Journal of King Saud University – Computer and Information Sciences 34 (2022) 4099–4108

Contents lists available at ScienceDirect

Journal of King Saud University – Computer and Information Sciences

journal homepage: www.sciencedirect.com

CyberAid: Are your children safe from cyberbullying?

Lee Jia Thun[a], Phoey Lee Teh[a,*], Chi-Bin Cheng[b]

[a] Department of Computing and Information Systems, School of Engineering and Technology, No 5, Jalan Universiti, Sunway City 47500, Malaysia
[b] Department of Information Management, Tamkang University, No 151, Yingzhuan Road, Tamsui District, New Taipei City 251301, Taiwan

ARTICLE INFO

Article history:
Received 22 October 2020
Revised 10 February 2021
Accepted 1 March 2021
Available online 11 March 2021

Keywords:
Machine language
Cyberbully
Textual analysis
Hate speech
Human computer interface
Mobile application

ABSTRACT

Researchers around the world have been implementing machine learning as a method to detect cyberbullying text. The machine is trained using features such as variations in texts, through social media context and interactions in a social network environment. The machine can also identify and profile users through gender or use of hate speech. In this study, we analysed different types of mobile applications that manage cyberbullying. This study proposes a mechanism, which combines the best cyberbullying detection features to fill the gaps and limitations of existing applications. The results of the study have shown that the proposed mobile application records a higher accuracy in detecting cyberbully than other available applications.

1. Introduction

Cyberbullying commonly occurs in social networking sites (Hosseinmardi et al., 2015). It is an act which utilises technology to harm others (Young et al., 2017). Statistically, 58% of students between grade 4 to 8 do not notify their parents when they encounter cyberbullying (Chen et al., 2012; i-SAFE Inc., 2019); while 54% of parents feel they are unable to monitor and protect their children from inappropriate online content. Several mobile applications have been developed to monitor interactions between adolescents (aged between 10 and 19 years) and general internet users. Some of the applications even enable parents to have full access to their children's online conversations with others. However, privacy issues pose a great challenge in managing and overseeing cyberbullying. One of the examples is My Mobile Watchdog (My Mobile, 2001). The parental monitoring application PocketGuardian (LLC, 2019) has since provided a feature that notifies parents when bullying content from or to their children is detected, but without showing the actual content.

Many studies have proposed methods on detecting cyberbullying on social media. However, these studies mainly focus on online features. Some studies focus on techniques, especially machine learning, to examine online content. In the case of machine learning, the features used to train algorithms are essential. Machine learning refer to the use of artificial intelligence (AI) and provides systems the ability to automatically learn and improve from experience without being explicitly programmed (Holzinger, 2016; Expert System Team, 2020). In the case of machine learning, the features used to train algorithms are essential and do not really fully implemented in their research but only involved tested model, making it hard to be adopted for cyberbullying monitoring. Cases has shown that adolescents do not notify their parents when they are being cyberbullied (i-SAFE Inc., 2019). It may then be too late to help them when cases happen.

Therefore, this study aims to examine methods of cyberbullying detection and to integrate them into a mobile application. Unlike the previous study, this study proposes an application that alerts parents if their child is a potential victim or perpetrator of cyberbullying. Our method incorporates multiple features including sentiment value, number of exclamation marks, number of personal pronouns, and account creation date to enable parents to easily identify their children without invading the children's privacy. The application collects sample content testing text/comments from tweets. Tweets that contain profane words are assumed to have a higher possibility of being a hate speech, that may lead to

* Corresponding author.
E-mail addresses: 17055351@imail.sunway.edu.my (L.J. Thun), phoeyleet@sunway.edu.my (P.L. Teh), Cbcheng@mail.tku.edu.tw (C.-B. Cheng).

Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

https://doi.org/10.1016/j.jksuci.2021.03.001
1319-1578/

cyberbullying (Teh et al., 2018). Samples of these texts are selected and then sent to parents through the application. To protect the privacy of the children, parents can only access the identified harmful content rather than the full online conversations.

2. Literature review

The literature review section is divided into three separate sub-sections. In the first sub-section, textual features are discussed, follow with network features and users features in the second and third sub-section. The existing cyberbullying applications and classifiers are discussed in fourth and fifth sub-sections.

To prepare the machine language in classifying the cyberbullying and no-cyberbullying features, explanations of each feature and how it works are detailed in the following sections.

2.1. Textual features

Textual features include 1) profane words, 2) punctuation marks 3) uppercase, 4) personal pronouns, 5) emotion text, and 6) hashtag and URLs.

Profanity, mostly indicates an offensive, impolite and rude attitude, making them an essential feature for cyberbullying detection. Chen et al. (2012) claimed that offensive sentences always contain pejoratives, profanities, or obscenities. They proposed a Lexical Syntactic Feature (LSF) language model which uses profane words on noswearing.com and urbandictionary.com to detect the offensiveness of sentences. Researchers (Dadvar et al., 2012; Wong & Teh, 2020) have used different sets of profanity keywords to detect foul words and to seek the gender difference in the use of vulgar terms. The latter is a feature that can improve the performance of machine learning algorithm in classifying cyberbullying text. A higher occurrence of curse words in online comments is more prone to be detected as being bullying messages. Study has performed the test by training the density of bad words in the model (Huang et al., 2014). The study by Huang et al. (2014) has proven that most highly ranked textual features for cyberbullying detection are to identify the use of bad words. Following that, Zhao et al. (2016) and Teh, Cheng, and Chee (2018) have also highlighted that cyberbullying messages often contain curse or insulting words. Therefore, the occurrence of these words is a reasonable determination of bullying content. Singh et al. (2019) claimed in their study that informal language consisting of swear words is frequently used to direct abuse to the victim. In a study by Foong and Oussalah (2017) and Novalita et al. (2019) both swear words and the lexical database were used to detect cyberbullying. Although profanity/swear/curse dictionaries contain a rich repository and are used by many studies, profane slangs are evolving at a fast pace and the words used by different generations of users may varies. This causes list-based detections to perform poorly (Sood et al., 2012). To understand the language used in cyberbullying, Kontostathis et al. (2013) and Teh et al. (2018) carried out a study identifying words used by bullies on Formspring.me and YouTube's comments section, respectively. The former established queries for cyberbullying content detection, while the latter formed a list of commonly used profane words and its hate categories. The results can not only provide future researchers a clearer understanding of the evolution of foul words used by different user groups over time, but also contribute to indicating the abuser's profile.

Punctuation marks refer to characters such as comma, colon, question mark, semicolon and so on. The use of punctuation marks such as exclamation marks often implies a strong feeling when expressing emotions. Punctuation marks also indicate shouting or speaking in high volume (Huang et al., 2014). In Chen et al. (2012), exclamation marks were used as a feature in the LSF model to detect the level of offensiveness of YouTube users. In the study, it was pointed that the use of such punctuations can emphasise the level of offensiveness intended in a comment. Teh et al. (2015), who carried out an analysis on the effects of exclamation marks in textual comments via texting with 12 online sentiment tools, highlighted that most of the tools produced no score to show different in the expression using the various number of exclamation marks count towards the original set of words, but with the human coder (study to ask human to rate and label the expression), the different number of exclamation marks used in the text show significantly different expression. Which means, through human rating study, they found that the number of exclamation marks could actually influence the sentiment value of the message.

The use of uppercase often exhibits strong emotions. Unusual capitalisation in text, excluding its use in the initial letters of first words and named entities, may strongly indicate cyberbullying (Foong and Oussalah, 2017). According to Chen et al. (2012), the use of uppercase can determine the volume and feelings of a person. In showing the effect of uppercase use in a text, the study claimed that the sentence "You are STUPID" is more offensive than "You are stupid". Huang et al. (2014) and Chatzakou et al. (2017) used uppercase as one of the textual features to train the classifier in their studies to detect cyberbullying on Twitter. Pak & Teh (2018), who studied the value of expression behind letter capitalisation in product reviews, concluded that the use of uppercase is able to enforce the different levels of expression. The uppercase tends to make a positive review more positive, and a negative review more negative.

There are three types of personal pronouns; first (i.e.: "I"), second (i.e.: "you") and third (i.e.: "he, Alice"). According to Dadvar et al. (2012), harassing posts often contain the use of personal pronouns, particularly third personal pronouns (Singh et al., 2019). Dadvar et al. (2012) highlighted that second personal pronouns play a significant role in detecting online harassment. This qualifies second personal pronouns as an individual attribute in training the classifier, while other pronouns are qualified as another attribute. A message containing attributes relating to cyberbullying with a second personal pronoun has a strong likelihood of being a harassment. The use of personal pronouns gives others an idea about the person that the message is directed at (Al-Garadi et al., 2016). Foong & Oussalah (2017) used the Linguistic Inquiry and Word Count (LIWC) feature to capture second personal pronouns and the total number of pronouns in a given text. Sarna & Bhatia (2017) and Novalita et al. (2019) used a different approach in their studies, where they used five combinations of personal pronouns, bad words and words expressing negative emotions to indicate direct/indirect bullying, such as: (first + negative emotion + second). For example, the phrase "I hate you" could represent direct bullying. The integration of this feature allows for better cyberbullying detection, as sometimes a bad word or a word expressing negative emotions cannot in itself be determined to be directed towards a person or a thing. With the use of personal pronouns, the target of the ill-intended message can be easily identified which can then help determine the occurrence of bullying.

Emotion text refers to the use of words indicating positive or negative feelings in a text. According to Sarna and Bhatia (2017), bullying behaviour can be strongly indicated by negative emotions that leave an impact on the victim. To this, Foong and Oussalah (2017) used the LIWC feature to capture negative emotions and words that convey sadness, anxiety and anger for their cyberbullying detection system. Sarna and Bhatia (2017) also pointed that bullying text does not always have the occurrence of bad words. To overcome this problem, they incorporated a number of words expressing positive emotions in a text to help identify non-bullying text that consists of bad words. Similar to the previously mentioned studies, Novalita et al. (2019) also took this feature into consideration and used an emotion word list for detection. It is useful in identifying serious bullying messages by calculating the occurrence of negative emotion words. It also helps in avoiding non-bullying messages consisting of bad words from being classified as cyberbullying.

URLs and hashtags can be useful in cyberbullying detection. They are often used to direct users to a web page/content. According to Al-Garadi et al. (2016), URLs can be used to measure the activeness of a user in an online environment. They claimed that those who are "considerably active in online environments were likely to engage in cyberbullying behaviour". Chatzakou et al. (2017) who did a study on cyberbullying detection on Twitter also mentioned that normal users tend to use fewer URLs in their tweets compared to aggressors and bullies. Hashtags are also one of the features that help in detecting cyberbullying. Normal users tend to use less hashtags in single tweets. Multiple hashtags are mostly used by bullies (Balakrishnan et al., 2019) to propagate their attacking messages to more persons or groups. Sarna and Bhatia (2017) and Novalita et al. (2019) did not treat URLs as a single feature. Instead, they combined it with other features such as personal pronouns and bad words in their cyberbullying detection model. Sarna and Bhatia (2017) mentioned that a message which consists of pronouns, URLs and bad words may have embarrassing content regarding the victim that is made public.

The aforementioned textual features are useful in identifying cyberbullying. However, Navarro and Jana (2012) highlighted that a person's sociability in the online environment (network feature) has a strong correlation with cyberbullying behaviour.

2.2. Network features

Network features include 1) the number of followers and followees, 2) number of likes, 3) the number of shared media and mentioned users, 4) account creation date, and 5) user features. Network features refer to attributes indicating the social ability of a person in an online environment. There are total of seven common network features

Number of followers and followees refers to the number of followers and followees indicates the amount of followers and followees that a user has in his/her social media account. Hosseinmardi et al. (2015), in their study on detecting cyberbullying on Instagram, mentioned that users with more followers tend to be more popular and are thus more prone to drawing negative comments from others. Chatzakou et al. (2017) also highlighted that normal users tend have friends as followers instead of strangers who may practise cyberbullying. It can be said that users with a high number of followers have higher popularity than normal users, and vice versa. By integrating this feature, the performance of cyberbullying detection can be enhanced. The proposed cyberbullying detection model by Hosseinmardi et al. (2015) which uses the Linear support vector machine classifier recorded an increased accuracy from 52% to 87% when textual features and image features were used with network features. However, not every network feature affects the detection.

Number of likes refers to the number of like that the user's post have. (Balakrishnan et al., 2019). Although this feature was included by Hosseinmardi et al. (2015) and Balakrishnan et al. (2019) in their detection models, the former observed that the correlation between this feature and cyberbullying is less significant. Thus, it may not be necessary to include the feature in building the detection model.

Number of Shared Media and Mentioned users refers the amount of posts a user has posted or shared on social media and the number of other users that are tagged in a post. (Balakrishnan et al., 2019). Although Hosseinmardi et al. (2015) mentioned that the number of shared media has no significant correlation with cyberbullying, this feature was included in the model proposed by Al-Garadi, Varathan and Ravana (2016), Chatzakou et al. (2017) and Balakrishnan et al. (2019). In particularly, Chatzakou et al. (2017) found that "bullies post less, participate in fewer online communities, and are less popular than normal users. Aggressors are relatively popular and tend to include more negativity in their post".

On the account creation date that refers to the labelled dataset on time period, Chatzakou et al. (2017) studied their labelled dataset on two time periods of the Twitter user accounts in their dataset. They found that approximately 38% of users who were detected as bullies at the earlier time period had deleted their Twitter accounts at the later time period. They claimed that the deletion was perhaps to prevent their accounts from being suspended temporarily or permanently by Twitter for being spam, fake or abusive. The deletion may also help bullies hide their identities. According to Ribeiro et al. (2017), bullies tend to have a later account creation date than normal users. To hide their true identities, bullies create other accounts instead of using their real accounts to cyberbully others. After a period of time, these later accounts will be deleted. Thus, the creation date of an account can provides useful information in classifying online bullying.

2.3. Users features

It indicates the profile of a user (e.g age and gender). Dadvar et al. (2012) showed that incorporating gender information in training classifiers through the use of the support vector machine model can improve cyberbullying detection. This is because there is a difference in the way a female bully and a male bully uses foul and inflammatory language online. The study showed that the mentioned approach can enhance the baseline detection by 39%. Other than that, Al-Garadi et al. (2016) included user features such as gender and age in their study. However, the information provided by users in an online environment can be inaccurate as they tend to exclude their personal information from social media. This poses a challenge in obtaining user features. To address this, Wong & Teh (2020) proposed to predict users' gender and age by forming a list of words most used by different genders. They also assumed that the first name registered in users' twitter accounts can be used to determine their gender.

All the aforementioned features used in previous cyberbullying detection models are useful in identifying harassment in an online environment. However, their proposed research method did not end with the implementation part and were not included with the integration. Hence, there is a need to implement the models in a way that allows users to use them anytime. This is to prevent bullying or manage the aftermath after cyberbullying cases are detected.

2.4. Existing (or Available) cyberbullying applications

There are several types of related application available to detect cyberbully online. Table 1 presents a comparison between these applications.

ReThink is a mobile application which sends users a warning message when users try to send texts consisting of harmful words ReThink—Stops Cyberbullying—Google Play App. (n.d.), 2020. Users have to download the application and change the keyboard setting to allow it to work. While the objective of the application is to minimise potential bullying behaviour, the algorithm behind the application is based on simple keystroke logging whereby the application only detects vulgar words found within a string of characters. The application is thus unable to detect harmful content where vulgar words are not used (Lempa, Ptaszynski and Masui, 2015).

Table 1
Comparison between applications.

Application name	Method	Limitation
ReThink	Detects vulgar words	Keyboard settings need to be changed to allow the application to work
Cyberbully Blocker	1) Classifies test using a brute force search algorithm that is trained with language modelling methods, or 2) Allows users to input texts and choose the cyberbullying detection method preferred	Developed to test the performance of used algorithms rather than to be applied to real-life scenarios
BullyBlocker	Computes based on indicators, and detects and evaluates offending words with other factors such as children's gender, age, etc. through TS algorithm	Only shows why someone is detected as a cyberbullying victim, and does not show the text detected
AbuSniff	Suggests user action from the set {"unfriend", "unfollowing", "restrict access", "sandbox", "ignore"} for features such as common photo count, mutual friend count, and so on using supervised learning algorithm	To detect potential cyberbully perpetrator, users need to log into their Facebook accounts in AbuSniff and manually fill out a questionnaire about a random person from their friends' list

Cyberbully Blocker is an Android mobile application which provides two methods of harmful messages detection. This application involves two methods in the development. The first method is to classify test using brute force search algorithm that is trained with language modelling methods. The second method works by using a list of seed word, with three categories to obtain the semantic orientation score and then maximise the relevance of categories (Lempa, Ptaszynski and Masui, 2015). The application allows users to input texts and choose the cyberbullying detection method they prefer. The application has a feedback feature which displays the result of detection to users. The limitation of the application is that it was developed to test the performance of used algorithms rather than being applied to real-life scenarios.

BullyBlocker is a mobile application that focuses on cyberbullying detection on the social networking site, Facebook. The application was designed mainly for parents and guardians of adolescents. Adolescents who are monitored by the application need to log into facebook account on the bullyblocker. A feature called Bullying Rank will display to parents to show result if the adolescent's FB posts contain any bully component. It is computed based on a series of complicated indicators as well as algorithms that will detect and evaluate offending words with other factors such as child's gender, age and so on. After detection, the application provides a list of helpful anti-cyberbullying resources such as anti-bullying organisations and hotlines to assist parents whose children are identifying if there is any potential of cyberbullying (Silva et al., 2018). While the application shows the reason why someone is detected as a cyberbullying victim only, it does not show the text detected. Parents or guardians will be unable to determine whether the alleged bullying text is detected accurately.

AbuSniff is a system that identifies Facebook friends perceived as strangers or abusive, and protects the user by unfriending, unfollowing or restricting the access to information for such friends" (Talukder and Carbunar, 2018). To use this system, users need to log into their Facebook accounts in the AbuSniff. The system lets users fill out a questionnaire about a random person from their friends list. Based on Abuse Prediction Module, which is one of the components of the system that trained on several features such as common photo count, mutual friend count and so on using supervised learning algorithm, it will suggest users the action from the set {"unfriend", "unfollowing", "restrict access", "sandbox", "ignore"}. Users can choose to ignore or proceed with the action suggested by the system (Talukder and Carbunar, 2018). This application is intended for teenagers to safeguard themselves and their own personal information against potential bullies. According to Talukder and Carbunar (2018), some users do not mind having people who have been detected as potential bullies in their friends' list. This attitude may make Facebook users more vulnerable to cyberbullying.

This paper aims to develop a mobile application with a two-fold function: detecting cyberbullying content on Twitter, and alerting adolescent users' parents and guardians towards whom the content is directed. The application involves the parents or guardians in monitoring their children or adolescents under their care to reduce cyberbullying instances and encourage them to instil knowledge of proper Internet use to the children.

2.5. Classifier

There are numerous classifier algorithms such as support vector machine (SVM) (Rafiq et al., 2015; De-La-Pena-Sordo et al., 2016; Tulkens et al., 2016; Shende and Deshpande, 2017) Naive Bayes (Nandhini and Sheeba, 2015; Srinidhi Skanda et al., 2017), logistic regression (LR) (Davidson et al., 2017; Srinidhi Skanda et al., 2017), decision tree (DT) (Kontostathis et al, 2013), K-means neural network (KNN) (Ozel et al, 2017), random forest (RF) (Al-Garadi, et al. 2016), and AdaBoost (AB) (Mukherjee et al., 2017; Ribeiro et al., 2017). Classification involves two phases: training and testing. Training consists of giving the data to the classifier which then reads and parses the data, while testing involves taking the data output from the training phase to predict offensive content (Shende and Deshpande, 2017).

Deep learning, also known as hierarchical learning, is a subtype of machine learning that is different from other type-specific algorithms such as supervised, US, semi-supervised, and is able to learn from data representation. It has been introduced to formally apply artificial intelligence (AI) (Gulcehre, 2015). Deep learning uses deep neural network (NN) to learn features from the input data using its built-in multiple staked layers. The algorithms of deep learning are able to create new features from the input data, which proves to be more effective. Studies of Agrawal and Awekar (2018), Alorainy et al. (2018), Pitsilis et al. (2018), and Zhang and Luo (2018) applied deep learning methodology. Table 2 presents an analysis and summary of classification techniques with its features.

In this study, the first objective is to discover a set of useful features in training a machine learning algorithm to detect cyberbullying content. The second is to integrate a machine learning model into a mobile application to help parents detect cyberbullying directed towards their children.

3. Methods

The application has two major components: the mobile application component and the machine learning component.

3.1. Mobile application

A mobile application was developed using Android Studio (as in Fig. 1). The interface file in the Android Studio Environment is responsible for the mobile application's graphical user interface (GUI). Each interface corresponds to one or more logical java files that handle the input of user and provide respective output. The mobile application was built by combining the two components.

Table 2
Analysis and summary of classification techniques.

References	Method [Classifier]	Features
(Pitsilis et al., 2018)	Deep Learning [Long term short-term memory (LSTM)]	User-related information, word frequency, vectorisation
(Agrawal and Awekar, 2018)	Deep Learning (BLSTM)	Word embedding, transfer learning
(Van Hee et al., 2018)	Supervised [(SVM)]	BoW, Subjectivity lexicon features, topic models, character N-gram, word N-grams, terms lists
(Alorainy et al., 2018)	Supervised learning [Multilayer perceptron (MLP), LR]	Othering lexicon + doc2vec
(Zhang and Luo, 2018)	Deep Learning [(Base + Gated Recurrent Units (GRU))]	Word embedding
(Founta et al., 2018)	Deep Learning	Metadata, word embedding (word2vec, GloVe)
(Watanabe et al., 2018)	Supervised [J48graft]	Semantic features, Unigrams features, Pattern features
(Ribeiro et al., 2017)	Semi-supervised [graphsage]	GloVe embedding, network/activity features
(Magu, et al., 2017)	Supervised [SVM]	BoW
(Gao and Huang, 2017)	Supervised [LR, Neural network (NN), Ensemble Model]	Character N-grams and word N-grams, user features, linguistic features, sentiment polarities, emoticons NRC LSTM with attention
(Srinidhi Skanda et al., 2017)	Supervised [LR]	Keyword embedding (Distributed memory for sentence vectors, keywords for stance)
(Ozel et al., 2017)	Supervised [NB, SVM, KNN, J48]	Chi-square (CHI2) and Information gain (IG), Emoticons
(Benigni, Joseph and Carley, 2017)	IVCC [Multiplex vertex]	Metadata: user account features, spectral and node metric representations of followings, mentions, and user-by-user (shared hashtag) networks
(Shende and Deshpande, 2017)	Supervised [SVM, NB]	Tokenization, term frequency, TF-IDF, n-gram
(Salguero and Espinilla, 2018)	Flexible text analyzer [Weka]	Ontology-based
(Vishwamitra et al., 2017)	[Pronunciation-based Convolutional neural network (PCNN)]	Keywords matching
(Davidson et al., 2017)	Supervised [LR]	N-gram, sentiment lexicon, TF-IDF, syntactic features
(Malmasi and Zampieri, 2017)	Supervised [SVM]	N-grams, character N-grams, word skip-grams
(Di Capua et al., 2016)	Unsupervised [Growing Hierrachical self organizing Map (GSHOM)]	Social features, semantic features, sentiment features, syntactic
(Rafiq et al., 2015)	Supervised [AdaBoost]	Metadata, n-gram
(De-La-Pena-Sordo et al., 2016)	Semi-supervised [Eucledian distance, SVM]	Syntactic features, statistical features, opinion features, n-grams
(Tulkens et al., 2016)	Supervised [SVM]	Dictionary (LIWC for Dutch, word2Vec)
(Zhao et al., 2016)	Supervised [Linear SVM]	BoW, Semantic-enhanced BoW, LSA, LDA,
(Al-Garadi et al., 2016)	Supervised [RF]	Metadata, vulgarity features, user features
(Liu and Forss, 2015)	Supervised [NB, SVM]	N-grams, BoW, topic features, semantic, senti-ment analysis, meta infor-mation, tf-idf
(Burnap and Williams, 2015)	Supervised [SVM]	BoW, Typed de-pendencies, *syn*-tactic, N-grams, lexicons
(Mukherjee et al., 2017)	Supervised [Linear SVM]	N-gram, User and media information
(Hosseinmardi et al., 2015)	Supervised [Linear SVM, LR]	N-gram features, social graph features
(Nandhini and Sheeba, 2015)	Unsupervised [NB]	BoW, grammatical features, fuzzy rules, genetic algorithms
(Dinakar et al., 2012)	Commonsense reasoning [SVM]	Commonsense reasoning
(Djuric et al., 2015)	Supervised [LR]	Paragraph2vec, CBoW, embed-ding
(Rafiq et al., 2015)	Supervised [NB, AdaBoost, Decision Tree, RF]	Meta information, N grams
(Reynolds et al., 2011)	Supervised [J48 decision tree]	Bad word lexical features
(Chen et al., 2012)	Supervised [SVM]	Lexical syntactic feature model (BoW, N-gram, sentiment analysis, syntactic features)
(Xiang et al., 2012)	Semi-supervised [LR]	Topical features, sentiment analysis

Overall, this study enhances reviewed existing mobile application and techniques by developing a machine learning model and integrating it into a mobile application that allow parents to use. Table 3 descript the software and tools used in development.

3.2. Machine learning model

Machine learning is to allow the application to classify cyberbullying tweets. It was not developed using Android Studio because it requires a large computing power for training. If it was developed in the same environment as the mobile application, the performance of the application would degrade. It was only after the machine learning model was trained that it could be integrated into the mobile application, which can be used by simply issuing a call to the model. To evaluate the trained model, standard matrices such as Precision, Recall and Accuracy were used. Table 4 provides description of the tools used in Machine Learning.

The first stage in creating the machine learning model is Twitter data collection for analysis. The tweets were crawled using several hashtags, including #cyberbullying, #bullying, #stopbullying and several cyberbullying-related keywords suggested by Kontostathis et al. (2013). A total of 5000 tweets were obtained and manually reviewed by developer. Tweets containing similar content with other tweets, languages other than English, or ambiguous meaning were removed from the dataset. The collection of tweets then underwent a labelling process based on the definition mentioned in Chatzakou et al. (2017) and Novalita et al. (2019). The tweets will undergo a labelling process: label "1″ indicates a tweet as being related to cyberbullying, while label "0" indicates it as a normal tweet. A tweet is labelled with "1" if it con-

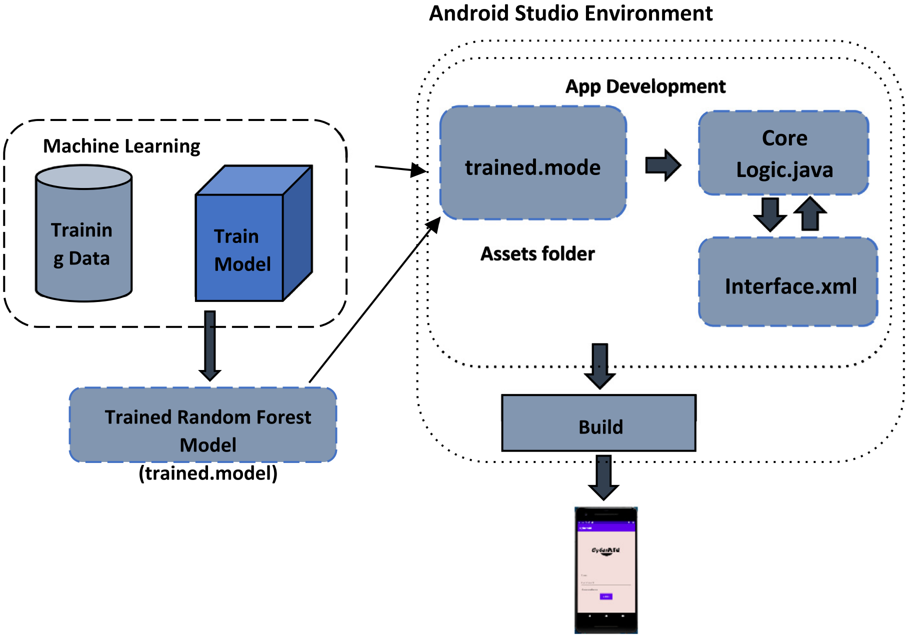

Fig. 1. Proposed framework.

Table 3
Description of the software and tools used for mobile application development.

Category	Software Name	Description
Programming Language	JAVA	A general-purpose programming language intended to let application developers write once, run anywhere.
Software	Android Studio	A software to build apps on every type of Android device.
Virtual Device	Pixel 2 API R	A configuration that defines the characteristics of an Android phone, tablet, Wear OS, Android TV, or Automotive OS device that are to be simulated in the Android Emulator.
Database	SQLite	SQLite is a C-language library that implements a small, fast, self-contained, high-reliability, full-featured, SQL database engine.
API	TwitterAPI	Twitter API provides the tools needed to contribute to, engage with, and analyse conversations on Twitter. It allows integration of Twitter to the mobile application.
Main Libraries	Stanford CoreNLP	A set of natural language analysis tools written in Java to retrieve the sentiment score of the sentence.
	wekaStripped	A port of weka 3 to the Android platform to implement machine learning.

Table 4
Description of the tools used for machine learning.

Category	Tool	Description
Programming language	Python	An interpreted, high-level and general-purpose programming language.
Software	Spyder	A free and open-source scientific environment written in Python, for Python, and designed by and for scientists, engineers and data analysts.
Main Libraries	Numpy	Enable numerical computing with Python.
	Pandas	An open-source data analysis and manipulation tool, built on top of the Python programming language.
	Scikit-learn	Tools for predictive data analysis.
	Imbalanced-learn	A python package offering several re-sampling techniques commonly used in datasets showing strong between-class imbalance.

tains a harmful message or is repeatedly sent to a specific user by the same person. Else, it is labelled with "0". After eliminating all unwanted tweets, a total of 1200 tweets were acquired whereby 1000 were related to non-cyberbullying and 200 were related to cyberbullying. The dataset was split into two sets whereby 75% was of the training set while 25% was of the testing set. All data was stored in a CVS file for further analysis. To handle the imbalanced dataset, Synthetic Minority Oversampling Technique (SMOTE) is applied (Brownlee, 2020). SMOTE works by selecting examples that are close in the feature space, drawing a line between the examples in the feature space and drawing a new sample at a point along that line (Brownlee, 2020).

In stage 2, the tweets is pre-processed where tweet components such as numbers, extra spaces, and "at" or "@" symbols (which indicate replies in a tweet). This is to eliminate irrelevant components that will slow down the process of feature extraction. The collected data was analysed using Spyder (Spyder Website, 2018), a development environment for Python programming language. To eliminate elements that would not help in cyberbullying detection and to reduce the dimension of feature vectors, the data underwent several pre-processes using the Python RegEx module and Python Natural Language Toolkit (NLTK) to remove extra white spaces, numbers, "@" symbols, and "RT"s or retweets. The purpose of this step was to clear the data from irrelevant components.

In stage 3, a set of features is extracted from the tweets to be analysed and chosen as useful features in training the model. The operations were performed to analyse and obtain a set of features that can train the machine learning algorithm for classification purposes. A total of 11 features were extracted for analysis. The obtained features are as follows:

(a) Sentiment Value - A cyberbullying-related text is likely to have negative sentiments, thus this feature may be useful in classifying cyberbullying tweets and distinguishing them from non-cyberbullying tweets. The sentiment value was obtained using VADER (Pandey, 2018), which is a sentiment analysis tool that calculates the polarity of text ranging from 1 (indicating positive sentiment) to 0 (indicating negative sentiment). The tool's ability to handle punctuations, exclamation marks, emoticons (Teh et al., 2016) and so on makes it an ideal tool for this study.

(b) Exclamation Mark Count - The usage of exclamation marks often shows strong feelings and implies "shouting" or speaking in high volume (Huang et al., 2014). This situation is likely to arise when a user is sending a harmful message to another. Thus, this feature is taken into consideration in training the machine learning model to detect cyberbullying.

(c) Hashtag Count - This feature counts the number of hashtags used in a tweet. Hashtags used in this study for data collection purposes were not considered.

(d) Profanity Count - According to Huang et al. (2014), a cyberbullying-related text often involves the usage of profane words. The present study made use of the words lists from noswearing.com and Teh et al. (2018) to search for profane words in tweets. As profanity is more associated with hatred, profanity-based methods in hate speech detection could be effective (Teh & Cheng, 2020). Hence, this feature is taken into consideration.

(e) Emoticon Count - This feature counts the number of emoticons used in a tweet. With the help of Python's library EMOT, which consists of the Unicode and text forms of emoticons, all emoticons were converted into text form for counting and vectorisation purposes. Similar to sentiment value, emoticon is the expression of emotion including hatred and is likely to arise as a harmful message. Furthermore, sociological studies have suggested that emotional information can be used to better understand bullying behaviour, a finding which leads to the selection of this feature.

(f) Word Count - This feature counts the total number of words in a tweet. Bullying is defined as an aggressive, intentional act or behaviour that is carried out by a group or an individual repeatedly and over time against a victim who cannot easily defend him or herself (Oweus, 1993). The word count feature is included through the counting of the total number of tweets that are able to represent repetitive action of an individual against a victim.

(g) Personal Pronoun Count - This feature counts the number of personal pronouns used in a tweet. Since a cyberbullying text is often directed at a particular person (victim), it might consist of more personal pronouns than normal text.

(h) Account Creation (days) - This feature counts the account creation in days by subtracting the date a cyberbullying tweet is posted from the account creation date. An account is considered newly created if the value obtained by the calculation is less than or equal 183 days, indicate half a year.

(i) Followers Ratio - This feature determines the ratio by dividing the number of a user's followers by the number of followees. The value acquired by the calculation will range from 0 to >1. A user with a followers ratio of more than 1 means that they have more followers than followees, indicating his/her popularity on social media.

(j) Combination of Personal Pronouns and Negative/Profane Words **-** This feature is to find a certain text pattern in a tweet that could result in cyberbullying. Below are the patterns that this study considered. The negative words list used was constructed by Huang et al. (2014).

1. Second personal pronouns (he, she, it, etc.) combine negative or profane word. (e.g.: He is such an idiot)
2. Third personal pronouns (they, them, etc.) combine negative or profane word. (e.g.: They looks ugly)
3. First personal pronouns (I, we, etc.) combine negative word and second or third personal pronouns. (e.g.: I hate you)

(k) Count Vectorisation - It is impossible for a machine learning algorithm to work directly with text. Thus the text has to be converted to a numeric form through vectorisation. The present study made use of Python Scikit-Learn's CountVectorizer to count the occurrence of each word, known as token, in a text and to use the value as its weight (Russell, 2017).

Stage 4 is to integrate a machine learning model into the mobile application. Stage 5 is to perform feature extraction to train the machine learning algorithm to classify cyberbullying and non-cyberbullying tweets. The dataset was split into two sets whereby 75% was of the training set while 25% was of the testing set. Before training was carried out, Synthetic Minority Oversampling Technique, a technique to handle imbalance dataset, was applied (Brownlee, 2020). The technique works by generating new data from the minority class via duplication. It does not provide the model with additional information but instead makes the two classes balanced. According to Novalita et al. (2019), an RF classifier's performance varies based on the parameters used, such as the depth of trees, the number of trees, and so on. To overcome inconsistencies, this study implemented Grid Search with Cross-Validation (GridSearchCV) by Scikit-Learn to obtain the best combination of parameters to be used in RF. GridSearchCV works by evaluating all possible combinations based on the pre-defined parameter, and shows the best result (Koehrsen, 2018). The best combinations of parameters were acquired, with n_estimators, min_samples_leaf, 3 and 1000; max_features, log2; criterion, gini; max_depth, 50; min_samples_split, 10 and random_state,15.

Stage 6 is to evaluate the classification performance of the machine learning algorithm. A prediction of class on test data was decided based on votes gathered from the different decision trees in the forest, where "1″ indicates cyberbullying and "0" indicates non-cyberbullying. 75% of the tweets are used for training while the remaining 25% are used for testing. The matrices used for evaluation purpose are Confusion Matrix, Accuracy, Precision and Recall. Table 5 shows the results and the explaination of the improvement of the model following it.

4. Result and analysis

To evaluate the performance of the proposed model, the model was tested with a 25% testing data by using several matrices, including Precision, Recall and Accuracy. These matrices were used because they could measure the effectiveness of the model even when the distribution classes were not balanced. This was particularly helpful in this study as the portion of the non-cyberbullying class was significantly more than that of the cyberbullying class (Balakrishnan et al., 2019). In particularly, Precision was calculated using the formula, Precision = $T_p / (T_p + F_p)$, while Recall was calculated using the formula, Recall = $T_p / (T_p + F_N)$. Accuracy meanwhile was calculated using the formula $(T_p + T_N)/(T_P + T_N + F_P + F_N)$ (Novalita et al., 2019).

From Table 5, random forest shows the highest accuracy score of 92%. The non-cyberbullying class achieved a 94% precision score and a 97% recall score, while the cyberbullying class only achieved a 79% precision score and a 61% recall score. To improve the performance of the model, an evaluation of the features used to train the model was carried out. The evaluation was done by implementing a method, namely feature_importance provided by Scikit-Learn, to

Table 5
Performance of the Machine Learning Model.

Features	Classes	Precision	Recall	Accuracy Score
Random Forest	0 (non-cyberbullying)	94%	97%	92%
	1 (cyberbullying)	79%	61%	
Decision Tree	0 (non-cyberbullying)	93%	94%	89%
	1 (cyberbullying)	64%	61%	
Decision Tree/Regression	0 (non-cyberbullying)	93%	92%	87%
	1 (cyberbullying)	56%	61%	
SVM with kernel = "rbf"	0 (non-cyberbullying)	94%	96%	91%
	1 (cyberbullying)	75%	61%	
SVM with kernel = "sigmoid"	0 (non-cyberbullying)	97%	90%	90%
	1 (cyberbullying)	60%	86%	
Gaussian Naïve Bayes	0 (non-cyberbullying)	88%	83%	76%
	1 (cyberbullying)	27%	36%	
Complement Naïve Bayes	0 (non-cyberbullying)	97%	87%	86%
	1 (cyberbullying)	52%	82%	
Naïve Bayes (Bernouli)	0 (non-cyberbullying)	97%	87%	86%
	1 (cyberbullying)	52%	82%	

calculate the score according to the Gini Impurity reduction used to choose the split points. Based on the score, it output the ranking of each feature indicating the importance of them in classification (Gluck, 2017). The five most useful features (e.g. Word count, Followers Ratio, Combination of Personal Pronouns, Profane Count and "bully") for classification based on the results produced by the method.

To enhance the proposed model and form a set of features useful for cyberbullying detection, further evaluation on other features not in the top five features ranking was carried out manually. By doing this, this study discovered that the features "Emoticon Count", "Exclamation Mark Count", "Account Creation (days)" did not contribute to classification. This is because the usage of emoticons and exclamation marks in the collected dataset was considerably less in cyberbullying and non-cyberbullying tweets. Ribeiro et al. (2017) pinpointed that hateful users are likely to have a later account creation date than normal users to hide their true identity. However, in this study, the percentage of users who are not involved in cyberbullying is almost the same as those who have a late account creation date. This may be due to some normal users not wanting to reveal their identity when protecting cyberbullying victims. Thus, these users create a new account to do so instead of using their own account. The account creation date feature was, therefore eliminated since it could not effectively differentiate the two classes in this study.

After removing those features, training and testing were carried out again to evaluate the performance of the new model using the same matrices as the previous model. It can be observed that the performance of the model has slightly improved where the precision of cyberbullying class has increased from 79% to 84%, and the recall rate rose from 61% to 70%. Overall, the accuracy score increased by 2%. It proved that the elimination of the three features improve performance.

Compared to the mobile application Rethink ReThink—Stops Cyberbullying—Google Play App. (n.d.), 2020, this study implemented a machine learning model instead of a keyword-based approach. Compared to Cyberbully Blocker (Lempa et al., 2015), this application provides a workable and user-friendly GUI for parents. Compared to BullyBlocker (Silva et al., 2018) which displays a score indicating whether or not a protected child is involved in cyberbullying, this proposed model allows parents to review the actual text detected as having cyberbullying sentiments, and to save it for later investigation. AbuSniff (Talukder & Carbunar, 2018), a mobile application that suggests a list of actions such as "unfollow", "unfriend", "ignore" to users whose Facebook friends are suspected to be involved in cyberbullying, is a self-monitoring tool rather than a tool that parents can use to monitor their child. Users who choose to ignore taking any action will put themselves at risk of cyberbullying. Therefore, to overcome such a problem, the proposed application helps parents detect cyberbullying activities from or towards their child.

5. Conclusion

This study has analysed various features and existing mobile applications that detect cyberbullying on social media. By evaluating the features, the results have shown that the study has identifed a set of useful features to detect cyberbullying and built a model based on those features. The results have also shown that the proposed mobile application is able to implement the machine learning model and provide a GUI allowing parents to use it for cyberbullying detection among their children. In summary, this study has proposed a mobile application which integrates a machine learning model that can assist parents in cyberbullying detection among their children.

In the future, other contexts such as an image as well as video can be taken into consideration for cyberbullying detection. Since Twitter allows users to post text with images and videos, it is crucial to also analyse the content of those images and videos to enhance the performance of detection. Moreover, the proposed mobile application is developed for Android device users, further work can be done by using frameworks such as React Native to build the application so that both Android and IOS users can get benefited.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Agrawal, S., & Awekar, A. (2018). Deep learning for detecting cyberbullying across multiple social media platforms. In Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). 10772 LNCS (Issue Table 2), 141–153. https://doi.org/10.1007/978-3-319-76941-7_11.

Al-Garadi, M.A., Varathan, K.D., Ravana, S.D., 2016. Cybercrime detection in online communications: the experimental case of cyberbullying detection in the Twitter network. Comput. Hum. Behav. 63, 433–443. https://doi.org/10.1016/j.chb.2016.05.051.

Alorainy, W., Burnap, P., Liu, H., & Williams, M. (2018). The Enemy Among Us: Detecting Hate Speech with Threats Based "Othering" Language Embeddings. ArXiv E-Prints, arXiv:1801.07495.

Balakrishnan, V., Khan, S., Fernandez, T., & Arabnia, H. R. (2019). Cyberbullying detection on Twitter using Big Five and Dark Triad features. Personality and Individual Differences, 141(September 2018), 252–257. https://doi.org/10.1016/j.paid.2019.01.024.

Benigni, M.C., Joseph, K., Carley, K.M., 2017. Online extremism and the communities that sustain it: Detecting the ISIS supporting community on Twitter. PLoS ONE 12 (12), 1–23. https://doi.org/10.1371/journal.pone.0181405.

Brownlee, J. (2020). SMOTE for Imbalanced Classification with Python. Machine Learning Mastery. https://machinelearningmastery.com/smote-oversampling-for-imbalanced-classification/.

Burnap, P., Williams, M.L., 2015. Cyber hate speech on Twitter: An application of machine classification and statistical modeling for policy and decision making. Policy Internet 7 (2), 223–242. https://doi.org/10.1002/poi3.85.

Chatzakou, D., Kourtellis, N., Blackburn, J., De Cristofaro, E., Stringhini, G., & Vakali, A. (2017). Mean birds: Detecting aggression and bullying on Twitter. WebSci 2017 - Proceedings of the 2017 ACM Web Science Conference, 13–22. https://doi.org/10.1145/3091478.3091487.

Chen, Y., Zhou, Y., Zhu, S., & Xu, H. (2012). Detecting offensive language in social media to protect adolescent online safety. Proceedings - 2012 ASE/IEEE International Conference on Privacy, Security, Risk and Trust and 2012 ASE/IEEE International Conference on Social Computing, SocialCom/PASSAT 2012, 71–80. https://doi.org/10.1109/SocialCom-PASSAT.2012.55.

Dadvar, M., Ordelman, R., De Jong, F., & Trieschnigg, D. (2012). Improved cyberbullying detection using gender information. Dutch-Belgian Information Retrieval Workshop, DIR 2012, May 2014, 23–26. http://purl.utwente.nl/publications/79872.

Davidson, T., Warmsley, D., Macy, M., & Weber, I. (2017). Automated hate speech detection and the problem of offensive language. Proceedings of the 11th International Conference on Web and Social Media, ICWSM 2017, 512–515.

De-La-Pena-Sordo, J., Pastor-Lopez, I., Ugarte-Pedrero, X., Santos, I., Bringas, P.G., 2016. Anomaly-based user comments detection in social news websites using troll user comments as normality representation. Logic J. IGPL 24 (6), 883–898. https://doi.org/10.1093/jigpal/jzw043.

Di Capua, M., Di Nardo, E., & Petrosino, A. (2016). Unsupervised cyber bullying detection in social networks. 2016 23rd International Conference on Pattern Recognition (ICPR), 432–437. https://doi.org/10.1109/ICPR.2016.7899672.

Dinakar, K., Picard, R., Lieberman, H., Jones, B., Havasi, C., Lieberman, H., Picard, R., Lieberman, H., 2012. Common sense reasoning for detection, prevention, and mitigation of cyberbullying. IJCAI Int. Joint Conf. Artif. Intell. 2 (3), 4168–4172. https://doi.org/10.1145/2362394.2362400.

Djuric, N., Zhou, J., Morris, R., Grbovic, M., Radosavljevic, V., & Bhamidipati, N. (2015). Hate Speech Detection with Comment Embeddings. Proceedings of the 24th International Conference on World Wide Web, 29–30. https://doi.org/10.1145/2740908.2742760.

Expert System Team (2020), What is Machine Learning? A Definition. http://www.expert.ai/blog/machine-learning-definition/.

Foong, Y. J., & Oussalah, M. (2017). Cyberbullying system detection and analysis. Proceedings - 2017 European Intelligence and Security Informatics Conference, EISIC 2017, 40–46. https://doi.org/10.1109/EISIC.2017.43.

Founta, A.-M., Chatzakou, D., Kourtellis, N., Blackburn, J., Vakali, A., & Leontiadis, I. (2018). A unified deep learning architecture for abuse detection. WebSci '19: Proceedings of the 10th ACM Conference on Web Science, 105–114. https://doi.org/10.1145/3292522.3326028.

Gao, L., & Huang, R. (2017). Detecting online hate speech using context aware models. Proceedings of Recent Advances in Natural Language Processing, 260–266. https://doi.org/10.26615/978-954-452-049-6_036.

Gluck, C. (2017). Running Random Forests? Inspect The Feature Importances With This Code. Medium Towards Data Science. https://towardsdatascience.com/running-random-forests-inspect-the-feature-importances-with-this-code-2b00dd72b92e.

Gulcehre, C. (2015). Deep Learning. Welcome to Deep Learning.

Holzinger, A., 2016. Interactive machine learning for health informatics: when do we need the human-in-the-loop?. Brain Inform. 3 (2), 119–131. https://doi.org/10.1007/s40708-016-0042-6.

Hosseinmardi, H., Mattson, S. A., Rafiq, R. I., Han, R., Lv, Q., & Mishra, S. (2015). Analyzing labeled cyberbullying incidents on the instagram social network. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 9471, 49–66. https://doi.org/10.1007/978-3-319-27433-1_4.

Huang, Q., Singh, V. K., & Atrey, P. K. (2014). Cyber bullying detection using social and textual analysis. SAM 2014 - Proceedings of the 3rd International Workshop on Socially-Aware Multimedia, Workshop of MM 2014, 3–6. https://doi.org/10.1145/2661126.2661133.

i-SAFE Inc. (2019). http://auth.isafe.org/outreach/media/media_student_behavior.

Koehrsen, W. (2018). Hyperparameter Tuning the Random Forest in Python - Towards Data Science. Medium Towards Data Science. https://towardsdatascience.com/hyperparameter-tuning-the-random-forest-in-python-using-scikit-learn-28d2aa77dd74.

Kontostathis, A., Reynolds, K., Garron, A., & Edwards, L. (2013). Detecting cyberbullying: Query terms and techniques. WebSci '13: Proceedings of the 5th Annual ACM Web Science Conference, May 2013, 195–204. https://doi.org/10.1145/2464464.2464499.

Lempa, P., Ptaszynski, M., & Masui, F. (2015). Cyberbullying Blocker Application for Android. 7th Language & Technology Conference, December, 408–412.

Liu, S., & Forss, T. (2015). Text Classification Models for Web Content Filtering and Online Safety. 2015 IEEE International Conference on Data Mining Workshop (ICDMW), 961–968. https://doi.org/10.1109/ICDMW.2015.143.

LLC, P. (2019). PocketGuardian Parents - Stop Cyberbullying - Apps on Google Play. https://play.google.com/store/apps/details?id=com.gopocketguardian.parentapp&hl=en.

Magu, R., Joshi, K., & Luo, J. (2017). Detecting the hate code on social media. Proceedings of the Eleventh International AAAI Conference on Web and Social Media (ICWSM 2017), 608–612.

Malmasi, S., & Zampieri, M. (2017). Detecting hate Speech in social media. RANLP 2017—Recent Advances in Natural Language Processing Meet Deep Learning, 467–472. https://doi.org/10.26615/978-954-452-049-6_062.

Mukherjee, S., Cain, N., Walker, J., White, D., Ray, I., & Ray, I. (2017). POSTER. Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services—MobiSys '15, 2559–2561. https://doi.org/10.1145/3133956.3138844.

My Mobile Watchdog—Features. (2001). https://www.mymobilewatchdog.com/features/.

Nandhini, B.S., Sheeba, J.I., 2015. Online social network bullying detection using intelligence techniques. Procedia Comput. Sci. 45 (C), 485–492. https://doi.org/10.1016/j.procs.2015.03.085.

Navarro, J.N., Jana, L.J., 2012. Going Cyber: Using Routine Activities Theory to Predict Cyberbullying Experiences. Sociological Spectrum 32 (1), 81–94. https://doi.org/10.1080/02732173.2012.628560.

Novalita, N., Herdiani, A., Lukmana, I., Puspandari, D., 2019. Cyberbullying identification on Twitter using random forest classifier. J. Phys. Conf. Ser. 1192 (1). https://doi.org/10.1088/1742-6596/1192/1/012029.

Oweus, D., 1993. Bullying at School: What We Know and What We Can Do. Blackwell, Oxford.

Ozel, S. A., Sarac, E., Akdemir, S., & Aksu, H. (2017). Detection of cyberbullying on social media messages in Turkish. 2017 International Conference on Computer Science and Engineering (UBMK), 366–370. https://doi.org/10.1109/UBMK.2017.8093411.

Pak, I., Teh, P.L., 2018. Value of expressions behind the letter capitalization in product reviews. ACM Int. Conf. Proc. Ser. 147–152. https://doi.org/10.1145/3185089.3185150.

Pandey, P. (2018). Simplifying Sentiment Analysis using VADER in Python (on Social Media Text). Medium Analytics Vidhya. https://medium.com/analytics-vidhya/simplifying-social-media-sentiment-analysis-using-vader-in-python-f9e6ec6fc52f.

Pitsilis, G. K., Ramampiaro, H., & Langseth, H. (2018). Detecting offensive language in tweets using deep learning. https://arxiv.org/pdf/1801.04433v1.pdf, 1–17.

Rafiq, R. I., Hosseinmardi, H., Han, R., Lv, Q., Mishra, S., & Mattson, S. A. (2015). Careful what you share in six seconds: Detecting cyberbullying instances in Vine. Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015, 617–622. https://doi.org/10.1145/2808797.2809381.

Reynolds, K., Kontostathis, A., & Edwards, L. (2011). Using machine learning to detect cyberbullying. Proceedings—10th International Conference on Machine Learning and Applications, ICMLA 2011, 2, 241–244. https://doi.org/10.1109/ICMLA.2011.152.

ReThink—Stops Cyberbullying—Google Play App. (n.d.). ReThink. Retrieved July 7, 2020, from https://play.google.com/store/apps/details?id=com.rethink.app.rethinkkeyboard&hl=en.

Ribeiro, M. H., Calais, P. H., Santos, Y. A., Almeida, V. A. F., & Meira, W. (2017). "Like Sheep Among Wolves": Characterizing Hateful Users on Twitter. http://arxiv.org/abs/1801.00317.

Russell. (2017). More NLP with Sklearn's CountVectorizer—Russell—Medium. https://medium.com/@rnbrown/more-nlp-with-sklearns-countvectorizer-add577a0b8c8.

Salguero, A., Espinilla, M., 2018. A flexible text analyzer based on ontologies: An application for detecting discriminatory language. Lang. Resour. Eval. 52 (1), 185–215. https://doi.org/10.1007/s10579-017-9387-6.

Sarna, G., Bhatia, M.P.S., 2017. Content based approach to find the credibility of user in social networks: an application of cyberbullying. Int. J. Mach. Learn. Cybern. 8 (2), 677–689. https://doi.org/10.1007/s13042-015-0463-1.

Shende, S.B., Deshpande, L., 2017. A computational framework for detecting offensive language with support vector machine in social communities. In: 2017 8th International Conference on Computing, Communication and Networking Technologies (ICCCNT), pp. 1–4. https://doi.org/10.1109/ICCCNT.2017.8204020.

Singh, V., Varshney, A., Akhtar, S. S., Vijay, D., & Shrivastava, M. (2019). Aggression Detection on Social Media Text Using Deep Neural Networks. January 2019, 43–50. https://doi.org/10.18653/v1/w18-5106.

Srinidhi Skanda, V., Anand Kumar, M., & Soman, K. P. (2017). Detecting stance in kannada social media code-mixed text using sentence embedding. 2017 International Conference on Advances in Computing, Communications and Informatics, ICACCI 2017, 964–969. https://doi.org/10.1109/ICACCI.2017.8125966.

Sood, S. O., Antin, J., & Churchill, E. F. (2012). Profanity use in online communities. Conference on Human Factors in Computing Systems—Proceedings, April, 1481–1490. https://doi.org/10.1145/2207676.2208610.

Spyder Website. (2018). Spyder. https://www.spyder-ide.org/.

Silva, Y.N, Deborah, L.H., Rich, C, 2018. BullyBlocker: Toward an Interdisciplinary Approach to Identify Cyberbullying. Social Network Analysis and Mining 8 (1), 1–15. https://doi.org/10.1007/s13278-018-0496-z.

Talukder, S., Carbunar, B., 2018. AbuSniff: Automatic detection and defenses against abusive facebook friends. 12th International AAAI Conference on Web and Social.

Teh, P.L., Cheng, C.B., Chee, W.M., 2018. Identifying and categorising profane words in hate speech. ACM Int. Conf. Proc. Ser. 65–69. https://doi.org/10.1145/3193077.3193078.

Teh, P.L., Cheng, C.B., 2020. Profanity and hate speech detection. Int. J. Inf. Manage. Sci. 31 (3), 227–246. https://doi.org/10.6186/IJIMS.202009_31(3).0002.

Teh, P.L., Rayson, P., Pak, I., Piao, S., 2015. Sentiment analysis tools should take account of the number of exclamation marks!!!. In: 17th International Conference on Information Integration and Web-Based Applications and Services. https://doi.org/10.1145/2837185.2837216.

Teh, P. L., Rayson, P., Pak, I., Piao, S., & Yeng, S. M. (2016). Reversing the polarity with emoticons. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 9612, 453–458. https://doi.org/10.1007/978-3-319-41754-7_48.

Tulkens, S., Hilte, L., Lodewyckx, E., Verhoeven, B., & Daelemans, W. (2016). A Dictionary-based approach to racism detection in Dutch social media. 2–8. https://arxiv.org/abs/1608.08738.

Van Hee, C., Jacobs, G., Emmery, C., DeSmet, B., Lefever, E., Verhoeven, B., De Pauw, G., Daelemans, W., Hoste, V., 2018. Automatic detection of cyberbullying in social media text. PLoS ONE 13 (10), 1–23. https://doi.org/10.1371/journal.pone.0203794.

Vishwamitra, N., Zhang, X., Tong, J., Hu, H., Luo, F., Kowalski, R., & Mazer, J. (2017). MCDefender: Toward effective cyberbullying defense in mobile online social networks. IWSPA 2017. Proceedings of the 3rd ACM International Workshop on Security and Privacy Analytics, Co-Located with CODASPY 2017, 37–42. https://doi.org/10.1145/3041008.3041013.

Watanabe, H., Bouazizi, M., Ohtsuki, T., 2018. Hate speech on Twitter: a pragmatic approach to collect hateful and offensive expressions and perform hate speech detection. IEEE Access 6, 13825–13835. https://doi.org/10.1109/ACCESS.2018.2806394.

Wong, S.C., Teh, P.L., 2020. How different genders use profanity on Twitter ?. In: Proceedings of the 2020 the 4th International Conference on Compute and Data Analysis, pp. 1–9. https://doi.org/10.1145/3388142.3388145.

Xiang, G., Fan, B., Wang, L., Hong, J., Rose, C., 2012. Detecting offensive tweets via topical feature discovery over a large scale twitter corpus. ACM Int. Conf. Proc. Ser. 1980–1984. https://doi.org/10.1145/2396761.2398556.

Young, R., Subramanian, R., Miles, S., Hinnant, A., Young, R., Subramanian, R., Miles, S., Hinnant, A., 2017. Social representation of cyberbullying and adolescent suicide: A mixed-method analysis of news stories social representation of cyberbullying and adolescent suicide: A mixed-method. Health Communication 32 (9), 1082–1092. https://doi.org/10.1080/10410236.2016.1214214.

Zhang, Z., & Luo, L. (2018). Hate speech detection: A solved problem? The challenging case of long tail on Twitter. 1(0), 1–5. https://arxiv.org/pdf/1803.03662.pdf.

Zhao, R., Zhou, A., & Mao, K. (2016). Automatic detection of cyberbullying on social networks based on bullying features. ACM International Conference Proceeding Series, 04-07-Janu, 1–6. https://doi.org/10.1145/2833312.2849567.