iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** IV **Month of publication:** April 2023

DOI: https://doi.org/10.22214/ijraset.2023.50529

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue IV Apr 2023- Available at www.ijraset.com

Natural Language Processing and Naïve Bayes Classifier Algorithm to Automate the Detection of Cyberbullying

Fagun N. Patel[1], Kashish N. Mehta[2], Shubbh R. Mewada[3]
[1,2,3]*Student, Department of Computer Engineering, SAL Institute of Technology and Engineering Research (Gujarat Technological University), India*

Abstract: ***The impact of social media on contemporary culture has been unprecedented, making it the most significant medium of our times. While it has had a positive effect on people's worldview, social media has also been linked to a rise in undesirable phenomena such as cyberbullying, cyberstalking, and cybercrime. Cyberbullying, in particular, can have a negative impact on individuals' mental health and has even been identified as the root cause of mental health issues in some cases. The proliferation of sexually explicit comments and the spread of rumors by multiple individuals are some of the negative influences that have been observed in the social media ecosystem. In recent years, academics have been increasingly concerned about the indicators of online harassment. Our goal is to develop a system that can detect instances of online abuse using Natural Language Processing (NLP) and Naïve Bayes, among other techniques.***

The cultural norms have shifted dramatically due to the rapid transmission of the COVID-19 virus, resulting in a rise in cyberbullying, especially among adolescents. The younger generation is more likely to engage in this practice, which has become more widespread with the stratospheric rise in popularity of various online engagement-promoting platforms. The COVID-19 pandemic has changed the way people interact online and has contributed to an increase in cyberbullying. As more people began working from home, bullying became a more significant concern.

Our proposed system includes modules for data cleansing, text mining, word embedding, and regression analysis, among others. We utilize the Lemmatization technique for text mining, which enhances the model's precision. We also utilize the Vader emotion for feature extraction, which generates word vectors that are scattered numerical representations of word attributes. Additionally, Naive Bayes is used for data categorization to prevent overfitting in the proposed model. This would help in creating vectors that connect words with similar meanings.

Keywords: Cyberbullying, automatic detection, Natural Language Processing, Naïve Bayes, TF-IDF, Word2Vec.

I. BACKGROUND OF THE WORK

Social media has become an incredibly popular online platform in recent years, offering both benefits and drawbacks for society at large. While the expansion of social media has increased access to information and employment opportunities, it has also increased the likelihood of abusive behavior, cyberbullying, and criminal activity. Unfortunately, the negative impacts of cyberbullying on mental health and psychological well-being cannot be overstated. As such, researchers have begun to develop methods for identifying instances of cyberbullying using natural language processing (NLP) and machine learning techniques.

Social media platforms allow users to share personal images, videos, and conversations with individuals from all over the world. A majority of social media users access these platforms via mobile devices, with popular platforms including Facebook, Twitter, Instagram, and TikTok. Social media has had an impact on a range of industries, from education and business to philanthropy, and has contributed to the expansion of the global economy.

Researchers have developed a number of deep learning systems for identifying instances of cyberbullying. For example, a deep neural network model can analyze data from the physical environment to identify instances of cyberbullying in real-time. The use of transfer learning and convolutional neural networks can also aid in the detection of cyberbullying, particularly through the use of word embedding techniques.

One of the challenges of identifying cyberbullying is the difficulty in addressing cross-modal linkages and structural correlations between social media sessions. Researchers have proposed innovative solutions to overcome these challenges, such as XBully, a method for identifying cyberbullying that reframes multi-modal social media data as a heterogeneous network.

Other researchers have explored the use of neural networks, including long-short term memory (LSTM) layers, for identifying cyberbullying phrases in text. Huang (2018) developed a unique neural network model that combined the convolutional architecture with the LSTM layout to detect cyberbullying phrases. The use of stacked core layers in network architecture also improved the performance of neural networks, while the addition of support vector machine-like performance using L2 weight regularization and a Hinge loss function improved the model's activation function.

The detection of cyberbullying is of significant public health concern, and an effective detection model is of crucial scientific interest. Researchers have developed supervised machine learning approaches for monitoring Twitter for cyberbullying, which have yielded promising results. For instance, Ghosh (2017) reported a F measure of 0.936% and a region under the receiver-operating characteristic curve of 0.943% using their innovative detection approach.

So, the identification of cyberbullying on social media platforms remains a challenging and complex problem. However, the development of machine learning and deep learning techniques has shown promise in detecting cyberbullying and addressing its negative impacts. Continued research and innovation in this area will be essential for safeguarding individuals' mental health and well-being on social media.

II. PROBLEM AREAS

In the course of reviewing various approaches and algorithms used in the development of models, several challenges were identified, which are addressed in this section through recommended solutions. One of the common techniques used in text mining is stemming, which involves removing the initial characters of a phrase. However, it has been found that this approach often leads to inaccurate outputs, thereby diminishing the predictability of the model. To address this issue, lemmatization has been recommended, as it improves word meaning by analyzing based on parts of speech, resulting in more accurate dictionary words.

Another technique commonly used in word embedding is the neural network-based word2vec method, which automatically identifies the associations between words in a dataset. However, it has been observed that word2vec faces significant challenges in managing out-of-vocabulary words, making it unsuitable for word embeddings. Additionally, training parameters for a new language depend on local information presented in a dataset, and a large number of texts are required for proper model training. To overcome these challenges, the suggested model for word embedding incorporates the TF-IDF approach, a statistical metric that assigns high weights to less frequent words and low weights to more frequent words, providing a numerical representation of each word's significance in the document.

Logistic regression is another commonly used technique in predictive modeling, which focuses on modeling the probability of a single outcome. However, logistic regression has several limitations, such as the requirement of a linear relationship between variables for accurate predictions. To improve the overall accuracy of the model and overcome the limitations of logistic regression, the proposed model employs the Naïve Bayes classifier algorithm technique, which minimizes data overfitting and is applicable to both discrete and continuous data. Additionally, the Naïve Bayes technique handles missing values in the dataset automatically.

So, the proposed solutions provide effective ways to overcome the challenges associated with common techniques used in the development of models. By implementing these solutions, researchers and practitioners can enhance the accuracy and predictability of their models, leading to better decision-making and outcomes.

III. METHODOLOGY

The proposed system consists of several modules, including data purification, text mining, word embedding, and regression analysis. The recommended approach for text mining includes the incorporation of lemmatization, which increases the precision of the model. The Vader sentiment analysis tool is useful for feature extraction. The suggested method employs Naive Bayes for data classification, which helps in reducing overfitting. It also uses sentiment analysis to generate word vectors, which are distributed numerical representations of word characteristics. These word attributes can be phrases that represent the context of the various words in our lexicon, aiding in the construction of vectors that connect words with similar meanings.

A. Collection of data

Cyberbullying is a complex phenomenon that can occur in various forms, such as sharing someone's hashtag without their consent, posting offensive video content, or using derogatory language. However, the majority of cyberbullying incidents are conducted through text messages. Fortunately, the Twitter Application Programming Interface (API) enables real-time access to social media text data. To extract valuable information from tweets, Tweepy, a Python application, can analyse them for relevant hashtags.

```
consumer_key = 'hC02vDfkSX1IBYOZPRjCpgXCm'
consumer_secret = 'Pm080QkpceLW3Q1RKBRIIZDI3Nn1khMogxUeB2KyJjFqIamr6i'
access_key = '1321309664772907008-NT3w9KrGuT4BzsCKHdiFEgq8HamZ9B'
access_secret = 'fkudHFKdUlbkaGLUwnNGeEU2MxpPVKC1nqkEGpVmAqk3S'
```

```
auth = tweepy.OAuthHandler(consumer_key, consumer_secret)
auth.set_access_token(access_key, access_secret)
api = tweepy.API(auth)

# Enter Hashtag and initial date
print("Enter Twitter HashTag to search for")
words = input()
print("Enter Date since The Tweets are required in yyyy-mm--dd")
date_since = input()
```

Fig. 1. Twitter API Syntax

Code snippets show that users will need to specify both the exact hashtag they wish to search for and the period of time that has elapsed since the last time the hashtag was used. The Twitter API will then collect all of the tweets in real time that use that hashtag.

B. Data Preparation

The utilization of Twitter's Application Programming Interface (API) is a powerful tool for extracting data that is both time- and hashtag-specific. To retrieve tweets published after a certain date, the Tweepy library can be employed to perform a search for the specific hashtag, utilizing the "date since" attribute. Through the Twitter API, it is possible to access a range of data associated with a user's profile, including their name, bio, location, list of followers and those they follow, as well as the number of tweets they have made. Additionally, the content of the tweets and the hashtags used since a specified date can be accessed via the Twitter API, further expanding the scope of available data for analysis.

```
Enter Twitter HashTag to search for
Tiktok
Enter Date since The Tweets are required in yyyy-mm--dd
2021-01-01
1
<tweepy.cursor.ItemIterator object at 0x0000028D54CDD280>
Done----------
Scraping has completed!
```

Fig. 2. Scrapping of the data based on Hashtag

```
import pandas as pd
data= pd.read_csv("scraped_tweets2.csv")
data.head()
```

	Unnamed: 0	username	description	location	following	followers	totaltweets	retweetcount	text
0	0	Alrightpunk	Indie author. Host of @TentaclesNot. Book, com...	UK	896	1167	82012	0	I can't fathom how TikTok works.\n\nAm I offic...
1	1	zah3200	Abbottsford Projects ✨ USAF ❗ Valdosta, GA. IG...	Somewhere	80	96	2017	0	Nomore Snapchat, No TikTok, IG business, Twitt...
2	2	Yerqiz0305	only being me 🦋 this is safe zone and no petty...	NaN	9	0	3231	33	From this video we can know:\n1. Yeri is such ...
3	3	larin_rivas	-hyunjin you are my sunshine☆,she/her, stay&ar...	Hyunjin&skz♡☆	60	197	10286	2393	[TIKTOK] 220808\n\n@.newhopeclub Tiktok Update...
4	4	on_hema	love Niki and Enhypen stay together ENGENES an...	pokhara	383	282	87427	2393	[TIKTOK] 220808\n\n@.newhopeclub Tiktok Update...

Fig. 3. Scrapped data from the twitter

Once the data has been extracted from Twitter, it is read into the data frame through the Pandas library's read csv () method.

C. Pre-Processing of Data

After the collection of data, a preprocessing step is initiated to eliminate irrelevant tweets that contain too many characters. To ensure that sentiment analysis can be performed accurately, the preprocessing stage involves eliminating or replacing offensive material, such as word contractions, emotion emojis, and inappropriate language. This step helps to ensure that the model is trained on meaningful data, improving its overall accuracy.

```
with open ('contractions.json', 'r') as f:
    contractions_dict = json.load (f)contractions = contractions_dict
```

Fig. 4. Syntax for the contraction:

1) *Conversion of emoji:* It is possible to take the sentiment analysis a step further by utilizing Python's regular expression module to determine the underlying sentiment conveyed by emojis. Emojis such as a grin, a laugh, and grief are all examples of emotions that fall on the positive end of the spectrum. On the other hand, negative emotions like rage and melancholy occupy the lower end of the emotional scale. By detecting these emotions through regular expressions, it is possible to gain a more nuanced understanding of the sentiment expressed in the tweets. This approach can be used to further refine the data cleansing process and improve the accuracy of the sentiment analysis.

```
import re
def emoji(tweet):
    # Smile -- :), : ), :-), (:, ( :, (-:, :') , :O
    tweet = re.sub(r'(:\s?\)|:-\)|\(\s?:|\(-:|:\'\)|:O)', ' positiveemoji ', tweet)
    # Laugh -- :D, : D, :-D, xD, x-D, XD, X-D
    tweet = re.sub(r'(:\s?D|:-D|x-?D|X-?D)', ' positiveemoji ', tweet)
    # Love -- <3, :*
    tweet = re.sub(r'(<3|:\*)', ' positiveemoji ', tweet)
    # Wink -- ;-), ;), ;-D, ;D, (;,  (-; , @-)
    tweet = re.sub(r'(;-?\)|;-?D|\(-?;|@-\))', ' positiveemoji ', tweet)
    # Sad -- :-(, : (, :(, ):, )-:, :-/ , :-|
    tweet = re.sub(r'(:\s?\(|:-\(|\)\s?:|\)-:|:-/|:-\|)', ' negetiveemoji ', tweet)
    # Cry -- :,(, :'(, :"(
    tweet = re.sub(r'(:,\(|:\'\(|:"\()', ' negetiveemoji ', tweet)
    return tweet
```

Fig. 5. Pseudocode

2) *Cleaning and Processing of the tweets:* Pre-processing and post-processing are utilised because the obtained Twitter data needs to be cleaned up, the words need to be eliminated or replaced, and the emotions need to be extracted.

text	hashtags	processed_tweet
I can't fathom how TikTok works.\n\nAm I offic...	[]	can fathom how tiktok work am officially old
Nomore Snapchat, No TikTok, IG business, Twitt...	['Focused']	nomore snapchat no tiktok ig business twitter ...
From this video we can know:\n1. Yeri is such ...	[]	from this video we can know yeri is such mood ...
[TIKTOK] 220808\n\n@.newhopeclub Tiktok Update...	[]	tiktok tiktok update with jake and jungwon enh...
[TIKTOK] 220808\n\n@.newhopeclub Tiktok Update...	[]	tiktok tiktok update with jake and jungwon enh...

Fig. 6. Processing and cleaning of the tweets

In order to capture the user's viewpoint on a particular topic, the processed tweets will be categorized into abusive phrases and positive terms in preparation for the following phase, which will involve the creation of a graphical representation of that sentiment. The re.i.e., regular expression library is utilized here to identify the underlying sentiment of each tweet. Positive sentiments, such as joy and laughter, are recognized using regular expressions, while negative sentiments, such as anger and sadness, are identified using different regular expressions. Once the sentiments have been identified, the tweets are grouped accordingly. In addition, any punctuation marks such as ([]) are removed so that only the words conveying meaning and emotion remain.

D. Lemmatization

In order to effectively model the data, it is crucial to take into account the contextual nuances of the tweets as well as the underlying etymologies of the words. To achieve this, a rule-based technique known as lemmatization is employed.

This process involves identifying the canonical, dictionary, or citation form of a given word, which is referred to as its lemma. By reducing all variations of a word to its lemma, the data can be standardized and made more amenable to analysis. For example, the lemma of the words "runs," "running," and "ran" is "run," since they all refer to the same underlying noun. By applying lemmatization to the data, we can more accurately capture the semantic content of the tweets and generate insights that are more robust and reliable.

```
from nltk.stem.wordnet import WordNetLemmatizer
lemmatizer = WordNetLemmatizer()
tokenized_tweet = tokenized_tweet.apply(lambda x: [lemmatizer.lemmatize(i) for i in x]) .
Here every processed tweet goes through a lemmatization and it convert it into a root word wherever
required.
```

Fig. 7. Pseudocode

1) *Stopwords Removal:* In the context of natural language processing, a crucial step in text analysis involves the removal of stopwords, which are frequently used, yet semantically uninformative words. In order to streamline the data cleansing process, the widely-used Natural Language Toolkit (NLTK) offers a pre-defined list of stopwords that can be imported using the corpus module. Following lemmatization, words that do not carry any emotional or contextual significance are identified and removed from the dataset based on their presence in the imported list of stopwords. This results in a more refined dataset that is better suited for subsequent stages of analysis.

```
stopwords = nltk.corpus. Stopwords.words('english')
for i in range(len(tokenized_tweet)):
    tokenized_tweet[i] = ' '.join(tokenized_tweet[i])
data['processed_tweet'] = tokenized_tweet
```

Fig. 8. Pseudocode

E. Vader Sentiment Analysis

The proposed approach incorporates the Valence Aware Dictionary for Sentiment Reasoning (VADER) model to analyze text and estimate the polarity (positive or negative) as well as the intensity of emotions. The VADER sentiment analyzer is applied to unlabelled text input by loading its features into the Natural Language Toolkit (NLTK) package's model. The dictionary's ability to transform lexical input into an emotion intensity measure is critical to the effectiveness of this method of sentiment analysis.

To obtain the final sentiment score, the values associated with each word in the text are summed. The Polarity Score of each tweet is calculated using the SentimentIntensityAnalyzer() method of the Vader module, which generates the tweet's negative, positive, and composite scores. This comprehensive approach to sentiment analysis allows for a deeper understanding of the emotions conveyed in the tweets and their corresponding intensity.

	processed_tweet	neg	pos	compound	sentiment
0	can fathom how tiktok work am officially old	0.000	0.000	0.0000	neutral
1	nomore snapchat no tiktok ig business twitter ...	0.172	0.203	0.1027	positive
2	from this video we can know yeri is such mood ...	0.084	0.000	-0.2960	negative
3	tiktok tiktok update with jake and jungwon enh...	0.000	0.000	0.0000	neutral
4	tiktok tiktok update with jake and jungwon enh...	0.000	0.000	0.0000	neutral

Fig. 9. Polarity Score of Sentiment

Once the VADER sentiment analyzer computes the composite score for each tweet, the sentiment of each tweet can be easily determined. Tweets that have a composite score of zero or greater are considered positive, tweets with a score less than zero are negative, and tweets with a score of exactly zero are considered neutral. By categorizing the tweets in this way, it becomes possible to create a pie chart that visually represents the sentiment of the collected tweets.

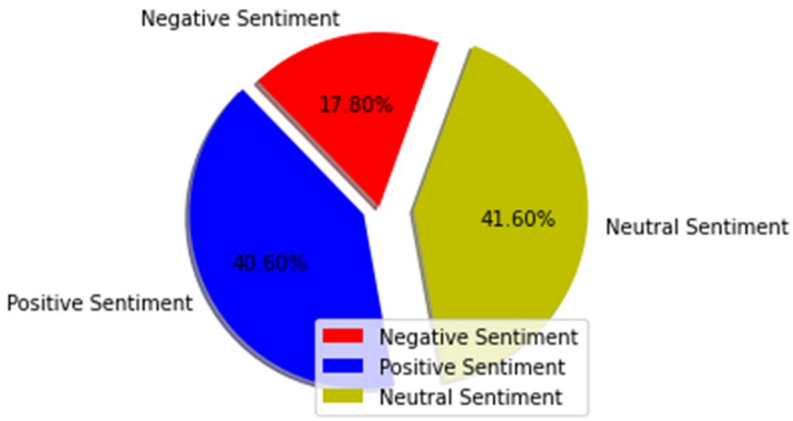


Fig. 10. Visualization of Harmful Words in Relation to Good and Neutral Terms

F. Naïve Bayes Classifier

In order to assess the performance of the model, it is essential to undertake several actions, such as applying the naive bayes method to estimate the probabilities of each characteristic belonging to a particular class, and to classify the data into distinct groups based on the provided attributes. This enables the subdivision of classes based on available attributes, which in turn allows for better classification accuracy. In the context of this approach, the naive Bayes classifier operates under the assumption that each characteristic can be interpreted independently of the others, which is made possible by the concept of conditional probability. This probability assumes that the likelihood of one feature is independent of the probability of any other feature, which in turn enhances the model's ability to accurately identify the features and improves its overall performance. The implementation code for the naive bayes classifier can be found further down in this article, as it is a crucial step towards determining the F1 score of the model.

	processed_tweet	sentiment
0	what look like scene from post apocalyptic mov...	negative
1	any romance book popular on tiktok is nothing ...	positive
2	just like love bomb we are one sooyoung 수영 yoo...	positive
3	whatever whatever whatever	neutral
4	your phone storage fill up and your requested ...	neutral
...	...	...
160	taylor is at on tiktok and it international ca...	negative
161	jeff satur why don you stay worldtour ver itun...	neutral
162	hezz_tiktok with mc minzy and eunji of brave g...	positive
163	just like love bomb we are one sooyoung 수영 yoo...	positive
164	since our dance prodigy ni ki is gaining lot o...	positive

Fig. 11. sentiment of the processed tweet

IV. EVALUATION

The evaluation of the model involves the use of a scoring formula that takes into account both correct and incorrect classification of data. By comparing the actual results with the expected results, we can determine the level of accuracy that the proposed model has in representing reality. This allows us to quantify the performance of the model in terms of its ability to correctly classify data, and thus assess its effectiveness in meeting the desired objectives. Various scoring metrics can be used to evaluate the model, such as F1 score, precision, recall, and accuracy, depending on the specific requirements of the problem being addressed.

```
tnb = TweetNBClassifier(df_train)
tnb = tnb.fit ()
predict = tnb.predict(df_test)
score = tnb.score(predict,df_test.sentiment.tolist())
print(score)
```

Fig. 12. Results

It is worth noting that models are evaluated based on a combination of functions, including TweetNBClassifier, predict, and score, to determine their accuracy. In the case of the suggested model, it has been found to have an accuracy rate of 78.3 percent, indicating that it is able to correctly classify sentiment in the majority of cases. It is important to note, however, that the accuracy of the model may vary depending on the specific dataset being analyzed and the criteria used for classification. Therefore, further testing and refinement may be necessary to optimize its performance for a particular use case.

V. CONCLUSION

The issue of cyberbullying has become increasingly prominent in modern society due to the widespread use of social media platforms. The harmful effects of cyberbullying on individuals and society as a whole have led to a call for the development of effective tools for detecting and combating this phenomenon. In response to this challenge, a research study was conducted with the primary objective of developing a software tool capable of automatically detecting signs of cyberbullying on the microblogging platform Twitter.

The proposed solution involves a combination of machine learning algorithms, including support vector machines and the Naive Bayes method. The Vader sentiment serves as the feature vector, replacing the TextBlob. The use of Vader sentiment allows for the analysis of emoticons, slang, conjunctions, and capitalization, among other features. Additionally, Vader sentiment does not require training data to perform accurately, unlike TextBlob, which eliminates unknown elements and only analyses words and phrases that can be assigned polarity.

The model is designed to improve in precision as more users employ it, allowing for the automatic recognition of cyberbullying signs and prompt action to be taken when necessary. The suggested method involves ignoring tweets that have been correctly labeled as positive and contain no bullying content. If the bot detects cyberbullying, it issues a warning and provides the user with the option to delete the offending tweet. Furthermore, misinterpreted tweets will receive a thumbs-down in response, and the classifier will flag similar tweets in the future as cyberbullying.

The proposed solution is flexible enough to incorporate input from all parties involved, as different types of cyberbullying require different responses. Nonetheless, improving the classifier's performance requires the inclusion of new data sources. The proposed method has the potential to be used to detect cyberbullying in additional languages, given the multilingual nature of social media. However, identifying cyberbullying based on audio or video content requires further investigation.

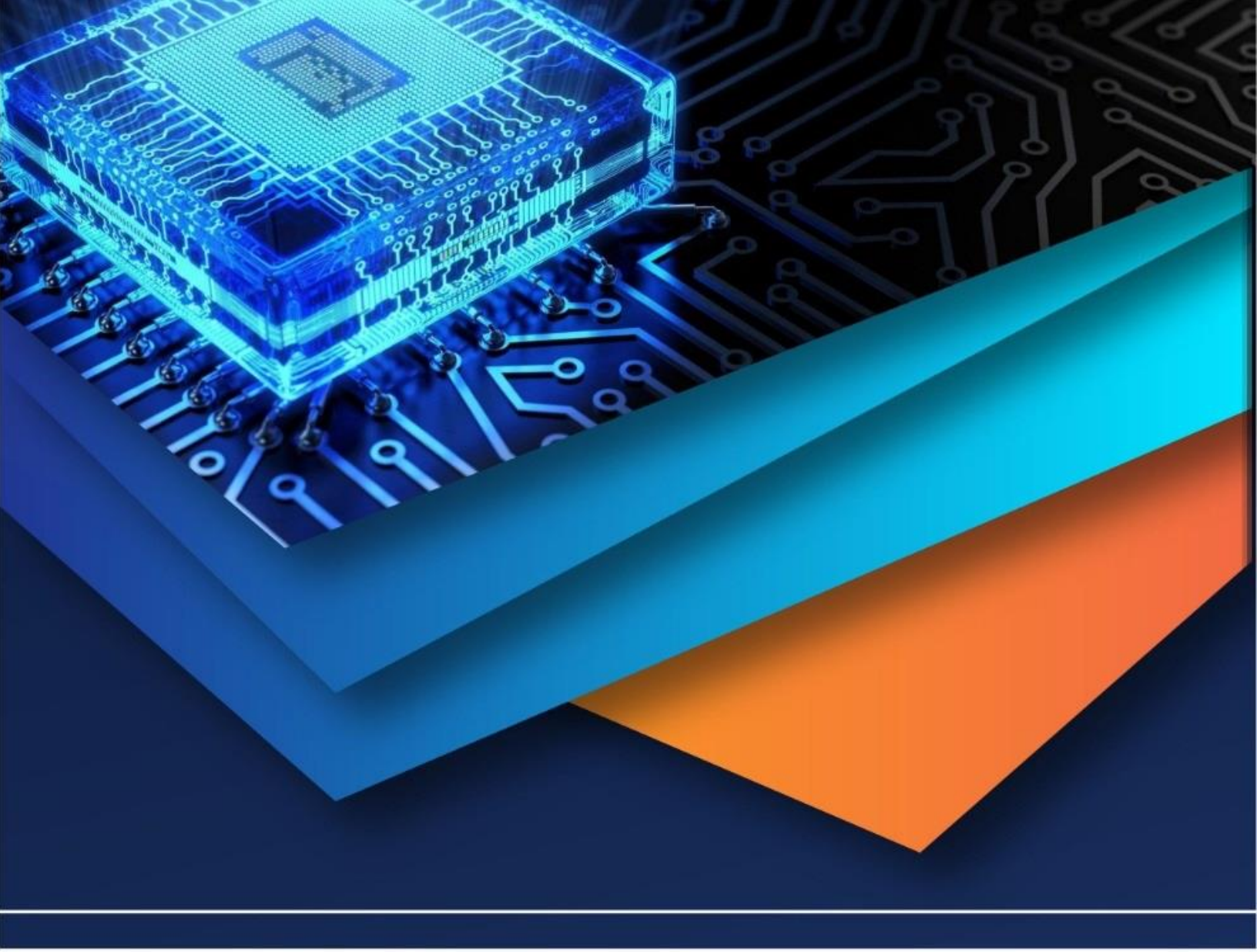
The proposed model was evaluated using a combination of functions, including TweetNBClassifier, predict, and score, and achieved an accuracy rate of 78.3 percent. The scoring formula compared the actual results to the anticipated results, taking into account both correct and incorrect tagging to determine how accurately the model represents reality.

In summary, the proposed model offers a promising solution to the challenge of detecting cyberbullying on social media. With the potential to incorporate new data sources and improve in precision as more users employ it, the model has the potential to significantly enhance moderation efforts and promote safer online interactions.

REFERENCES

[1] Adnan, 2019. An analytical study of information extraction from unstructured and multidimensional big data. Journal of Big Data , pp. 85-91.
[2] Agarwal, 2015. Research on Data Preprocessing and Categorization Technique for Smartphone Review Analysis. International Journal of Computer Applications , pp. 30-36.
[3] Agrawal, 2018. Deep learning for detecting cyberbullyingacross multiple social media platforms. s.l., Springer, pp. 141-153.
[4] Ajlan, 2018. Deep learning algorithm for cyberbul-lying detection. International Journal of Advanced Computer Scienceand Applications, pp. 1-9.
[5] Al-garadi, 2016. Cybercrimedetection in online communications: The experimental case of cyberbul-lying detection in the twitter network. Computers in Human Behavior, pp. 433-443.
[6] Anon, 2012. Stemming Algorithms: A Comparative Study and their Analysis.. International Journal of Applied Information Systems,, pp. 7-12.
[7] Anon, 2012. Stemming Algorithms: A Comparative Study and their Analysis.. International Journal of Applied Information Systems, pp. 7-12.
[8] Breiman, 2001. Naïve BayesS, Berkeley, CA: Statistics Department University of California..
[9] Campbell, 2012. Online social networking and the experience of cyber-bullying.. Studies in Health Technology and Informatics, pp. 212-217.
[10] Ghosh, 2017. Toward multimodal cyberbullying detection. s.l., s.n., pp. 2090-2099.
[11] Huang, 2018. Weakly supervised cyberbullying detectionusing co-trained ensembles of embedding models. s.l., IEEE, pp. 479-486.
[12] Jabbar, 2020. Empirical evaluation and study of text stemming algorithms. Springer, pp. 5559-5588.
[13] Karjaluoto, 2015. An-tecedents of social media b2b use in industrial marketing context:customers' view. Journal of Business & Industrial Marketing, pp. 1-8.
[14] Khyani, 2021. An Interpretation of Lemmatization and Stemming in Natural Language Processing. Journal of University of Shanghai for Science and Technology, pp. 350-357.
[15] Koto, 2015. A Comparative Study on Twitter Sentiment Analysis: Which Features are Good?. Passau, Germany, s.n., pp. 453-457.
[16] Peng, 2002. An Introduction to Logistic Regression Analysis and Reporting. The Journal of Educational Research, pp. 3-14.
[17] Peng, 2002. An Introduction to Logistic Regression Analysis and Reporting.. The Journal of Educational Research, Volume 96, pp. 3-14.

[18] Petcharat, 2017. A Corpus-Based Study of English Synonyms. Language Education and Acquisition Research Network Journal, pp. 1-15.
[19] Qaiser, 2018. Text Mining: Use of TF-IDF to Examine the Relevance of Words to Documents.. International Journal of Computer Applications , pp. 1-8.
[20] Qaiser, 2018. Text Mining: Use of TF-IDF to Examine the Relevance of Words to Documents.. International Journal of Computer Applications, Volume 181, pp. 1-8.
[21] Rajasekaran, 2018. Review on automatic text summarization. International Journal of Engineering & Technology, pp. 456-460.
[22] Ramachandra, 2020. Automated cyberbullying detection insocial media using an svm activated stacked convolution lstm network. s.l., s.n., pp. 170-174.
[23] Singh, 2015. VECTOR SPACE MODEL: AN INFORMATION RETRIEVAL. International Journal of Advanced Engineering Research and Studies, pp. 1-3.
[24] Skorkovská, 2012. Application of Lemmatization and Summarization Methods in Topic Identification Module for Large Scale Language Modeling Data Filtering.. Brno, Czech Republic, s.n.
[25] Skorkovská, 2012. Application of Lemmatization and Summarization Methods in Topic Identification Module for Large Scale Language Modeling Data Filtering.. Brno, Czech Republic, s.n.
[26] Talib, 2016. Text Mining: Techniques, Applications and Issues. International Journal of Advanced Computer Science and Applications, pp. 1-5.
[27] Vandebosch, 2014. Cyberbullying on social network sites. anexperimental study into bystanders' behavioural intentions to help thevictim or reinforce the bully. Computers in Human Behavior, pp. 259-271.
[28] Wang, 2020. Multi-modal cy-berbullying detection on social networks. s.l., IEEE, pp. 1-8.
[29] Webster, 1992. Tokenization as the initial phase in NLP. s.l., DBLP, pp. 1-8.
[30] Zhang, 2015. Using Word2Vec to process big text data. s.l., s.n.
[31] Zhang, 2015. Using Word2Vec to process big text data.. San Jose, CA, IEEE International Conference..



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY