ISSN 2278-3091
Volume 13, No.5, September – October 2024
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse011352024.pdf
https://doi.org/10.30534/ijatcse/2024/011352024

A Comparative Study of Transformer-based Models for Hate-Speech Detection in English-Kiswahili Code-Switched Social Media Text

Fredrick Ng'ang'a Njung'e[1], Aaron Mogeni Oirere[2], Rachael Njeri Ndung'u[3]
[1]Department of Information Technology, Murang'a University of Technology, Kenya, fredrickngang a@mut.ac.ke
[2]Department of Computer Science, Murang'a University of Technology, Kenya, amogeni@mut.ac.ke
[3]Department of Information Technology, Murang'a University of Technology, Kenya, rndungu@mut.ac.ke

Received Date : August 2, 2024 Accepted Date: September 20, 2024 Published Date: October 06, 2024

ABSTRACT

The transformer architecture, first introduced in 2017 by researchers at Google, has revolutionized natural language processing in various tasks, including text classification. This architecture formed the basis of future models such as those used in hate speech detection in code-switched text. In this research, we conduct a comparative study of transformer-based models for hate speech detection in English-Kiswahili code-switched text. First, the models were compared as feature extractors using a traditional classifier and then as end-to-end classifiers. The three multilingual transformer-based models compared include mBERT, mDistilBERT and XLM-RoBERTa, using SVM as the traditional classifier for the extracted features. The HateSpeech_Kenya dataset, sourced from Kaggle, was utilized in this study. As a feature extractor, mBERT's hidden states trained the highest-performing SVM with an accuracy of 0.5461 and a macro f1 score of 0.40. Among the three models evaluated, XLM-RoBERTa achieved the highest accuracy of 0.6069 and a macro f1 score of 0.49 on a balanced dataset. In contrast, mBERT achieved the highest accuracy of 0.7820 and a macro f1 score of 0.53 on an imbalanced dataset. The comparative study establishes that using transformer-based models as end-to-end classifiers generally performs better than using them as feature extractors with traditional classifiers. This is because directly training the models allows them to learn more task-specific features. Furthermore, the varying performance across balanced and imbalanced datasets highlights the need for careful model selection based on the dataset characteristics and specific task requirements.

Key words: Code-Switching, English-Kiswahili, Hate Speech, Multilingual Language Understanding, Text Classification, Transformers.

1. INTRODUCTION

Social media platforms have given individuals unique opportunities to express themselves, connect with others, and engage in online discourse. While this has ushered in an era of free expression, it has also presented a significant challenge of regulating the vast expanse of the internet, particularly considering the anonymity it affords users [1], [2]. This challenge is evident in the increased incidences of hate speech, cyberbullying, and abusive content on social media platforms [1], [2]. Regulatory commissions rely on public outrage and manual observation of social media posts to identify hate speech, which has proven challenging and inefficient [1], [2]. These approaches are particularly unfeasible given the large amount of data available on these platforms.

In the past, research efforts have mainly targeted monolingual contexts focused primarily on English [3]. This focus has sidelined multilingual and multicultural societies, such as those prevalent in many African countries like Kenya [1], [2], [3]. In such societies, social media users often converse by seamlessly blending multiple languages, including English and resource-scarce languages like Kiswahili. This phenomenon is known as Code-Switching [4].

Transformers are deep neural networks whose architecture was introduced in Vaswani et al.'s "Attention is All You Need" paper [5]. The main novelty of the Transformer architecture is the Self-Attention Mechanism, which replaces the recurrence of RNNs and the convolutions of CNNs [5]. This distinctive feature enables transformer-based models to capture complex relationships and dependencies inherent in the input data. This ability to weigh and prioritize different input data elements enhances the model's effectiveness at capturing patterns and semantic structures [6]. The architecture comprises Encoder-Decoder Stacks, Attention Mechanism, Positional Encoding, Feed-Forward Layer, Residual Connection and Layer Normalization [5]. This

architecture has influenced the development of subsequent models used in hate speech detection in code-switched text, such as the models included in this study [7], [8], [9].

Thus, the primary goal of this research was to conduct a comparative study of three transformer-based models as feature extractors and end-to-end classifiers for hate-speech detection in English-Kiswahili code-switched social media text. The rest of the paper is organized as follows: related works are discussed in section 2, the research methodology is described in section 3, the results of this study are outlined in section 4, the results are discussed in section 5, and finally, a conclusion is given in section 6.

2. RELATED WORKS

Ombui *et al.'s* study on hate speech detection in English-Kiswahili code-switched text compared nine classification models including Naïve Bayes, Support Vector Machine, Linear logistic regression, K-Nearest Neighbor, Random Forest bagging technique, Decision Tree, Hierarchical Attention Network, Convolutional Neural Network, and Extreme Gradient Boosting algorithm against features like count vectors, TF-IDF, N-Grams and word embeddings [1]. The findings of this study showed that classical machine learning models have traditionally relied on handcrafted features like TF-IDF and N-Grams, which often struggle to capture the intricacies of codeswitched text [1], [2], [3], [10]. On the other hand, contemporary deep learning approaches utilize pre-trained word embeddings from static representation models such as word2vec and GloVe [1], [3]. These static representation models do not offer contextual embeddings, which are critical when dealing with code-switched data.

Neeraj *et al.* conducted a comparative analysis of transformer-based models for identifying hate speech in Hinglish code-mixed conversational datasets [11]. The study experimented with Google MuRIL, XLM-RoBERTa, and Indic-BERT models alongside an ensemble of these models [11]. The dataset was sourced from Hate Speech and Offensive Content (HASOC) shared tasks, focusing on hate speech and offensive language in the code-mixed text [11]. The experimental setup included a batch size of 64, a maximum sequence length of 512 tokens, and early stopping on the validation loss for 10 epochs [11]. The initial learning rate was set to 2e-5 using the Adam optimizer. The results demonstrated that Google MuRIL outperformed other models with an accuracy of 0.60 and an F1 macro score of 0.56 [11].

Supriya *et al.* evaluated pre-trained transformer-based models for hate speech and offensive content identification across English, Indo-Aryan, and code-mixed (English-Hindi) languages [12]. Utilizing the Hugging Face transformers library and PyTorch for implementation, the study set a maximum sequence length of 128 tokens, with an AdamW optimizer at a learning rate of 2e-5, a dropout rate of 0.1, and a batch size of 16 [12]. The models were trained on Google Colab using GPU processing. The multilingual BERT model, fine-tuned on preprocessed code-mixed data, achieved a macro F1 score of 0.6795 [12]. The findings in this study demonstrated the effectiveness of transformer-based models in handling code-mixed text and provided valuable insights into optimizing models for such tasks.

Aryan *et al.* compared multiple pre-trained BERT models for code-mixed Hindi-English data, focusing on several downstream tasks, including sentiment analysis, emotion recognition, and hate speech identification [13]. The study compared multilingual and code-mixed models, including HingBERT, HingRoBERTa, and HingRoBERTa-Mixed, while non-code-mixed models included ALBERT, BERT, and RoBERTa [13]. The experiment was set up by tuning hyperparameters using WandB, with learning rates ranging from 1e-6 to 1e-4, epochs from 1 to 5, and batch sizes from 32 to 64 [13]. The results indicated that HingBERT-based models outperformed vanilla BERT models on code-mixed text, achieving state-of-the-art results on respective datasets [13]. The study's findings highlighted the superior performance of models specifically pre-trained on actual code-mixed text.

In this study, we extend these works by comparing three transformer-based models, mBERT, XLM-RoBERTa, and mDistilBERT, in the context of English-Kiswahili code-switched hate speech detection. The objective was to determine the effectiveness of these models as feature extractors and end-to-end classifiers in handling code-switched text and to identify the model that achieves the highest accuracy and macro F1 score.

3. METHODOLOGY

This section outlines the transformer-based models used for this comparative study, the experimental configuration used, a description of the dataset, the training procedure, and the evaluation metrics used to measure performance.

3.1 Transformer Models

3.1.1 Multilingual BERT

Multilingual BERT (mBERT) is a transformer-based model designed by Google. It extends the BERT model by handling multiple languages within a single model [14]. It has a multi-layer bidirectional transformer encoder architecture, which enables good contextual understanding across different languages. mBERT has 12 encoder layers and 179 million parameters, which makes it effective for multilingual tasks [14].

3.1.2 Cross-Lingual Multilingual RoBERTa

Facebook AI introduced cross-lingual multilingual RoBERTa (XLM-RoBERTa) as an improvement of the RoBERTa model, which removed some shortcomings of the BERT

architecture [15]. Due to its cross-lingual optimization, the model can effectively handle a diverse set of languages. XLM-RoBERTa comprises 12 encoder layers and has 279 million parameters, a strong foundation for multilingual language understanding [15].

3.1.3 Multilingual BERT

DistilBERT is a smaller variant of BERT that is computationally friendly, though it contains all the relevant characteristics of its large counterpart [16]. Multilingual DistilBERT (mDistilBERT) extends the DistilBERT model to process multilingual data, making it a lightweight yet powerful tool for handling diverse linguistic data. mDistilBERT has moderate complexity and is adapted to achieve high performance and high efficiency with 6 encoder layers and 135 million parameters [16]. Table 1 below summarizes the architectures of the three transformer-based models used in this study in terms of the number of parameters, number of layers and the type of architecture.

Table 1: Transformer-based Models used in the Study

Transformer	Parameters	Layers	Type of Architecture
mBERT	179M	12	Encoder
XLM-RoBERTa	279M	12	Encoder
mDistilBERT	135M	6	Encoder

3.2 Experimental Setup

This subsection describes the experimental setup of this study in terms of the materials, dataset, preprocessing steps, hyperparameters, training steps and evaluation.

3.2.1 Experimental Materials

All the experiments were conducted on an HP EliteBook with an Intel(R) Core(TM) i7-4600U CPU @ 2.10 GHz, 8Gb RAM, and an Intel(R) HD Graphics Processing Unit. Also, Google Colab Pro was used to utilize hardware acceleration with the help of GPU, the NVIDIA L4 GPU. The experiments were implemented using PyTorch framework version 2.4, the Hugging Face Transformers Library version 4.44.2, and CUDA version 12.6.1.

3.2.2 Dataset Description

The HateSpeech_Kenya dataset, available on Kaggle, was developed by researchers at Africa Nazarene University, Kenya. It was first presented in their paper "Building and Annotating a Codeswitched Hate Speech Corpora." [17] This dataset includes 48,057 tweets and was manually classified into hate speech, offensive, or neither. At least three novice annotators labelled each tweet, with the majority vote determining the final label. Label 0 was assigned to the Neither class, 1 to the Offensive class, and 2 to the Hate Speech class [17]. This data was collected during the Kenya presidential elections in August 2017 and the repeat election in October 2017 [17]. A custom crawler was employed to overcome the two-week data collection limitation by the Twitter API, collecting tweets during the three months leading up to the general elections and two weeks after the repeat election results were announced. This time was historically marked by an increase in the levels of online hate speech [17]. The dataset was cleaned by removing all URLs and substituting all mentions with USERNAME plus a number. Table 2 below displays the distribution of instances in the HateSpeech_Kenya dataset across the three classes i.e., Hate Speech, Offensive Language and Neither.

Table 2: Distribution of the HateSpeech_Kenya Dataset

Label	No. of Instances
Hate Speech	3181
Offensive	8543
Neither	36333
Total	**48057**

3.2.3 Data Preprocessing

The process of preparing the data for model training involved several steps. First, the dataset was split into three, i.e., the training, validation, and test sets in a ratio of 70:20:10, respectively. This division allowed the model's performance to be evaluated at different training and testing stages. Under-sampling was also applied to solve the problem of class imbalance. This technique aimed at reducing the number of samples in the majority class to correspond to the number of samples in the minority class, thus ensuring a balanced representation of each class. This was important to prevent the models from being biased towards the majority class. The subsequent step was tokenization, where the input sequences were converted into PyTorch tensors using the corresponding tokenizer for each model provided by the Hugging Face library. This transformation enabled the transformer models to process the raw text data. Each tweet was tokenized into a sequence of tokens, which were then fed into the models for further processing. Finally, feature extraction was done by passing the tokenized text through the transformer models to capture the hidden states from the models' intermediate layers. These hidden states were used as features to train the Support Vector Machine (SVM) to predict the labels of the tweets.

3.2.4 Hyperparameters

A uniform set of hyperparameters was used during the training process to make the results of all the models presented in the study comparable [18]. The Support Vector Classifier (SVC) was set up with a radial basis function (RBF) kernel, and the parameters C were set to 0.1 and gamma to 0.1. Furthermore, the probability argument was also set to true to obtain probability estimates. For the transformer-based models, a low learning rate of 1e-5 was used to maintain the learned weights and ensure stable training. The AdamW optimizer was chosen based on its ability to handle the complexities of transformer-based models. The models were trained for 5 epochs, with a batch size of 8, for training and evaluation to minimize memory usage rates while ensuring training is still efficient. In the training process, 500 warmup

steps were used to exponentially increase the learning rate at the beginning of training, which helped stabilize model training. A weight decay of 0.1 was used to perform regularization and prevent the model from overfitting. The evaluation strategy was to save the best model and load it at the end of the training based on performance metrics evaluated at each epoch.

3.2.5 Training

The baseline pretrained transformer-based models were loaded from the Hugging Face hub, which provides access to a wide array of state-of-the-art models. The models were initialized with pre-trained weights and then fine-tuned to adapt to the specific task. The training loop and the training arguments were configured using the Trainer class of the Hugging Face Transformers library. This class simplifies the process of training, validation, and hyperparameter optimization, allowing a streamlined workflow. This training setup provided an effective way of fine-tuning the models on the HateSpeech_Kenya dataset.

3.2.6 Evaluation

The models were evaluated based on their accuracy, precision, recall and F1 scores. These metrics offered a fine level of detail of the model's performance, including aspects such as false positives and false negatives. Since this is a multi-class classification task, accuracy and macro F1 score were used for model comparison. The macro F1 score was calculated as the average of the F1 scores for each class, treating all classes equally regardless of their support. Accuracy measured the overall correctness of the model, while precision and recall evaluated the ability of the model to identify positive instances correctly and the coverage of actual positive instances, respectively. The F1-score, a harmonic mean of precision and recall, balanced these two metrics to provide a single performance measure.

4. RESULTS

This section describes the results of the transformer-based models as feature extractors with an SVM classifier and end-to-end classifiers. The evaluation metrics used are accuracy and macro F1 score, allowing better comparison between the models' performance.

4.1 SVM Classifier using Hidden States

Table 3: Results for the SVM Classifier using Hidden States

Transformer	Accuracy	Macro Precision	Macro Recall	Macro F1 Score
mBERT	0.5461	0.42	0.51	0.40
XLM-RoBERTa	0.4800	0.41	0.51	0.38
mDistilBERT	0.5387	0.43	0.52	0.40

Table 3 above shows the results of training an SVM classifier on features extracted from the three transformer-based models. The best-performing classifier was learned using the features extracted from the mBERT model. This classifier achieved the highest accuracy of 0.5461 and a macro F1 score of 0.40, making it the best performer overall. The classifier learned using the features extracted from the mDistilBERT model had a slightly lower accuracy of 0.5387 but matched the mBERT model with a macro F1 score of 0.40, placing it in second place. XLM-RoBERTa model learned the lowest performing classifier with an accuracy of 0.4600 and a macro F1 score of 0.38, making it the least effective model in this comparison. These results show that larger models, like XLM-RoBERTa, do not translate to better performance as feature extractors. In contrast, smaller models like mBERT and mDistilBERT may extract features for traditional classifiers more effectively.

4.2 Transformers as End-to-End Classifiers

Table 4: Results for Transformers as End-to-End Classifiers

Transformer	Accuracy	Macro Precision	Macro Recall	Macro F1 Score
mBERT	0.5605	0.47	0.62	0.46
XLM-RoBERTa	0.6069	0.48	0.61	0.49
mDistilBERT	0.5982	0.46	0.60	0.46

As in table 4 above, the XLM-RoBERTa model achieved the highest accuracy of 0.6069 and the highest macro F1 score of 0.49, making it the top performer overall for this comparison. The second-best performing model was the mDistilBERT model, which had an accuracy of 0.5982 and a macro F1 score of 0.46. The mBERT model came in third, with the lowest accuracy of 0.5605. However, the mBERT model matched the mDistilBERT model with a similar macro F1 score of 0.46.

4.3 Transformers as End-to-End Classifiers on an Imbalanced Dataset

Table 5: Results for Transformers on an Imbalanced Dataset

Transformer	Accuracy	Macro Precision	Macro Recall	Macro F1 Score
mBERT	0.7820	0.62	0.50	0.53
XLM-RoBERTa	0.7679	0.57	0.48	0.49
mDistilBERT	0.7785	0.61	0.47	0.50

When trained on an imbalanced dataset, the models had varied performance, as shown in table 5 above. The mBERT model was the best overall performer, achieving the highest accuracy of 0.7820 and the highest macro F1 score of 0.53. The mDistilBERT model came in second, with an accuracy of 0.7785 and a macro F1 score of 0.50. The XLM-RoBERTa

model was the least effective model in this comparison, with an accuracy of 0.7679 and a macro F1 score of 0.49. Despite being a lightweight model, mDistilBERT's close performance to mBERT shows its effectiveness while maintaining a low computational cost.

5. DISCUSSION

5.1 SVM Classifier Using Hidden States

The transformer-based models demonstrated varying performance when used as feature extractors for SVM classifiers. The SVM classifiers trained on features extracted from mBERT and mDistilBERT showed similar effectiveness, achieving a moderate balance between precision and recall. The similar performance of the two models shows that both can capture relevant features from the input data, which the SVM classifier can effectively utilize. On the other hand, XLM-RoBERTa was the least proficient in this setting, indicating that it might not be as effective as a feature extractor as the other models. The lower accuracy and macro F1 score showed that the features extracted by XLM-RoBERTa were less discriminative, leading to the low performance of the SVM classifier.

5.2 Transformers as End-to-End Classifiers

Overall, the models performed better as end-to-end classifiers than their SVM counterparts, demonstrating that direct training on the classification task allows them to learn more task-specific features. The XLM-RoBERTa model stood out as the highest-performing model for this case, showing its robust ability to handle the balanced dataset. Its superior performance can be attributed to its ability to capture complex patterns in multilingual and cross-lingual contexts, which are crucial for accurately identifying hate speech and offensive content in code-switched data. The mDistilBERT model followed closely, demonstrating a competitive performance. This model's efficiency and effectiveness show that it balances computational resource requirements and classification performance well, making it a practical choice for real-world applications where resources might be limited. The mBERT model, while still effective, was slightly less robust than the other two models. Although it showed significant improvement over its SVM counterpart, its lower performance metrics indicated a need for further optimization in handling the dataset's multilingual and code-switched nature.

5.3 Transformers as End-to-End Classifiers on an Imbalanced Dataset

The end-to-end classifiers showed distinct behaviour when trained on the imbalanced dataset. The mBERT model was the most robust, effectively handling imbalanced data and achieving the best overall performance. Its ability to maintain high accuracy and macro F1 score demonstrates that it can robustly learn from skewed class distributions, which is crucial for real-world applications where data is often imbalanced. The mDistilBERT model also performed well, although slightly behind the mBERT. This performance demonstrates that while mDistilBERT is still efficient, specific characteristics of the BERT architecture might give it an edge in dealing with imbalanced datasets. While the XLM-RoBERTa model is still effective, it showed more sensitivity to class imbalance. Its lower performance metrics indicate a potential area for further enhancement, such as incorporating techniques specifically designed to mitigate the effects of class imbalance. This sensitivity highlights the challenges even advanced models face when dealing with skewed data distributions and underscores the need for continuous refinement in model training and data handling strategies.

6. CONCLUSION

The comparative study highlights the superiority of transformer-based models as end-to-end classifiers over traditional classifiers with extracted features. Among the models evaluated, the XLM-RoBERTa model was the best performer on the balanced dataset, making it the most suitable choice for scenarios where class distribution is even. On the other hand, the mBERT model was the most effective model on the imbalanced dataset, demonstrating strong overall performance and robustness in handling class imbalance. The results of this study underscore the versatility and effectiveness of transformer-based models in the domain of hate speech detection. Additionally, the study shows that while different models and configurations offer unique strengths, there is no one-size-fits-all solution. The choice of model and training strategy should be guided by the specific requirements of the task, such as the nature of the dataset and the computational resources available. The study also highlights the potential for fine-tuning these models with various architectural modifications and hyperparameter optimizations to enhance their performance further. Future work could explore further enhancements to these models, including techniques for addressing class imbalance and optimizing feature extraction processes.

REFERENCES

1. E. Ombui, L. Muchemi, and P. Wagacha, **"Hate Speech Detection in Code-switched Text Messages,"** 3rd Int. Symp. Multidiscip. Stud. Innov. Technol. ISMSIT 2019 - Proc., no. April 2020, 2019, doi: 10.1109/ISMSIT.2019.8932845.
2. S. K. Mugambi, **"Sentiment analysis for hate speech detection on social media: TF-IDF weighted N-Grams based approach,"** 2017, [Online]. Available: https://suplus.strathmore.edu/handle/11071/5657
3. Moy, Tian Xiang, Mafas Raheem, and Rajasvaran Logeswaran. **"Hate speech detection in English and non-English languages: A review of techniques and challenges."** Technology (2021). DOI: 10.14704/WEB/V18SI05/WEB18272

4. Gimode, Jescah Khadi. **"A socio-pragmatic and structural analysis of code-switching among the Logoli speech community of Kangemi, Nairobi, Kenya."** PhD diss., University of South Africa, 2015. https://core.ac.uk/download/pdf/43177683.pdf
5. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). **Attention is all you need.** Advances in neural information processing systems, 30.
6. Ghosh, Koyel, and Apurbalal Senapati. **"Hate speech detection: a comparison of mono and multilingual transformer model with cross-language evaluation."** In Proceedings of the 36th Pacific Asia Conference on Language, Information and Computation, pp. 853-865. 2022.
7. Mukherjee, Swapnanil, and Sujit Das. **"Application of transformer-based language models to detect hate speech in social media."** Journal of Computational and Cognitive Engineering (2022). DOI: 10.47852/bonviewJCCE2022010102
8. Srikissoon, Trishanta, and Vukosi Marivate. **"Combating Hate: How Multilingual Transformers Can Help Detect Topical Hate Speech."** EPiC Series in Computing 93 (2023):203-215. https://2wvvw.easychair.org/publications/download/28NM
9. Mustafa Farooqi, Zaki, Sreyan Ghosh, and Rajiv Ratn Shah. **"Leveraging Transformers for Hate Speech Detection in Conversational Code-Mixed Tweets."** arXiv e-prints (2021): arXiv-2112.
10. Ababu, Teshome Mulugeta, and Michael Melese Woldeyohannis. **"Afaan Oromo hate speech detection and classification on social media."** In Proceedings of the thirteenth language resources and evaluation conference, pp. 6612-6619. 2022. https://aclanthology.org/2022.lrec-1.712
11. Singh, Neeraj Kumar, and Utpal Garain. **"An Analysis of Transformer-based Models for Code-mixed Conversational Hate-speech Identification."** In Forum for Information Retrieval Evaluation (Working Notes) (FIRE). CEUR-WS. Org. 2022.
12. Chanda, Supriya, S. Ujjwal, Shayak Das, and Sukomal Pal. **"Fine-tuning pre-trained transformer-based model for hate speech and offensive content identification in English, Indo-Aryan and code-mixed (English-Hindi) languages."** In Forum for Information Retrieval Evaluation (Working Notes) (FIRE), CEUR-WS. Org. 2021.
13. Patil, Aryan, Varad Patwardhan, Abhishek Phaltankar, Gauri Takawane, and Raviraj Joshi. **"Comparative Study of Pre-Trained BERT Models for Code-Mixed Hindi-English Data."** In 2023 IEEE 8th International Conference for Convergence in Technology (I2CT), pp. 1-7. IEEE, 2023.
14. Devlin, J., 2018. **Bert: Pre-training of deep bidirectional transformers for language understanding.** arXiv preprint arXiv:1810.04805.
15. Conneau, A., 2019. **Unsupervised cross-lingual representation learning at scale.** arXiv preprint arXiv:1911.02116.
16. Sanh, V., 2019. **DistilBERT, A Distilled Version of BERT: Smaller, Faster, Cheaper and Lighter.** arXiv preprint arXiv:1910.01108.
17. Ombui, E., Muchemi, L. and Wagacha, P., 2021. **Building and annotating a codeswitched hate speech corpora.** Int. J. Inf. Technol. Comput. Sci, 3, pp.33-52.
18. Muia, C.M., Oirere, A.M. and Ndungu, R.N., 2024. **A Comparative Study of Transformer-based Models for Text Summarization of News Articles.** https://doi.org/10.30534/ijatcse/2024/011322024