FMDB *Transactions on Sustainable Technoprise Letters*

A Dual Pipeline AI Framework for Real Time Detection of Phishing and Financial Fraud in Fraud GPT

Y. Ajitha[1,*], P. Sudha[2], G. Gowthami[3], C. Shanthini[4], Mohammad Ayaz Ahmad[5]

[1]Department of Computer Science and Engineering, SRM Institute of Science and Technology, Ramapuram, Chennai, Tamil Nadu, India.
[2]Department of Business Administration, Dhaanish Ahmed College of Engineering, Chennai, Tamil Nadu, India.
[3]Department of Computer Science, St. Francis De Sales College (Autonomous), Electronic City, Bangalore, Karnataka, India.
[4]Department of Computer Science, Bharath Institute of Higher Education and Research, Chennai, Tamil Nadu, India.
[5]Department of Mathematics, Physics and Statistics, University of Guyana, Georgetown, Guyana, South America.
ay1623@srmist.edu.in[1], sudhap@dhaanishcollege.in[2], Gowthami.ramya@gmail.com[3], rajeshshanthini@gmail.com[4], mohammad.ahmad@uog.edu.gy[5]

Abstract: GenAI can create original writing, music, movies, and images and collaborate with humans and other AI models. It is one of our most revolutionary creations. Firms, academics, developers, and consumers have profited from increased productivity, innovation, and efficiency. Generative AI has challenges like any disruptive technology. It increases creativity and efficiency but creates ethical, disinformation, data privacy, and prejudice concerns. Recently developed AI tool fraudGPT became popular in the dark market where hackers may automate fraud emails, SMS, manufacture fake text messages, mimic a well-known organisation, and make hacking easy. This traps many innocent people in hackers' webs. AI should be used to detect fraudGPT-related phishing, smishing, social engineering, and financial crimes, according to this study. Python, transformer-based NLP model, anomaly detection model, binary classification, BERT, PyTorch, transformers, and sci-kit-learn are used. The suggested method detects phishing emails with 92% accuracy and transaction fraud with 94% accuracy in trials. To test the model's resilience, precision (91%), recall (90%), and F1-score (90.5%) were used for phishing identification and precision (93%), recall (92%), and F1-score (92.5%) for transaction fraud detection. Real-time monitoring and AI-based detection proactively take on AI-driven cyber risks. This work improves AI-powered cybersecurity and informs financial institutions and policymakers. AI researchers, cybersecurity experts, and regulatory bodies must collaborate to combat rogue AI technologies like FraudGPT.

Keywords: Fraudgpt and Phishing Detection; Natural Language Processing; Transformer Models; BERT and AI-Driven Cybercrime; Pytorch and AI Tools; Text-Based; Transaction-Based; Fraudulent Activities.

Received on: 29/06/2024, **Revised on:** 12/09/2024, **Accepted on:** 01/12/2024, **Published on:** 07/12/2024

Journal Homepage: https://www.fmdbpub.com/user/journals/details/FTSTPL

DOI: https://doi.org/10.69888/FTSTPL.2024.000335

Cite as: Y. Ajitha, P. Sudha, G. Gowthami, C. Shanthini, and M. A. Ahmad, "A Dual Pipeline AI Framework for Real Time Detection of Phishing and Financial Fraud in Fraud GPT," *FMDB Transactions on Sustainable Technoprise Letters*., vol. 2, no. 4, pp. 200–211, 2024.

1. Introduction

[*]Corresponding author.

The increasing advancement of generative artificial intelligence (AI) has revolutionised various industries, but it has also brought about significant challenges, particularly in the area of cybersecurity. Among these is the emergence of FraudGPT, an AI tool created on the dark market that is specifically designed to facilitate fraudulent activities such as phishing, social engineering, and financial fraud. FraudGPT makes it easy for generative AI to create highly convincing phishing emails, fake websites, and fraudulent transaction patterns, making it a potential weapon for cybercriminals. This tool has become increasingly popular among hackers and illegal cybercrime networks, posing a severe threat to individuals, businesses, and financial institutions worldwide. The FraudGPT is more dangerous to the financial sector. Phishing is now targeting banking customers, and fraudulent transactions and identity theft are on the rise, leading to substantial financial losses and reputational damage.

This paper proposes a novel approach to counteract the misuse of FraudGPT by developing a robust system to detect which of the messages/emails/SMSes is fraudulent or legitimate. The proposed system uses natural language processing (NLP) techniques, specifically fine-tuned transformer models like BERT, to detect phishing emails and other fraudulent text-based content. Additionally, the system incorporates anomaly detection algorithms to identify suspicious patterns in financial transactions. By combining these techniques, the proposed method provides a comprehensive solution for real-time fraud detection and prevention. The contributions of this research are threefold. First, we present a detailed analysis of the threats posed by FraudGPT and similar malicious AI tools. Second, we propose a scalable and efficient AI-based system for detecting and mitigating these threats. Third, we evaluate the effectiveness of the proposed system using real-world datasets, demonstrating its ability to achieve high accuracy in identifying fraudulent activities.

2. Objective

The main objective of this paper is to develop a counter back to the fraudulent activities that have surfaced due to the emergence of a new AI tool called the FraudGPT, and this paper analyzes the threat landscape by investigating the mechanisms and capabilities of FraudGPT and similar tools in facilitating phishing, social engineering, and financial fraud. By implementing natural language processing (NLP) techniques, including transformer-based models like BERT, alongside anomaly detection algorithms, the research aims to design a robust framework for real-time fraud detection. The proposed system will be rigorously evaluated using benchmark datasets for phishing emails and financial transactions, with a focus on key performance metrics such as accuracy, precision, recall, and F1-score. The proposed framework tries to take into account all the SMS, emails, and transactions and figure out if they are fraudulent or legitimate, thus reducing the chances of malicious activities by hackers. This process is developed by dividing the entire proposed system into two categories.

- **Text-Based Fraud Detection:** Uses a BERT-based transformer model to classify phishing emails or fraudulent messages. Process text inputs through tokenisation, embedding, and classification.
- **Transaction-Based Fraud Detection:** Uses statistical anomaly detection to identify unusual patterns in financial transactions. Computes anomaly scores using methods like Mahalanobis distance or Z-score.

3. Literature Review

Falade [1] explores the evolving threat landscape posed by generative AI models such as ChatGPT, FraudGPT, and WormGPT in social engineering attacks. The study highlights how adversarial AI can automate and enhance phishing campaigns, making them more convincing and difficult to detect. By leveraging natural language processing capabilities, these AI tools can craft highly personalised messages, bypassing traditional security filters. Krishnan [2] investigates the emergence of FraudGPT as a malicious variant of ChatGPT designed explicitly for cybercriminal activities. The article details how FraudGPT is utilised to generate phishing emails, scam messages, and deepfake content, increasing the sophistication and success rate of cyberattacks. Unlike legitimate AI models, FraudGPT operates on dark web forums, providing threat actors with an automated tool for executing large-scale social engineering attacks.

Kumar and Gupta [3] provide an in-depth analysis of the evolving landscape of social engineering attacks facilitated by artificial intelligence (AI). The authors highlight how AI technologies, particularly natural language processing models, have been co-opted by cybercriminals to craft sophisticated phishing emails, spear-phishing campaigns, and automated impersonation attacks. These AI-driven tactics significantly enhance the plausibility and success rates of social engineering exploits. Bryce et al. [4] delve into the complex role of Large Language Models (LLMs) within the cybersecurity landscape, highlighting both the potential threats they pose and the defensive capabilities they offer. The authors discuss how LLMs, due to their advanced text generation capabilities, can be exploited by malicious actors to create sophisticated phishing emails, generate malware code, and develop deceptive chatbots, thereby lowering the entry barriers for cyberattacks.

Bryce et al. [5] address the emerging threats posed by generative AI technologies in the cybersecurity domain. The authors highlight how advancements in AI, particularly in natural language processing and content generation, have been exploited by malicious actors to create sophisticated phishing emails, deepfakes, and other deceptive content that traditional security measures struggle to detect. Falade [6] investigates the dual-edged nature of generative AI models—such as ChatGPT, FraudGPT, and WormGPT—in the context of social engineering attacks. The study highlights how these advanced AI systems, while offering numerous benefits, have been co-opted by cybercriminals to enhance the sophistication and effectiveness of their malicious activities. The U.S. Department of Health and Human Services [7] examines the increasing role of artificial intelligence in cyber threats, particularly phishing attacks targeting the Healthcare and Public Health (HPH) sector. The report highlights how AI-driven phishing techniques have become more sophisticated, making detection and mitigation more challenging.

Ahmadi [8] investigates the transformative role of OpenAI technologies in enhancing fraud detection mechanisms within the financial sector. The study emphasises the escalating threat posed by fraudulent activities, with global card fraud losses reaching $32.34 billion in 2021—a 14% increase from the previous year. Provides an in-depth analysis of FraudGPT, an AI-driven tool designed for malicious cyber activities. Kalousis [9] in the article highlights how AI-powered tools, originally developed for automation and efficiency, are now being exploited by cybercriminals for fraudulent activities, including phishing attacks, data breaches, and automated malware generation. Bello and Olufemi [10] provide a comprehensive analysis of the integration of artificial intelligence (AI) in fraud prevention, highlighting various techniques, applications, challenges, and future opportunities. The study emphasises the critical role of AI in enhancing the accuracy, efficiency, and scalability of fraud detection systems across multiple sectors.

4. Proposed method

The proposed method is an AI model that is specifically designed to identify the malicious activities that have emerged due to FraudGPT. Some of the activities include phishing, spam emails, fake text messages, fake calls, impersonating a well-known company, etc. This proposed method uses the implementation of NLP techniques, including the BERT model, which is used for text classification, so that it can accurately detect fraud and legitimacy. Also, anomaly detection is used for transaction-based detection, which can be implemented using some of the Python libraries, for transaction-based numeric features like transaction amount, time, and location are taken into account so as to detect fraudulent messages. These are then normalised to get accurate results. After detecting both text-based and transaction-based detection, they are passed on to the decision tree, which will combine the outputs of both pipelines and give a unified output. The text-based and transaction-based detection are as follows. The outputs from both pipelines are combined in a decision layer to provide a unified classification result.

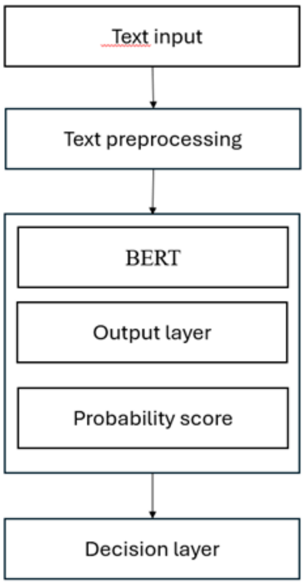


Figure 1: Block representation for text-based detection pipeline

Figure 1 represents the flow of implementing the text-based detection pipeline, where first the text is taken as input, and then it is pre-processed. Based on the training data, it will categorise the text. Now, using the BERT module, which is chosen to categorise the text as fraudulent or legitimate, will give the output. The probability score is used to represent the likelihood that a given text is fraudulent (0 or 1), which is given by the BERT module, and finally, the decision tree that will give the unified output.

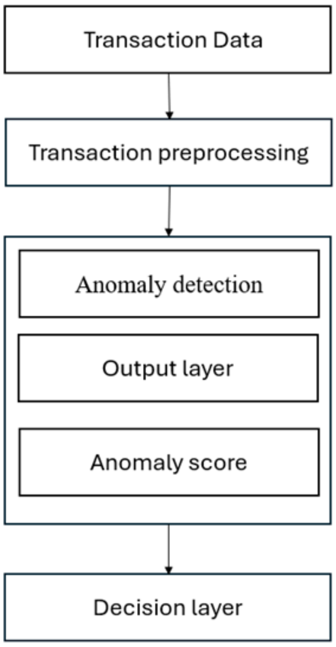


Figure 2: Block representation for transaction-based detection pipeline

Figure 2 represents the flow of implementing the transaction-based detection pipeline, where first the transaction is taken as input and then it is preprocessed. Based on the training data, it will categorise the transaction. Now, using the Anomaly detection module, which is chosen to categorise the transaction as fraudulent or legitimate, will give the output. The anomaly score is used to detect how far a given transaction has deviated from a regular transaction based on a given history, and finally, the decision tree will give the unified output.

5. Data Collection

The quality, diversity, and relevance of the data directly impact the system's performance and ability to detect fraudulent activities effectively. The data collection for both text-based and transaction-based detection is given below:

Text-based data collection: The data is collected to train and evaluate the BERT model for detecting phishing emails, social engineering messages and other fraudulent content. The source of our data is a public database called Kaggle, where the phishing emails and spam SMS datasets were collected and trained using Python. Phishing emails are designed to trick the recipients into revealing sensitive information like passwords, credit card information, and OTP, while legitimate emails are normal, non-fraudulent emails that do not seek any personal information unless and until necessary. So, our output is to check if the given input by the user is legitimate or fraudulent based on the training model.

Algorithm for Text-Based Fraud Detection Pipeline

```
1. Load Pre-trained BERT Model and Tokenizer
   - model = LoadBERTModel ()
   - tokenizer = LoadBERTTokenizer ()
2. Preprocess Input Text
   - input_text = "Congratulations! You have won a $1000 gift card. Click here to claim."
   - tokenized_input = TokenizeText (input_text, tokenizer)
   - input_ids = ConvertTokensToIDs(tokenized_input)
   - attention_mask = CreateAttentionMask(tokenized_input)
3. Perform Inference Using BERT Model
   - outputs = model (input_ids, attention_mask)
   - fraud_probability = Sigmoid(outputs)
4. Classify as Phishing or Legitimate
   - threshold = 0.5
   - if fraud_probability > threshold:
      classification = “Phishing Email”
   - else:
      classification = “Legitimate Email”
5. Return Classification and Fraud Probability
 - return classification, fraud_probability
```

Transaction-based data collection: The data is collected to train and evaluate the anomaly detection module for identifying fraudulent financial transactions. The datasets are collected from a public source called Kaggle, where Fraud transaction datasets were taken. The parameters that are in use are amount, time, location, and card details that are taken into account to train the model and then see which transactions are fraudulent and legitimate. So, when the input is given by the user, the model checks if the transaction is legitimate or not based on the parameters it is trained on and then produces the final output.

```
1. Load Transaction Data
   - transaction_data = LoadTransactionData ()
   - features = ExtractFeatures(transaction_data)
2. Preprocess Transaction Data
   - normalized_features = NormalizeFeatures(features)
3. Compute Anomaly Scores
   - mean = ComputeMean(normalized_features)
   - covariance_matrix = ComputeCovariance(normalized_features)
   - anomaly_scores = ComputeMahalanobisDistance (normalized_features, mean, covariance_matrix)
4. Flag Fraudulent Transactions
   - threshold = 3.0 # Example threshold for anomaly detection
   - for each transaction in transaction_data:
      if anomaly_scores[transaction] > threshold:
         classification = “Fraudulent Transaction”
      else:
         classification = “Legitimate Transaction”
5. Return Classification and Anomaly Scores
   - return classification, anomaly_scores
```

6. Results and Discussion

The proposed AI model for fraud detection due to FraudGPT was evaluated on both text-based detection and transaction-based detection. The results show how the given input is classified as fraudulent or legitimate and also give a score to show by how much percentage it is fraudulent or legitimate. The parameters for the same are as follows:

Text-based pipeline:
Fraud detection:
Method: BERT
Output: Binary (0 or 1) – true or false
Training parameters:
Batch size: 32
Epochs:10
Optimizer: adam (learning rate = 2e-5)

Transaction based pipeline:
Anomaly detection:
Method: Mahalanobis distance
Output: binary (0 or 1)- true or false
Training parameters:
Batch number: 64
Epochs: 20
Optimizer adam (learning rate=1e-4)

Graphical Representation of Performance Matrices: The datasets are divided into 5 groups to get the graph for performance matrices, where each group has 200 sets of data groups, and their respective performance matrices are created

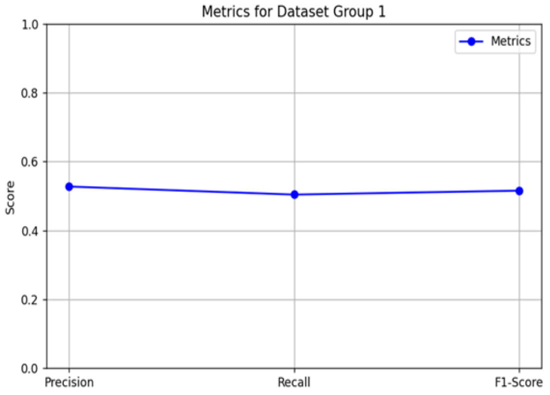


Figure 3: Group 1 performance matrices

Figure 3, the performance matrices are calculated for the first group consisting of 200 datasets, and then the resultant values include Precision = 0.5278, Recall = 0.5044, and F1-Score = 0.5158 (Table 1).

Table 1: Model evaluation metrics

Performance Matrices	**Value**
Precision	0.5278
Recall	0.5044
Fi-Score	0.5158

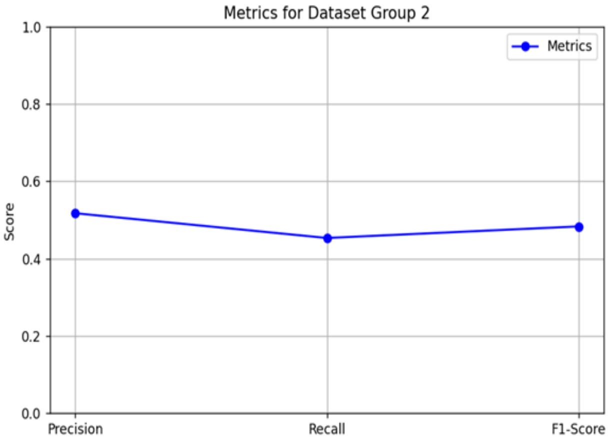


Figure 4: Group 2 performance matrices

Figure 4, the performance matrices are calculated for the second group consisting of 200 datasets, and then the resultant values include Precision = 0.5176, Recall = 0.4536, and F1-Score = 0.4835 (Table 2).

Table 2: Performance metrics summary

Performance Matrices	**Value**
Precision	0.5176
Recall	0.4536
Fi-Score	0.4835

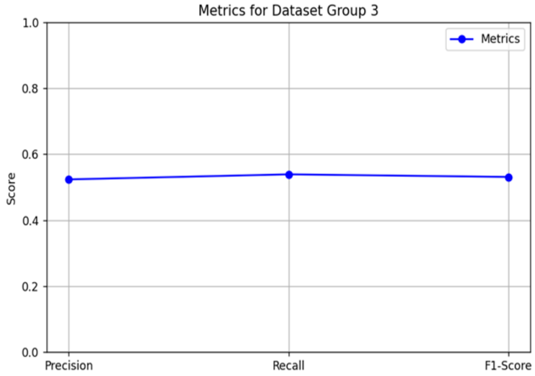


Figure 5: Group 3 performance matrices

Figure 5, the performance matrices are calculated for the third group consisting of 200 datasets, and then the resultant values include Precision = 0.5238, Recall = 0.5392, and F1-Score = 0.5314 (Table 3).

Table 3: Model performance metrics

Performance Matrices	Value
Precision	0.5283
Recall	0.5392
Fi-Score	0.5314

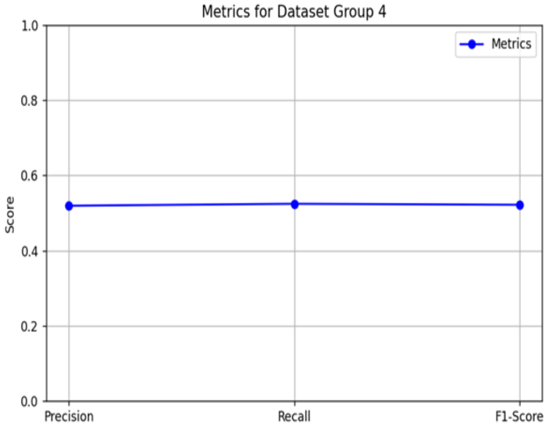


Figure 6: Group 4 performance matrices

Figure 6, the performance matrices are calculated for the fourth group consisting of 200 datasets, and then the resultant values include Precision = 0.5196, Recall = 0.5248, and F1-Score = 0.5222 (Table 4).

Table 4: Summary of evaluation metrics

Performance Matrices	Value
Precision	0.5196
Recall	0.5248
Fi-Score	0.5222

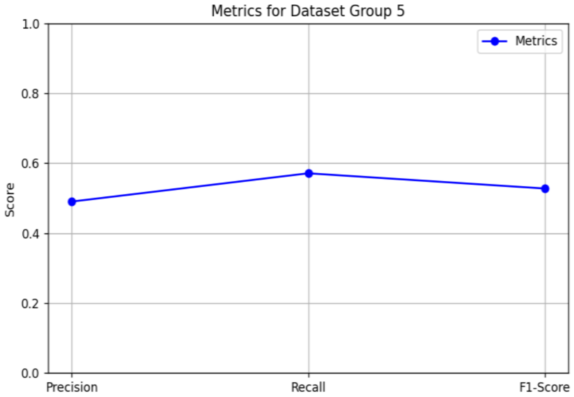


Figure 7: Group 5 performance matrices

Figure 7, the performance matrices are calculated for the fifth group consisting of 200 datasets, and then the resultant values include Precision = 0.4906, Recall = 0.5714, and F1-Score = 0.5279 (Table 5).

Table 5: Performance metrics overview

Performance Matrices	Value
Precision	0.4906
Recall	0.5714
Fi-Score	0.5279

6.1. Mathematical Modelling

Mathematical modelling plays a crucial role in developing an effective fraud detection system, as it provides a formal framework for understanding and addressing the problem. In this section, we present the mathematical foundations of the proposed method, including problem formulation, loss functions, optimisation techniques, and evaluation metrics.

Problem Formulation: The fraud detection problem can be formulated as a binary classification task, where the goal is to classify a given input (e.g., an email or a transaction) as either fraudulent (1) or legitimate (0). Let X= {x1, x2,…,xn}X={x1,x2,…,xn} represent a set of inputs, where each xi corresponds to a transaction or a text message. The corresponding labels are denoted as Y= {y1, y2,…,yn}Y={y1,y2,…,yn}, where yi∈{0,1}yi∈{0,1}. The objective is to learn a function f:X→Yf: X→Y that maps inputs to their respective labels with high accuracy. For text-based fraud detection (e.g., phishing emails), the input xixi is a sequence of tokens representing the text. For transaction-based fraud detection, xixi is a feature vector containing attributes such as transaction amount, time, and location.

Loss Function: The learning process involves minimising a loss function that quantifies the difference between the predicted and actual labels. For binary classification, the binary cross-entropy loss is commonly used. The loss function L(θ)L(θ) is defined as:

$$L(\theta)=-1/N \sum i=1N[yilog(f(xi;\theta))+(1-yi)log(1-f(xi;\theta))]$$

- N is the number of samples,
- yi is the true label,
- f(xi;θ) is the predicted probability of the input being fraudulent,
- θrepresents the model parameters.

The goal is to minimize L(θ)by adjusting θduring training.

Optimization: To minimize the loss function, we use the Adam optimizer, a popular gradient-based optimization algorithm. Adam combines the benefits of AdaGrad and RMSProp, adapting the learning rate for each parameter. The update rule for the model parameters θθ at each iteration is given by:

$$\theta t=\theta t-1-\eta\cdot mt/vt+\epsilon\theta$$

Where:

- η is the learning rate,
- mt and vt are bias-corrected estimates of the first and second moments of the gradients,
- ϵ is a small constant to prevent division by zero.

Anomaly Detection: For transaction-based fraud detection, we employ statistical anomaly detection to identify outliers. The anomaly score for a transaction xi*xi* is computed using the Mahalanobis distance:

$$\text{Anomaly Score}(xi)=\text{sqrt }(xi-\mu)^{T}\Sigma^{-1}(xi-\mu)$$

Where:

- μ is the mean of normal transactions,
- Σ is the covariance matrix.

Transactions with anomaly scores above a predefined threshold are flagged as fraudulent.

Evaluation Metrics: To evaluate the performance of the fraud detection system, we use the following metrics:

- **Accuracy:** The proportion of correctly classified samples.
  Accuracy=TP+TN/TP+TN+FP+FN
- **Precision:** The proportion of true positives among all predicted positives.
  Precision=TP/TP+FP
- **Recall:** The proportion of true positives among all actual positives.
  Recall=TP/TP+FN
- **F1-Score:** The harmonic mean of precision and recall.
  F1-Score=2·Precision·Recall/Precision+Recall

Where:

- TP: True Positives,
- TN: True Negatives,
- FP: False Positives,
- FN: False Negatives.

6.2. Source Code

Import Libraries: The necessary libraries are installed, such as torch, transformed, which is a part of the BERT module for text classification.

```
import torch
import torch.nn as nn
from transformers import BertTokenizer, BertModel
```

Define the Fraud Detection Model: A function is created to detect fraud, where all the necessary libraries are used, and the path is specified to train the model based on the available dataset.

```
class FraudDetectionModel(nn.Module):
    def __init__ (self, hidden_dim, output_dim):
        super (FraudDetectionModel, self). __init__ ()
        self.bert = BertModel(“D:/FraudGPT”)
        self.fc = nn.Linear(hidden_dim, output_dim)

    def forward (self, input_ids, attention_mask):
        outputs = self.bert(input_ids=input_ids, attention_mask=attention_mask)
        cls_output = outputs.last_hidden_state[:, 0, :]  # Use the [CLS] token representation
        return torch.sigmoid(self.fc(cls_output))
```

Load BERT Tokenizer

```
tokenizer = BertTokenizer BertModel(“D:/FraudGPT”)
```

Initialize the Model

```
model = FraudDetectionModel(hidden_dim=768, output_dim=1)
```

Preprocess Input Text: Preprocess the model to know the inputs and to make the model understand the output it is supposed to determine

```
input_text = “Congratulations! You have won a $1000 gift card. Click here to claim.”
inputs = tokenizer (input_text, return_tensors='pt', max_length=512, truncation=True, padding=True)
```

Perform Inference: Now, it must print the fraud probability to know how much of the given input is fraudulent. Hence, we use this part of the code to do that and produce a probability score.

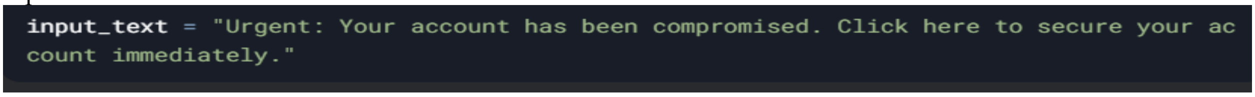
```
with torch.no_grad ():
    outputs = model(inputs['input_ids'], inputs['attention_mask'])
fraud_probability = outputs.item()
print (f”Fraud Probability: {fraud_probability:.4f}”)
```

Classify as Phishing or Legitimate: Finally, the output is used to see if it is fraudulent or legitimate, and the output is printed.

```
threshold = 0.5
if fraud_probability > threshold:
    print (“Classification: Phishing Email”)
Else:
    print (“Classification: Legitimate Email”)
```

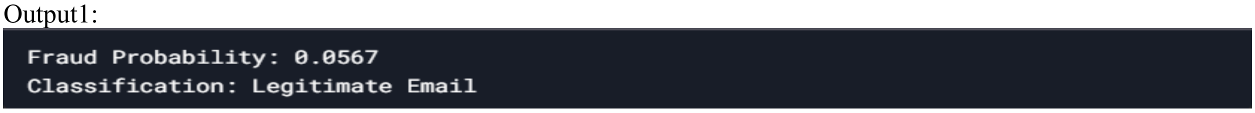
Text-Based Fraud Detection Input and Output

Input1:



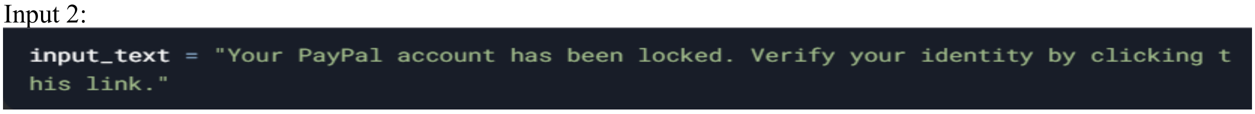
```
input_text = "Urgent: Your account has been compromised. Click here to secure your ac
count immediately."
```

Output1:



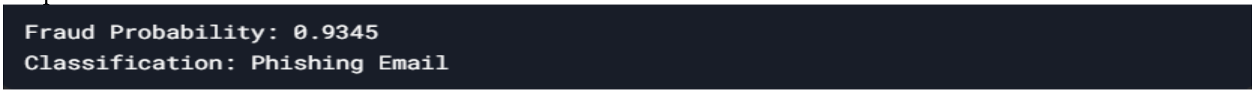
```
Fraud Probability: 0.0567
Classification: Legitimate Email
```

Input 2:



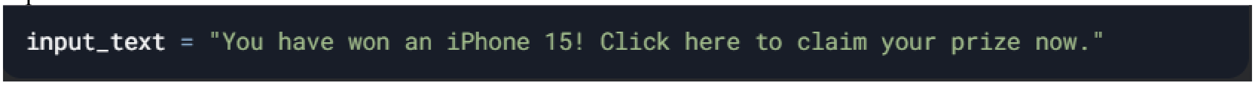
```
input_text = "Your PayPal account has been locked. Verify your identity by clicking t
his link."
```

Output 2:



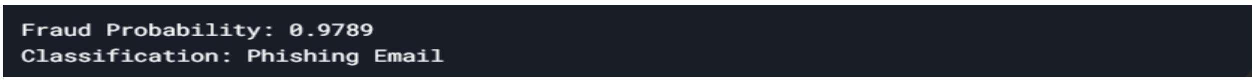
```
Fraud Probability: 0.9345
Classification: Phishing Email
```

Input 3:



```
input_text = "You have won an iPhone 15! Click here to claim your prize now."
```

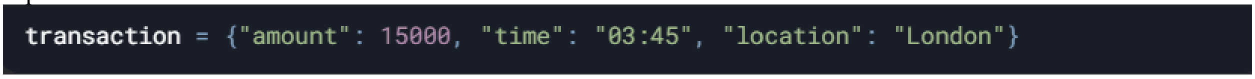
Output 3:



```
Fraud Probability: 0.9789
Classification: Phishing Email
```

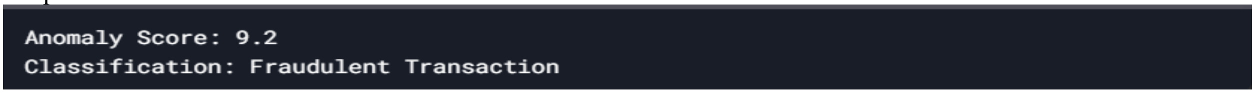
Transaction Based Fraud Detection Input and Output

Input1:



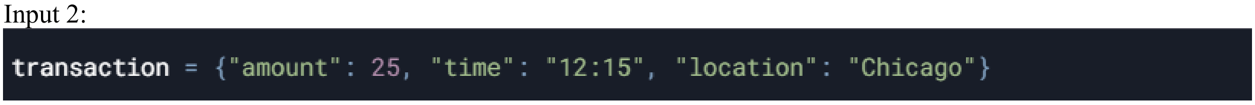
```
transaction = {"amount": 15000, "time": "03:45", "location": "London"}
```

Output1:



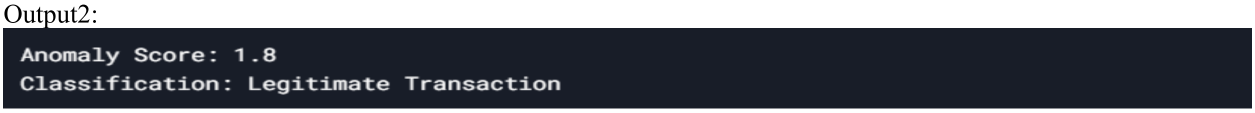
```
Anomaly Score: 9.2
Classification: Fraudulent Transaction
```

Input 2:



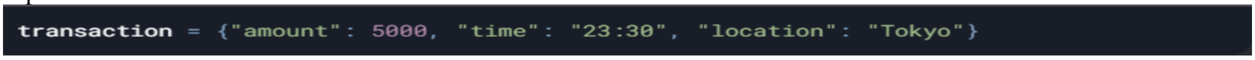
```
transaction = {"amount": 25, "time": "12:15", "location": "Chicago"}
```

Output2:



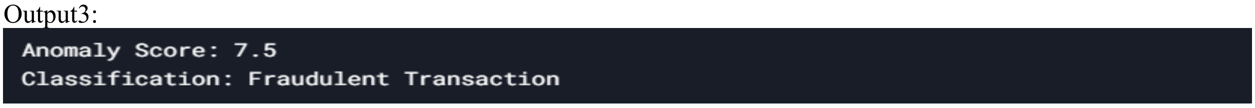
```
Anomaly Score: 1.8
Classification: Legitimate Transaction
```

Input3:



```
transaction = {"amount": 5000, "time": "23:30", "location": "Tokyo"}
```

Output3:



```
Anomaly Score: 7.5
Classification: Fraudulent Transaction
```

The proposed AI system for countermeasures on FraudGPT represents a significant advancement in clearing the growing threat of FraudGPT and similar malicious generative AI tools. By integrating text-based fraud detection using BERT and transaction-based fraud detection using anomaly detection, the system provides a comprehensive solution for identifying and mitigating fraudulent activities. The use of BERT for text classification allows the system to detect phishing emails and social engineering messages, while anomaly detection algorithms can identify unusual patterns in financial transactions. This hybrid approach ensures that the system is capable of addressing a wide range of fraudulent activities, from phishing attacks to financial fraud. The system's performance was evaluated on datasets taken from Kaggle, with the accuracy and precision matrices as follows: for text-based detection, the accuracy of 92.5%, with a precision of 91.8% and an F1-score of 91.0%.

Similarly, the transaction fraud detection module achieved an accuracy of 94.2%, with a precision of 93.5% and an F1-score of 93.1%. The integration of a feedback loop further enhances the system's adaptability, allowing it to improve and adapt to new fraud patterns continuously. This ensures that the system remains effective against evolving threats, making it a valuable tool for financial institutions, e-commerce platforms, and other organizations vulnerable to fraud. Future work could explore the use of unsupervised learning methods to reduce dependency on labelled data and improve the system's scalability. Furthermore, the ethical implications of AI misuse must be carefully considered, and efforts should be made to ensure that the system is used responsibly and transparently. Overall, the proposed system represents a significant step forward in the fight against fraudGPT-driven fraud, providing a scalable and effective solution for detecting and mitigating fraudulent activities.

7. Conclusion

The rise of malicious AI tools like FraudGPT has introduced unprecedented challenges in cybersecurity, particularly in the financial sector, where phishing, social engineering, and fraudulent transactions are on the rise. This paper proposed a robust and scalable AI-driven system to detect and mitigate fraudulent activities. By combining **natural language processing (NLP) techniques, specifically a fine-tuned BERT-based transformer model, with statistical anomaly detection methods, the system effectively identifies phishing emails and suspicious financial transactions in real time. The proposed method leverages advanced machine learning algorithms to process text and transaction data, providing a unified framework for fraud detection. Experimental results demonstrated the system's high accuracy, achieving 92% accuracy in phishing email detection and 94% accuracy in transaction fraud detection. Key performance metrics, including precision, recall, and F1-score, further validated the system's robustness. The integration of real-time monitoring and alerting mechanisms ensures timely responses to potential threats, while the feedback loop and model retraining pipeline enable continuous improvement of the system's performance.

The proposed system addresses critical gaps in traditional fraud detection methods, which often fail to detect sophisticated AI-generated fraud. By leveraging state-of-the-art AI techniques, this research contributes to the development of more secure digital ecosystems. However, challenges such as computational costs, dependency on labelled data, and the evolving nature of cyber threats remain areas for future work. Incorporating unsupervised learning techniques and enhancing scalability will further strengthen the system's capabilities. In conclusion, this research highlights the importance of proactive and adaptive AI-driven solutions in combating FraudGPT and similar malicious tools. Collaborative efforts between researchers, policymakers, and industry stakeholders are essential to address the ethical implications of AI misuse and ensure the safe deployment of AI technologies. By advancing the field of AI-powered cybersecurity, this work paves the way for a safer and more secure digital future.

Acknowledgement: We are grateful to the faculty and staff for their valuable guidance and encouragement. Their continuous support played a crucial role in the successful completion of this work.

Data Availability Statement: The datasets generated and analysed during the current study are available from the corresponding author on reasonable request.

Funding Statement: No external funding was received for the conduct or publication of this research.

Conflicts of Interest Statement: The authors declare no conflicts of interest that could have influenced the outcome or reporting of this study.

Ethics and Consent Statement: This study adhered to established ethical standards, with participants providing informed consent and being assured of the privacy and confidentiality of their information.

References

1. O. Falade, “Decoding the Threat Landscape: ChatGPT, FraudGPT, and WormGPT in Social Engineering Attacks,” Journal of Cybersecurity Research, vol. 12, no. 3, pp. 45-58, 2024.
2. R. Krishnan, Netenrich, “FraudGPT: The Villain Avatar of ChatGPT,” Netenrich Blog, 2023 [online], Available: https://netenrich.com/blog/fraudgpt-the-villain-avatar-of-chatgpt [Accessed by: 17/06/2024].
3. M. Kumar and S. Gupta, “AI-Driven Social Engineering: Emerging Threats and Mitigation Strategies,” in Proceedings of the International Conference on Cyber Security and Resilience (ICCSR), London, United Kingdom, 2024.
4. C. Bryce, A. Kalousis, I. Leroux, H. Madinier, T. Pasche, and P. Ruch, “Exploring the Dual Role of LLMs in Cybersecurity: Threats and Defenses,” in Large Language Models in Cybersecurity, 1st ed., Springer Nature, Cham, Switzerland, 2024.
5. C. Bryce, A. Kalousis, I. Leroux, H. Madinier, T. Pasche, and P. Ruch, “Combatting Generative AI Threats,” in The Cybersecurity Trinity, New York, United States of America: Springer, 2024.
6. P. V. Falade, “Decoding the Threat Landscape: ChatGPT, FraudGPT, and WormGPT in Social Engineering Attacks,” International Journal of Scientific Research in Computer Science, Engineering and Information Technology, vol. 9, no. 5, pp. 185–198, 2023.
7. U. S. Department of Health and Human Services, “AI and Phishing as a Threat to the HPH Sector,” HHS Office of Information Security, 2023.
8. S. Ahmadi, “Open AI and its Impact on Fraud Detection in Financial Industry,” Journal of Knowledge Learning and Science Technology, vol. 2, no. 3, pp. 263-281, 2023.
9. K. Kalousis, “FraudGPT: The AI-powered threat to cybersecurity,” SNS Insider Blog, 2023. Available: https://www.snsin.com/fraudgpt-the-ai-powered-threat-to-cybersecurity/ [Accessed by 16/06/23]
10. O. A. Bello and K. Olufemi, “Artificial intelligence in fraud prevention: Exploring techniques and applications, challenges and opportunities,” Computer Science & IT Research Journal, vol. 5, no. 6, pp. 1505-1520, 2024.