International Journal of Applied Mathematics
Volume 38 No. 6s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

DETECTION OF MACHINE-GENERATED TEXT BY INTEGRATING ROBERTA EMBEDDINGS WITH TOPOLOGICAL FEATURES

Rejimoan R[1*], Gnanapriya B[2], Jayasudha J S[3]

[1*]Department of Computer Science and Engineering, Annamalai University, Chidambaram

rejimoan@gmail.com

[2]Department of Computer Science and Engineering, Annamalai University, Chidambaram

priyamvatha.joey@gmail.com

[3]Department of Computer Science, Central University of Kerala

jayasudhajs@gmail.com

Abstract

In the contemporary digital landscape, language generation models have experienced an explosive surge in popularity, driven by remarkable advancements in Artificial Intelligence (AI) and Natural Language Processing (NLP). As a result, distinguishing between human-generated and machine- generated text has become increasingly complex. The pervasive presence of highly advanced language models and hence machine- generated content has heightened concerns surrounding the spread of misinformation and the proliferation of deceptive and plagiarized content. To address this pressing challenge, an innovative solution exists in harnessing the combined power of the RoBERTa (Robustly Optimized BERT Approach) model and TDA (Topological Data Analysis) features to develop a model capable of discerning between human and machine-generated text effectively. The idea is to capture semantic differences in text belonging to these two classes, as identified by RoBERTa, and integrate it with the structural and geometrical properties of the associated attention maps as learned from TDA to give rise to a model that outperforms any of these approaches taken individually. Through this endeavor, a valuable tool could be provided across various domains, including academia, enabling the detection of AI-generated content and fostering a safer and more trustworthy digital environment.

Index Terms—AI Text, Machinte Generated Text Detection, TDA, RoBERTa, TDA-BERTa, Topological Features, Deep Learning, Text Classification, Embeddings.

I. INTRODUCTION

Recent advancements in Text Generation Models (TGMs) have begun to pose a formidable challenge to the authenticity and trustworthiness of digital content. In addition to straight-forward text generation, they can be used for a plethora of other purposes such as code generation [1] and fake news generation [2]. The introduction of the transformer architecture [3] has led to a proliferation of Large Language Models (LLMs) that can mimic human writing with remarkable precision. Bidirectional Encoder Representation Transformers (BERT)

[4] and Robustly Optimized BERT Approach (RoBERTa)

Received: August 10, 2025

International Journal of Applied Mathematics
Volume 38 No. 6s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

[5] are some of the transformer-based models that are at the forefront of the field today.

Moreover, the advent of ChatGPT through the introduction of GPT-2 [6], GPT-3 [7], and more recently GPT-4 [8] has further exacerbated the issue. This underscores the urgent need for robust mechanisms to distinguish between human-authored and machine-generated text across diverse applications.

Deceptive AI-generated content poses a significant danger to society as it has become increasingly indistinguishable from human-generated content [9] and propagates quickly [10], leading to widespread misinformation. Models such asGROVER [2] can generate fake news that is virtually indistinguishable from real news, and some can generate reviews that closely resemble human-generated ones [11]. The rise of AI text generators has also led to their widespread, albeit often misapplied, use in academia, resulting in increased plagiarism that is challenging to detect. Due to the importance of detecting machine-generated content, various methods of detection have been proposed. For instance, fine-tuning RoBERTa for the detection of machine-generated text [12] has yielded highly accurate results, including in the detection of fake tweets [13]. The high performance of RoBERTa can be attributed to its ability to capture semantic differences between AI-generated and human-generated text.

Another approach, as proposed by [14], involves deriving topological features from the attention maps of the text to classify it into either of the two classes(AI generated & Human generated) based on its structural and geometrical properties, which also aid in capturing syntactical properties. It is based on the fact that attention maps generated by the transformer model can be represented as weighted bipartite graphs and thus can be efficiently investigated with Topological Data Analysis (TDA). TDA methods are known to effectively capture surface and structural patterns in data which, are crucial to the task of detecting machine generated text.

Combining these two models and leveraging their respective strengths to develop a unified model has the potential to outperform each individually. This synthesis could yield an improved model for the detection of machine-generated text, particularly since RoBERTa currently demonstrates state-of- the-art performance in this task. Therefore, developing a model that outperforms the individual models under consideration would significantly advance the field. To achieve this, the task here is to classify a given text into either of two classes (machine-generated or human-generated) and to evaluate the performance of the aforementioned models individually and in combination to determine which approach yields the best results.

The contributions of the paper can be summarized as:

i) Synthesize a novel model (TDA-BERTa) by combining topological features derived from attention maps with RoBERTa embeddings.

ii) Analyze the performance of the individual and combined models through comparative analysis to determine the optimal model and not the improvement in performance.

Received: August 10, 2025

International Journal of Applied Mathematics
Volume 38 No. 6s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

II. Literature Review

Reference [2] introduces GROVER, a model designed to detect fake news generated by Large Language Models (LLMs) like GPT-2. It utilizes a discriminator network and adversarial training to achieve high accuracy, highlighting its potential to combat misinformation. It underscores the importance of ongoing research in this area and concludes that the most effective way to detect text generated by a Text Generation Models (TGMs) is by using that TGM itself.

Reference [12] proposes a staged release strategy for LLMs to manage social impact, emphasizing risk assessment and collaboration. It fine-tunes a RoBERT a model to detect text from large LLMs like GPT-2, aiming to prevent misinformation spread. It advocates for cautious LLM release to maximize societal benefit. It contradicts the conclusion of [2] that the most effective way to detect text generated by a TGM is by using that TGM itself.

Reference [14] introduces a novel approach to detect artificial text by analyzing the topology of attention maps generated by transformers. By leveraging techniques from topological data analysis (TDA), the method extracts topological features from these maps, which are then used to train a classifier for detecting artificial text. The approach demonstrates high accuracy on benchmark datasets, offering promising prospects for TDA's application in natural language processing tasks.

Reference [15] underscores the importance of detectors in mitigating the misuse of Text Generative Models (TGMs), which produce human-like text. Despite the critical need for reliable detectors, there is a lack of comprehensive surveys in this rapidly evolving field. It addresses this gap by providing a critical analysis of existing detection methods, focusing on English-language text, and conducting an in-depth error analysis of state-of-the-art detectors to inform future research directions.

Reference [16] highlights concerns about the rapid progress of Text Generative Models (TGMs) and their potential misuse, including the spread of fake news and misinformation. To address this, visual tools like Giant Language Model Test Room (GLTR) have been developed, leveraging simple statistical methods to detect machine-generated content. GLTR's effectiveness has been demonstrated in human-subject studies, showing a significant increase in accuracy compared to unaided detection, indicating its potential for various applications.

Reference [17] presents DetectGPT, a novel zero-shot Machine-Generated Text (MGT) detection method based on probability curvature theory. Unlike traditional methods, DetectGPT doesn't require training a separate classifier or collecting datasets. Instead, it leverages the observation that MGTs often exhibit distinct probability curvatures compared to human-written text. By analyzing the curvature of probability distributions generated by pre-trained language models, DetectGPT can effectively differentiate between MGT and human-written text without explicit training or dataset collection.

Received: August 10, 2025

International Journal of Applied Mathematics
Volume 38 No. 6s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Reference [18] evaluates a Transformer-based model's ability to differentiate between text generated by ChatGPT and human-authored content, particularly in short text scenarios. Two tests are conducted, one using ChatGPT-generated text and another with rewritten human-authored reviews. The optimized Transformer model outperforms a perplexity-based strategy, with the SHapley Additive exPlanations (SHAP) framework providing explanations for the model's decisions. Notably, ChatGPT's writing style is characterized by formality, politeness, impersonality, and a focus on general concepts.

III. Methodology

The objective is to implement and analyze individual RoBERTa and TDA-based models, as well as their combined form(TDA-BERTa), to evaluate their performance and determine which performs better. For this purpose, the task is defined as detecting whether a given text is machine- generated or not. Initially, a dataset is collected and undergoes preprocessing tailored to each approach. Two pipelines are established for preprocessing: one for RoBERTa and one for TDA. The RoBERTa model is implemented by taking a pre-trained RoBERTa model (base) from Hugging Face, and further training it and making it capable of text classification by adding a classification head. The TDA features considered here include topological features based on properties like Betti numbers of attention maps generated from the text, and template(distance-to-pattern) features, which capture attention patterns by calculating the Frobenius distance between tokens of different types. Barcode features, as used in [14], are omitted due to their high computational requirements and negligible impact on performance, as noted in [14]. Once the TDA features are derived and stored, a classifier is trained on these features to perform prediction. Unlike [14] which employed a logistic regression model as classifier, the one used here is an Artificial Neural Network(ANN) as it can capture information from the features more effectively. The model combining these two approaches is created by combining the embeddings produced by RoBERTa with the topological features derived using TDA.

A. Datasets Used

The Hugging Face library was used to gather datasets [19] for the study. The primary dataset (table I) contained two sub- sets: ”research_abstracts_labeled” and ”wiki_labeled.” These datasets contained labeled research abstracts and Wikipedia articles, both human-generated and AI-generated. The datasets were combined using the library's functionality, resulting in a final dataset with 320,000 rows. The label ”1” indicates machine-generated text, while ”0” represents human-generated text. Due to the high computational resource requirements in extracting the TDA features, the training dataset is taken as a subset (10,000 samples) of the larger dataset [19]. For testing purposes, a subset of [19] (2,000 samples) will be kept aside as a testing set. Henceforth all mentions of training and testing sets shall refer to these subsets respectively. An independent dataset is also custom-built to assess whether the model generalizes well to unseen data with out-of-domain characteristics.

Received: August 10, 2025

International Journal of Applied Mathematics
Volume 38 No. 6s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Table I PRIMARY DATASET CONSIDERED

Dataset	Features	Number of Rows
Wiki-labelled Dataset	[title, label, text, word count]	300,000
Research-labelled Dataset	[title, label, text, word count]	20,000
Combined Dataset	[label, Text]	320,000
Training Set Used	[label, Text]	10,000
Testing Set Used	[label, Text]	2,000

The custom dataset (table II) containing human-generated and AI-generated text was collected to assess whether the model generalizes well to unseen data with out-of-domain characteristics. Human-generated text was sourced from articles and books published before 2017, predating the advent of large language models (LLMs). This ensured that the human text did not contain influences from LLMs. On the other hand, AI- generated text was obtained by prompting ChatGPT with various inputs to generate responses. The final dataset comprised 200 rows, with 100 examples of each: 100 human-generated text samples and 100 AI-generated text samples. This balanced dataset was designed to test the model's performance on an independent set of data.

Table Ii Custom-Built Independent Dataset

Text Type	Collection Method
Human-generated	Sourced from articles and books published before 2017 to avoid influence from large language models (LLMs)
AI-generated	Obtained by prompting ChatGPT with various prompts to generate responses

B. RoBERTa Model

1) ***Data Preprocessing:*** The preprocessing for RoBERTa primarily focuses on tokenization and conversion of raw text into token IDs suitable for input to the model. The input text is encoded by tokenizing it using the RoBERTa tokenizer and then subsequently converting the tokenized text into token IDs, incorporating special tokens like **[CLS]** and **[SEP]** to delineate the beginning and end of sequences. Additionally, attention masks are generated to indicate the presence of valid tokens within each input sequence.

Received: August 10, 2025

International Journal of Applied Mathematics
Volume 38 No. 6s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

*2) **Model Archtecture:*** For utilizing RoBERTa model architecture in AI text detection, the pretrained RoBERTa- base model is obtained from Hugging Face. The input data, consisting of tokenized word IDs, attention masks, and token type IDs, is passed through multiple layers of the RoBERTa model, resulting in embeddings as output. These layers implement self-attention mechanisms, facilitating the analysis of contextual relationships and dependencies. Following this, flatten layers convert the embeddings into a one-dimensional array. Dropout layers are subsequently applied to regulate overfitting by randomly deactivating a proportion of neuronsduring training. The processed data then flows through fully connected layers equipped with Rectified Linear Unit (ReLU) activation functions, allowing for the extraction of hierarchical representations from the text, which are crucial for classification tasks. Finally, a softmax layer is employed to generate probability distributions across various categories, enabling the model to effectively classify text into distinct classes. The architecture of the model developed is shown in fig. 1.

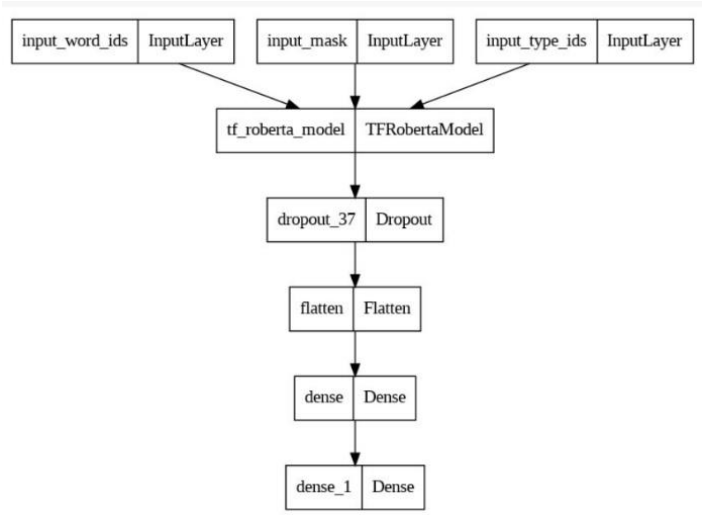


Fig. 1. RoBERTa Model Architecture Used For Detection of Machine-Generated Text.

The contextualized representations obtained as embeddings using RoBERTa's advanced capabilities aid in accurately detecting machine-generated text.

3) Training: The model is trained using Google's Tensor Processing Unit (TPU) for fast training. The training set after preprocessing is fed to the model in order to train it. To monitor validation accuracy and prevent overfitting, a portion of the training set (20%) is designated as the validation set. Early Stopping, with a monitoring period of 3 epochs, is also employed during training. The various parameters and hyperparameters used are listed in table III.

TABLE III SUMMARY OF ROBERTA MODEL TRAINED

rameter / Hyperparameter	**lue**
umber of epochs	
tch size	

Received: August 10, 2025

International Journal of Applied Mathematics
Volume 38 No. 6s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

earning rate	-5
ptimizer	dam
egularization	2 regularization with factor 5e-4
oss function	parse categorical crossentropy
etrics	ccuracy
ainable Parameters	4,978,050

C. TDA

Text classification using TDA involves first generating attention matrices for the input text, storing it and then subsequently deriving the required TDA features from it. Following the approach in [14], BERT model is used to generate attention matrices for the text after having tokenized (token size taken

is 128 as increasing token size leads to a considerable increase in computation time and memory requirements) it. The TDA features derived are :

i) Topological features like Betti numbers of the undirected graph & the number of edges, strongly connected com- ponents and simple directed cycles in the directed graphs formed from the attention maps.

ii) Template(distance-to-pattern) features, which capture attention patterns by calculating the Frobenius distance

Template Features To derive distance-to-pattern features from attention matrices, the attention graph is first represented as an incidence matrix $A = (a_{ij})$, where $a_{ij} = 1$ for all edges (ij) in E and 0 otherwise. Two graphs, E and E', are considered with the same set of vertices and their corresponding incidence matrices A and A'. The distance between these graphs is calculated using the Frobenius norm of the difference between their incidence matrices, normalized by the norms of the matrices (eq. (1)) [14]:

between tokens.

$$d(E, E') = \frac{\|A - A'\|_F}{\|A\|_F^2 + \|A'\|_F^2} = \frac{\sum_{ii}(a_{ii} - a'_{ij})^2}{\sum_{ij}(a_{ij})^2 + \sum_{ij}(a'_{ij})^2} \quad (1)$$

*1) **Data Preprocessing:*** Preprocessing data for TDA [14] involves additional steps to prepare the text data for subsequent topological analysis. The model begins by performing text cleaning which removes entity mentions and corrects errors present in the text. This ensures that the input data is free from noise and inconsistencies, facilitating accurate analysis. Subsequently, the cleaned text is tokenized using BERT tokenizer to generate input sequences suitable for computing attention matrices required for TDA.

Received: August 10, 2025

International Journal of Applied Mathematics
Volume 38 No. 6s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

2) ***Feature Extraction:*** The TDA feature extraction process begins with computing attention weights from the pre- processed text. These attention weights represent the significance of token interactions within the text. These attention weights are then utilized to construct attention
matrices, where each element represents the attention from one token to another. Subsequently, various topological and Template(distance-to-pattern-based) features are derived from these attention weights, capturing distinct aspects such as self- attention, attention to neighboring tokens, and attention to specific token types like commas and periods. These features provide valuable insights into the structural and semantic characteristics of the text, facilitating deeper analysis and interpretation within the context of TDA.

Topological Features: Topological features are derived from the attention matrices [14] by first fixing a set of thresholds $T = \{t_1, t_2, ..., t_k\}$, where $0 < t_1 < t_2 < ... < t_k < 1$, determining the strength of connections between vertices in the attention graph. For each attention head h and corresponding weights $W_{attn} = (w_{attnij})$, a weighted directed graph $h_s(t)$ is constructed for a given text sample s and each threshold level t in T . This graph captures the relationships between different parts of the text sample based on attention weights. Various topological features are then extracted from the constructed graphs, including the first two Betti numbers of the undirected graph $h_s(t)$, and the number of edges, strongly connected components, and simple directed cycles in the directed graph $h_s(t)$. To obtain the complete set of topological features for a given text sample s and attention head h, the features extracted from all threshold levels t are concatenated. Each attention head h in the set HM of chosen attention heads is iterated over, and for each threshold level t in T ,the attention graph $h_s(t)$ is calculated and extracts the corresponding topological features $f(h_s(t))$. Finally, the concatenated features for all attention heads and threshold levels are returned, representing the attention patterns of the model for a given text sample. This distance metric ranges between 0 and 1. For un- weighted graphs, the distance can be expressed as (eq. (2)) [14]:

$$d(E, E') = \frac{|E \Delta E'|}{|E| + |E'|} \qquad (2)$$

where $E \Delta E'$ represents the symmetric difference of sets E and E'.

Distances from the given graph to specific attention patterns as graph features is then considered. These patterns include attention to the previous token, attention to the next token, attention to the [CLS]-token (denoting the beginning of the text), attention to the [SEP]-token (used to separate segments), and attention to punctuation marks. Each distance-to-pattern feature $d_i(E)$ is calculated as the distance $d(E, P_i)$, where P_i represents the specific attention pattern.

D. Training

The derived features are stored as NumPy arrays once computed. These saved features are

Received: August 10, 2025

International Journal of Applied Mathematics
Volume 38 No. 6s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

then utilized to train an Artificial Neural Network (ANN), enabling it to make predictions based on the extracted features. The ANN hence effectively classifies the text into the two predefined classes. The training set is further split into a validation set(20% of training set) and Early stopping is implemented to monitor validation accuracy. The various parameters and hyperparameters used are listed in table IV.

TABLE IV SUMMARY OF TDA-BASED ANN TRAINED

Specification	**Value**
Model Architecture	Sequential
Layers	Dense, Dropout, Dense, Dropout, Dense
Activation Functions	ReLU (Dense layers), Softmax (Output layer)
Regularization	L2 regularization with factor 5e-4
Trainable Parameters	46,059,354
Optimizer	Adam
Learning Rate	1e-5
Loss Function	Sparse categorical crossentropy
Metrics	Accuracy
Number of Epochs	100
Batch Size	32

E. TDA-BERTa Model

*1) **Data Preprocessing:*** The data is fed into two pipelines one that preprocesses data for RoBERTa and another for TDA. They follow the same functioning as when used individually.

F. Model Architecture

The preprocessed text for RoBERTa, including input IDs, token type IDs (None for RoBERTa), and attention masks, is passed through a pretrained RoBERTa model to generate con- textual embeddings. Simultaneously, TDA features are derived from the preprocessed text using previously discussed methods and then concatenated into a single feature array. These processed embeddings and features are then concatenated. The model's classification head comprises a dropout layer and two dense layers, facilitating the classification of the text as either "machine-generated" or "human-generated". The architecture of the model developed is shown in fig. 2.

Received: August 10, 2025

International Journal of Applied Mathematics
Volume 38 No. 6s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

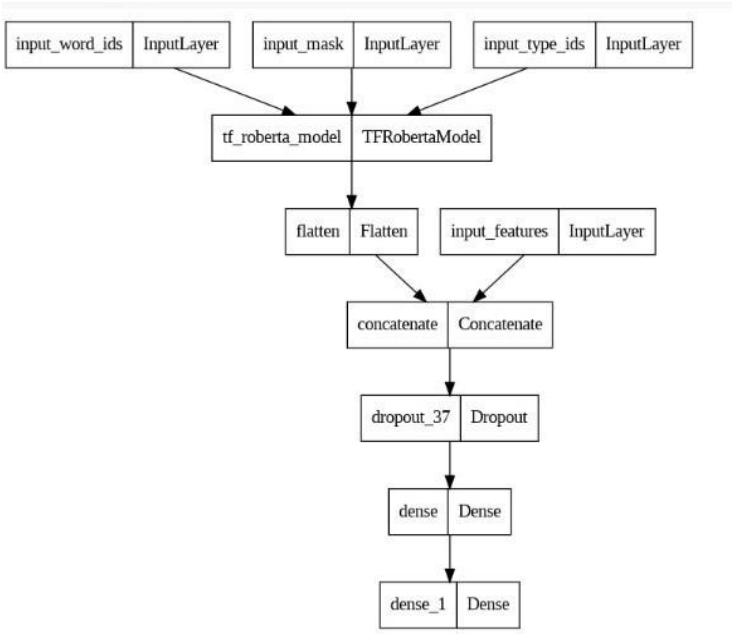


Fig. 2. TDA-BERTa Model Architecture

1) ***Training:*** The model is trained using Google's Tensor Processing Unit (TPU) for fast training. The training data is fed to the two preprocessing pipelines to facilitate its input into the model. The saved TDA features are also loaded from memory and passed into the model. To monitor validation accuracy and prevent overfitting, a portion of the training set (20%) is designated as the validation set. Early Stopping, with a monitoring period of 3 epochs, is also employed during training. The various parameters and hyperparameters used are listed in table V.

IV. Results And Discussion

Comparative analysis of the performance of the three models under study—RoBERTa, TDA-based ANN, and the com- bination of the two (TDA-BERTa)—is performed by testing them on the testing set and the independent dataset. The results reveal that TDA-BERTa outperforms both the individual models on almost all the metrics of performance. It has the highest values for accuracy, recall, precision, F1 Score, and ROC-AUC Score (fig. 3 & fig. 4). This demonstrates that TDA-BERTa not only performs better on testing set derived from its native dataset (table VI) but also generalizes well enough to outperform its constituent models on unseen and out-of-domain data (table VII). The higher AUC score (fig. 5) indicates that the TDA-BERTa is better at distinguishing between the positive and negative classes, making it more effective for the detection of machine generated text from human generated text.

TABLE V Summary of TDA-BERTa model training

Parameter / Hyperparameter	Value
Number of epochs	10

Received: August 10, 2025

International Journal of Applied Mathematics
Volume 38 No. 6s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Batch size	64
Learning rate	1e-5
Regularization	L2 regularization with factor 5e-4
Optimizer	Adam
Loss function	Sparse categorical crossentropy
Metrics	Accuracy
Trainable parameters	176,526,338

TABLE VI PERFORMANCE METRICS FOR MODELS ON THE TESTING SET

Metric	RoBERTa	TDA-based ANN	TDA-BERTa
Accuracy	0.9920	0.9420	**0.9965**
Precision	0.9892	0.9523	**0.9970**
Recall	0.9950	0.9315	**0.9960**
F1 Score	0.9921	0.9418	**0.9965**
ROC-AUC Score	0.9996	0.9851	**0.9999**

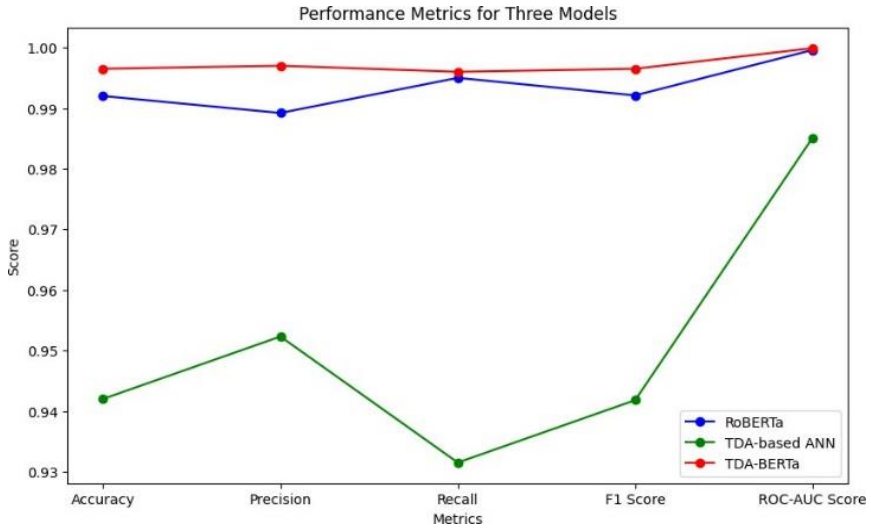


Fig. 3. Comparative Analysis of Models on Testing Set

TABLE VII PERFORMANCE METRICS FOR MODELS ON THE INDEPENDENT DATA SET

Metric	RoBERTa	TDA-based ANN	Combined Model
Accuracy	0.8141	0.7487	**0.8693**
Precision	0.7299	0.7119	**0.7937**

Received: August 10, 2025

International Journal of Applied Mathematics
Volume 38 No. 6s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Recall	**1.0000**	0.8400	**1.0000**
F1 Score	0.8439	0.7706	**0.8850**
ROC-AUC Score	0.9752	0.7784	**0.9971**

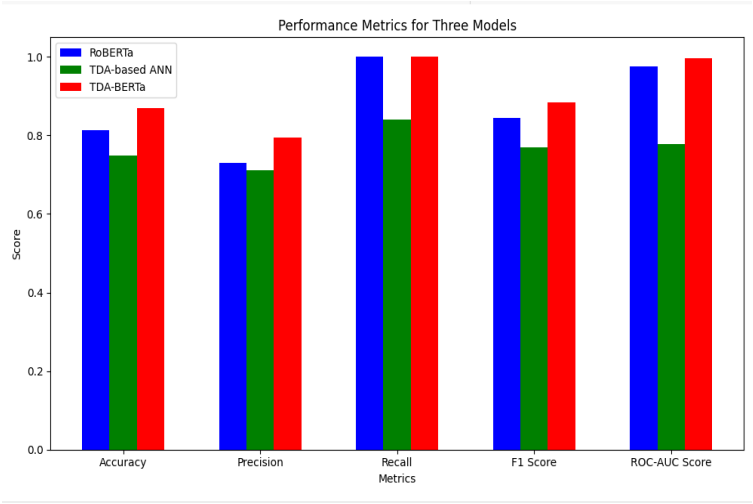
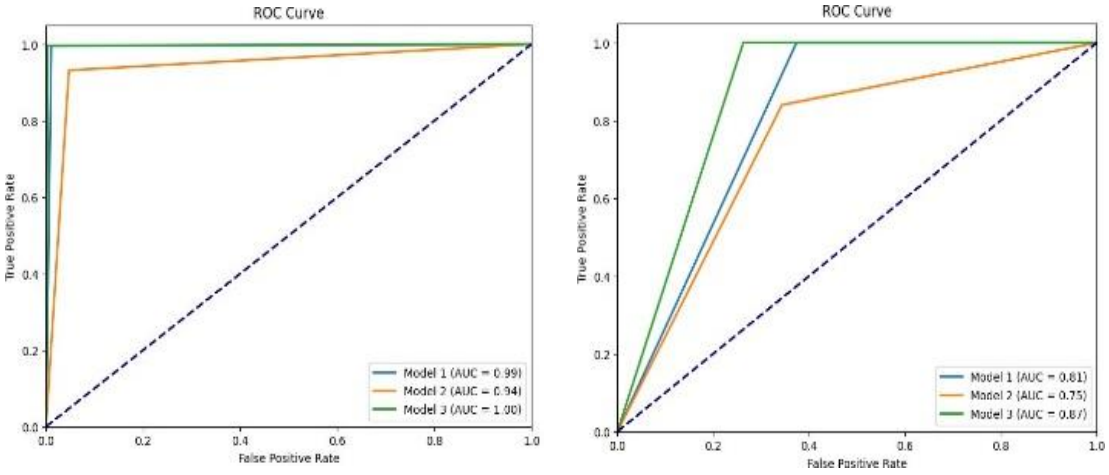


Fig. 4. Comparative Analysis of Models on Independent Set



(a) (b)

Fig. 5. (a) AUC Scores of Models on The Testing Set (b)AUC Scores of Models on The Independent Set

V. Conclusion And Future Work

The primary aim of this paper is to assess the performance of a model that integrates topological features with contextualized embeddings from RoBERTa, with the goal of advancing the field of Machine-Generated Text Detection. The study demonstrates promising results, as the developed model, TDA- BERTa, outperforms the individual approaches upon which it is built, as evidenced by the conducted testing. Particularly notable is its enhanced ability to generalize to unseen data. However, it is acknowledged that for the model to be

Received: August 10, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

considered exceptional, further testing on larger datasets is necessary, given the relatively small size of the out-of-domain independent dataset used in this study. Future avenues for exploration include increasing token length, employing RoBERTa instead of BERT for deriving TDA features, and exploring alternative methods for integrating embeddings and topological features beyond simple concatenation. With increased computational resources, the application of this approach could be further investigated.

REFERENCES

[1] Keskar, Nitish Shirish, et al. ”Ctrl: A conditional transformer language model for controllable generation.” arXiv preprint arXiv:1909.05858 (2019).

[2] Zellers, Rowan, et al. ”Defending against neural fake news.” Advances in neural information processing systems 32 (2019).

[3] Vaswani, Ashish, et al. ”Attention is all you need.” Advances in neural information processing systems 30 (2017).

[4] Devlin, Jacob, et al. ”Bert: Pre-training of deep bidirectional trans- formers for language understanding.” arXiv preprint arXiv:1810.04805 (2018).

[5] Liu, Yinhan, et al. ”Roberta: A robustly optimized bert pretraining approach.” arXiv preprint arXiv:1907.11692 (2019).

[6] Radford, Alec, et al. ”Language models are unsupervised multitask learners.” OpenAI blog 1.8 (2019): 9.

[7] Brown, Tom, et al. ”Language models are few-shot learners.” Advances in neural information processing systems 33 (2020): 1877-1901.

[8] Achiam, Josh, et al. ”Gpt-4 technical report.” arXiv preprint arXiv:2303.08774 (2023).

[9] Cooke, Nicole A. Fake news and alternative facts: Information literacy in a post-truth era. American Library Association, 2018.

[10] Vosoughi, Soroush, Deb Roy, and Sinan Aral. ”The spread of true and false news online.” science 359.6380 (2018): 1146-1151.

[11] Adelani, David Ifeoluwa, et al. ”Generating sentiment-preserving fake online reviews using neural language models and their human-and machine-based detection.” Advanced information networking and applications: Proceedings of the 34th international conference on advanced information networking and applications (AINA-2020). Springer Inter- national Publishing, 2020.

[12] Solaiman, Irene, et al. ”Release strategies and the social impacts of language models.” arXiv preprint arXiv:1908.09203 (2019).

[13] Fagni, Tiziano, et al. ”TweepFake: About detecting deepfake tweets.” Plos one 16.5 (2021): e0251415.

Received: August 10, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

[14] Kushnareva, Laida, et al. ”Artificial text detection via examining the topology of attention maps.” arXiv preprint arXiv:2109.04825 (2021).

[15] Jawahar, Ganesh, Muhammad Abdul-Mageed, and Laks VS Laksh- manan. ”Automatic detection of machine generated text: A critical survey.” arXiv preprint arXiv:2011.01314 (2020).

[16] Gehrmann, Sebastian, Hendrik Strobelt, and Alexander M. Rush. ”Gltr: Statistical detection and visualization of generated text.” arXiv preprint arXiv:1906.04043 (2019).

[17] Mitchell, Eric, et al. ”Detectgpt: Zero-shot machine-generated text detection using probability curvature.” International Conference on Machine Learning. PMLR, 2023.

[18] Mitrovic´, Sandra, Davide Andreoletti, and Omran Ayoub. ”Chatgpt or human? detect and explain. explaining decisions of machine learn- ing model for detecting short chatgpt-generated text.” arXiv preprint arXiv:2301.13852 (2023).

[19] Sivesind, Nicolai Thorer, and Andreas Bentzen Winje. ”Human-vs- Machine.” Hugging Face, 2023.

[20] Bakhtin, Anton, et al. ”Real or fake? learning to discriminate machine from human generated text.” arXiv preprint arXiv:1906.03351 (2019).

[21] Ippolito, Daphne, et al. ”Automatic detection of generated text is easiest when humans are fooled.” arXiv preprint arXiv:1911.00650 (2019).

[22] Deng, Yuntian, et al. ”Residual energy-based models for text generation.” arXiv preprint arXiv:2004.11714 (2020).

[23] OpenAI, T. B. ”Chatgpt: Optimizing language models for dialogue. OpenAI.” (2022).

[24] He, Xinlei, et al. ”Mgtbench: Benchmarking machine-generated text detection.” arXiv preprint arXiv:2303.14822 (2023).

[25] Islam, Niful, et al. ”Distinguishing Human Generated Text From Chat- GPT Generated Text Using Machine Learning.” arXiv e-prints (2023): arXiv-2306.

[26] Crothers, Evan, Nathalie Japkowicz, and Herna L. Viktor. ”Machine- generated text: A comprehensive survey of threat models and detection methods.” IEEE Access (2023).

[27] Ibrahim, Karim. ”Using AI-based detectors to control AI-assisted plagia- rism in ESL writing: “The Terminator Versus the Machines”.” Language Testing in Asia 13.1 (2023): 46.

[28] Wang, Zecong, et al. ”Implementing BERT and fine-tuned RobertA to detect AI generated news by ChatGPT.” arXiv preprint arXiv:2306.07401 (2023).

[29] Gambini, Margherita, et al. ”Detecting Generated Text and Attributing Language Model Source with Fine-tuned Models and Semantic Under- standing.” (2023).

Received: August 10, 2025