TweepFake: Detecting Deep Fake Tweets

Shaden Dhafer Alhashysh[1], Hadeel Hassan Althbiti[2], Zainab Adnan Alsaggaf[3],
Sabah Abdulgader Baothman[4], Reem Saleh Saeed Almalki[5], Asail Mashhour Alamoudi[6],
Seham Khaldun Nahlawi[7], Shahad Maher Maqram[8]

Department of Computer Science, King Abdulaziz University, Jeddah, Saudi Arabia

smalhashysh@stu.kau.edu.sa[1], halthbiti0002@stu.kau.edu.sa[2], zalialsaggaf@stu.kau.edu.sa[3],
sadbullahbaothman@stu.kau.edu.sa[4], rsaeedalmalki@stu.kau.edu.sa[5], aeidahalamoudi@stu.kau.edu.sa[6],
shasannahlawi@stu.kau.edu.sa[7], ssaeedmagram@stu.kau.edu.sa[8]

Abstract: The proliferation of deepfake tweets poses significant challenges to the integrity of information on social media platforms like Twitter. This study aims to develop and evaluate various machine-learning models to detect deepfake tweets effectively. We utilized a comprehensive dataset of 25,572 tweets, equally divided between human-generated and bot-generated content, sourced from diverse accounts using multiple generative techniques. Our preprocessing involved cleaning the tweet text by removing URLs, mentions, special characters, and irrelevant white spaces and converting all text to lowercase. The dataset was split into 81% for training, 9% for validation, and 10% for testing.

***Index Terms*—Deepfake tweets, Twitter, machine learning, RoBERTa, SVM, BERT, BiLSTM, disinformation detection.**

I. Introduction

The rapid advancement of artificial intelligence and machine learning technologies has led to the proliferation of deepfake content, posing significant threats to the integrity of information on social media platforms. Deepfake tweets, which involve sophisticated algorithms to generate misleading or false content, have become increasingly prevalent on Twitter. These tweets can spread misinformation, manipulate public opinion, and undermine trust in digital communication.

Detecting deepfake tweets is crucial for maintaining the credibility of information shared on social media. Traditional methods of fake news detection, such as manual verification and basic machine learning models, have proven inadequate in addressing the complex and evolving nature of deepfake content. Therefore, there is a pressing need for advanced machine-learning approaches to accurately identify and mitigate the spread of deepfake tweets

This study focuses on developing and evaluating several state-of-the-art machine learning models for detecting deepfake tweets on Twitter. We employ a comprehensive dataset of 25,572 tweets, equally divided between human-generated and bot-generated content, sourced from diverse accounts using various generative techniques. Our methodology involves extensive preprocessing of the tweet text to enhance data quality and ensure uniformity, followed by applying advanced classification algorithms.

We explore the efficacy of four prominent machine learning models: RoBERTa, Support Vector Machine (SVM), BERT, and Bidirectional Long Short-Term Memory (BiLSTM) networks. Each model offers unique strengths in handling the challenges associated with deepfake detection. RoBERTa, an optimized version of BERT, excels in capturing contextual nuances and generalizing across diverse datasets. SVM is renowned for its robustness in high-dimensional spaces, while BERT's bidirectional processing provides a deeper understanding of linguistic patterns. BiLSTM further enhances detection accuracy by considering context in both forward and backward directions.

By systematically evaluating these models, this study aims to identify the most effective approach for detecting deepfake tweets and highlight the importance of continuous refinement in detection algorithms. Our findings contribute to the broader effort of combating disinformation on social media, ensuring the integrity and reliability of digital communication.

II. Related Work

Ahmed et al. (2021) developed a methodology for classifying fake news on Twitter using machine learning algorithms. They collected a dataset from Kaggle and Twitter API, comprising 4048 news articles, of which 1865 were fake and 2118 were genuine. The preprocessing steps involved removing stop words, tokenization, stemming, and converting text to English. Feature extraction was performed using TF-IDF, and Naïve Bayes and Passive Aggressive classifiers were employed for classification. The Passive Aggressive classifier yielded superior results to Naïve Bayes, achieving an accuracy of 0.78, precision of 0.61, recall of 0.69, and F1-score of 0.54. In contrast, the Naïve Bayes classifier attained an accuracy of 0.73, precision of 0.63, recall of 0.67, and F1-score of 0.51. These findings underscore the effectiveness of the Passive-aggressive classifier in identifying fake news tweets [1].

Khanam et al. (2021) developed a methodology for fake news detection utilizing supervised machine learning algorithms applied to the LIAR-PLUS Master dataset. The approach encompassed dataset collection, preprocessing, feature selection, and model training. Following noise removal and part-of-speech tagging in the preprocessing phase features such as word count, average word length, and part-of-speech tags were selected for extraction. The dataset was divided into training and testing sets, with classification models including XGBoost, Random Forests, Naive Bayes, K-Nearest Neighbors (KNN), Decision Tree, and SVM. Results, analyzed using confusion matrices, revealed varying accuracies for each algorithm. XGBoost exhibited the highest accuracy at 75.34%, followed closely by SVM and Random Forests with 73.21% and 72.89% accuracy, respectively. Naive Bayes performed moderately well, with an accuracy of 70.98%, while KNN and Decision Tree achieved 68.76% and 71.45% respectively. These findings highlight the efficacy of machine learning approaches in detecting fake news [2].

Li et al. (2023) developed a methodology for detecting deepfake texts by examining texts written by machines and humans using various classification algorithms. They collected a dataset from multiple sources, including Reddit, BBC news, and scientific articles, comprising 447,674 human-written and machine-generated text instances. Preprocessing steps involved punctuation normalization, line-break removal, and filtering out texts that were too long or too short. Feature extraction was performed using advanced linguistic analysis and machine learning models. Four detection methods were employed: PLM-based (Longformer), FastText, GLTR, and DetectGPT. The Longformer classifier yielded superior results, achieving an average recall (AvgRec) of 90.53% and an AUROC of 0.99 in in-domain settings. Longformer achieved an AvgRec of 68.40% and an AUROC of 0.93 in out-of-distribution settings. In contrast, the FastText classifier attained an AvgRec of 78.80% and an AUROC of 0.83 in in-domain settings, and an AvgRec of 63.54% and an AUROC of 0.72 in out-of-distribution settings. DetectGPT performed well in in-domain settings but showed significant performance deterioration in cross-domain scenarios. These findings underscore the effectiveness of the Longformer classifier in identifying deepfake texts across diverse domains and LLMs [3].

Roy et al. (2018) developed a deep-learning framework for detecting and classifying fake news into fine-grained categories. They employed a novel ensemble architecture combining Convolutional Neural Network (CNN) and Bi-directional Long Short Term Memory (Bi-LSTM), which processed features through a Multi-layer Perceptron (MLP) for final classification. Tested on the LIAR dataset, which includes 12.8K annotated statements, their model achieved an accuracy of 44.87%, outperforming the existing state-of-the-art methods. The ensemble approach of CNN for feature extraction and Bi-LSTM for capturing sequential dependencies improved classification accuracy, demonstrating its potential in handling complex fake news categorization tasks [4].

Lai et al. (2022) integrates Natural Language Processing (NLP) and Machine Learning (ML) techniques to classify fake news. It uses traditional ML models: Binomial Logistic Regression, Naive Bayes, Support Vector Machines, Random Forest, and neural network models: Convolutional Neural Networks and Long Short-Term Memory Networks. The dataset comprises fake and true news articles from a Kaggle dataset and scraped articles from various news sources, totaling 24,204 fake news articles and 22,970 true news articles. Preprocessing involves tokenization, with Term Frequency-Inverse Document Frequency (TF-IDF) for ML models and Word2Vec embeddings for neural networks. The results show that all traditional ML models achieved over 85% accuracy, while neural network models achieved over 90% accuracy. Neural network models outperformed traditional ML models by approximately 6% in precision, indicating their superior performance in fake news classification [5].

Wang et al. (2019) employ a hierarchical classifier to categorize fake news into five types: factual, propaganda, hoax, irony, and two subcategories of propaganda. The methodology integrates logistic regression and deep neural network techniques, analyzing linguistic cues, sentiment, subjectivity, stance, and social media dissemination patterns. The datasets used include the SHPT and PolitiFact datasets, supplemented by related tweets to enhance classification accuracy. The findings indicate that this fine-grained classification approach outperforms traditional binary classifiers, providing deeper insights into the nature and spread of misinformation on social media [6].

The article by M. Khalid et al. (2024) delves into the analysis of sentiments in deepfake tweets, highlighting the potential risks of deepfakes and the consequences of comprehending public sentiment. It presents a novel Sentiment Majority Voting Classifier (SMVC) to enhance sentiment analysis accuracy by aggregating results from three lexicon-based models: TextBlob, VADER, and AFINN. TextBlob provides a user-friendly API for everyday natural language processing tasks. At the same time, VADER is a rule-based sentiment analysis tool tailored for social media, and AFINN utilizes pre-computed sentiment scores for words. The study also explores transfer learning techniques to improve the performance of machine learning models. Four models: Logistic Regression, SVM, Random Forest, and Naive Bayes. Were evaluated, with Logistic Regression achieving the highest accuracy at 98.9%. The dataset comprised tweets related to deep fakes, which were collected and annotated for sentiment analysis. The authors recommend further research to refine these techniques and examine their materiality in other domains [7].

Prior research by Cso (2021) on identifying generated media, including GAN-generated images and AI-generated texts, has significantly advanced while facing ongoing challenges. For instance, GANprintR and ensemble detection have exhibited high precision in spotting fake images in controlled settings using datasets such as CelebA and FFHQ. However, their effectiveness diminishes when confronted with real-world data, unknown generative models, and post-processing variations. Similarly, studies examining text detection, like those assessing the reliability of RoBERTa fine-tuned on GPT-2 generated datasets, illustrate the difficulty of distinguishing AI-generated texts from authentic ones, particularly with social media's diverse and concise text formats standard. These findings emphasize the potential of automated detection tools in supporting intelligence and cybersecurity endeavors, underscoring their limitations and the pressing need for further development to bolster their resilience and dependability in practical scenarios. The research underscores the importance of creating adaptable and robust detection systems to manage generated media's dynamic and growing nature effectively [8].

In this study by Kirn et al. (2022), researchers used a dataset of over 10,000 tweets, including human-written and bot-generated tweets, to train and test models for detecting fake tweets. They converted the tweets into numerical vectors using feature extraction techniques like TF-IDF and Doc2vec. Various classifiers, including Support Vector Machines (SVM) and neural networks, were employed to detect deepfake tweets. The models were evaluated through cross-validation and tested on a labeled dataset. The results showed with models achieving 95% accuracy in identifying fake tweets [9].

Pu et al.'s study (2022) focused on detecting deepfake text generated by advanced language models like GPT-2 and GPT-3. They collected synthetic text from various online services and evaluated the effectiveness of different detection methods. The study highlighted that many existing defenses show significant performance degradation when applied to real-world data compared to their original test settings. They found that defenses using semantic information in the text, such as entity-based features, provided better robustness and generalization performance [10].

Fagni et al. (2021) introduced a dataset named TweepFake for detecting deepfake tweets, which includes 25,572 tweets equally divided between human and machine-generated content. They evaluated 13 detection methods, including logistic regression, random forest, SVM, BERT, and RoBERTa. Their findings revealed that the RoBERTa model achieved the highest accuracy of 89.6%, demonstrating the challenges posed by sophisticated generative models like GPT-2 in creating human-like text and the effectiveness of transformer-based models in detecting deepfake tweets. The dataset and code are publicly available to support further research in this area [11].

Murayama et al. (2021) developed a point process model named TiDeH (Time-Dependent Hawkes process) to understand and predict the spread of fake news on Twitter. The proposed model views the spread as a two-stage process: initially, fake news spreads like ordinary news; later, the recognition of its falsity spreads as a separate news item. They validated the model using two datasets of fake news on Twitter. The TiDeH model achieved a mean absolute error of 36.9 and 2.40 in the RFN and Tohoku datasets, respectively, demonstrating its superiority over current methods in predicting the evolution of fake news spread. The model effectively infers the correction time when users start recognizing the falsity, providing insights into the dynamics of fake news dissemination and potential for detection and mitigation applications [12].

Sadiq et al. (2023) leverage Deep Learning (DL) and FastText embeddings to detect deepfake tweets on social media. The study employs a simple Convolutional Neural Network (CNN) architecture using FastText embeddings to classify tweets as human-generated or bot-generated. The dataset includes 25,572 tweets from the TweepFake dataset, comprising 12,786 human-generated tweets and 12,786 machine-generated tweets from various models including GPT-2, RNN, and others. Preprocessing involves tokenization, case conversion, and removal of unnecessary elements. The methodology utilizes CNN with FastText embeddings and compares performance with several machine learning models (Decision Tree, Logistic Regression, AdaBoost Classifier, etc.) and other deep learning models (LSTM, CNN-LSTM). Results indicate that CNN with FastText embeddings achieves the highest accuracy of 93%, outperforming traditional machine learning models and other deep learning models. This demonstrates the effectiveness of the proposed method for deepfake tweet detection, offering a significant improvement over existing state-of-the-art approaches [13].

Gambini et al. (2022) explore the capabilities of state-of-the-art deepfake social media text detectors in recognizing tweets generated by GPT-2 and GPT-3 models. They utilize and optimize various Transformer-based language models, including GPT-2, BERT, RoBERTa, and BERTweet, The dataset comprises tweets from human and bot accounts, including those generated by older techniques like RNNs, resulting in a diverse set for training and testing. Preprocessing involves standard text preprocessing techniques tailored to social media text. The results show that hyper-parameter optimization generally balances the performance across human and machine-generated texts, with the BERTweet model achieving the highest accuracy on human-written tweets (94.7%) and significant improvement on GPT-2 tweets (80.2%). An ensemble method combining BERTweet, BART, and TwitterRoBERTa further enhances detection performance, particularly in realistic settings with low false positive rates. However, all detectors exhibited a marked decrease in accuracy when tested on GPT-3 generated tweets, highlighting the superior human-like quality of GPT-3 texts and the need for new detection approaches to handle such advanced generative models. The study underscores the ongoing challenge of deepfake detection in social media and calls for further research to address the detection of increasingly sophisticated machine-generated texts [14].

Rupapara et al. (2021) discuss sentiment analysis in the context of deep fake technology using machine learning and deep learning models. It compares the performance of various classifiers and models such as SVM, LR, GNB, ETC, GBM, ADA, LSTM, GRU, Bi-LSTM, and CNN+LSTM. The study highlights the effectiveness of a stacked Bi-directional LSTM (SBi-LSTM) model in accurately classifying sentiments in deep fake tweets, achieving an accuracy of 0.92 [15].

Ajik et al. (2023) ffocuson utilizing optimized CNN and LSTM techniques fto detectfake news through data cleaning, preprocessing, feature extraction, and hyperparameter tuning. By employing the HyperOpt technique for model optimization, the study demonstrates enhanced performance of the optimized models in comparison to non-optimized ones. The findings underscore the significance of effective strategies in the ongoing battle against fake news dissemination [16].

III. Dataset

We have integrated a dataset of real Deepfake tweets sourced from actual posts on Twitter [17]. This dataset consists of tweets from a diverse array of 23 bots and 17 human accounts using various generation techniques such as Markov Chains, RNNs, RNNs combined with Markov models, LSTMs, and GPT-2. Additionally, they included a random selection of tweets from the human accounts emulated by the bots to ensure the dataset's completeness and balance. The resulting dataset contains 25,572 tweets equally divided between human-generated and bot-generated tweets. This resource is crucial for researchers and practitioners in their ongoing quest to develop effective Deepfake detection and mitigation strategies.

IV. Methodology

This section outlines the methods and techniques used in our research to detect Deepfake tweets. We describe the preprocessing steps applied to the dataset, the methodology for splitting the data, and the proposed models implemented for detection.

A. The prepossessing stage

In the preprocessing step of our Deepfake tweet dataset, we began by dropping unnecessary columns to simplify the analysis. We kept only the tweet's text and the account type, indicating whether a human or a bot generated the tweet. Then, we applied a cleaning method to the tweet text to enhance the quality and consistency of the data. This involved removing URLs, mentions, special characters, and irrelevant white spaces, which could introduce noise into the analysis. Also, we converted all text to lowercase to ensure uniformity, as text in different cases could be treated as distinct tokens by the model. This comprehensive preprocessing step was crucial in preparing the data for effective training, validation, and testing of our Deepfake detection models, ensuring that the text input was clean, consistent, and ready for further analysis.

B. The splitting methodology

To effectively train, validate, and test our Deepfake detection models, we have split the dataset of 25,572 tweets in a structured manner. The dataset is divided into 81% for training, 9% for validation, and 10% for testing. Specifically, 20,712 tweets are used for training, providing a comprehensive base for the model to learn from various human and bot-generated tweets. The validation set comprises 2,302 tweets, allowing us to fine-tune the model and ensure it generalizes well to new data. The remaining 2,558 tweets are reserved for testing, providing an unbiased evaluation of the model's performance. This systematic split ensures robust training, effective validation, and accurate testing, leading to reliable and effective Deepfake detection capabilities.

C. The proposed methods

This section provides details of the proposed methods for detecting deepfake tweets.

1) RoBERTa

RoBERTa (Robustly Optimized BERT Approach) is an enhanced version of the BERT model developed by Facebook AI, designed to address some of its predecessor's limitations. It builds upon BERT's language masking strategy and incorporates several vital modifications: it removes the next-sentence prediction task, trains on significantly larger datasets (over 160GB compared to BERT's 16GB), and uses larger mini-batches and higher learning rates. These changes allow RoBERTa to better generalize to downstream tasks, outperforming BERT on various NLP benchmarks [18]. In the context of deep fake detection on Twitter, RoBERTa's robustness and improved training make it particularly effective. By fine-tuning RoBERTa on datasets of real and synthetic tweets, it becomes adept at distinguishing between authentic and AI-generated content. This capability is crucial for identifying disinformation and fake accounts on social media platforms. The model's architecture and training process enable it to capture subtle linguistic nuances, making it a powerful tool for maintaining the integrity of information on Twitter and other social media sites [8].

2) SVM

Support Vector Machine (SVM) is a supervised learning algorithm used for classification and regression tasks, and it is predominantly applied to classification problems. The main objective of SVM is to find the optimal decision boundary that separates data points from different classes in an n-dimensional space. This decision boundary is known as a hyperplane. SVM identifies the most crucial data points, known as support vectors, which are the points closest to the hyperplane from each class. These support vectors are instrumental in defining the position and orientation of the hyperplane. By maximizing the margin, or the distance between the hyperplane and the support vectors, SVM ensures that the decision boundary is as robust and distinct as possible. This helps accurately categorize new data points for the future [19]. SVM is particularly effective for fake tweet detection due to its ability to handle high-dimensional spaces, versatility with different kernel functions, robustness to overfitting, and scalability with sparse data. It performs well with limited labeled data and can be adapted to handle imbalanced datasets. AItsstrong theoretical foundations ealso nsure globally optimal solutions, and its quick classification capabilities make it suitable for real-time applications. These strengths combine to make SVM a powerful tool for discerning subtle patterns and maintaining robustness against overfitting and class imbalance in the dynamic environment of social media [20].

3) BERT

Bidirectional Encoder Representations from Transformers (BERT) is a pre-trained language representation model developed by Google, primarily used for natural language processing (NLP) tasks. BERT revolutionized NLP by introducing a bidirectional training approach to language modeling. Unlike traditional models that process text sequentially (left-to-right or right-to-left), BERT processes text bi-directionally, simultaneously considering a word's left and right context. This approach enables BERT to capture deeper linguistic information and context, enhancing its ability to understand the nuances of language. BERT's architecture is based on the Transformer model, which uses self-attention mechanisms to process words about all other words in a sentence. This allows BERT to understand the contextual relationships between words more effectively. The model is pre-trained on two tasks: Masked Language Modeling (MLM) and Next Sentence Prediction (NSP). In MLM, random words in a sentence are masked, and BERT learns to predict these masked words based on their context. In NSP, BERT learns to understand the relationship between pairs of sentences by predicting if one sentence follows another in a text. The pre-training phase provides BERT with a strong understanding of language, which can be fine-tuned for specific NLP tasks such as question answering, sentiment analysis, and text classification by adding a simple output layer. BERT's bidirectional context understanding enables it to capture the nuanced linguistic patterns that distinguish authentic and AI-generated tweets, making it effective for deep fake detection. Its fine-tuning ability for specific NLP tasks allows BERT to accurately classify and identify disinformation on social media platforms, ensuring the integrity of information on Twitter [21].

4) BiLSTM

Bidirectional Long Short-Term Memory (BiLSTM) is a Recurrent Neural Network (RNN) that outperforms the standard LSTM by consisting of two LSTMs and processing the input data forward and backward. The final output is then created by combining the outputs of the two LSTM networks. This bidirectional processing gives the network a more comprehensive understanding of the sequential data, making it particularly useful for tasks where context from past and future states is important. BiLSTM has achieved state-of-the-art outcomes on various tasks, such as text summarization, speech recognition, and machine translation [22]. Since BiLSTMs can extract contextual information from tweets' past and future words, they are especially useful for "tweepfake" detection, making detecting subtle cues and patterns indicative of deception easier. The model's accuracy in identifying tweets as authentic or fraudulent is improved by this bidirectional processing, which raises the standard of social media content analysis.

V. Results and Discussion

The performance of each model is evaluated using accuracy and other metrics. Subsequent subsections discuss the details of the experiments.

A. RoBERTa Result

TABLE I
RoBERTa Model Training

EXP	Epochs	LR	E. Stop.	Dropout	Reg.	Batch Norm.
1	10	2e-5	Yes	No	No	No
2	10	1e-5	Yes	Yes	Yes	No
3	10	1e-5	Yes	Yes	Yes	Yes

TABLE II
RoBERTa Model Training Results

EXP	Accuracy	Precision	Recall	F1
1	0.8213	0.8227	0.8213	0.8211
2	0.8241	0.8253	0.8240	0.8239
3	0.8241	0.8299	0.8240	0.8233

The results of three different experiments were analyzed to determine the best performance. The comparison was based on accuracy, precision, recall, and F1 score. Experiment 1 had the lowest values in all metrics, making it the least effective. Experiment 3 had the highest precision, indicating a strong ability to identify positive instances correctly. However, the recall and F1 scores for Experiment 3 were slightly lower compared to Experiment 2. Experiment 2 demonstrated the best overall balance, with the highest F1 score, which combines precision and recall into a single measure. This indicates that Experiment 2 performed best in identifying positive instances while minimizing errors. Therefore, Experiment 2 is the best experiment due to its superior overall performance.

B. SVM Result

TABLE III
SVM Training

EXP	C	Kernel	gamma
1	[0.1,1,10]	['rbf','linear']	[1, 0.1, 0.01]
2	[0.1,1,10,100]	'rbf'	[1, 0.1, 0.01, 0.001]
3	[0.1,1,10,100]	'rbf'	[scale,auto]
4	[0.1,10]	'rbf'	['scale','auto']

TABLE IV
SVM Training Results

EXP	Best-Parameters	Accuracy	Precision	Recall	F1
1	[10,'linear',0.1]	0.66	0.59	0.49	0.59
2	[10,'rbf',1]	0.61	0.60	0.60	0.59
3	[10,'rbf','scale']	0.69	0.70	0.70	0.70
4	[0.1, 'rbf','auto']	0.57	0.50	0.53	0.50

The results of four different experiments were analyzed and compared to determine the best performance based on accuracy, precision, recall, and F1 score. Experiment 4 had the lowest accuracy, precision, and F1 score values, making it the least effective overall, despite having a recall higher than Experiment 1. Experiment 3 excelled with the highest accuracy, precision, recall, and F1 score, indicating a strong overall performance and a high ability to identify positive instances correctly. While Experiment 2 showed balanced precision and recall, its F1 score and overall metrics were lower than Experiment 3. Experiment 1 also lagged with lower accuracy and recall despite having a precision and F1 score similar to Experiment 2. Therefore, Experiment 3 is the best experiment due to its superior performance across all metrics, making it the most effective in accurately identifying positive instances and minimizing errors.

C. *BERT Result*

TABLE V
BERT MODEL TRAINING

EXP	Epochs	LR	Batch Size	Dropout	E. Stop.	Adam
1	7	5e-5	16	No	Yes	Yes
2	7	2e-5	32	No	Yes	Yes
3	7	1e-5	32	Yes	Yes	Yes

TABLE VI
BERT MODEL TRAINING RESULTS

EXP	Accuracy	Precision	Recall	F1
1	0.7982	0.7998	0.7982	0.7980
2	0.8072	0.8094	0.8072	0.8069
3	0.8021	0.8042	0.8021	0.8018

The results of three experiments were analyzed to evaluate the performance of the BERT model. Experiment 1, with a learning rate of oe-5 and a batch size of o6, showed the lowest accuracy, precision, recall, and F1 score among the three experiments. Experiment 2, which used a learning rate 2e-5 and a batch size 32, achieved the highest scores across all metrics. However, Experiment 3, despite having a lower learning rate of 1e-5 and including dropout, did not perform as well as Experiment 2 but was still better than Experiment 1. Importantly, Experiment 3 did not exhibit overfitting, making it the best configuration for the BERT model. Including dropout and a lower learning rate in Experiment 3 contributed to a more generalized model, balancing accuracy, precision, recall, and F1 score without overfitting.

D. *BiLSTM Result*

Four experiments were conducted with different parameters in each. Table VII shows the configurations used in each experiment.

TABLE VII
BiLSTM MODEL TRAINING

EXP	Epochs	LR	E. Stop.	Dropout	Reg.	Batch Norm.
1	30	0.001	No	No	No	No
2	30	0.001	Yes	1 Layer	No	Yes
3	20	0.001	Yes	2 Layers	Yes	No
4	16	0.0001	Yes	2 Layers	Yes	No

The results of the BiLSTM model training are summarized in Table VIII. It can be observed that with the progression of experiments, the accuracy of the model improves while the loss decreases. Specifically, Experiment 4 shows the highest accuracy at 0.68 and the lowest loss at 0.60, indicating that the combination of 16 epochs, a learning rate of 0.0001, early stopping, and L2 regularization with dropout layers contributed to better performance.

TABLE VIII
BiLSTM MODEL TRAINING RESULTS

EXP	Accuracy	Loss
1	0.60	0.80
2	0.62	0.69
3	0.65	0.66
4	0.68	0.60

Overall, Table IX shows a comparison of the best accuracy reached for each model. RoBERTa demonstrated the best performance among the evaluated models with an accuracy of 82%, followed by SVM (69%), BERT (80%), and BiLSTM (68%).

TABLE IX
ACCURACY COMPARISON OF MODELS

Model	Accuracy
RoBERTa	82%
SVM	69%
BERT	80%
BiLSTM	68%

VI. CONCLUSION AND FUTURE WORK

In conclusion, we have used four AI models, RoBERTa, SVM, BERT, and BiLSTM, with a dataset of real Deepfake tweets to classify the tweets as either written by a bot or a human. We standardized the preprocessing across all models to facilitate a fair comparison. RoBERTa achieved the highest accuracy at 82.41%, outperforming other models. This underscores the importance of selecting suitable models for specific tasks. Challenges included varying model performance and the impact of preprocessing techniques. For future work, we suggest expanding the dataset for better generalizability, incorporating multi-modal data for additional context, and experimenting with ensemble methods to improve accuracy. Further research into advanced preprocessing techniques and model interpretability could enhance performance and trust in AI-driven classification systems. To ensure reproducibility and support further research, all code implementations and experimental configurations are publicly available in our GitHub repository: https://github.com/Zinab0/TweepFake.

References

[1] Nikam, S. S., & Dalvi, R. (2020, October). Machine learning algorithm based model for classification of fake news on twitter. In 2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC) (pp. 1-4). IEEE.

[2] Khanam, Z., Alwasel, B. N., Sirafi, H., & Rashid, M. (2021, March). Fake news detection using machine learning approaches. In IOP conference series: materials science and engineering (Vol. 1099, No. 1, p. 012040). IOP Publishing.

[3] Li, Y., Li, Q., Cui, L., Bi, W., Wang, L., Yang, L., Shi, S., & Zhang, Y. (2023, May 22). Deepfake text detection in the wild. arXiv.org. https://arxiv.org/abs/2305.13242

[4] Roy, A., Basak, K., Ekbal, A., & Bhattacharyya, P. (2018). A deep ensemble framework for fake news detection and classification. arXiv preprint arXiv:1811.04670.

[5] Lai, C. M., Chen, M. H., Kristiani, E., Verma, V. K., & Yang, C. T. (2022). Fake news classification based on content level features. Applied Sciences, 12(3), 1116.

[6] Wang, L., Wang, Y., De Melo, G., & Weikum, G. (2019). Understanding archetypes of fake news via fine-grained classification. Social Network Analysis and Mining, 9, 1-17.

[7] M. Khalid, A. Raza, F. Younas, F. Rustam, M. G. Villar, I. Ashraf, and A. Akhtar, "Novel Sentiment Majority Voting Classifier and Transfer Learning-Based Feature Engineering for Sentiment Analysis of Deepfake Tweets," IEEE Access, vol. 12, pp. 67117–67129, 2024.

[8] Cso, "Detecting Generated Media: A Case Study on Twitter Data," Policy Commons, 19-Oct-2021. [Online]. Available: https://policycommons.net/artifacts/4780722/detecting-generated-media/5617018/.

[9] Kirn, Hina, Anwar, Muhammad, Sadiq, Ashina, Zeeshan, Hafiz M, Mehmood, Imran, and Butt, Rizwan Aslam. "Deepfake tweets detection using deep learning algorithms." Engineering Proceedings, vol. 20, no. 1, 2022, pp. 2. MDPI.

[10] Pu, Jiameng, Sarwar, Zain, Abdullah, Sifat Muhammad, Rehman, Abdullah, Kim, Yoonjin, Bhattacharya, Parantapa, Javed, Mobin, and Viswanath, Bimal. "Deepfake text detection: Limitations and opportunities." In *2023 IEEE Symposium on Security and Privacy (SP)*, 2023, pp. 1613-1630. IEEE.

[11] Fagni, Tiziano, Falchi, Fabrizio, Gambini, Margherita, Martella, Antonio, and Tesconi, Maurizio. "TweepFake: About detecting deepfake tweets." PLOS ONE, vol. 16, no. 5, 2021, e0251415.

[12] Murayama, Taichi, Wakamiya, Shoko, Aramaki, Eiji, and Kobayashi, Ryota. "Modeling the spread of fake news on Twitter." PLOS ONE, vol. 16, no. 4, 2021, e0250419.

[13] S. Sadiq, T. Aljrees and S. Ullah, "Deepfake Detection on Social Media: Leveraging Deep Learning and FastText Embeddings for Identifying Machine-Generated Tweets," in IEEE Access, vol. 11, pp. 95008-95021, 2023, doi: 10.1109/ACCESS.2023.3308515.

[14] Margherita Gambini, Tiziano Fagni, Fabrizio Falchi, and Maurizio Tesconi. 2022. On pushing DeepFake Tweet Detection capabilities to the limits. In Proceedings of the 14th ACM Web Science Conference 2022 (WebSci '22). Association for Computing Machinery, New York, NY, USA, 154–163. https://doi.org/10.1145/3501247.3531560

[15] Rupapara V, Rustam F, Amaar A, Washington PB, Lee E, Ashraf I. Deepfake tweets classification using stacked Bi-LSTM and words embedding. PeerJ Comput Sci. 2021 Oct 21;7:e745. doi: 10.7717/peerj-cs.745.

[16] Ajik, Emmy & Obunadike, Georgina & Echobu, Faith. (2023). Fake News Detection Using Optimized CNN and LSTM Techniques. Journal of Information Systems and Informatics. 5. 1044-1057. 10.51519/journalisi.v5i3.548.

[17] Tizfa, "GitHub" https://github.com/ShadenDhafer/AItopics/blob/ae3fd570b6a409b2a3e84691580534ec34db9592/RoBERTa_TweepFake.ipynb

[18] "RoBERTa." https://huggingface.co/docs/transformers/model_doc/roberta.

[19] Steinwart, I., & Christmann, A. (2008). Support vector machines. Springer Science & Business Media.

[20] Meyer, D., & Wien, F. T. (2001). Support vector machines. R News, 1(3), 23-26.

[21] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

[22] A. Aziz Sharfuddin, M. Nafis Tihami and M. Saiful Islam, "A Deep Recurrent Neural Network with BiLSTM model for Sentiment Classification," 2018 International Conference on Bangla Speech and Language Processing (ICBSLP), Sylhet, Bangladesh, 2018, pp. 1-4, doi: 10.1109/ICBSLP.2018.8554396.