*. Research press.

[www.stompoutbullying.org](http://www.stompoutbullying.org). (n.d.). *Cyber-Mobbing: A New Form of Cyberbullying Affecting Teens*. [online] Available at: <https://www.stompoutbullying.org/blog/cyber-mobbing> [Accessed 29 Jun. 2023].

Yang, Y. and Yang, Y. (2017). Chapter 4 Ensemble Learning. In: *Temporal Data Mining Via Unsupervised Ensemble Learning*. [online] Elsevier, pp.35–56. doi:<https://doi.org/10.1016/B9780128116548.00004X>.

Yao, M., Chelmiss, C. and Zois, D. (2019). Cyberbullying Ends Here: Towards Robust Detection of Cyberbullying in. In: *The World Wide Web Conference*. [online] p.34273433. doi:<https://doi.org/10.1145/3308558.3313462>.

Zeera Talat and Hovy, D. (2016). Hateful symbols or hateful people? Predictive features for hate speech detection on twitter