TOXIC COMMENT CLASSIFICATION USING NLP AND LOGISTIC REGRESSION

ABSTRACT

People use social media to express their ideas and emotions as much as ever before. It has had a huge impact on how people communicate with one another. When it first came out it seemed like it was going to make communication better, however over time a new problem developed; the toxic comments. There are many types of toxic comments. Most of them appear to be minor, some appear to be very harsh and direct, and all of these types create environments where many people don't want to participate.

Toxic comments can take many forms, including but not limited to: hate speech, cyberbullying and profanity. As a result of toxic comments, most people will not express themselves freely. Therefore, the objective of this project is to develop an automated detection system that identifies toxic comments (or at least those that have potential) using natural language processing and machine learning technologies.

Once the text data is cleaned (primarily of characters we don't need), features such as TF-IDF, n-grams, and a few sentiment-related characteristics are gathered from it. Since the dataset is not perfectly balanced, the SMOTE algorithm is utilized to adjust it. Once the dataset is adjusted, a logistic regression model is then built to determine if a particular comment is toxic or not. While the model is relatively simple compared to other models, it has proven effective enough for this application.

The goal is simple. If we could detect potentially toxic comments sooner rather than later, online communities would be safer and more welcoming places. People could express themselves without fear of being attacked without a valid reason.

I.INTRODUCTION

The fast development of social media sites has changed how individuals interact with each other, as everyone gets an opportunity to express their views and opinions globally. But with this freedom, there have also been some negative effects of the growth of toxic comments, such as hate speech, cyber bullying, insults and offensive language. This may lead to emotional strain and less user interaction and a hostile atmosphere. Due to this, the issue of ensuring that online spaces are safer and more inclusive has become a major problem. The conventional moderation system is slow and can be inconsistent particularly with massive content that is uploaded daily. Moderators can overlook abusive remarks or respond slower thereby letting toxic content remain on the page as long as it ought. As a result, interest in automatically detecting toxic comments on the basis of Natural Language Processing and machine learning has increased. These methods are a potentially scalable and trustworthy method of processing huge volumes of text and assist in detecting harmful behaviour faster. Labelled comments can be trained on the supervised learning models to learn various patterns of toxicity. These models can distinguish toxic and non toxic comments very well by considering such features as offensive words, sentiment, context, and structure of the sentence. The logistic regression approach to a binary classification is one of the standard machine learning techniques applied in this project to automatically identify toxic comments. Logistic regression is effective with most text classification tasks, as it is easy, effective and easy to interpret. The model is then trained on labeled comment data to be able to learn the distinction between harmful and safe language. The project utilizes the NLP steps as well, such as cleaning of the text and tokenization, TF IDF based feature engineering and text vectorization of the text to enhance the performance of the model. Measures of the accuracy of the system will be done in terms of precision, recall and F1 score in order to gain an insight on the effectiveness of the system in detecting toxic content. The project will facilitate the continued struggle towards creating safer and more friendly online communities by coming up with this automated system. Since the digital communication keeps expanding, addressing toxic comments is more significant than ever. This article is an attempt at demonstrating how machine learning and NLP techniques may be applied to curb online harassment, hate speech and other undesirable behaviour, making the internet a healthier and more pleasant experience to users.

II . PROBLEM STATEMENT

The problem with toxic comments is that they are hard to handle, they come in large numbers and they take various forms. Their different tone, structure and purpose result in manual moderation that is slow and unequally applied, particularly where new text is constantly uploaded. It requires a stable classification system that can differentiate between harmful comments and a normal conversation with defined limits of decision and predictable performance. The logistic regression offers an organised method of modelling this split by connecting textual features with the likelihood of toxicity. The model can understand linguistic patterns and provide definite toxicity probabilities with the data prepared and the features clearly defined, so that the process of decision-making will be consistent and predictable. The central issue covered in this project is that it is necessary to have a scalable and interpretable system that would be able to classify toxic comments. The aim is to come up with a logistic regression-based model that makes use of cleaned and structured text features, is able to deal with class imbalance, and provides stable predictions that can be used in real-time or near-real-time moderation..

III. LITERATURE SURVEY

The emergence of user-generated content on social networks has led to the rapid increase of toxic comments, which makes it a popular issue to research. Recent studies have investigated a variety of deep-learning systems and transformer-based models, and a significant number of such systems are still troubled by issues of class imbalance, small linguistic clues, and data reliance. This section examines three pertinent studies that offer useful baselines and drawbacks of the existing toxic comment detection systems. The limitations are also a reason to consider simpler, interpretable and computationally efficient methods like Logistic Regression that is employed in the current work.

A. BiLSTM-CRF Detection of Toxic Spans (SemEval-2021)

In SemEval-2021 Toxic Spans Detection task, the system was created using the BiLSTM-CRF with the use of GloVe Twitter embeddings to determine the particular parts of the comment that are toxic. As opposed to whole-comment classifiers, this was more difficult as it required localization of spans at a fine-level. The authors use SemEval-2021 Toxic Spans as their training data, where thousands of annotated Reddit and Wikipedia comments have toxic character spans manually labeled. Their model scored approximately 61 percent on F1-score, and the inclusion of ToxicBERT did not significantly improve the results, indicating that it is challenging to locate multi-word toxic spans. The paper identifies one of the weaknesses that are shared by most studies: recurrent models are usually capable of capturing isolated toxic words but not longer contextual patterns. The difficulties render the work a helpful deep-learning baseline and demonstrate that even relatively complicated models may fail in real-world toxicity tasks.

B. CNN-BiLSTM-basedMulti-Label Bengali Toxic Comment Classification (2023).

Belal et al. suggested a two-stage Bengali toxic comment detection architecture, which involves LSTM with BERT embedding to perform binary classification and CNN-BiLSTM network with attention to perform multi-label classification. Their experiments were based on a modified Bengali toxic comment dataset of 16,073 annotated samples divided into a number of toxicity types. Although the binary LSTM classifier was successful (89.42% accuracy), the multi-label CNN-BiLSTM had a lower accuracy of approximately 78.92% that is substantially less than transformer-based fine-tuned systems. Big problems of dataset imbalance and small access to high-quality Bengali toxicity corpora were also mentioned by the authors. These limitations tend to result in less generalization and inaccurate classification of the minority toxicity classes. Although they can use superior architectures, the research demonstrates that classical deep-learning models can be underperforming without large and balanced datasets, thus they can be used as baselines to compare simpler models like Logistic Regression.

C. SS-BERT: Bias-Aware Toxicity Detection (2021).

Zhao et al. developed SS-BERT, a form of BERT that minimizes the identity-term bias: a widespread problem where identity words (Black, Asian, Muslim) are falsely considered toxic. The authors evaluated the performance of the model on a number of datasets such as Twitter hate-speech corpora, Wiki Toxicity dataset, and other identity-sensitive collections. However, SS-BERT did not show significant improvements over the baseline BERT despite being transformer-based. Baseline F1 scores were 0.58 on some datasets, primarily because of label noise, vague contexts and overreliance on identity terms. The experiment shows that even the sophisticated models have serious drawbacks in case datasets are biased. This is why SS-BERT is a significant contemporary benchmark, which helps to understand that the performance of models can decrease based on the characteristics of datasets, which also justifies the usefulness of simpler and interpretable models such as Logistic Regression to perform controlled, binary toxicity classification problems.

To conclude, the studied articles reveal that models like BiLSTM-CRF, CNN-BiLSTM, and SS-BERT still have such issues as the imbalance in the dataset, the presence of subtle contextual toxicity, and identity-term bias, which result in the unequal performance of the models across various datasets. These results suggest that a more complex architecture does not necessarily lead to a more stable and interpretable architecture. Conversely, the current study incorporates a Logistic Regression model backed by organized preprocessing and feature-based methods, which provides a viable and clear method of detecting toxic comments. In general, the literature indicates that properly designed classical NLP models such as Logistic Regression can still be useful and efficient despite the further development of more complicated architectures.

IV.SYSTEM ARCHITECTRE

The system will be created to categorize the comments as Toxic or Non-Toxic with the help of NLP techniques and the help of Logistic Regression. The processing begins with a labelled corpus and proceeds to preprocessing, vectorization and classification phases.

Labelled Corpus

The dataset will already have comments that are identified as Toxic or Non-Toxic. It is primarily stored in CSV format and is used as the primary source of training. These labels cause the model to acquire typical patterns that tend to occur in abusive language.

Data Preprocessing

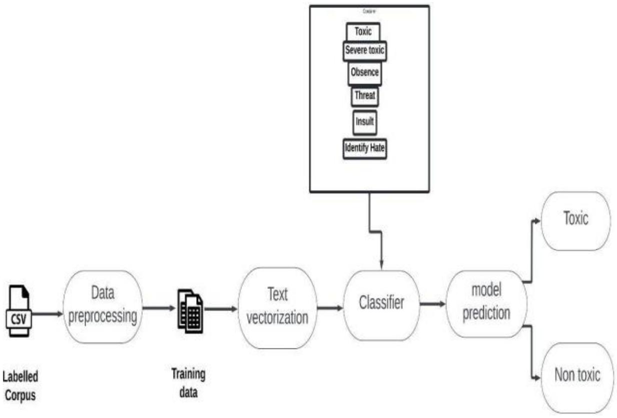
Raw text usually contains noise which influences the model performance. Preprocessing entails the elimination of symbols, stop words and other elements that do not contribute much meaning. Lowercasing is applied to maintain consistency, and lemmatization is applied to reduce words to their base form. Following this, the text is made cleaner and appropriate to the following steps.

Text Vectorization

The model requires that the text is cleaned and converted into numbers. The reason behind the use of TF-IDF here is that it assists in identifying words that are significant by comparing the frequency of the word in a comment to that of the entire dataset. This enables this model to target terms that typically exhibit toxic behaviour**.**

Classifier

Classification is done with the help of Logistic Regression. It is trained on TF-IDF characteristics and predicts a comment as being toxic. The model is not complicated and is also easy to comprehend. It provides specific decision limits and is suitable in tasks that are binary as in this case**.**



Model Prediction

New comments are processed and vectorized in the same way. The trained model makes a prediction on whether the comment is toxic or not toxic. This enables quick detection and is practically real time based on deployment**.**

Output

The output of the system is binary:1 for Toxic and 0 for non toxic comments.

V. METHODOLOGY

Problem Framing

The task is to determine whether a comment is toxic or non-toxic. The comments that are toxic normally include insults, hate or aggressive tone and the non-toxic ones remain neutral. The difficulty arises in the mixed writing styles, sarcasm and the lack of balance between the two classes. All this has to be dealt with by the model with stability in predictions.

Text Preprocessing

The text is purged of noise, such as symbols, redundant spaces and words. All this is converted to lowercase. Lemmatization breaks words down into simple base form and this simplifies the process of detecting patterns. The tokenization is used to divide the text into smaller portions which the model can read more effectively.

Feature Extraction

TF-IDF converts the filtered text into useful numeric data. N-grams capture short word patterns which in many cases contain contextual hints. Sentiment scores introduce an emotional color, which at times brings out latent toxicity. These features combined with each other provide the model with a more comprehensive perspective of the comment.

Handling Imbalance

The number of toxic comments is smaller and, therefore, the dataset remains unbalanced by default. Synthetic toxic samples are formed with the help of SMOTE. This will assist the model to have more examples and less bias in predicting everything as non-toxic. It causes classification that is more stable.

Model Training

Logistic Regression is then trained using the extracted features to learn the likelihood of toxicity. It identifies patterns in words, emotion and little phrases. The model is not complicated, and it is also effective in binary decisions. It also gives interpretable results that are simple to comprehend.

Model Evaluation

The model is measured using accuracy, precision, recall and F1-score. Precision measures the number of correct prediction toxic comments. Test Recall Checks The number of real toxic comments the model actually picked. F1 maintains a balance in the presence of both matter.

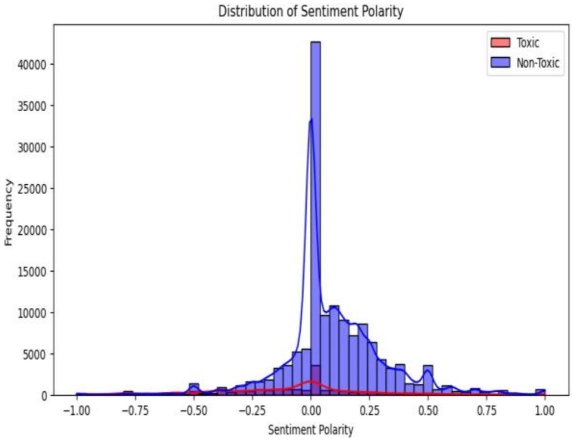
Real-Time Classification

New comments are processed through cleaning and feature extraction. The model subsequently gives a label of toxicity almost immediately. This assists platforms to filter harmful material quicker. It also minimizes manual moderation.

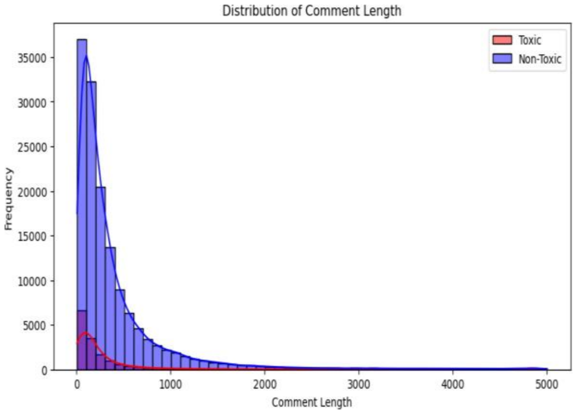
Continuous Improvement

The model is re-trained using newer comments because language continuously evolves. This assists it to detect new patterns of toxicity that were not present previously. Cross-validation makes the model to remain stable. The moderation system is effective in the long-term since it is updated regularly**.**

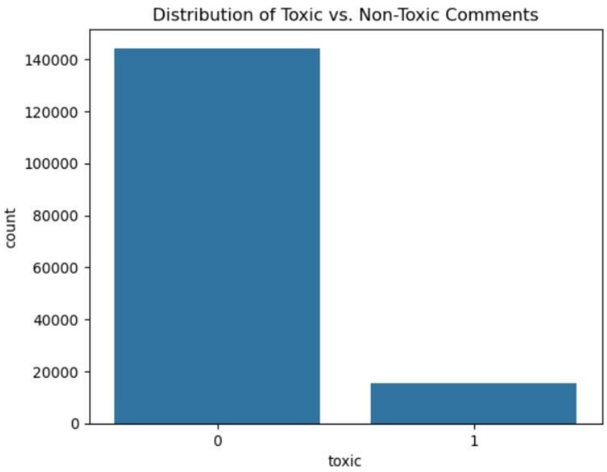
Distribution of Sentiment Polarity:



Distribution of Comment length



Distribution of Toxic vs Non-Toxic Comment :



VIII.RESULTS AND DISCUSSION

1.1. Evaluation Metrics:

The evaluation of the performance of the **Logistic Regression model** for toxic comment classification is performed using the following metrics:

- **Precision**: The proportion of true positive predictions (toxic comments predicted as toxic) out of all predicted toxic comments.

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

- **Recall**: The proportion of true positive predictions out of all actual toxic comments (true positives

+ false negatives).

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

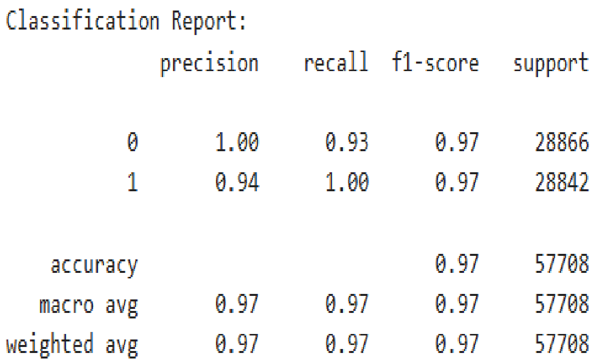
- **F1-Score**: The harmonic mean of precision and recall, offering a balance between them.

$$\text{F1-Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

Where:

- TP: True Positives (correctly predicted toxic comments).
- TN: True Negatives (correctly predicted non-toxic comments).
- FP: False Positives (non-toxic comments incorrectly predicted as toxic).
- FN: False Negatives (toxic comments incorrectly predicted as non-toxic).

Evaluation Metrics

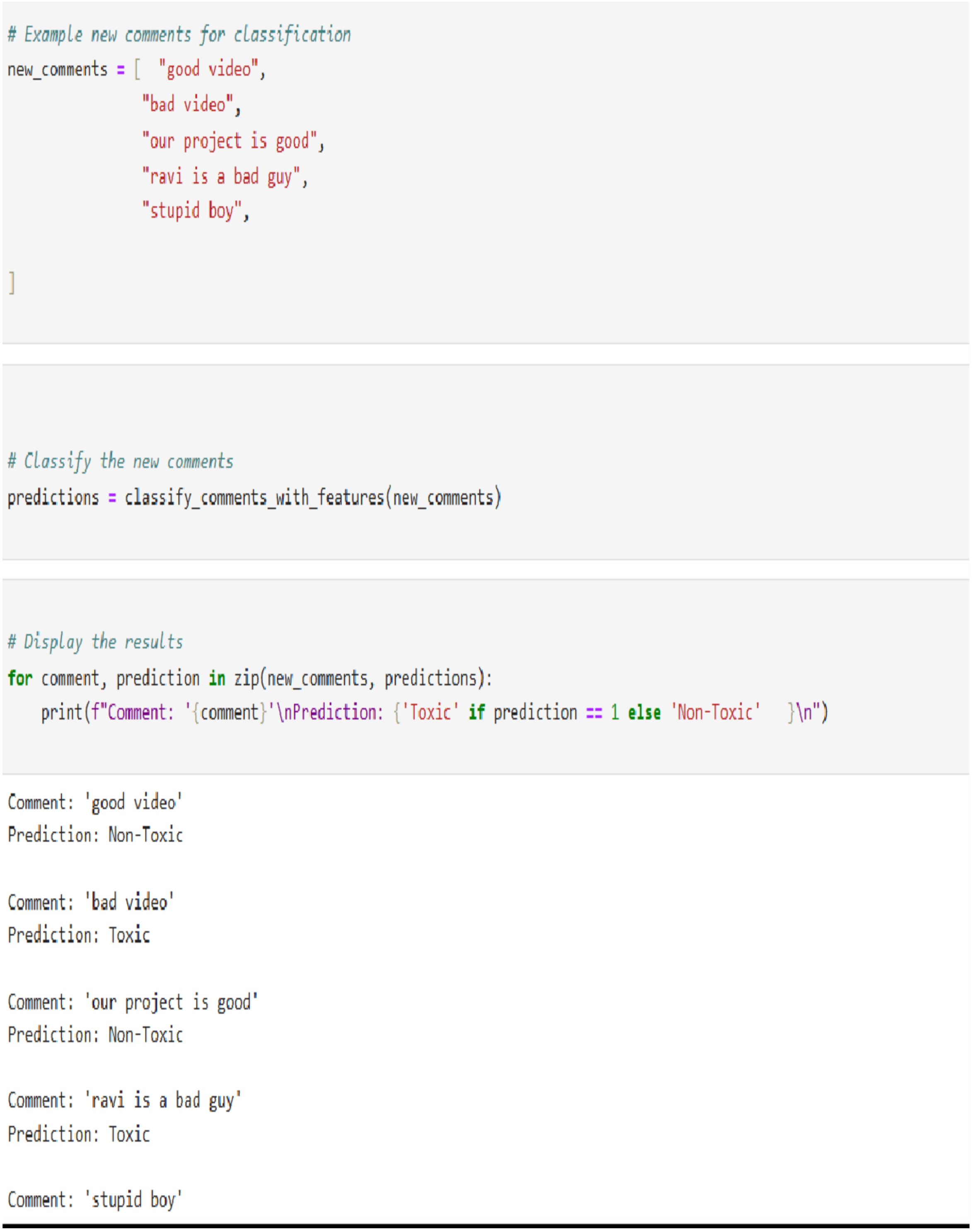


```
Classification Report:
              precision    recall  f1-score   support

           0       1.00      0.93      0.97     28866
           1       0.94      1.00      0.97     28842

    accuracy                           0.97     57708
   macro avg       0.97      0.97      0.97     57708
weighted avg       0.97      0.97      0.97     57708
```

Result:



```
# Example new comments for classification
new_comments = [  "good video",
                "bad video",
                "our project is good",
                "ravi is a bad guy",
                "stupid boy",


]
```

```
# Classify the new comments
predictions = classify_comments_with_features(new_comments)
```

```
# Display the results
for comment, prediction in zip(new_comments, predictions):
    print(f"Comment: '{comment}'\nPrediction: {'Toxic' if prediction == 1 else 'Non-Toxic'   }\n")
```

```
Comment: 'good video'
Prediction: Non-Toxic

Comment: 'bad video'
Prediction: Toxic

Comment: 'our project is good'
Prediction: Non-Toxic

Comment: 'ravi is a bad guy'
Prediction: Toxic

Comment: 'stupid boy'
```

IX.CONCLUSION

The development of a machine learning-based system to detect toxic content proved to be a viable solution to enable faster content moderation in digital platforms. Jigsaw Toxic Comment dataset was cleansed with such steps as lowercasing, punctuation elimination, stop words removal and lemmatization. TF-IDF and n-grams were used to identify helpful text patterns, and Logistic Regression provided a straightforward and quick solution to binary classification. The model was stable and operated well in real-time predictions based on the accuracy, precision, recall, and F1-score.

In general, it can be concluded that machine learning can indeed be useful in minimizing harmful information on the Internet in the case of its application. As the system continues to get improved (including more in-depth language models), more resistant to imbalance and more specific categories of toxicity), it can become even more reliable and appropriate towards larger platforms.

X.REFERENCES

1. Jigsaw. (2020). *Toxic Comment Classification Challenge*. Kaggle Dataset. https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge
2. Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press. ISBN: 978-0521865715
3. Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics, 5*, 135–146. https://doi.org/10.1162/tacl_a_00051
4. Jurafsky, D., & Martin, J. H. (2019). *Speech and Language Processing*. Pearson. ISBN: 978-0131873216
5. Davidson, T., Warmsley, D., Macy, M., & Weber, I. (2017). Automated Hate Speech Detection and the Problem of Offensive Language. *Proceedings of the 11th International AAAI Conference on Web and Social Media (ICWSM).*
6. Hosmer, D. W., Lemeshow, S., & Sturdivant, R. X. (2013). *Applied Logistic Regression*. Wiley. ISBN: 978-0470582473
7. Powers, D. M. W. (2011). Evaluation: From Precision, Recall, and F-Measure to ROC, Informedness, Markedness, and Correlation. *Journal of Machine Learning Technologies, 2(1)*, 37–63.
8. He, H., & Garcia, E. A. (2009). Learning from Imbalanced Data. *IEEE Transactions on Knowledge and Data Engineering, 21(9)*, 1263–1284.
9. Scikit-learn Documentation. (n.d.). https://scikit-learn.org/stable/
10. Pandas Documentation. (n.d.). https://pandas.pydata.org/docs/
11. NumPy Documentation. (n.d.). https://numpy.org/doc/
12. McKinney, W. (2017). *Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython*. O'Reilly Media. ISBN: 978-1491957660
13. Schmidt, A., & Wiegand, M. (2017). A Survey on Hate Speech Detection using Natural Language Processing. *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media.* https://doi.org/10.18653/v1/W17-1101
14. Mittelstadt, B. D., Allo, P., Taddeo, M., Wachter, S., & Floridi, L. (2016). The Ethics of Algorithms: Mapping the Debate. *Big Data & Society, 3(2),* 1–21. https://doi.org/10.1177/2053951716679679
15. Liu, B. (2015). *Sentiment Analysis: Mining Opinions, Sentiments, and Emotions*. Cambridge University Press. ISBN: 978-1107017894
16. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of