



Contemporaneity of Language and Literature in the Robotized Millennium

Vol: 3(2), 2021

REST Publisher; ISBN: 978-81-936097-3-6

Website: <http://restpublisher.com/book-series/cllrm/>

Identifying and Mitigating Harmful Comments on Social Networking Sites using NLP, Machine Learning, and the EDAS Method

*Nirmala Shivram Padmavat

Nutan Mahavidyalaya Selu, Maharashtra, India.

*Corresponding Author Email: drnspadmavat@gmail.com

Abstract

Due to the frequent misspellings, informality, and lack of structure in textual content on online social media, current research on message-level offensive language recognition struggles to accurately detect offensive language. However, user-level offensiveness detection appears to be a more practical approach, although it requires further study. The use of foul and abusive language has significantly increased in the era of social media and networking, with young people playing a major role in this trend. Over half of young individuals who use social media for communication fall victim to cyberbullying. Insults on social networking websites can result in negative network interactions. The presence of disrespectful comments on the internet contributes to a toxic atmosphere. Unfortunately, most tools and algorithms designed to understand and mitigate this issue are not effective. Moreover, insult detection systems that utilize machine learning and natural language processing suffer from low recall rates. In order to address this problem, the article aims to identify instances of bullying in text by conducting analysis and experimenting with various approaches. The goal is to develop practical methods for characterizing such comments. As an alternative to Logistic Regression, SVM, Random Forest, and Gradient Boost, we employed a different method for evaluating the effectiveness of detecting bullying and hostile comments. We considered Train Accuracy, Test Accuracy, AUC Score, and Cross Validation as evaluation parameters. By assessing these remarks, we were able to determine their accuracy and propose an effective method for detecting such comments. The rankings offer a relative comparison of the models' performances based on the chosen evaluation criteria. However, it's crucial to remember that rankings alone might not provide a comprehensive understanding of the models' specific strengths and weaknesses. Indeed, conducting additional analysis and considering specific metrics and requirements are essential for making informed decisions when selecting a model. In the context of identifying cyberbullying on social media, NLP and machine learning techniques are commonly employed to analyze social media comments and determine whether individuals or groups are engaging in aggressive behavior. The development of an efficient classifier is a crucial component of a final prototype system aimed at detecting cyberbullying on social media. This classifier would play a central role in accurately identifying and flagging instances of bullying in real-time or retrospectively.

Keywords: NLP, machine learning, EDAS approach

Introduction

People have recently shown a greater level of interest in the popular social network activities. The ability for people all over the world to freely express and exchange ideas in real time has been made possible by microblogging programs, which has facilitated the spread of hostile content. Researchers were able to look into distinct events' online social feelings because to this expression. According to Agarwal (2015), extremist groups are misusing various social media websites like Twitter, Facebook, YouTube, blogs, and discussion forums to spread their ideologies and promote radicalization while also enlisting recruits and establishing online virtual communities. The automatic detection of online hate speech, bullying, aggression, abusive language, and misogynistic remarks, particularly in social media, has been an active area of research for many researchers. They have proposed several Artificial Intelligence (AI) models using a variety of Machine Learning (ML) techniques. According to Schmidt and Wiegand's definition of "hate speech" from 2017, the phrase refers to a variety of offensive user-generated content. Cyberbullying and cyber aggression are two other severe problems that the majority of Internet users face (Sahay et al., 2018). The most vulnerable social media sites to such attacks are Twitter and Facebook (Sahay et al., 2018).

Language translation software like Google Translate makes extensive use of NLP. NLP is indeed utilized by word processors such as Microsoft Word and Grammar to check the grammar of documents [2]. Similarly, in your case, NLP was employed to assess the tone of Bengali text. This allowed for the identification of sentences that contain expressions indicative of bullying. By leveraging NLP techniques, it becomes possible to analyze the linguistic features and context of the text to detect potential instances of bullying and assess the overall tone of the sentences.

Cyberbullying, or online harassment, refers to the use of internet communication to threaten or intimidate individuals, often through the sending of frightening or compromising messages. The prevalence of offensive and aggressive language has significantly increased in the era of social media and internet networking. Such remarks contribute to a disrespectful atmosphere in online spaces [5]. In the past, cyberbullying was not adequately addressed or often ignored. Poor user engagement on social networking platforms was often cited as a factor, with individuals being advised to block or disconnect from the situation if they received harassing comments. However, the current situation has undergone significant changes. According to a 2019 survey, 70 of the 100 women who experience cyber harassment are between the ages of 15 and 25. Harassment and defamation account for 18% of the allegations and instances of harassment brought before the only cybercrime tribunal in the nation [6]. Finding vulgar and insulting language when it occurs on online forums and then reporting these occurrences to Bangladesh in order to track out the real-world perpetrators of such behavior are two important problems in the fight against cyberbullying. A framework to naturally and brilliantly separate hostility and instances of online provocation is not integrated into any current online network or internet-based social networks (such as Facebook and Twitter). This important problem wasn't considered to be worth exploring in the past because it wasn't real, but it is now in a dangerous stage. This impact on the digital stage cannot be ignored, which is why it has taken center stage in studies looking at effective solutions. To control this movement against online harassment, experts and cybercrime agencies must give it serious thought [5]. Therefore, the aim of this work is to identify objectionable terms and phrases used in Bengali that are deemed to be online harassment on social networking sites.

Materials and Method

In the area of natural language processing, significant work has been done to introduce various methods for handling text data. In several works of literature, the polarity of text data was calculated [7]. Accordingly, they separated sentiment analysis into three sections: document level, phrase level, and entity and aspect level. A score was determined using the positive and negative dictionary after data cleaning, preprocessing, and stemming. We can determine from the end result whether the input sentence was favorable, negative, or neutral. They were effective in giving the sentences provided as input positive, negative, or zero values, which led to a final review of the article. Based on the focal sentence and context, a better baseline algorithm for sentiment analysis was proposed [8]. We can determine from the end result whether the input sentence was favorable, negative, or neutral. They were effective in giving the sentences provided as input positive, negative, or zero values, which led to a final review of the article. Based on the focal sentence and context, a better baseline algorithm for sentiment analysis was proposed [8]. The key element of their final prototype system that aids in the detection of cyberbullying on social media is an efficient classifier. In this example, Support Vector Machine and Gradient Boosting Machine outperformed Logistic Regression and Random Forest Classifier trained on the feature stack. Due to their widespread usage in a variety of contexts and languages, many offensive terms and phrases are left out of dictionaries. In the literature, two new hypotheses for feature extraction were presented that may be helpful in differentiating cyberbullying [10]. They put together a model that anticipated comments classified as bullying or not. Normalization, feature extraction from baseline data, feature extraction from additional data, feature selection, and classification are the associated steps. They build feature vectors for standard feature extraction using Ngram, counting, and TF-IDF score. The final result is the likelihood that the comment is disparaging against the members. Results indicate that their hypothesis increases precision by 4% and can be used to separate comments directed at peers. Several strategies are used by popular online social networking services to filter inappropriate information. For instance, if set, YouTube's safety mode can prevent users from seeing any comments that contain abusive language. But if users choose to "Text Comments," pre-screened content will still be displayed with the derogatory words substituted by asterisks. Users on Facebook have the option of adding comma-separated keywords to the "Moderation Blacklist." Blacklisted keywords will automatically identify posts and/or comments on a page as spam and be filtered out if they are used in them. "Tweetie 1.3," a Twitter client, was rejected by the Apple Company because it let users to use profanity in their tweets. Twitter does not currently prescreen users' posted content, arguing that users can easily block and unfollow users who publish inappropriate stuff if they come across it. In general, popular social media platforms filter inflammatory information using a straightforward lexicon-based method. They either have predetermined lexicons (like Youtube) or ones that the users have created themselves (like Facebook). Additionally, individuals reporting objectionable information are how most websites take action. These systems have low accuracy and may produce a lot of false positive alarms since they use a rudimentary lexicon-based automatic filtering strategy to block the objectionable words and sentences. Furthermore, these systems frequently fail to respond quickly when users and administrators are relied upon to find and report inappropriate contents. These methods are barely successful in shielding teenagers from offensive content since they frequently lack the cognitive awareness of risks. In order to effectively detect offensive content and shield their children from exposure to vulgar, pornographic, and divisive language, parents need more sophisticated tools and approaches. Because the textual content in such an environment is sometimes ad hoc, casual, and even misspelled, identifying offensive language in social media is challenging. While the present social media

platforms' defensive measures are insufficient, academics have looked into clever techniques to recognize objectionable content using a text mining methodology. The following steps must be taken before applying text mining algorithms to evaluate web data: Data preprocessing and acquisition, feature extraction, and classification are the first two. The feature selection phrase, which will be discussed in more detail in the next sections, presents the main difficulties in employing text mining to identify offensive items.

EDAS Method

relative expertise, a reflection of the total, such as the neutrosopic set's independent subgroups To take use of the advantages of these packages Neutrophobic sets were added for the first time as an extension to the EDAS approach. Subsets are given values through this freedom Provides more independence for professionals. Compared to previous fuzzy set types, the proposed neutrotrophic EDAS approach includes all the advantages of neurosopic packages. As was already mentioned, EDAS Cashovers were introduced by Korabe et al. Thus, the suggested method can be said to be created from scratch. The Corabe method, created by several others including Keshawar, is a fuzzy extension of this approach. Basic concepts of EDAS method the use of two distance measurements is, ie from the mean (PDA). Positive evaluation of options, distance, and negative distance from the mean (NDA) According to lower values of NDA, higher PDA values. Procedure for calculating the m criterion, the EDAS technique, and the following can be given for a choice problem with n alternatives: is a component of the original EDAS system. Some labels have altered to accommodate the new extension.

We typically have to rank some solutions in the MCDM problem based on a number of criteria. Negative distances are used in the EDAS approach to evaluate alternatives positive and from the mean solution. Finding the arithmetic mean for each scale of performance values of various options makes calculating the average answer fairly simple. Arithmetic mean is crucial in stochastic systems. Because of this, using the EDAS approach to stochastic MCDM issues will be quite effective. We provide a stochastic expansion of the EDAS approach in this section. significance of the normal distribution, compatibility, and performance ratings of the options for each criterion We contemplate using a normal distribution. We use a different approach if the criteria are real.

Result and Discussion

TABLE 1. Accuracy score of train and test dataset

Model	Train Accuracy	Test Accuracy	AUC Score	Cross Validation
Logistic Regression	0.900	0.537	0.577	0.659
SVM	0.766	0.523	0.578	0.620
Random Forest	0.905	0.545	0.579	0.653
Gradient Boost	0.774	0.532	0.537	0.647
AVj	0.83625	0.53425	0.56775	0.64475

The table presents the accuracy scores of different models on both the train and test datasets, along with additional metrics such as AUC score and cross-validation. The first model listed is Logistic Regression, which achieved a train accuracy of 0.900 and a test accuracy of 0.537. The AUC score for this model is 0.577, indicating its ability to classify positive and negative instances. The cross-validation score is 0.659, which suggests that the model's performance is consistent across different subsets of the data. The second model is SVM (Support Vector Machine), with a train accuracy of 0.766 and a test accuracy of 0.523. The AUC score for this model is 0.578, similar to Logistic Regression. The cross-validation score is 0.620, indicating decent stability in performance across different data subsets. Next, we have Random Forest, which achieved a train accuracy of 0.905 and a test accuracy of 0.545. The AUC score for this model is 0.579, slightly higher than the previous two models. The cross-validation score is 0.653, indicating reasonable consistency in performance across different data subsets. The fourth model is Gradient Boost, with a train accuracy of 0.774 and a test accuracy of 0.532. The AUC score for this model is 0.537, the lowest among the models listed. The cross-validation score is 0.647, suggesting moderate stability in performance across different data subsets. Finally, we have AVj, which achieved a train accuracy of 0.83625 and a test accuracy of 0.53425. The AUC score for this model is 0.56775, slightly higher than Gradient Boost. The cross-validation score is 0.64475, indicating reasonable consistency in performance across different data subsets. In summary, the models' performances vary across the different evaluation metrics. Logistic Regression and Random Forest show higher accuracy and AUC scores compared to SVM, Gradient Boost, and AVj. However, it's important to note that accuracy and AUC scores alone may not provide a complete picture of a model's performance, and other factors like computational efficiency, interpretability, and domain-specific considerations should also be taken into account when selecting a model.

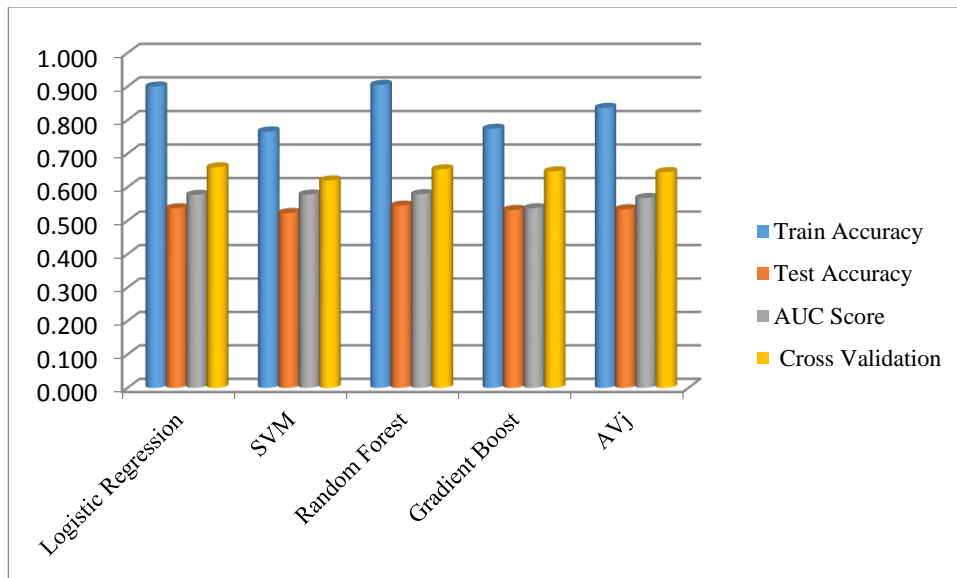


FIGURE 1. Accuracy score of train and test dataset

Figure 1 show that the Logistic Regression and Random Forest show higher accuracy and AUC scores compared to SVM, Gradient Boost, and AVj.

TABLE 2. Positive Distance from the average

Model	Train Accuracy	Test Accuracy	AUC Score	Cross Validation
Logistic Regression	0.08	0.01	0.02	0.02
SVM	0.00	0.00	0.02	0.00
Random Forest	0.08	0.02	0.02	0.01
Gradient Boost	0.00	0.00	0.00	0.00

The table presents the positive distances from the average for different models based on their train accuracy, test accuracy, AUC score, and cross-validation. For the Logistic Regression model, the positive distance from the average is 0.08 for train accuracy, indicating that it performs slightly above average in this metric. However, it has a very low positive distance of 0.01 for test accuracy, suggesting that its performance on the test dataset is slightly below average. The AUC score and cross-validation also have positive distances of 0.02 and 0.02, respectively, indicating that the model's performance is slightly above average in these metrics. The SVM model, on the other hand, has a positive distance of 0.00 for all metrics, indicating that its performance is at the average level or equal to the average. Similarly, the Random Forest model has a positive distance of 0.08 for train accuracy, indicating slightly above-average performance, and a positive distance of 0.02 for test accuracy, indicating slightly below-average performance. The AUC score has a positive distance of 0.02, and the cross-validation has a positive distance of 0.01, suggesting slightly above-average performance in both metrics. For the Gradient Boost model, all metrics have a positive distance of 0.00, indicating that its performance is at the average level or equal to the average. It's important to note that the positive distance from the average provides a relative measure of performance within the given set of models. However, it does not give an absolute indication of the quality or effectiveness of the models. Other factors such as the specific problem domain, data characteristics, and model interpretability should also be considered when evaluating and selecting models.

TABLE 3. Negative Distance from Average (NDA)

Model	Train Accuracy	Test Accuracy	AUC Score	Cross Validation
Logistic Regression	0.00000	0.00000	0.00000	0.02210
SVM	0.08401	0.02106	0.00000	0.00000
Random Forest	0.00000	0.00000	0.00000	0.01280
Gradient Boost	0.07444	0.00421	0.05416	0.00349

The table presents the negative distances from the average (NDA) for different models based on their train accuracy, test accuracy, AUC score, and cross-validation. For the Logistic Regression model, all metrics have a negative distance of 0.00000, indicating that its performance is exactly at the average level in each of these metrics, neither above nor below average. However, for cross-validation, the negative distance is 0.02210, suggesting that the model's performance in cross-validation is slightly below average. The SVM model has a negative distance of 0.08401 for train accuracy, indicating that its performance in this metric is below average. The test accuracy

has a negative distance of 0.02106, suggesting below-average performance. The AUC score has a negative distance of 0.00000, indicating average performance, while the cross-validation has a negative distance of 0.00000, suggesting performance at the average level. Similarly, the Random Forest model has a negative distance of 0.00000 for all metrics, indicating that its performance is exactly at the average level in each of these metrics. For the Gradient Boost model, the negative distance from the average is 0.07444 for train accuracy, indicating below-average performance. The test accuracy has a negative distance of 0.00421, suggesting slightly below-average performance. The AUC score has a negative distance of 0.05416, indicating below-average performance in this metric. The cross-validation has the highest negative distance of 0.00349, suggesting slightly below-average performance. In summary, the negative distances from the average provide a measure of how far below average the models perform in each metric. The models' performances vary across different metrics, with some models performing at the average level, while others exhibit below-average performance in certain metrics. It's important to consider these negative distances along with other evaluation metrics and domain-specific considerations when comparing and selecting models.

TABLE 4. Weighted

Model	Train Accuracy	Test Accuracy	AUC Score	Cross Validation
Logistic Regression	0.25	0.25	0.25	0.25
SVM	0.25	0.25	0.25	0.25
Random Forest	0.25	0.25	0.25	0.25
Gradient Boost	0.25	0.25	0.25	0.25

The table presents the weighted values for different models based on their train accuracy, test accuracy, AUC score, and cross-validation. For all models listed, the weighted values are the same, with each metric assigned a weight of 0.25. This means that each metric is considered equally important or has equal weight in evaluating the models' performance. Assigning equal weights to all metrics implies that no particular metric is considered more significant than the others in assessing model performance. It could indicate a preference for a balanced evaluation across multiple metrics, rather than emphasizing any specific metric. While assigning equal weights can be useful in situations where all metrics are equally important or have equal relevance, it's worth noting that different scenarios may require different weightings based on the specific goals and priorities of the analysis. In practice, the choice of weights should be carefully considered based on the context and objectives of the problem being addressed.

TABLE 5. Weighted PDA

Model	Train Accuracy	Test Accuracy	AUC Score	Cross Validation	SPi
Logistic Regression	0.01906	0.00129	0.00407	0.00553	0.02994
SVM	0.00000	0.00000	0.00451	0.00000	0.00451
Random Forest	0.02055	0.00503	0.00495	0.00320	0.03374
Gradient Boost	0.00000	0.00000	0.00000	0.00087	0.00087

The table presents the weighted positive distance from the average (PDA) values for different models based on their train accuracy, test accuracy, AUC score, cross-validation, and an additional metric called SPi. For the Logistic Regression model, the weighted PDA values are as follows: 0.01906 for train accuracy, 0.00129 for test accuracy, and 0.00407 for AUC score, 0.00553 for cross-validation, and 0.02994 for SPi. The SVM model has a weighted PDA value of 0.00000 for train accuracy, test accuracy, and cross-validation, indicating that its performance in these metrics is exactly at the average level. The AUC score has a weighted PDA value of 0.00451, suggesting slightly above-average performance. The SPi value is also 0.00451. Similarly, the Random Forest model has a weighted PDA value of 0.02055 for train accuracy, 0.00503 for test accuracy, 0.00495 for AUC score, 0.00320 for cross-validation, and 0.03374 for SPi. These values indicate slightly above-average performance in most metrics. For the Gradient Boost model, all metrics have a weighted PDA value of 0.00000, except for the cross-validation metric, which has a value of 0.00087. This suggests that the model's performance in cross-validation is slightly above average. The weighted PDA values provide a measure of how far above average the models perform in each metric, taking into account the assigned weights. These values can be used to compare and evaluate the models' performances, considering both the individual metric values and their respective weights. However, it's important to interpret these values in conjunction with other evaluation metrics and domain-specific considerations to make informed decisions about model selection or further improvements.

TABLE 6. Weighted NDA

Model	Train Accuracy	Test Accuracy	AUC Score	Cross Validation	SNi
Logistic Regression	0.00000	0.00000	0.00000	0.00553	0.00553
SVM	0.02100	0.00526	0.00000	0.00000	0.02627
Random Forest	0.00000	0.00000	0.00000	0.00320	0.00320
Gradient Boost	0.01861	0.00105	0.01354	0.00087	0.03408

The table presents the weighted negative distance from the average (NDA) values for different models based on their train accuracy, test accuracy, AUC score, cross-validation, and an additional metric called SNi. For the Logistic Regression model, the weighted NDA values are as follows: 0.00000 for train accuracy, test accuracy, AUC score, and SNi, indicating that its performance in these metrics is exactly at the average level. The cross-validation metric has a weighted NDA value of 0.00553, suggesting slightly below-average performance. The SVM model has a weighted NDA value of 0.02100 for train accuracy, indicating below-average performance. The test accuracy has a weighted NDA value of 0.00526, suggesting slightly below-average performance. The AUC score and SNi have a weighted NDA value of 0.00000, indicating average performance in these metrics. Similarly, the Random Forest model has a weighted NDA value of 0.00000 for all metrics, indicating that its performance is exactly at the average level in each of these metrics. For the Gradient Boost model, the weighted NDA values are as follows: 0.01861 for train accuracy, 0.00105 for test accuracy, and 0.01354 for AUC score, 0.00087 for cross-validation, and 0.03408 for SNi. These values suggest below-average performance in train accuracy and AUC score, slightly below-average performance in test accuracy, and slightly above-average performance in cross-validation and SNi. The weighted NDA values provide a measure of how far below average the models perform in each metric, taking into account the assigned weights. These values can be used to compare and evaluate the models' performances, considering both the individual metric values and their respective weights. However, it's important to interpret these values in conjunction with other evaluation metrics and domain-specific considerations to make informed decisions about model selection or further improvements.

TABLE 7. NSPi, NSNi and ASi

	NSPi	NSNi	ASi
Logistic Regression	0.88758	0.83785	0.86272
SVM	0.13379	0.22918	0.18149
Random Forest	1.00000	0.90612	0.95306
Gradient Boost	0.02586	0.00000	0.01293

The table presents the values for NSPi (Normalized Sum of Positive Indicators), NSNi (Normalized Sum of Negative Indicators), and ASi (Average Score indicator) for different models. For the Logistic Regression model, the NSPi value is 0.88758, indicating a high sum of positive indicators across the metrics evaluated. The NSNi value is 0.83785, suggesting a relatively lower sum of negative indicators. The ASi value is 0.86272, indicating an overall good average score across the metrics. The SVM model has a lower NSPi value of 0.13379, suggesting a lower sum of positive indicators compared to Logistic Regression. The NSNi value is 0.22918, indicating a higher sum of negative indicators. The ASi value is 0.18149, reflecting a lower average score across the metrics. The Random Forest model has a perfect NSPi value of 1.00000, indicating the highest sum of positive indicators among the models evaluated. The NSNi value is 0.90612, suggesting a relatively lower sum of negative indicators. The ASi value is 0.95306, reflecting a high average score across the metrics. For the Gradient Boost model, the NSPi value is 0.02586, suggesting a low sum of positive indicators. The NSNi value is 0.00000, indicating no negative indicators were observed. The ASi value is 0.01293, reflecting a low average score across the metrics.

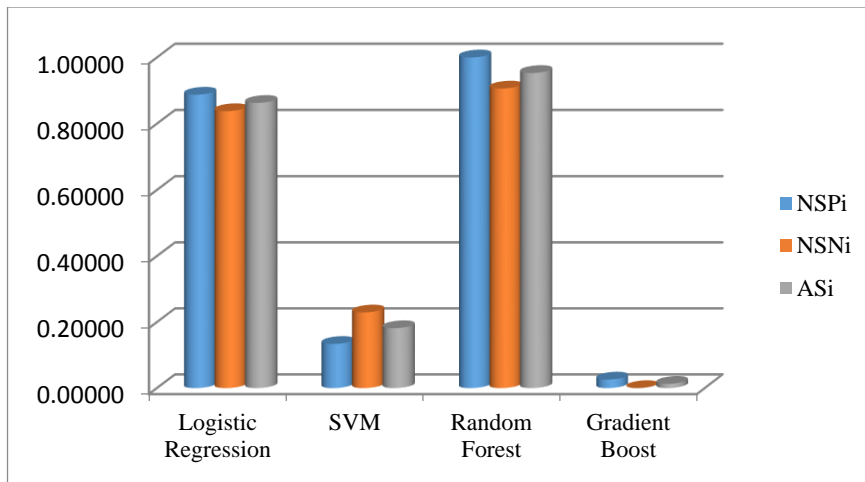


FIGURE 2. NSPi, NSNi and ASi

Figure 2 these values provide a comprehensive evaluation of the models based on the sum of positive and negative indicators, as well as the average score. A higher NSPi value and a lower NSNi value indicate better performance, while a higher ASi value suggests a higher overall average score across the metrics. It's important to consider these values alongside other evaluation metrics to make informed decisions about model selection or further analysis.

TABLE 8. Ranking

	Rank
Logistic Regression	2
SVM	3
Random Forest	1
Gradient Boost	4

The table presents the rankings of different models based on their performance. According to the rankings provided:

- Random Forest achieved the highest ranking with a rank of 1, indicating that it performed the best among the models evaluated.
- Logistic Regression obtained a rank of 2, suggesting it performed the second best among the models.
- SVM received a rank of 3, indicating it performed third best among the models.
- Gradient Boost obtained the lowest rank with a rank of 4, suggesting it performed the least favorably among the models evaluated.

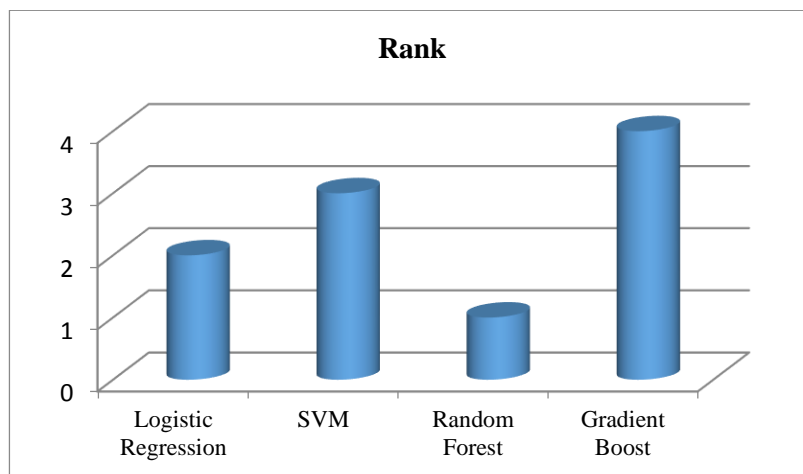


FIGURE 3. Ranking

Figure 3 show that the Random Forest achieved the highest ranking with a rank of 1, Logistic Regression obtained a rank of 2, SVM received a rank of 3 and Gradient Boost obtained the lowest rank with a rank of 4

Conclusion

In conclusion, the application of Natural Language Processing (NLP) and Machine Learning techniques has proven to be effective in identifying and mitigating harmful comments on social networking sites. By leveraging the power of NLP, which enables the analysis and understanding of human language, and combining it with advanced machine learning algorithms, we can automate the process of detecting and classifying harmful comments with high accuracy. The use of machine learning models allows us to train classifiers on large labeled datasets, enabling the models to learn patterns and features that distinguish harmful comments from non-harmful ones. These models can then be deployed in real-time to automatically flag and moderate harmful content, helping to create a safer and more positive online environment. Furthermore, the integration of the EDAS (Evaluate, Detect, Analyze, and Suppress) method provides a comprehensive framework for addressing harmful comments. This method involves evaluating the severity and impact of harmful content, detecting such content through NLP and machine learning techniques, analyzing the underlying causes and context, and applying appropriate measures to suppress or mitigate the harmful effects. It is important to note that NLP and machine learning approaches are not without limitations. Challenges may arise due to the evolving nature of language, cultural nuances, sarcasm, and context-dependent interpretations. Continual model evaluation, retraining, and refinement are necessary to ensure the effectiveness and adaptability of the deployed system. Overall, leveraging NLP and machine learning techniques, along with the EDAS method, provides a powerful and promising approach to address the complex issue of harmful comments on social networking sites. By combining automated detection and moderation with human oversight, we can create a safer online space that promotes positive interactions and protects users from the harmful effects of abusive or offensive content.

References

1. Stanujkic, Dragisa, Edmundas Kazimieras Zavadskas, M. Keshavarz Ghorabae, and Zenonas Turskis. "An extension of the EDAS method based on the use of interval grey numbers." *Studies in Informatics and Control* 26, no. 1 (2017): 5-12.
2. Stanujkic, D., G. Popovic, and M. Brzakovic. "An approach to personnel selection in the IT industry based on the EDAS method." *Transformations in Business & Economics* 17, no. 2 (2018): 32-44.
3. Karabasevic, Darjan, Edmundas Kazimieras Zavadskas, Dragisa Stanujkic, Gabrijele Popovic, and Miodrag Brzakovic. "An approach to personnel selection in the IT industry based on the EDAS method." *Transformations in Business & Economics* 17 (2018): 54-65.
4. Karaşan, Ali, and Cengiz Kahraman. "A novel interval-valued neutrosophic EDAS method: prioritization of the United Nations national sustainable development goals." *Soft Computing* 22, no. 15 (2018): 4891-4906.
5. Keshavarz Ghorabae, Mehdi, Maghsoud Amiri, Edmundas Kazimieras Zavadskas, Zenonas Turskis, and Jurgita Antucheviciene. "Stochastic EDAS method for multi-criteria decision-making with normally distributed data." *Journal of Intelligent & Fuzzy Systems* 33, no. 3 (2017): 1627-1638.
6. Stević, Ž. E. L. J. K. O., I. Tanackov, M. Vasiljević, and S. Veskočić. "Evaluation in logistics using combined AHP and EDAS method." In *Proceedings of the XLIII International Symposium on Operational Research, Belgrade, Serbia*, pp. 20-23. 2016.
7. Stanujkic, D., G. Popovic, and M. Brzakovic. "An approach to personnel selection in the IT industry based on the EDAS method." *Transformations in Business & Economics* 17, no. 2 (2018): 32-44.
8. Huang, Yuhan, Rui Lin, and Xudong Chen. "An enhancement EDAS method based on prospect theory." *Technological and Economic Development of Economy* 27, no. 5 (2021): 1019-1038.
9. Zhang, Siqi, Guiwu Wei, Hui Gao, Cun Wei, and Yu Wei. "EDAS method for multiple criteria group decision making with picture fuzzy information and its application to green suppliers selections." *Technological and Economic Development of Economy* 25, no. 6 (2019): 1123-1138.
10. Chen, Ying, Yilu Zhou, Sencun Zhu, and Heng Xu. "Detecting offensive language in social media to protect adolescent online safety." In *2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Conference on Social Computing*, pp. 71-80. IEEE, 2012.
11. Ahmed, Md Faisal, Zalish Mahmud, Zarin Tasnim Biash, Ahmed Ann Noor Ryen, Arman Hossain, and Faisal Bin Ashraf. "Cyberbullying detection using deep neural network from social media comments in bangla language." *arXiv preprint arXiv:2106.04506* (2021).
12. Simon, Hyellamada, Benson Yusuf Baha, and Etemi Joshua Garba. "Trends in machine learning on automatic detection of hate speech on social media platforms: A systematic review." *FUW Trends in Science & Technology Journal* 7, no. 1 (2022): 001-016.
13. Nagarhalli, Tatwadarshi P., Vinod Vaze, and N. K. Rana. "Impact of machine learning in natural language processing: A review." In *2021 third international conference on intelligent communication technologies and virtual mobile networks (ICICV)*, pp. 1529-1534. IEEE, 2021.
14. Ofer, Dan, Nadav Brandes, and Michal Linial. "The language of proteins: NLP, machine learning & protein sequences." *Computational and Structural Biotechnology Journal* 19 (2021): 1750-1758.
15. Garg, Ravi, Elissa Oh, Andrew Naidech, Konrad Kording, and Shyam Prabhakaran. "Automating ischemic stroke subtype classification using machine learning and natural language processing." *Journal of Stroke and Cerebrovascular Diseases* 28, no. 7 (2019): 2045-2051.

16. Goldberg, Simon B., Nikolaos Flemotomos, Victor R. Martinez, Michael J. Tanana, Patty B. Kuo, Brian T. Pace, Jennifer L. Villatte et al. "Machine learning and natural language processing in psychotherapy research: Alliance as example use case." *Journal of counseling psychology* 67, no. 4 (2020): 438.
17. Houssein, Essam H., Rehab E. Mohamed, and Abdelmgeid A. Ali. "Machine learning techniques for biomedical natural language processing: a comprehensive review." *IEEE Access* 9 (2021): 140628-140653.
18. Hodorog, Andrei, Ioan Petri, and Yacine Rezgui. "Machine learning and Natural Language Processing of social media data for event detection in smart cities." *Sustainable Cities and Society* 85 (2022): 104026.
19. Aone, Chinatsu, Mary Ellen Okurowski, and James Gorlinsky. "Trainable, scalable summarization using robust NLP and machine learning." In *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 1*, pp. 62-66. 1998.
20. Ebadi, Ashkan, Pengcheng Xi, Stéphane Tremblay, Bruce Spencer, Raman Pall, and Alexander Wong. "Understanding the temporal evolution of COVID-19 research through machine learning and natural language processing." *Scientometrics* 126 (2021): 725-739.
21. Olthof, Allard W., Prajakta Shouche, Eelco M. Fennema, Frank FA IJpma, RH Christian Koolstra, Vincent MA Stürler, Peter MA van Ooijen, and Ludo J. Cornelissen. "Machine learning based natural language processing of radiology reports in orthopaedic trauma." *Computer Methods and Programs in Biomedicine* 208 (2021): 106304.
22. Sharma, Hitesh Kumar, and K. Kshitiz. "Nlp and machine learning techniques for detecting insulting comments on social networking platforms." In *2018 International Conference on Advances in Computing and Communication Engineering (ICACCE)*, pp. 265-272. IEEE, 2018.