Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Elsevier hereby grants permission to make all its COVID-19-related research that is available on the COVID-19 resource centre - including this research content - immediately available in PubMed Central and other publicly funded repositories, such as the WHO COVID database with rights for unrestricted research re-use and analyses in any form or by any means with acknowledgement of the original source. These permissions are granted for free by Elsevier for as long as the COVID-19 resource centre remains active.

Engineering Applications of Artificial Intelligence 115 (2022) 105283

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

Identification of cyber harassment and intention of target users on social media platforms

S. Abarna [a,*], J.I. Sheeba [a], S. Jayasrilakshmi [a], S. Pradeep Devaneyan [b]

[a] *Department of Computer Science and Engineering, Puducherry Technological University, India*
[b] *Department of Mechanical Engineering, Sri Venkateshwaraa College of Engineering and Technology, Puducherry, India*

ARTICLE INFO

Keywords:
Cyberbullying
Fast text
Word2vec
Natural language processing
Insulting vocabulary
Intention detection
Buzzwords

ABSTRACT

Due to Coronavirus diseases in 2020, all the countries departed into lockdown to combat the spread of the pandemic situation. Schools and institutions remain closed and students' screen time surged. The classes for the students are moved to the digital platform which leads to an increase in social media usage. Many children had become sufferers of cyber harassment which includes threatening comments on young students, sexual torture through a digital platform, people insulting one another, and the use of fake accounts to harass others. The rising effort on automated cyber harassment detection utilizes many AI-related components Natural language processing techniques and machine learning approaches. Though machine learning models using different algorithms fail to converge with higher accuracy, it is much more important to use significant natural language processes and efficient classifiers to detect cyberbullying comments on social media. In this proposed work, the lexical meaning of the text is analysed by the conventional scheme and the word order of the text is performed by the Fast Text model to improve the computational efficacy of the model. The intention of the text is analysed by various feature extraction methods. The score for intention detection is calculated using the frequency of words with a bully-victim participation score. Finally, the proposed model's performance is measured by different evaluation metrics which illustrate that the accuracy of the model is higher than many other existing classification methods. The error rate is lesser for the detection model.

1. Introduction

With the rapid growth in the usage of social media platforms, Cyberharassment has become one of the major important issues in our society. Online Harassment causes many negative consequences that highly affect the victims due to the high frequency provided by Information and Communication Technologies (ICT) (López-Vizcaíno and Nóvoa, 2021). The percentage of people who experienced cyberharassment during their lifetime has increased to 36% in 2019 from 18% in 2007, due to the high use of mobile devices and social networks by children and teenagers. According to the research conducted by Bozyiğit et al. (2021a) around three billion people use social platforms for communication. It is indistinct that social media platforms afford various advantages and also some malicious activities such as cyberharassment. It is a cybercrime where a harasser can share negative posts, personal information, false content, or messages about the victim to humiliate or threaten the victim repeatedly overtime on the digital platform. The early detection and termination of cyberharassment in social networks help to reduce its adverse effects on the victims and to identify the harasser.

The study of the cyber-harassment field comes under Psychology, Information Technology (IT), Education, and Behavioural Science (BS) (Elsafoury et al., 2021). Over the past decade, the automated detection of cyberharassment specifically on the subject of detective work on cyberharassment from social media networks like Twitter, YouTube, and Instagram using predictable Machine-learning models, deep learning models, and rule-based models. Based on the national Pew research centre survey in 2020, the percentage of online harassment takes place on different social media platforms such as Instagram (63%), Twitter (24%), WhatsApp (34%), Facebook (46%), Telegram (18%), and Snapchat (39%). Among all, Instagram ranked high in the occurrence of cyber harassment where people can post images and videos followed by comments. The survey (García-Díaz and Cánovas-García, 2020) exposed that women folk were about double as possible as men to state that they had been targeted as a result of their gender. Young women, usually undergo sexualized forms of harassment. In Jain et al. (2020) explains that expressive and social difficulties occur not only among victims but correspondingly among harasser victims also.

A study by the Pew Research Centre presented that around 59% of U.S. teens have generally familiar with at least one of six types of

[*] Corresponding author.
E-mail addresses: abarna@pec.edu (S. Abarna), sheeba@pec.edu (J.I. Sheeba), jayasrilakshmis97@pec.edu (S. Jayasrilakshmi), pr.signs@gmail.com (S.P. Devaneyan).

https://doi.org/10.1016/j.engappai.2022.105283
Received 9 May 2022; Received in revised form 13 July 2022; Accepted 27 July 2022
Available online 10 August 2022
0952-1976/© 2022 Elsevier Ltd. All rights reserved.

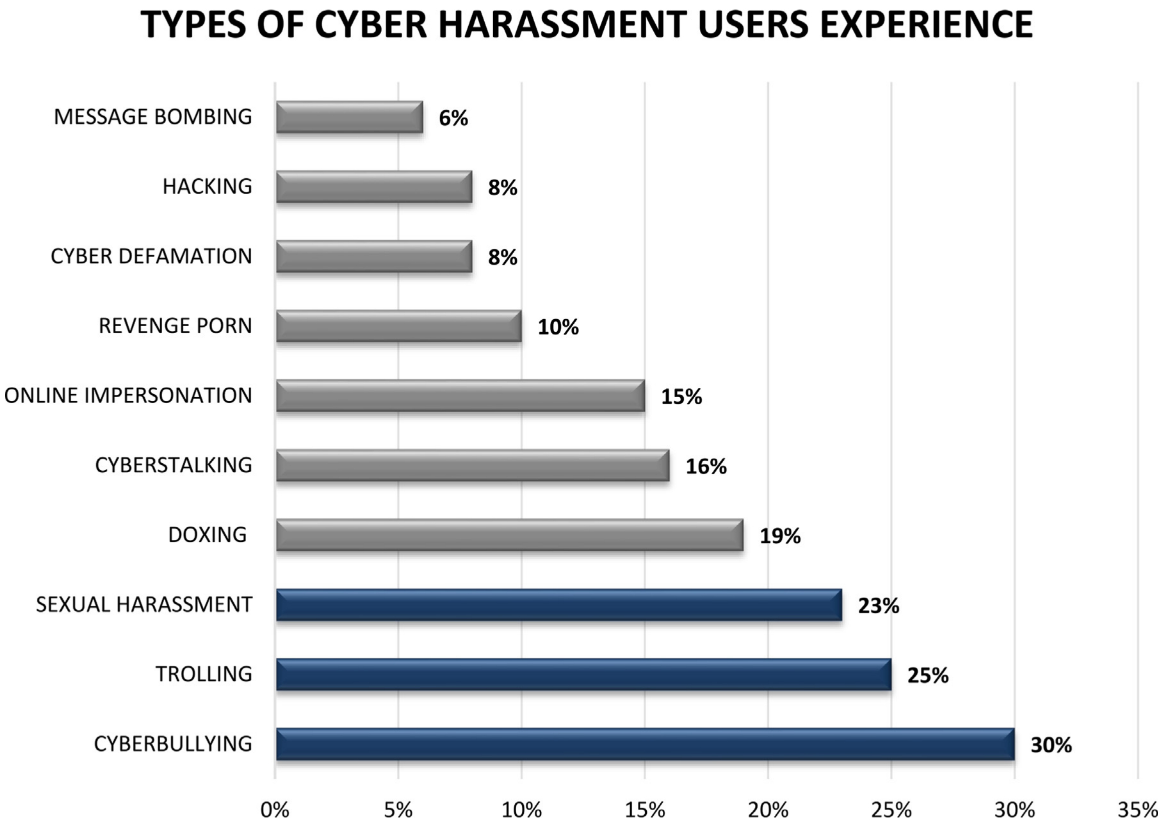


Fig. 1. Percentage of different Types of Cyberharassment.

online offensive behaviours (KunWang et al., 2020a,b). Such cyberharassment may have solemn significance, including anxiety, suicide, psychosomatic, depression, negative emotions symptoms. From Fig. 1, the number of teens experiencing cyberharassment had remained increased by (**32%**). More than half of US adults who use the internet have dealt with different types of Cyberharassment reporting cyberbullying (**30%**), trolling (**25%**), sexual harassment (**23%**), doxing (**19%**), cyberstalking (**16%**), online impersonation (**15%**), revenge porn (**10%**), cyber defamation (**8%**), hacking (**8%**), and message bombing (**6%**). It is clear that cyberbullying, trolling and sexual harassment remain in the top position over the post-covid-19 pandemic lockdown. In Ducange and Fazzolari (2018) the monitoring and the harassment classification are supported by the comments endlessly mined from a set of public pages and publicly available social networking platforms (e.g., Facebook, Instagram Twitter etc.).

The goal of cyberharassment research is to detect and classify offensive language in text content using machine learning. The features used to detect online harassment are divided into four categories: content, sentiment, user, and network-based features (Elsafoury et al., 2021).

1. Keywords, spelling, punctuations, profanity, pronouns, and document length are examples of content-based lexical features that can be extracted from a document.
2. Phrases, Keywords, and symbols (such as Emoticons) are examples of emotion-based features that can be used to determine the sentiments expressed in a document. To advance the performance of cyberbullying detection framework, sentiment-based analysis is frequently combined with variables such as pronoun usage and TFIDF.
3. Information about users' social media networks such as the count of followers adores and opinions received, or the number of individuals they follow is an example of user-based features. Age, race, gender, and sexual orientation are all factors to consider.
4. The utilization indicators taken from the online social network such as frequency of posting, number of friends, followers, uploads, likes, and so on are examples of network-based features.

There are many methods for detecting online harassment using machine learning techniques but with certain limitations: Use of a single classifier may have its limitations which results in low performance, Imbalance of dataset, Detection of variants of harassment words are not considered, Buzzwords are increasing. Hence the collection of new abusive words should be increased. From the survey, Fig. 2 describes that users' age is taken horizontally and the percentage of how much they are affected by different categories of cyber harassment is taken vertically. The graph shows that users between the age of 15 to 35 are affected more by trolling and cyberstalking where they are being threatened repeatedly over the internet (Thun et al., 2021). People above the age of 35 years are experiencing Doxing is when an individual's personal information is posted online to make others harass. We propose a conventional method of detecting cyberharassment in the Instagram text comments dataset in this research. Instagram is a social media platform in which users can share images followed by their corresponding comments either publicly or privately.

The main contribution of the proposed work is as follows:

- To build a lexical-based conventional model for analysing the morphology and word order of the harassment words in text comments by making use of Similarity measures and the Fast Text model.
- To find the targeted groups and intention behind the particular textual comment extracted by NLP using the feature extraction method.
- To diminish the loss function and time intricacy of the detection model using fast text supervised algorithm combined with word similarity measure.

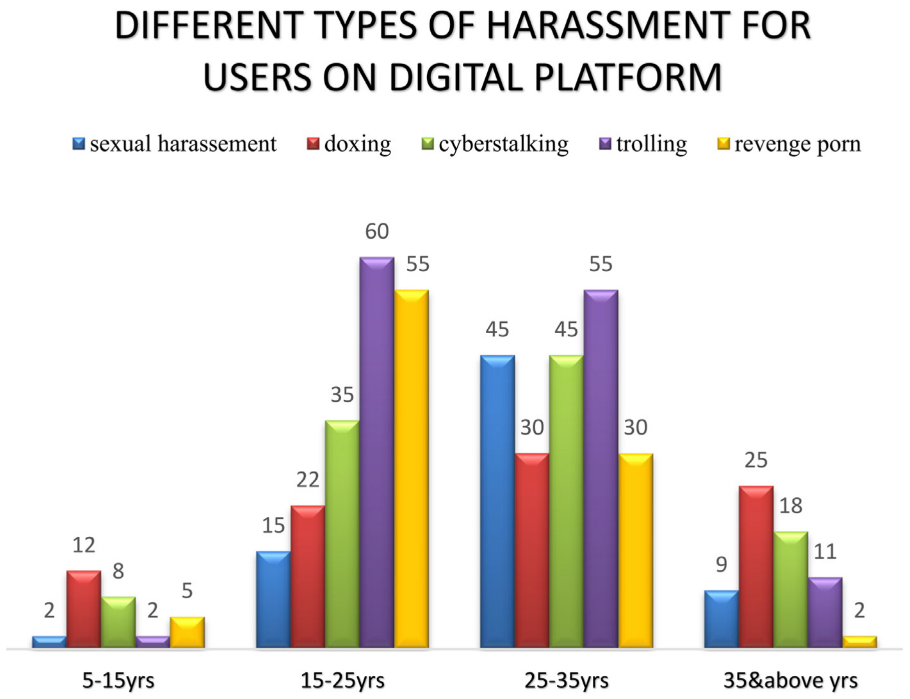


Fig. 2. Impact of cyberharassment on age diversity.

This work is divided into four sections: Section 2 discusses the background and corresponding works; Section 3 discusses the proposed framework model and Section 4 discusses the performance metrics for classifying output.

2. Background and related work

Many research studies have been conducted over the last decade on cyber harassment detection in order to control or reduce abuse in the social network community. With the rapid growth in the research field of cyber harassment, many machine learning techniques and natural language processes for identifying harassment on social network platforms have been proposed.

Text representation and word embeddings methods

García-Díaz et al. (2021) evaluate a model for detecting sexism in the Spanish language based on typical Word Embeddings and Linguistic Features (AWE+LF) using a new dataset made up of Spanish-language tweets. With a complete accuracy of 85.175 per cent with SMO, the model surpasses baseline models based on BoW, linguistic features (LF), and the Average of Word Embeddings (AWE). Furthermore, the model assesses three corpus subsets: oppressive messages directed at appropriate women folk (VARW); verbal distinctions between Spanish spoken in Europe and Latin America (SELA); and transcripts addressing overall misogynistic personalities like scorn, dominance, sexual harassment, and stereotyping (DDSS).

Li et al. (2021) many models have been proposed to learn effective word embeddings, such as word2vector, GloVe, ELMo, non-negative sparse embedding (NNSE), Bert and transformer-based methods have achieved great successes on many NLP tasks to learn word representation. Pasupa and Na Ayutthaya (2019) word embedding, which aids in modelling the meaning of each word, POS-tag, which aids in modelling the grammatical function of words, and sentic, which aids in modelling the emotion of words. According to the experiment's findings, The CNN model and each feature worked together to get the optimum outcome and produced the best F1 score of 0.817 at p 0.01, which was much higher than that of any other model. AlKhwiter and Al-Twairesh (2021) presented two supervised POS taggers that were created using Bidirectional Long Short-Term Memory (Bi- LSTM) models with conditional random fields. The accuracy of the Bi-LSTM-based POS tagger is 96.5 per cent. Tag-Guided Hyper- RecNN/TreeLSTM (TG-HRecNN/TreeLSTM), which incorporates a hyper network into RecNNs to employ Part-of-Speech (POS) tags on words, is suggested by Shen et al. (2020) and phrases as inputs with dynamism and produce the parameters for the semantic composition. In skip-gram based feature extraction method is projected to characterize high-dimension word vectors based on word2vec. For each word term, it is separated into n-gram characters to characterize the word direction and also resolves the challenges out of language words Hou and Maa (2020).

Machine learning and Deep learning methodologies in cyberharassment detection

Tolba et al. (2021) assess the possibilities of adopting such hybrid methods to efficiently switch the task of detecting cyber harassment by means of exceedingly asymmetric Twitter facts and choose the most appropriate grouping for the envisioned use. Three evaluation metrics that are frequently employed for unbalanced classification have been discussed in relation to the results of a comprehensive comparison investigation. The optimum feature representation was discovered to be Glove, and the top-performing combinations included LSTM and BLSTM with cost-sensitive knowledge and VL approach.

In López-Vizcaíno et al. (2021a) explores the prediction of fake news before it has been propagated on social media. For this purpose, it used a theory-driven model that represents the news content at four language levels (lexicon, syntax, semantic and discourse-level) reaching 88% correctness and outperforming all baselines in existing work.

More emotions that may be related between positive and negative classes cannot really be captured by binary classifiers for the classification of hate speech. As a result, to address issues with hate speech classification on Twitter, a probabilistic clustering model was created by Ayo et al. (2021) with an F1-score of 91.5, the probabilistic bunching model for categorizing hatred language outperformed related models. In addition, the generated model suggests a better test with an AUC of 0.9645 when associated to analogous techniques. Sadiq et al. (2021) solve the issue of automatic aggression detection on the collection of tweets from online trolls and uses a Multilayer Perceptron to feed it significant manually created features. They tested with a cutting-edge CNN-LSTM and CNN-BiLSTM deep neural network combination, and together have shown promising outcomes. According to statistical findings, the suggested typical system achieved the finest and accurately distinguishes hostile conduct in 92 percent of cases. Rajesh and Sharanya (2020) Negative opinions on social media are found and prevented using natural language processing and profanity detection library functions to potential halt and lower the rate of cyberbullying occurring on social media. Sainju et al. (2021) gave background information on the people tweeting about bullying, their motivations for doing so, and the times of day they most frequently do so. It emphasizes the fact that consumers come to share both their offline and online experiences. In Aguado and Julian (2019) sentiment analysis is a line of research that tries to evaluate the detection of opinions, sentiments, appraisals, and attitudes in different kinds of media. Current deep learning methods attempt to achieve remarkable performance to address this issue in a new way Wang et al. (2019). For example, the convolutional neural network (CNN) and recurrent neural networks (RNN) are used for such text classification tasks, which adopt totally different ways of understanding natural languages. Most existing intention detection models rely on a large number of well-established annotated datasets (Xue and Ren, 2021).

In Martín and Fernández-Isabel (2020) knowledge Based Systems manage knowledge to solve general tasks and to support the decision-making, learning and action processes achieved by humans to provide a complete architecture to gather, store, inference, interpret and communicate knowledge.

In Tran and Luong (2019) is dedicated to the Vietnamese language which poses several challenges. The task is to deeply analyse user utterances to identify intention implied by user notes and analyse the content to extract useful contexts for AI bots. Natural language understanding (NLU) is serious to the performance of goal-oriented verbal discourse systems. In Chen et al. (2019) typically contain the intent classification and slot filling tasks, aiming to form a semantic parse for user utterances (see Table 1).

3. Proposed work

In our proposed model, the text data is collected and pre-processed. The features are extracted by various mechanisms suitable for the model. The extracted features are detected using similarity measures to find the intention of the harassment comments. The Fast Text model also classifies the text comments. Thus, the results from both the conventional and fast text models are compared by their probability of getting that label to get the desired results. Finally, the model is evaluated by various performance metrics.

3.1. Exploration of dataset

In this proposal work, we used one of the crucial social network platforms called Instagram where the harassment had increased.

Instagram: It is a social media website where users may share, likes, and comment on the photo. Instagram has been identified as one of the five social media networks with the topmost proportion of users reporting online harassment. (Chelmis and Yao) provides dataset with the following information for each media session: userid, number of likes, days passed from posting, likes score, type of post, number of tags, number of comments, date posted, date and time of posting the media. In total, this dataset contains 155 260 users' Instagram posts along with their likes score, number of days from posting etc. The average number of words in each comment is 12. There are 2218 sessions, 2754 harassment content and 8203 Nonharassment content. With an imbalanced dataset, there are 75 percent Nonharassment comments and 25 percent harassment remarks in the training data. Training data is used to identify whether or not a person is a bully and to generate test data. The test data is used to determine how well the classifier performs in classifying correctly.

3.2. Process model for detecting cyberharassment

Based on the survey, many machine learning models for detecting online harassment are performed using NLP (Natural Language processing). Since the role of NLP in text, data processing plays a major role in classification but the traditional machine learning pipeline gives a lower performance (Tseng et al., 2019). Hence the probability-based detection mechanism is proposed in this work. It uses a word embedding model for representing all the vocabulary in the n-dimensional space. The fast text model is used to find the word order of the text to get the lexical meaning in the sentence. It computes in lesser time due to the hierarchical tree structure that works behind it. The intention of the harassment comment is determined using POS (Parts of speech) tagging in the NLP step. It uses a mapping function for finding the target users associated with the intention behind it. Due to the imbalance of the text dataset, the model may not perform well due to which fast text model is used which reduces the impact of the imbalanced dataset over harassment comments. Fig. 3 describes the process model for detecting cyber harassment on the Instagram platform along with its intention finding.

3.2.1. Conventional method of detecting harassment (word similarity)

The conventional flow of detecting online harassment along with its intention is considered a key part of the prevention of harassment on social media platforms. Therefore, in the traditional method, the raw data from social media consists of different unwanted text content.

Fig. 3, shows the conventional model for detecting cyber-harassment and its intention on the Instagram social media platform. For building the prediction model, the dataset is divided into 80% of training and 20% of testing. In training dataset, it contains irrelevant characters which need some data cleaning process. The pre-processed comments are given to feature extraction. In feature extraction, the targeted users and their intention behind each comment are extracted using a mapping function. The vocabulary of words is determined by the Bag Of words technique. The vector representation of each word in the sentences is computed using the word2vec model which is trained using a 1.6 million twitter corpus. The similarity measure (Ristanti et al., 2019) computes the comparison among each word in the training model and the words in the Buzzwords list. The threshold value is fixed based on the analyses of the labelled dataset. During preprocessing some words are removed due to which the contextual meaning of the sentence is not considered. Hence the raw textual data is given to the Fast Text model along with the label. The probability of the label for each sample is computed by comparing the value from both conventional and Fast Text modules. For evaluating the machine learning model, testing data is cross validated. The following are the steps involved in developing the proposed model:

Step 1*:* The noise in the raw textual data may lead to misclassification when determining the intention from textual data. It mentions to rather that is not related to word-based social language, and it can take many forms, such as URLs, unusual letterings, punctuation, the usage of comments, square brackets, and blank spaces. Hence many preprocessing techniques are needed to normalize the text data in a suitable for feature extraction. The main purpose of preprocessing is to derive insights from noisy text data before transferring them to the machine learning model. The python library NLTK is used for data cleaning which consists of many pre-trained models for pre-processing such as wordnet, regular expression etc., The techniques used in this model are listed as follows:

- Converting to lowercase
- Remove punctuations, symbols, URLs, extensions
- Tokenisation
- Lemmatization

Table 1
Summary of the existing work for cyberharassment detection.

Author's	Methodology	Measures	Dataset	Features	PROS & CONS
Sanchez-Medina et al. (2020)	Occupational Safety and Health (OSH) standards	Preventing sexual cyberbullying	organizations	Textual, User-based	To examine how the association between a dark personality and sexual cyberbullying activities may be influenced by factors including gender, age, or socioeconomic class.
Yuvaraj et al. (2020)	Deep decision trees and multi-feature artificial intelligence for classification	Automatic detection of cyberbullying	Twitter	Textual, User-based	With its increased text classification accuracy, the innovative Deep classifier's accuracy in classification is validated.
Calvo-Morata et al. (2021)	Conectado, A Solemn Game Against Bullying	Focuses on evaluating the game's efficacy and analysing the data produced during the game.	Social media	Textual, Sentimental, Contextual	To surge alertness on bullying and cyberbullying to school students
Balakrishnan et al. (2019)	Big Five and Dark Triad features	Cyberbullying recognition	Twitter	Textual, User-based	Indication showing connection among handler characters and cyberbullying enactment to detect online bullying patterns.
Bozyigit et al. (2021b)	AdaBoost	Cyberbullying detection	Social media	Textual	To provide an effective estimate of performance on the variant containing social media features.
Lopez-Vizcaíno et al. (2021b)	Supervised learning method	Cyberbullying detection	Vine social networks	Textual	Proposed dual clusters of features named threshold and dual for two early detection methods.
Eronen et al. (2021)	Feature Density (FD) using different linguistically-backed feature pre-processing methods	Automatic cyberbullying detection	Social media Yelp business review	Textual, Sentimental, Contextual	To estimate dataset complexity
Chia et al. (2021)	Feature Engineering techniques and Machine Learning	To explore the irony and satire	Twitter	Textual, User-based	To evaluate the properties of irony and sarcasm recognition in cyberbullying detection tasks.
Ireland et al. (2021)	Supervised machine learning	Automated detection and prevention systems	Twitter	Textual, Sentimental, User-based	Bullies' relative popularity, collective and automated efficacy, and incident interpretation
Dennehy et al. (2020)	The Critical Appraisal Skills Program assessment tool	Used to assess the calibre of the studies that were included.	Social media	Textual, Sentimental, Contextual	Intent, repetition, accessibility, anonymity, and disclosure barriers were five major ideas that were found.
Chatzakou et al. (2019)	Robust methodology	To separate bullies and attackers from regular Twitter users	Twitter	Textual, User-based, Network-based	Classify the accounts with an accuracy of over 90% and an AUC.
Tahmasbi and Rastegari (2018)	Socio-contextual approach	To create and test a model for automatic detection	Twitter	Textual, Sentimental, Contextual	It gives information about several scenarios to detect cyberbullying.

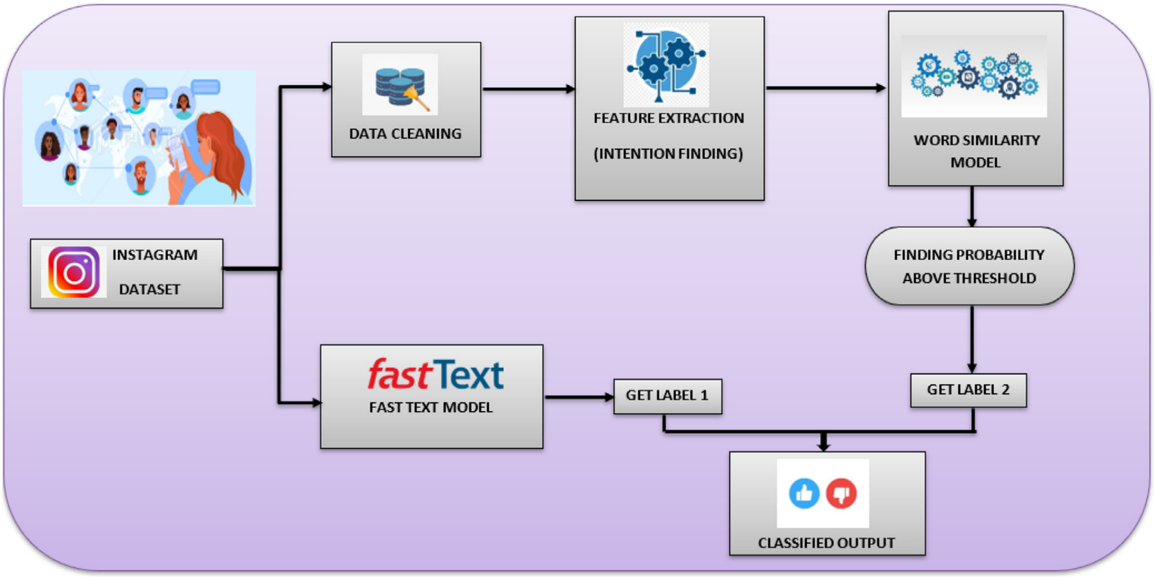


Fig. 3. The process for detecting online harassment along with its intention model.

Table 2
Sample example for the data cleaning process.

Sample text: @jackSimson_$Your mother looks so bitch UGLY ass nappy which make scenes nigga**#	
Techniques	Specified outcome
Lowercase conversion	@jacksimson_$your mother looks so bitch ugly ass nappy which make scenes nigga**#
Removal of symbols	jacksimson your mother looks so bitch ugly ass nappy which make scenes nigga
After tokenization	[‘jacksimson’, ‘your’, ‘mother’, ‘looks’, ‘so’, ‘bitch’, ‘ugly’, ‘ass’, ‘nappy’, ‘which’, ‘make’, ‘scenes’, ‘nigga’]
After lemmatization	[‘jacksimson’, ‘your’, ‘mother’, ‘**look**’, ‘so’, ‘bitch’, ‘ugly’, ‘ass’, ‘nappy’, ‘which’, ‘make’, ‘**scene**’, ‘nigga’]

Table 3
Sample working example for feature extraction.

Sample Text: “Everybody makes mistakes and he did so what like shit it”	
Techniques	Specified Outcome
Tagged tokens	[(Everybody/NN), (makes/VBP). (mistakes/NN), (and/CC), (he/PRP), (did/VBG), (so/CC), (what/WDT), (like/NN), (shit/FW), (it/PRP)]
Filter ‘NN’ & ‘PRP’	[(Everybody/NN), (mistakes/NN), (he/PRP), (like/NN), (it/PRP)]
Finding target user	[(he/PRP), (it/PRP)] → ‘Individual person’
Mapping function for finding intention	‘Individual person’ → ‘sexual attention’ ‘entertainment’ ‘popularity’

Table 4
Pseudocode for finding targeted users.

Input: ti ∀ i⟶ tokens,tags

Output: (C1,T1) where T⟶ targeted user

```
Start
Postag:
tu = []
dict = {tokens;tags}
su = [NN/singular]
for i in cleaned_text:
if(tag = NN&PRP)
for (ti in su)
targetuser.append(“individual”)
else
targetuser.append(“community”)
End
```

Table 5
Pseudocode for determining intention behind each text group.

Input: ti ∀ i⟶ tokens,tags

Output: (C1,I1) where T⟶ intention

```
Start:
ID1 = {i1: “sexual attention”, i2: “entertainment”}
ID2 = {i3: “popularity”, i4: “trolling”}
for x in targetuser:
if(x == “individual”):
TU.map(ID1.values())
Else:
TU.map(ID2.values())
Print(TU)
End
```

Table 2 gives sample example for the pre-processing steps. In order to treat two different words, like ‘ugly’ and ‘UGLY’, convert the words into lowercase. When analysing the text, the irrelevant characters such as [‘’, “,/, {}, @, #, < , > , *,), ˆ, (,] are removed from the text. To build the machine learning model, the words in the sentences are separated into tokens. Tokenization splits the text into chunks of words or sentences that help in analysing the sequence of words in the text. In order to represent the textual corpus, lemmatizing the words will convert the word into roots format like ‘scenes’ are converted to ‘scene’.

Step 2: From the cleaned textual corpus, POS tagging from the NLTK library helps to define their main context, functions, and usage in a sentence which helps to find the targeted user groups. The POS tags are the properties of the words such as Nouns, Verbs, Adjectives and Adverbs. The advanced type of this tagging is NER (Named Entity Recognition) which helps to extract key information to understand the text. It is a natural language processing (NLP) technology that can automatically scan full articles and classify essential things in the text into predetermined categories shown in Table 3.

The list of tagging along with its abbreviations are predefined in the POS tagging mechanism (Shen et al., 2020). The targeted groups can be determined by applying the following statements in Table 4.

After finding the targeted users, it is mapped to the intention behind each group. In order to get the motivation behind each harassment text, mapping is done by using regular expressions and mapping functions in Table 5. For each sentence

$$\{S_{j\epsilon D} = [I_{j\in S}]\} \tag{1}$$

In Eq. (1) where S(j) denotes each sentence, I(j) denote the intention of that sentence.

Step 3: In order to extract the targeted users from the textual content, nouns and pronouns are not removed by stop word removal. Word embedding techniques are used for representing all the words into vector space for building the classifier model. Hence, the Bag of words (BOW) method is used to analyse the collection of words in a training corpus. All the irrelevant characters are removed by the list of stop words from the NLTK library. There are 891 common words including ‘also’, ‘around’, ‘besides’, ‘further’, ‘though’ are collected together. After pre-processing, the textual features from Instagram data are extracted using a vocabulary of known words in BOW (Hou and Maa, 2020). Each sentence in the corpus is considered a document. Create a dictionary for mapping the words and their count for constructing vocabulary as a Counter. Each sentence in the corpus is added to the counter function to create a list of the unique bag of words. In Fig. 6, shows an example of the BOW example for three sentences in the training corpus. By running the cleaned comments on the counter function, there is a vocabulary called My Corpus () of 62,345 unique words based on the text comments in the dataset. In figure shows the sample of the top 20 words in positive comments.

Step 4: The vocabulary of unique words is constructed in a corpus. Fig. 4 shows some harassment words in the vocabulary and Fig. 5 shows some non-harassment words with higher frequency. In order to find the harassment words in the training sample, word embedding techniques represent the context of individual words present in a vocabulary. It also creates an association between two words through their word vectors. Word2Vec is used to create dense word embeddings where the corpus is given without any label information. The purpose of the embedding model is to identify more Buzzwords in the training sample. Thus, Word2vec is trained using an additional corpus of 1.6million

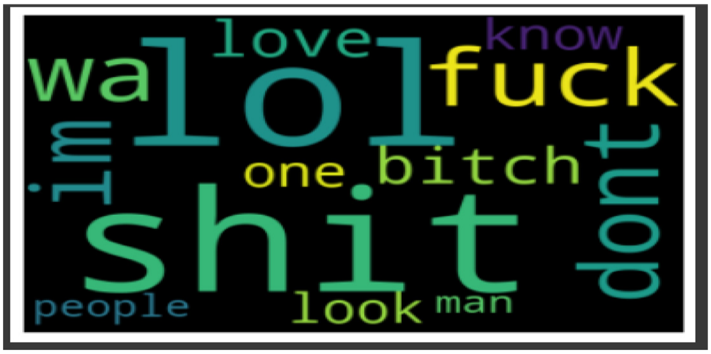


Fig. 4. Word cloud sample for harassment words with higher frequency in the vocabulary.

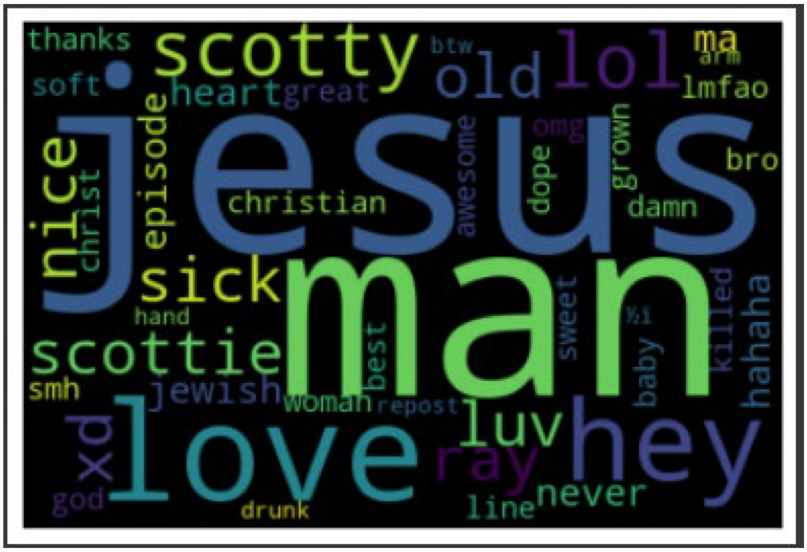


Fig. 5. Word cloud sample for non-harassment words with higher frequency in the vocabulary.

tweets which contains many harassment words with its dense vectors. It also computes the similarity between two words and finds the most similar words present in the vocabulary. The Genism library computes vector representations of the words in the training corpus.

There are two different architecture models CBOW (Continuous Bag of Words) and the Skip-gram model. The word embedding model works in the following scheme:

1. The cleaned corpus is passed as input to the BOW model.
2. Count vectorizer () will construct the vocabulary of unique words with lossy counting.
3. The sentence window size (n-gram) is chosen for separating each word and the contextualized meaning is determined by Dynamic scaling, pruning and subsampling.
4. The words in the vocabulary are passed to CBOW as input vectors {x1, x2, x3.… xn} where each word vector is mapped in the hidden layer {h1, h2, h3.…..hn}.
5. The vector values of each word are optimized by the Hierarchical SoftMax layer as the skip-gram model {y1, y2, y3, .… yn}.
6. If the loss occurs, the learning parameter changes the weighted values by negative sampling and backpropagate.
7. Finally, the vector representation of each sentence is given as the final product from the output layer {y1′, y2′, y3′, .… yn′}.

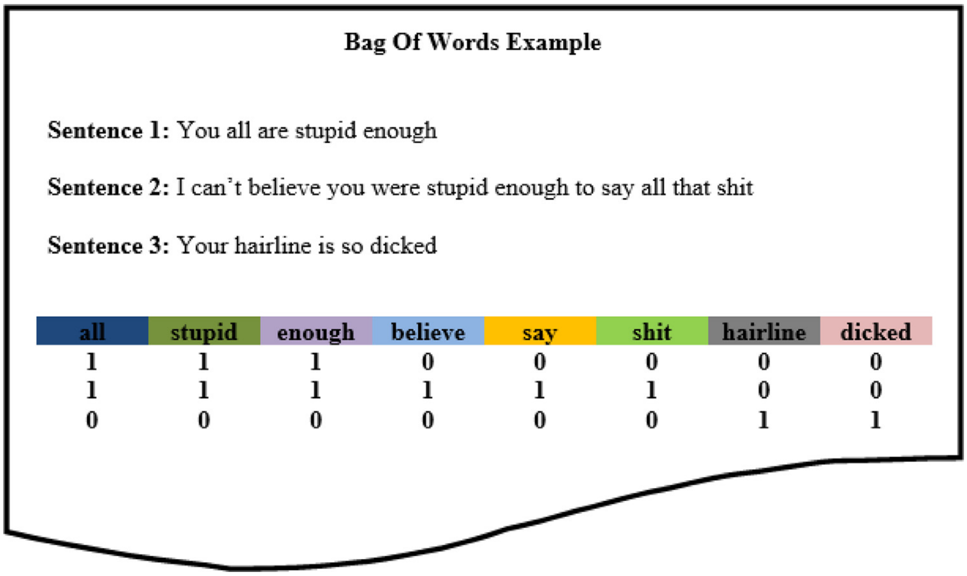
The CBOW architecture will take in the context words as input (i) and predict our target word (t). The context words are given as input to an embedding layer where some random weights are initialized. In a lambda layer, the average embeddings are given out as vectors. Finally, the weighted average embeddings are passed to the SoftMax layer to predict the target word. But it is not considered, only the first hidden layer in the model is taken.

To determine contextually similar words and to generate sequence embeddings for similar meaning words, Fig. 7 shows how the skip-gram model predicts multiple words from a single word, where (i, t) is the input to the embedding layer signifies (i) as input and (t) as the label. To get dense word embeddings, both (t) and (i) pairs are passed to individual separate embedding layers. The dot product value of these two embeddings is computed in the 'merge layer'. From the sigmoid layer, it outputs either 0 or 1 based on the dot product value. The predicted label is compared with the actual label, if the loss occurs it will backpropagate and change the epoch values.

Step 5: In this step, find the word comparison among individual word in the vocabulary and the buzz words list. Every sentence in the training corpus is compared with every word in the buzzwords. The higher value of the similarity measure will be considered as the sentence with harassment in the sample. Let the words in a sentence be {w1, w2, w3,…wm}, each sentence in the sample be {s1, s2, s3.…sn} and words in the buzzwords list are {b1, b2, b3,.…bn}.

For training sample $D_j \in D, j = 1, 2, \ldots..M$ where 'M' is the entire count of sentences in the corpus. The word vector for individual word 'w' is denoted as $\gamma(w)$. The possibility for declaring each sentence 's1' as harassment is denoted by calculating cosine similarity such as:

$$\text{SEW} = (max_{b(i)\epsilon B, n=1,2\ldots n}(sim(\gamma\left(w_m\right), \gamma\left(b_n\right)))$$



Bag Of Words Example

Sentence 1: You all are stupid enough

Sentence 2: I can't believe you were stupid enough to say all that shit

Sentence 3: Your hairline is so dicked

all	stupid	enough	believe	say	shit	hairline	dicked
1	1	1	0	0	0	0	0
1	1	1	1	1	1	0	0
0	0	0	0	0	0	1	1

Fig. 6. BOW example for sample sentences.

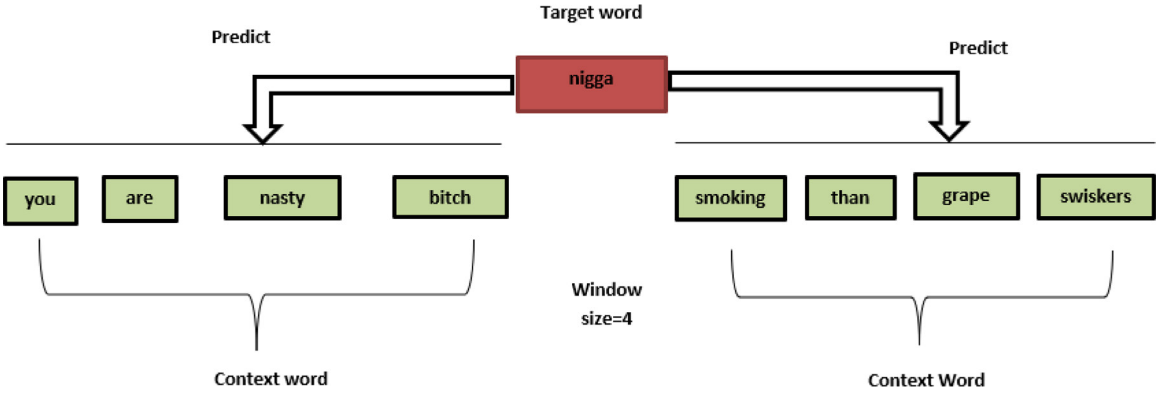


Fig. 7. Skip-gram example for finding target vector.

$$\mathrm{P[s(n)]} = \{max_{D(j)\epsilon D, j=1,2,..M}(\mathrm{SEW})\}\tag{2}$$

In Eq. (2) Where SEW denotes the similarity value for each word in a sentence with the buzzwords list. P[s(n)] denotes the possibility of labelling each sample into harassment type.

$$\begin{aligned}\mathrm{Sim(X, Y)} &= \{\frac{Sw.Bw}{\|Sw\| * \|Bw\|}\}\\ &= \frac{\sum_{k=1}^{n}(Sw(k)) * Bw(k)}{\sqrt{\sum_{t=1}^{n}(Sw(t))2 * \sqrt{\sum_{t=1}^{n}(Bw(t))2}}}\end{aligned}\tag{3}$$

In Eq. (3) where 'Sw' denotes a word in the sample and 'Bw' denotes a word in the buzzwords list. After calculating the possibility for harassment, the D_j is marked as a normal sentence by (1-P[s(n)]. Based on the harassment content in the training sample, the threshold (T) for separating positive and negative samples is determined by the sampling technique. If the value of P[s(n)] > (T) then that sentence is labelled as a Positive sentence that is a harassment sample whereas if P[s(n)] < (T) then it is labelled as a Negative sentence. Summarizing the overall process of this module is as follows:

Table 6 shows the process of the conventional intention detection model which takes the training dataset, buzzwords list etc as inputs. It returns the predicted label along with its probability as output.

Table 6
Input and output for the conventional intention detection model.

Input	Training dataset S = {s1, s2, s3,....s(m)}, buzzwords list B = {b1, b2, b3, ...b(n)}, 1.6 million tweets for training word2vec
Output	Results R = {[S1 label, P[s(n)], I(S1)]},.......[Sn label, P[s(n)] I(Sn)]} where S1 label and P[s(n)] indicate the labelling and possibility of being Positive (harassment sentence) for each sentence. I(S1) denote the intention behind every comment in the sample of being harassment.

3.2.2. *Pretrained language model scheme (fast text method)*

In the conventional method, some words are being missed due to the stop word removal process. Therefore, its losses some contextual meaning in the sentence. This leads to the classification of sentences into the wrong label. For example, only with one word like 'bit*h' no correct decision can be made because it may refer to 'bit', 'bitter', 'bi**h'. Taking the semantic meaning of the adjacent words in the particular sentence gives the correct classified result. To rectify this issue, the word embedding language model is developed by Facebook research called the Fast Text model (FT) (KunWang et al., 2020a,b). It is an open-source library for converting words into vector representation and text classification with probability. Fast Text uses the hierarchical SoftMax to increase the computation time when dealing with a larger dataset. To improve model efficiency without sacrificing accuracy, FT employs a bag of n-grams. It employs hashing techniques to maintain a fast and memory-efficient mapping of the n-grams.

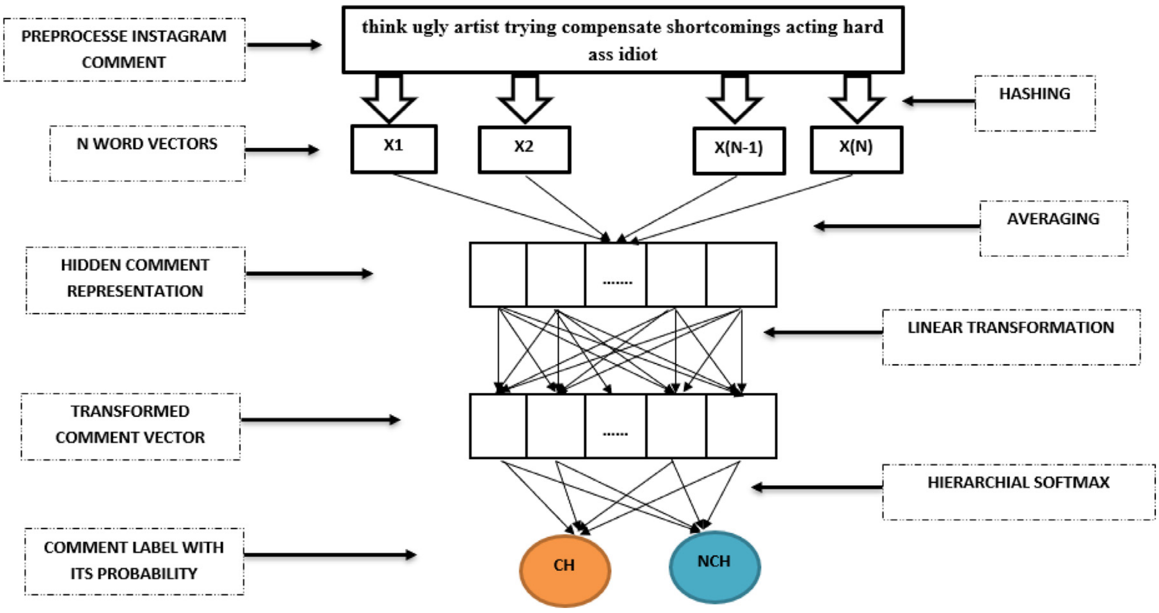

PREPROCESSE INSTAGRAM COMMENT
think ugly artist trying compensate shortcomings acting hard ass idiot
HASHING
N WORD VECTORS
X1
X2
X(N-1)
X(N)
AVERAGING
HIDDEN COMMENT REPRESENTATION
....
LINEAR TRANSFORMATION
TRANSFORMED COMMENT VECTOR
......
HIERARCHIAL SOFTMAX
COMMENT LABEL WITH ITS PROBABILITY
CH
NCH

Fig. 8. Sample working procedure for Fast Text classification model.

The working procedure of the FT language model for classification shown in Fig. 8 is as follows:

(1) Initially, the Instagram text comments prefix with its label are converted into word vector representations which are averaged into text embedding and fed to a linear classifier.
(2) The features are extracted from text embedding into the hidden layer.
(3) The probability distribution over pre-defined classes is determined by the hierarchical SoftMax layer.
(4) Huffman Coding Tree is used to reduce the computational complexity to O[mlog(n)], where n is the count of categories and m is the measurement of textual illustration.

Fast Text supports both word embedding techniques CBOW and Skip-gram models. It can be able to find the semantic similarities between two words. When it comes to training word vector models, FT is extremely fast. It can train approximately one billion words in less than ten minutes. Deep neural networks are used to build models. It captures the denotation of postfixes/prefixes in the corpus for a specific word. It can also generate word embeddings for rare words in the training sample. These approaches use a direct classifier to train the typical model. In the Linear classification module, the textual comments and their equivalent tags are characterized as similar vectors. In other words, the vector associated with the text comment is nearer to its label. FT uses the hierarchical SoftMax function to discover the probability score of a precise tag represented in a binary tree. A probability is represented by each node in the binary tree. The probability of finding a label for each sentence can be computed by the following equation:

$$P\{S_{i\in D}\} = [(\frac{-1}{S}\sum_{s=1}^{m} x_{s\,\log(\delta M1*M2*y_s)})] \quad (4)$$

In Eq. (4) Where ‘D’ represents the Instagram dataset, ‘S’ represents all the rows of sentences, ‘x_n’ is the label for n^{th} sentences, ‘δ’ is Hierarchical softmax function, M1 and M2 are weight matrices and ‘ y_s’ is the normalized bag of features of s^{th} sentences. A label is characterized by the probability laterally the track to that label. This shows that the leaf nodes of the binary tree signify the labels by speeding up the searching time.

The Fig. 9., shows that Fast Text uses Huffman trees can able to manage imbalanced classes. The frequently appearing words are present in the lower depth of the tree. Node 1 denotes the most frequently occurring word in the corpus. For example, the word ‘bitch’ occurs with a count of 46 ranks top. The probability of finding label for the word can be calculated as follows:

$$\text{P[‘bitch’, Positive]} = \sigma\ (\ \vartheta\ (\text{‘bitch’}) * \text{height}) \quad (5)$$

In Eq. (5) Where ‘ϑ’ represent the vector representation of the particular word, ‘Positive’ indicates the prefix label, ‘σ’ is the optimizing function and ‘height’ is the depth of the node in the tree. Hence this technique greatly diminishes the time intricacy of the training sample. Some words in the vocabulary are projected into the space vector representation. It shows that due to the imbalance of the dataset with lesser harassment words, the embeddings are scattered around the space. The x-axis represents the frequency of each word in the training quantity. Each word like ‘bitch’ is surrounded by ‘ugly’, ‘shit’, ‘unfair’, ‘nappy’ etc. The vector values of these words are between (−3.299 to 4.573).

3.2.3. Finding intention detection score (IDS)

From the Instagram dataset, it consists of userid, sender, receiver and comment columns with their severity measure. After the data cleaning process (stop word removal), the words are assigned in the bag of words vocabulary. Each word is set to sequential tokens. Construct a list of 349 buzzwords. Assign each comment to its sender and receiver in SDict{} and RDict{} are given below. The comments in both the dictionary are same feature. The mapping function set words for each category of intentions such as sexual attention, low self-esteemed, entertainment, popularity and threat individual etc. The threshold value (k) of 0.5 is fixed for determining the severity of harassment type and can able to prevent the bully from further conversation. In Table 7, shows the pseudocode for calculating word frequency in the BOW process. The probability of each harassment word (w_i) is at least 0.1 by finding word similarity between each word and each buzzword in the list. Each user has the possibility of being bully and also victim. The bully score is attributed as how many comments the particular userid send is labelled as harassment type. Likewise, the victim score is how many harassments comment are received by an userid.

$$\text{SDict} = \{\text{S1} : \text{C1}, \text{S2} : \text{C2}, \ldots\ldots\text{Sn} : \text{Cn}\}$$

$$\text{RDict} = \{\text{C1} : \text{R1}, \text{C2} : \text{R2}, \ldots\ldots\text{Cn} : \text{Rn}\}$$

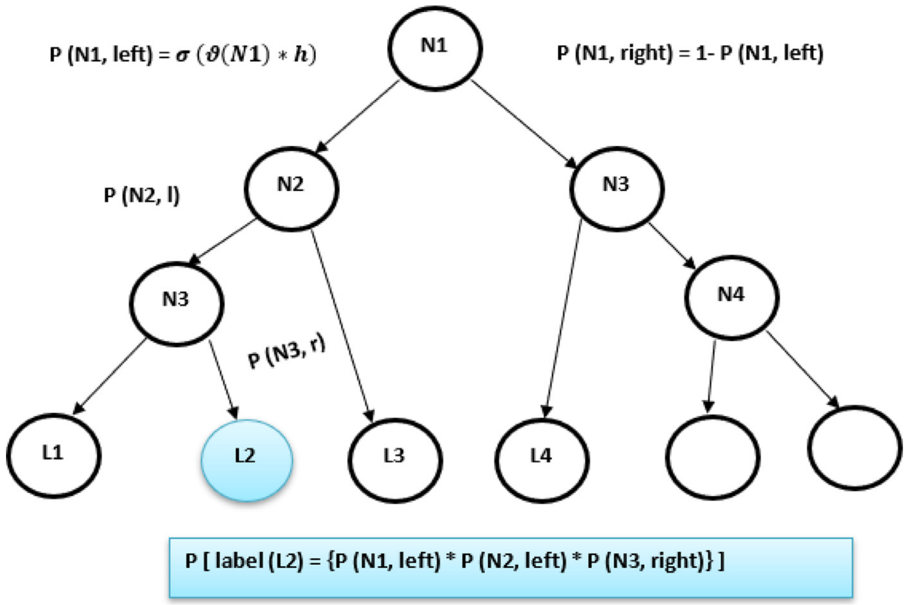

P (N1, left) = σ (ϑ(N1) * h)
N1
P (N1, right) = 1- P (N1, left)
P (N2, l)
N2
N3
N3
N4
P (N3, r)
L1
L2
L3
L4
P [label (L2) = {P (N1, left) * P (N2, left) * P (N3, right)}]

Fig. 9. Text classification using Huffman tree with probability.

Table 7
Pseudocode for calculating word frequency in the BOW.

```
Input: Ti ∀ i©words in Ci
Start:
for j in C:
If j in bw[]:
w(i) = 0.1
if j in l1:
w(i) + = 1
elseif j in l2:
w(i) + = 1
else
return(w(i))
End
```

where ‘S1’ and ‘R1 are the sender and receiver of the particular comment ‘C1’. The generalized method in (6) to find the intention score of each comment by a user is as follows:

$$\begin{aligned} \text{IDS} &= \max\{\frac{\rho}{2}(\|S\|^2 + \|C\|^2 - \|R\|^2) + (\frac{1}{2}[\sum_{W_{i\in f(C)}} (S_{score}(U_i) \\ &\quad + W_i - R_{score}(U_i))^2])\} \\ &= \max\{\text{range}(0,1)\} \end{aligned} \tag{6}$$

where (ρ) denotes the optimizer, ‘S’, ‘R’, ‘C’ represents a particular user (U_i) being a sender, receiver and the comment labelled as harassment. (W_i) denotes every word in a comment C. The intentions score includes how many times a particular user sends a text message, score of each word in the vocabulary is subtracted by the victim score the same user. Taking the maximum of the value gives the intention score since each comment may have various types of intention. If the IDS value is above ‘k’ (IDS > 0.5), then the user is marked as severe type and the particular userid is blocked further.

3.2.4. Computing label by comparing the probability values

In this module, both the trained model is integrated. The probability values for the corresponding label for each sentence is compared to get the final label either positive or negative. For example, consider 25th sentence is processed by Conventional scheme and fast Text model. The result from both the model consists for < label, probability > respectively. Finally, Table 8 shows the classification result for sample ‘s’ is obtained as follows:

Table 8
Pseudocode for combining WS and FT module.

```
Input : <l1,p1>;<l2,p2>
Start:
l1 ⟵ (WS)
l2 ⟵(FT)
If l1 = l2:
{
l1
}
Elseif l1 ≠l2:
{
P(l1)>p(l2)
l1
Else:
{
l2
}
End
```

3.2.5. Sample workflow of the model using an example

In this section. Let us consider the following text as an example for showing the working flow of the proposed model as given in Table 9.

Raw input data:

I1: “@skeeze_dez listenup Bitch! I will just F**k @on you worthless shit”

I2: “Like I said #killyourself you dumb like you took like a damn shit and take a shower cause you are attitude stink you stanky ugly Cunt__@#”

I3: “When you broke ass learn to spell and get to coked up crack headed get to asshole out of a dawn apartment and for starters at least I wear name brand you stupid hoe”

4. Analysis of result

In this section, the performance of the proposed model is estimated using different machine learning metrics such as Correctness (precision), recall, accuracy and F1-score. The effectiveness of the model was observed in this work by employing several assessment procedures to estimate how effectively the model can discriminate harassment from non-harassment. To understand the performance of competing models,

Table 9
Sample workflow of the model using an example.

Steps	Input data	Desired Outcome
After preprocessing	I1	‘skeezedez’, ‘listen’, ‘up’, ‘bitch’, ‘I’, ‘will’, ‘just’, ‘fuck’, ‘on’, ‘you’, ‘worthless’, ‘shit’
	I2	‘like’, ‘i’, ‘said’, ‘kill’, ‘yourself’, ‘you’, ‘dumb’, ‘like’, ‘you’, ‘took’, ‘like’, ‘a’, ‘damn’, ‘shit’, ‘and’, ‘take’, ‘a’, ‘shower’, ‘cause’, ‘youre’, ‘attitude’, ‘stink’, ‘you’, ‘stanky’, ‘ugly’, ‘cunt’
	I3	‘when’, ‘you’, ‘broke’, ‘ass’, ‘learn’, ‘to’, ‘spell’, ‘and’, ‘get’, ‘to’, ‘coked’, ‘up’, ‘crack’, ‘headed’, ‘get’, ‘to’, ‘asshole’, ‘out’, ‘of’, ‘a’, ‘dawn’, ‘apartment’, ‘and’, ‘for’, ‘starters’, ‘at’, ‘least’, ‘i’, ‘wear’, ‘name’, ‘brand’, ‘you’, ‘stupid’, ‘hoe’
After feature extraction (POS tagged) (extract ‘PRP’ & ‘NN’)	I1	*S1 = {‘PRP’-> ‘Individual’; Intention (I1)->Sexual attention (bitch, asshole, fuck)}* [‘skeezedez/NN’] [‘i/PRP’]
	I2	[‘attitude/NN’] [‘yourself/PRP’]
	I3	[‘apartment/NN’, ‘name/NN’, ‘brand/NN’] [‘you/PRP’]
Building vocabulary (BOW method)	I1	‘skeezedez’, ‘listen’, ‘bitch’, ‘fuck’, ‘worthless’, ‘shit’
	I2	‘kill’, ‘shower’, ‘attitude’, ‘stink’, ‘stanky’, ‘ugly’
	I3	‘broke’, ‘learn’, ‘coked’, ‘crack’, ‘headed’, ‘asshole’, ‘apartment’, ‘starters’, ‘brand’, ‘stupid’
Vector representation (word2vec model)	I1	[0.7357,0.1835,0.9826,0.9402,0.7411,0,1]
	I2	[−0.1985,0,0.1925,0.6217,0.4298,0.8605]
	I3	[0,0.2314,−0.1934,0.7384,0.1392,0.9325,0.5687,0.4218,0.2165,1]
Cosine similarity measure	I1	0.5021, −0.1972, 1, 0.9811, 0, 0.8090 ‘positive’, 1, ‘sexual attention’
	I2	0.8320, 0.3217, 0.5532, −0.2188, 0.6321, 1 ‘positive’, 1, ‘trolling’
	I3	0.3772, 0.4287, 0.3219, 0.5079, −0.1338, −0.9782, −1.0558, −0.4156, 0.5008,1 ‘positive’, 1, ‘sexual attention’
Training Fast Text model	I1	‘positive’, (0.8357)
	I2	‘positive’, (0.9218)
	I3	‘positive’, (0.9745)
Classified output	I1	‘positive’, 1, ‘sexual attention’, ‘IDS’
	I2	‘positive’, 1, ‘trolling’, ‘IDS’
	I3	‘positive’, 1, ‘sexual attention’, ‘IDS’

it is critical to review standard assessment metrics used in the research community (Sainju et al., 2021).

Precision estimates the proportion of appropriate comments among accurate positive (*tp*) and wrong positive (*fp*) comments fitting to an exact group.

$$\text{Precision} = \frac{tp}{(tp + fp)} \tag{7}$$

In Eqs. (7) & (8) where *tp* signifies correct positive, *tn* is a correct negative, *fp* denotes incorrect positive, and *fn* is an incorrect negative.

Specificity and Sensitivity are popular metrics for calculating True positive rate and True negative rate. Sensitivity (8) can able to correctly identify the harassment words in the given list. Specificity (9) can identify the words that are not included in the harassment list. Fig. 10, shows the sensitivity score for number of training samples for different experimenting algorithms such as J48, NB, MLP, LR. It shows that for intention detection model, the sensitivity remains higher and consistent.

$$\text{Sensitivity} = \frac{tp}{(tp + fn)} \tag{8}$$

$$\text{Specificity} = \frac{tn}{(tn + fp)} \tag{9}$$

In Eq. (10), ***Recall*** computes the relation of retrieved relevant comments over the total number of relevant comments.

$$\text{Recall} = \frac{tp}{(tp + fn)} \tag{10}$$

In Eq. (11), ***F-Measure*** provides a way to syndicate precision and recall into a solitary portion that captures both properties.

$$\text{F1} = \frac{2 * \text{P} * \text{R}}{\text{P} + \text{R}} \tag{11}$$

where ‘P’ denotes precision and ‘R’ signifies recall.

Mean Squared Error (MSE) is the difference between classification of harassment words by the model and the actual observed harassment words from the input. It is represented in (12).

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^{N} \left(HW_{observed,i} - HW_{model,i} \right)^2 \tag{12}$$

where $(HW_{observed})$ is manually observed classified harassment words, and (HW_{model}) is model classified harassment words for ‘*i*’ input comment. *(N)* is the number of input textual comment.

In Eq. (13), ***Mathews Correlation Coefficient (MCC)*** used in machine learning as a degree of eminence of two-fold and multi class classification. It finds the correlation between true values and predicted values.

$$\text{MCC} = \frac{tp * tn - fp * fn}{\sqrt{(tp + fp)(tp + fn)(tn + fp)(tn + fn)}} \tag{13}$$

From Fig. 11. It shows the Mean squared error rate for different samples of training data. It specifies that training the model with additional data will leads to lesser error rate. The MSE decreases after 5 K from 0.47 to 0.24.

From Fig. 12. It shows the Mathews Correlation coefficient (MCC) used in Eq. (13), macro-average precision and F1 score for intention detection model, J48, Naïve Bayes, Logistic regression, Multi layer perceptron, Support vector machine, BiLSTM and Random Forest. Among all the algorithms, our proposed model shows higher percentage of F1 score value and for MCC, values close to 1.

We have experimented with existing model such as Logistic Regression (LR), J48-Decision Tree (DT), Multi-Layer Perceptron (MLP), Random Forest (RF), Bidirectional Long Short-Term Memory (Bi-LSTM) and Naïve Bayes (NB). From Table 10, it shows that the detection of cyber harassment model with higher F1-score for the proposed work with 63%. It is performed on the Instagram training sample of 80%.

The Fig. 13 shows the balance of harassment content in the dataset. The positive comment is labelled as ‘CH’ (Cyber-harassment) and the

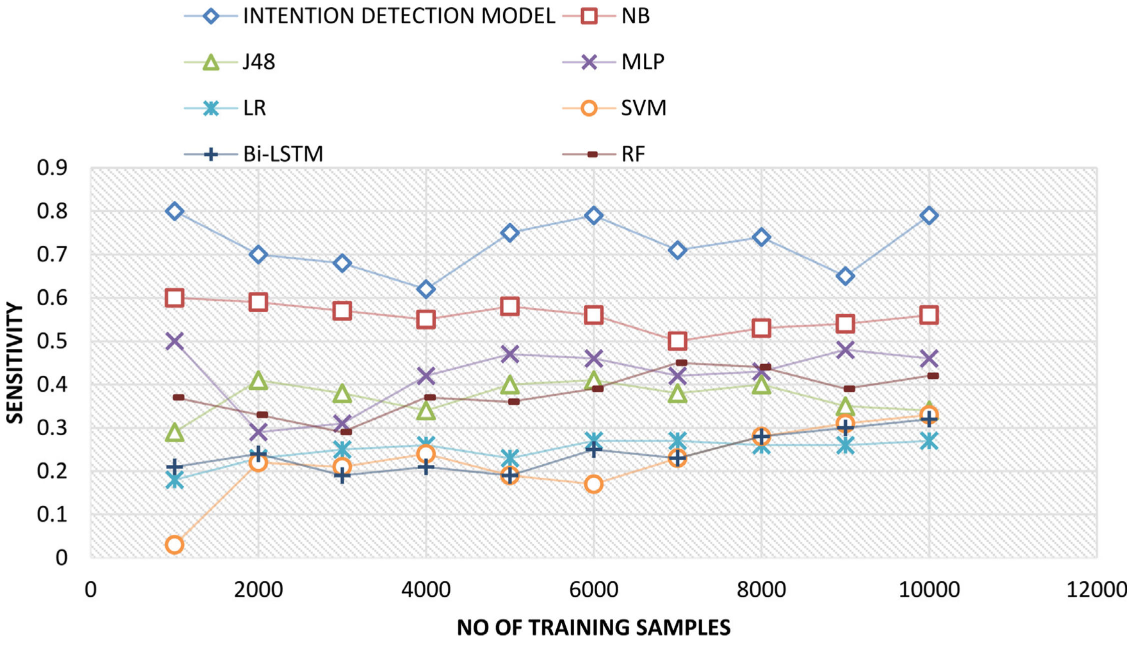


Fig. 10. Performance evaluation based on sensitivity values.

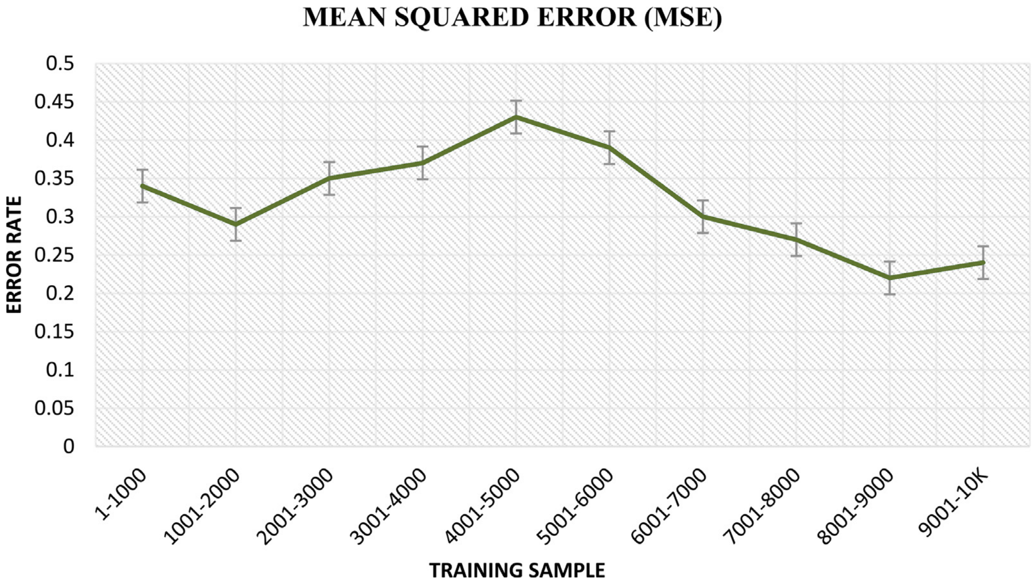


Fig. 11. Mean Squared Error rate for training sample.

Table 10
Training sample of harassment and non-harassment classes.

Algorithm used	Recall	Precision	F1 Score
J48	0.8280	0.9049	0.2944
Naïve Bayes	0.8352	0.8644	0.2021
Logistic regression	0.8188	0.8742	0.6168
MLP	0.8451	0.8933	0.5286
RF	0.8240	0.8355	0.4921
SVM	0.8625	0.7499	0.5044
Bi-LSTM	0.7103	0.6228	0.4369
Intention detection model	**0.8942**	**0.9145**	**0.6327**

negative comments are labelled as ‘NCH’ (Non-cyber harassment). It is nearly around 2k harassment sentences are present in the Instagram. Due to the imbalance of the dataset, the machine learning model is failed to give higher accuracy. But in our proposed work, the language word embedding model called fast Text are capable of managing an imbalance dataset since it deals with rare or different words in the sample vocabulary.

ROC Curve (Receiver the Operating Characteristic Curve) is a graph that shows the correct positive rate for the numerous potential data against the false-positive rate. ROC reveals the trade-off between sensitivity and specificity (lesser specificity and higher sensitivity).

From Fig. 14. it shows ROC curve for bag of words in the vocabulary for J48, naïve bayes, logistic regression, multi-layer perceptron, SVM, RF, bi-LSTM and the intention detection model. Among all these, IDM shows higher performance when compared to other classifiers.

From Table 11, it illustrates the time intricacy of the finest and worst techniques used for experimenting in relations to training and prediction time. It indicates that our proposed model had achieved the best prediction time in lesser seconds. Because it uses a pre-trained language model Fast text. The MLP takes the worst prediction time with

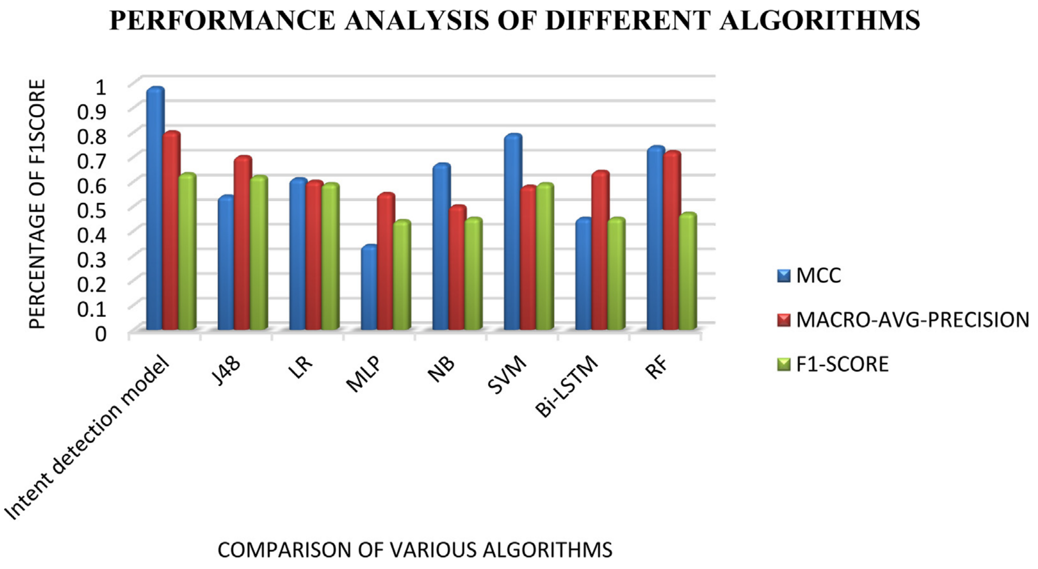


Fig. 12. Performance analysis of various algorithms.

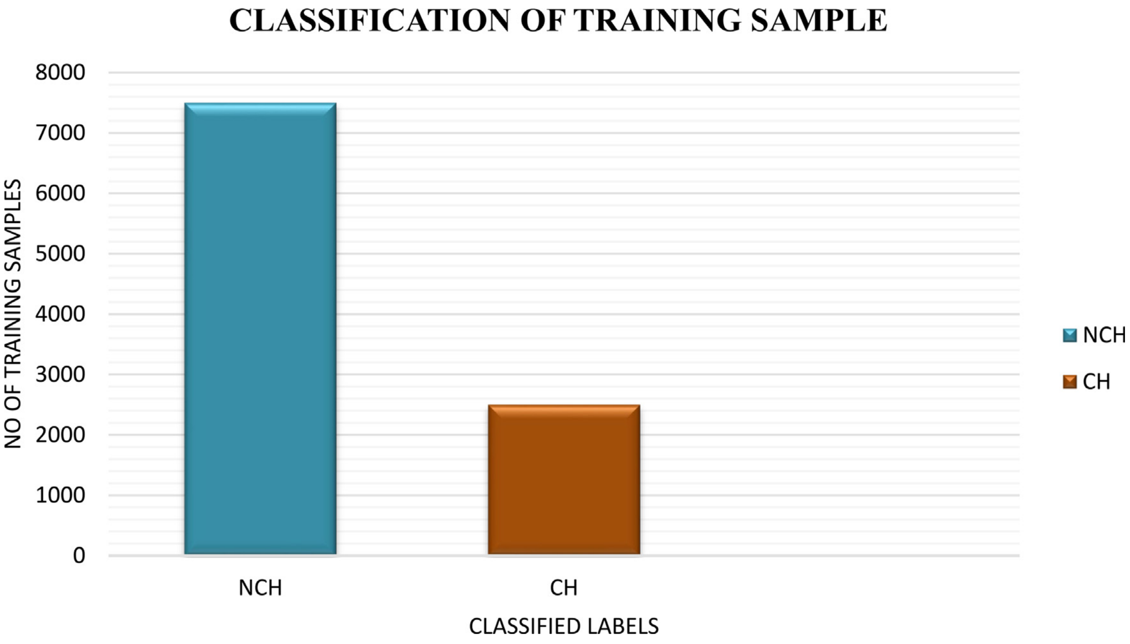


Fig. 13. No of training samples classifies into two labels.

Table 11
Training/Prediction time for two classes of the dataset.

S.no	Algorithms used	Training/prediction time
1	J48	2.57 s
2	Naïve Bayes	2.63 s
3	Logistic regression	3.01 s
4	MLP	5.94 s
5	SVM	2.83 s
6	RF	2.09 s
7	Bi-LSTM	4.70 s
8	**Intention detection model**	**0.14 s**

more than 5 s respectively. There were slight differences in the J48 and Naïve Bayes algorithms as shown in the analysis. From Table 12., it shows the statistical distribution of Instagram text comments with around 2k of harassment comments. Due to this imbalance nature, the fast text method is used for better prediction.

Table 12
Statistical distribution of Instagram dataset.

Dataset used	Total comments	CH	NCH
Instagram comments	10,957	2754	8203

5. Conclusion and future enhancements

In this research work, a systematic method of detecting cyber harassment and the intention behind each comment is analysed and experimented. The lexical and semantic meaning is detected by the conventional method using word embedding techniques such as word2vec. It gives higher accuracy when compared to traditional method. The intention of the harassment text is determined by POS tagging and mapping function. The pre-trained word embeddings language models called Fast Text are used for extracting contextual meaning of each word without losing exact information. The intention detection score

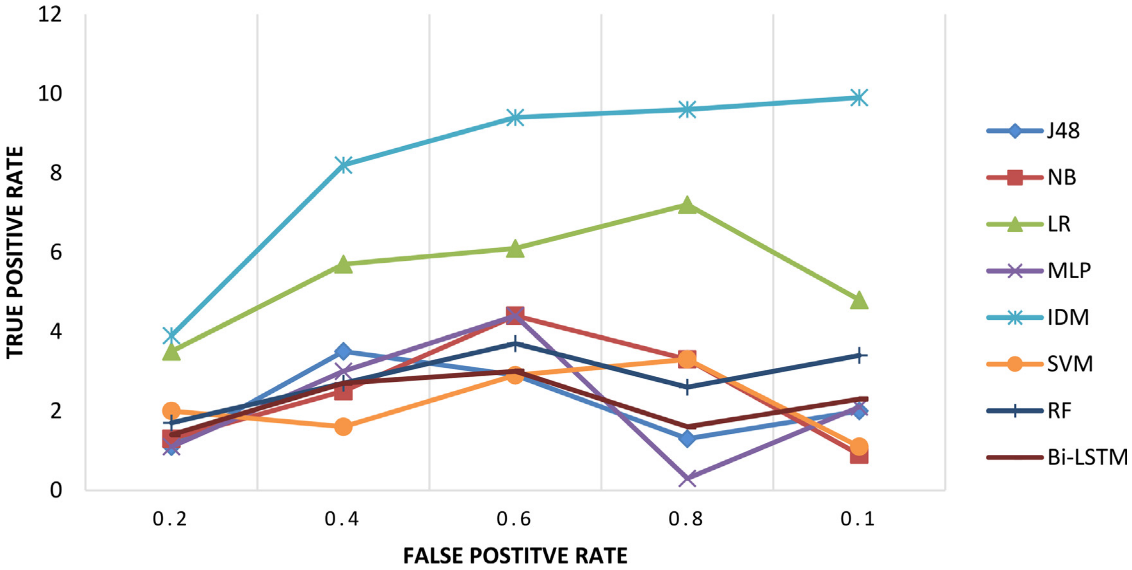


Fig. 14. ROC curve for bag of words for different algorithms.

is obtained by calculating the particular user's bully score and victim score. It indicates the severity of the motive behind the textual comment. The probability-based classification methods are used by combining Word similarity and fast text model along with intention. This model shows better results when compared to J48, NB, SVM, RF, bi-LSTM and MLP neural network. By adding fast text, the time complexity of the model is lesser due to memory management.

In future work, the integration of multiple machine learning classifiers can be used to advance the competence of the model. The bias and variance for single classifiers are larger since every method have some limitations. The ensemble based deep learning techniques may reduce the variance by considering the outliers. The neural networks architecture can be used to increase the efficacy of the classification model. the This will produce better results for detecting and classification of different variants of cyber harassment. Instead of collected data, the dynamic dataset can be used in order to solve the real-world detection of cyberharassment. Some security policies and rules can be incorporated for preventing harassment from social media platforms.

CRediT authorship contribution statement

S. Abarna: Conceptualization, Methodology, Data curation, Writing - original draft. **J.I. Sheeba:** Supervision, Validation. **S. Jayasrilakshmi:** Software, Writing - review & editing. **S. Pradeep Devaneyan:** Visualization, Investigation.

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.

Data availability

Data will be made available on request.

References

Aguado, G., Julian, V., 2019. A multi-agent system for guiding users in on-line social environments. Eng. Appl. Artif. Intell. http://dx.doi.org/10.1016/j.engappai.2020.103740.

AlKhwiter, Wasan, Al-Twairesh, Nora, 2021. Part-of-speech tagging for arabic tweets using CRF and Bi-LSTM. Comput. Speech Lang. http://dx.doi.org/10.1016/j.csl.2020.101138.

Ayo, Femi Emmanuel, Folorunso, Olusegun, Ibharalu, Friday Thomas, Osinuga, Idowu Ademola, Abayomi-Alli, Adebayo, 2021. A probabilistic clustering model for hate speech classification in twitter. Expert Syst. Appl. http://dx.doi.org/10.1016/j.eswa.2021.114762.

Balakrishnan, Vimala, Khan, Shahzaib, Fernandez, Terence, Arabnia, Hamid R., 2019. Cyberbullying detection on Twitter using big five and dark triad features. Int. J. Pers. Indiv. Differ. Elsevier.

Bozyigit, Alican, Utku, Semih, Nasibov, Efendi, 2021b. Cyberbullying detection: Utilizing social media features. Int. J. Expert Syst. Appl. Elsevier.

Bozyiğit, Alican, Utku, Semih, Nasibov, Efendi, 2021a. Cyberbullying detection: Utilizing social media features. J. Sci. Direct Expert Syst. Appl. http://dx.doi.org/10.1016/j.eswa.2021.115001.

Calvo-Morata, Antonio, Alonso-Fernandez, Cristina, Freire, Manuel, Martínez-Ortiz, Ivan, Fernandez-Manjon, Baltasar, 2021. Creating awareness on bullying and cyberbullying among young people: Validating the effectiveness and design of the serious game conectado. Int. J. Telemat. Inform. Elsevier.

Chatzakou, Despoina, Leontiadis, Ilias, Blackburn, Jeremy, Cristofaro, Emiliano De, Stringhini, Gianluca, Vakali, Athena, Kourtellis, Nicolas, 2019. Detecting cyberbullying and cyberaggression in social media. ACM Trans. Web 13 (3), 17.

Chelmis, Charalampos, Yao, Mengfan, 2019. Minority report: Cyberbullying prediction on instagram. In: 11th ACM Conference on Web Science. ACM.

Chen, Qian, Zhuo, Zhu, Wang, Wen, 2019. BERT for joint intent classification and slot filling. arXiv:1902.10909v1 [cs.CL].

Chia, Zheng Lin, Ptaszynski, Michal, Masui, Fumito, Leliwa, Gniewosz, Wroczynski, Michal, 2021. Machine learning and feature engineering-based study into sarcasm and irony classification with application to cyberbullying detection. Int. J. Inf. Process. Manage. Elsevier.

Dennehy, Rebecca, Meaney, Sarah, Walsh, Kieran A., Sinnott, Carol, Cronin, Mary, Arensman, Ella, 2020. Young people's conceptualizations of the nature of cyberbullying: A systematic review and synthesis of qualitative research. Int. J. Aggress. Violent Behav. Elsevier.

Ducange, Pietro, Fazzolari, Michela, 2018. An effective decision support system for social media listening based on cross-source sentiment analysis models. Eng. Appl. Artif. Intell. http://dx.doi.org/10.1016/j.engappai.2018.10.014.

Elsafoury, Fatma, Katsigianni, Stamos, Pervez, Zeeshan, Ramzan, Naeem, 2021. When the timeline meets the pipeline: A survey on automated cyberbullying detection. IEEE Access http://dx.doi.org/10.1109/ACCESS.2021.3098979.

Eronen, Juuso, Ptaszynski, Michal, Masui, Fumito, Smywiński-Pohl, Aleksander, Leliwa, Gniewosz, Wroczynski, Michal, 2021. Improving classifier training efficiency for automatic cyberbullying detection with feature density. Int. J. Inf. Process. Manage. Elsevier.

García-Díaz, José Antonio, Cánovas-García, Mar, 2020. Detecting misogyny in Spanish tweets. An approach based on linguistics features and word embeddings. Future Gener. Comput. Syst. http://dx.doi.org/10.1016/j.future.2020.08.032.

García-Díaz, José Antonio, Cánovas-García, Mar, Colomo-Palacios, Ricardo, Valencia-García, Rafael, 2021. Detecting misogyny in Spanish tweets. An approach based on linguistics features and word embeddings. Future Gener. Comput. Syst. http://dx.doi.org/10.1016/j.future.2020.08.032.

Hou, Zhihao, Maa, Kun, 2020. Attention-based learning of self-media data for marketing intention detection. Eng. Appl. Artif. Intell. http://dx.doi.org/10.1016/j.engappai.2020.104118.

Ireland, Leanna, Hawdon, James, Huang, Bert, Peguero, Anthony, 2021. Preconditions for guardianship interventions in cyberbullying: Incident interpretation, collective and automated efficacy, and relative popularity of bullies. Int. J. Comput. Hum. Behav. Elsevier.

Jain, Ojasvi, Gupta, Muskan, Satam, Sidh, Panda, Siba, 2020. Has the COVID-19 pandemic affected the susceptibility to cyberbullying in India? J. Comput. Hum. Behav. Rep. http://dx.doi.org/10.1016/j.chbr.2020.100029.

KunWang, Cui, Yanpeng, Hu, Jianwei, Zhang, Yu, Zhao, Wei, Feng, Luming, 2020a. Cyberbullying detection, based on the fast text and word similarity schemes. ACM Trans. Asian Low-Resour. Lang. Inf. Process. 20 (1), 6. http://dx.doi.org/10.1145/3398191, 15 pages.

KunWang, Cui, Yanpeng, Hu, Jianwei, Zhang, Yu, Zhao, Wei, Feng, Luming, 2020b. Cyberbullying detection, based on the fasttext and word similarity schemes. ACM Trans. Asian Low-Resour. Lang. Inf. Process. 20 (1), 6. http://dx.doi.org/10.1145/3398191, (November 2020), 15 pages.

Li, Shuangyin, Pan, Rong, Luo, Haoyu, 2021. Adaptive cross-contextual word embedding for word polysemy with unsupervised topic modeling. J. Knowl.-Based Syst. http://dx.doi.org/10.1016/j.knosys.2021.106827.

López-Vizcaíno, Manuel F., Nóvoa, Francisco J., 2021. Early detection of cyberbullying on social media networks. Eng. Appl. Artif. Intell. http://dx.doi.org/10.1016/j.future.2021.01.006.

López-Vizcaíno, Manuel F., Nóvoa, Francisco J., Carneiro, Victor, Cacheda, Fidel, 2021a. Early detection of cyberbullying on social media networks. J. Sci. Direct Future Gener. Comput. Syst. http://dx.doi.org/10.1016/j.future.2021.01.006.

Lopez-Vizcaíno, Manuel F., Novoa, Francisco J., Carneiro, Victor, Cacheda, Fidel, 2021b. Early detection of cyberbullying on social media networks. Int. J. Future Gener. Comput. Syst. Elsevier.

Martín, Alejandro G., Fernández-Isabel, Alberto, 2020. Suspicious news detection through semantic and sentiment measures. Eng. Appl. Artif. Intell. http://dx.doi.org/10.1016/j.engappai.2021.104230.

Pasupa, Kitsuchart, Na Ayutthaya, Thititorn Seneewong, 2019. Thai sentiment analysis with deep learning techniques: A comparative study based on word embedding, POS-tag, and sentic features. Sustainable Cities Soc. http://dx.doi.org/10.1016/j.scs.2019.101615.

Rajesh, Sudha, Sharanya, B., 2020. Recognition and prevention of cyberharassment in social media using classification algorithms. Mater. Today: Proc. http://dx.doi.org/10.1016/j.matpr.2020.10.502.

Ristanti, Putri Yuni, Wibawa, Aji Prasetya, Pujianto, Utomo, 2019. Cosine Similarity for Title and Abstract of Economic Journal Classification. Research Gate Conference Paper, http://dx.doi.org/10.1109/ICSITech46713.2019.8987547.

Sadiq, Saima, Mehmoodb, Arif, Ullah, Saleem, Ahmad, Maqsood, Choi, Gyu Sang, On, Byung-Won, 2021. Aggression detection through deep neural model on Twitter. Future Gener. Comput. Syst. http://dx.doi.org/10.1016/j.future.2020.07.050.

Sainju, Karla Dhungana, Mishra, Niti, Kuffour, Akosua, Young, Lisa, 2021. Bullying discourse on Twitter: An examination of bully-related tweets using supervised machine learning. Comput. Hum. Behav. http://dx.doi.org/10.1016/j.chb.2021.106735.

Sanchez-Medina, Agustin J., Galvan-Sanchez, Inmaculada, Fernandez-Monroy, Margarita, 2020. Applying artificial intelligence to explore sexual cyberbullying behavior. Heliyon Elsevier.

Shen, Gehui, Deng, Zhi-Hong, Huang, Ting, Chen, Xi, 2020. Learning to compose over tree structures via POS tags for sentence representation. Expert Syst. Appl. http://dx.doi.org/10.1016/j.eswa.2019.112917.

Tahmasbi, Nargess, Rastegari, Elham, 2018. A socio-contextual approach in automated detection of public cyberbullying on Twitter. ACM Trans. Soc. Comput. 1 (4), 15.

Thun, Lee Jia, Teh, Phoey Lee, Cheng, Chi-Bin, 2021. Cyberaid: Are your children safe from cyberbullying? J. King Saud Univ. Comput. Inf. Sci. http://dx.doi.org/10.1016/j.jksuci.2021.03.001, science direct.

Tolba, Marwa, Ouadfel, Salima, Meshoul, Souham, 2021. Hybrid ensemble approaches to online harassment detection in highly imbalanced data. Expert Syst. Appl. http://dx.doi.org/10.1016/j.eswa.2021.114751.

Tran, Oanh Thi, Luong, Tho Chi, 2019. Understanding what the users say in chatbots: A case study for the Vietnamese language. Eng. Appl. Artif. Intell. http://dx.doi.org/10.1016/j.engappai.2019.103322.

Tseng, Shu-Cih, Lu, Yu-Ching, Chakraborty, Goutam, et al., 2019. Comparison of sentiment analysis of review comments by unsupervised clustering of features using LSA and LDA. IEEE Trans..

Wang, Yufeng, Maa, Kun, Garcia-Hernandez, Laura, 2019. A CLSTM-TMN for marketing intention detection. Eng. Appl. Artif. Intell. http://dx.doi.org/10.1016/j.engappai.2020.103595.

Xue, Siyuan, Ren, Fuji, 2021. Intent-enhanced attentive bert capsule network for zero-shot intention detection. J. Neurocomput. http://dx.doi.org/10.1016/j.neucom.2021.05.085.

Yuvaraj, Natarajan, Chang, Victor, Gobinathan, Balasubramanian, Pinagapani, Arulprakash, Kannan, Srihari, Dhiman, Gaurav, Rajan, Arsath Raja, 2020. Automatic detection of cyberbullying using multi-feature based artificial intelligence with deep decision tree classification. Int. J. Comput. Electr. Eng. Elsevier.