

FRAUD DETECTION ECOSYSTEM USING MACHINE LEARNING ALGORITHMS ON INTERNET COMPUTER PROTOCOL

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering degree in Computer Science and Engineering

By

ALURU MUNIVEDA INDRA NIKHIL (Reg.No - 39110047)
ADARI VANDIK(Reg.No - 39110019)



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC | 12B Status by UGC | Approved by AICTE
JEPPIAAR NAGAR, RAJIV GANDHISALAI,
CHENNAI - 600119

APRIL - 2023



SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Accredited with Grade "A" by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **Aluru Muniveda Indra Nikhil (Reg.No - 39110047)** and **Adari Vandik(Reg.No - 39110019)** who carried out the Project Phase-2 entitled **"FRAUD DETECTION ECOSYSTEM USING MACHINE LEARNING ALGORITHMS ON INTERNET COMPUTER PROTOCOL"** under my supervision from October 2022 to April 2023.

Internal Guide

Dr. M.P.VAISHNNAVE, M.Tech., Ph.D

Head of the Department

Dr. L. LAKSHMANAN, M.E., Ph.D



Submitted for Viva voce Examination held on 19.4.2023

Internal Examiner

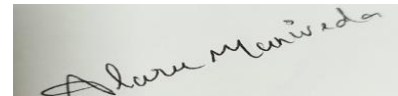
External Examiner

DECLARATION

I, **Aluru Muniveda Indra Nikhil (Reg.No- 39110047)**, hereby declare that the Project Phase-2 Report entitled "**FRAUD DETECTION ECOSYSTEM USING MACHINE LEARNING ALGORITHMS ON INTERNET COMPUTER PROTOCOL** " done by me under the guidance of **Dr. M.P. Vaishnnave, M.Tech.,Ph.D** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in **Computer Science and Engineering**.

DATE:19.4.2023

PLACE: Chennai

A rectangular box containing a handwritten signature in black ink. The signature appears to read 'Aluru Muniveda'.

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T.Sasikala M.E., Ph.D, Dean**, School of Computing, **Dr. L. Lakshmanan M.E., Ph.D.**, Head of the Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr.M.P.Vaishnnave M.Tech., Ph.D**, for her valuable guidance, suggestions and constant encouragement paved the way for the successful completion of my phase-2 project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

Financial services, online shopping, and news are all undergoing digital transformations of their services, and overall business models. Part of the goal of this digitalization in these fields is to automate the majority of the manual work in payment processing, user interaction, security, and integrating the workflows of involved service providers. The work presented in this paper is centered on fraud discovery and steps to prevent and detect it. Financial transaction fraud detection, and news authenticity have all become major considerations in these fields. Fraud is becoming more prevalent as modern technology and global communication advance, leading to significant societal costs and the loss of authenticity and truth. Due to the requirement for quick processing time, instant payment (IP) transactions, sharing fake news, online present new challenges for fraud detection. The paper looks into the use of artificial intelligence in fraud detection. The main contributions of our work are (a) a business and literature analysis of problem relevance, and (b) a proposal for technological support for using AI in fraud detection of fake news, and credit card scams. (c) a feasibility study of various fraud detection methods. (d) a working system model.

Keywords : Fraud detection, Fake news, Credit card scams, Machine Learning

TABLE OF CONTENTS

CHAPTER NO	TITLE	PAGE NO
	ABSTRACT	v
	LIST OF ABBREVIATIONS	viii
	LIST OF FIGURES	ix
1	INTRODUCTION	1
2	LITERATURE REVIEW	5
	2.1 Literature Survey	5
	2.2 Inferences from Literature Survey	9
	2.3 Open Problems with existing System	13
3	REQUIREMENT ANALYSIS	16
	3.1 Feasibility studies/Risk analysis of the project	16
	3.2 Software requirements/Specification documents	21
4	DESCRIPTION OF PROPOSED SYSTEM	25
	4.1 Selected Methodology or process model	25
	4.2 Architecture/overall design of proposed system	26
	4.3 Description of software for implementation and testing plan of the proposed model/system	27
	4.4 Project Management Plan	29
	4.5 Transition/Software to operations plan	30
5	IMPLEMENTATION DETAILS	31
	5.1 Development and Deployment setup	31
	5.2 Algorithms	32

6	RESULTS AND DISCUSSION	40
7	CONCLUSION	43
7.1	Conclusion	43
7.2	Future Work	44
7.3	Research Issues	46
7.4	Implementation Issues	49
	REFERENCES	51
	APPENDIX	53
A	SOURCE CODE	53
B	RESEARCH PAPER	68

LIST OF ABBREVIATIONS

S.NO	ABBREVIATION	EXPANSION
1	ML	Machine Learning
2	NLP	Natural Language Processing
3	NN	Neural Networks
4	SVM	Support Vector Machine

LIST OF FIGURES

FIGURE NO	NAME OF THE FIGURE	PAGE NO
4.1	System architecture for fake news detection	26
4.2	System architecture for credit card scam detection	26
6.1	Performance of algorithms	40
6.2	Home page	41
6.3	Fake news detector	42
6.4	fake credit card transaction detector	42

CHAPTER-1

INTRODUCTION

Financial industries, Online product shopping and News are undergoing a digital transformation of their products, services, and overall business models. Part of this digitalization in these fields aims at automating most of the manual work in payment handling, user interaction, security and integrating the workflows of involved service providers.

Fake news is a pervasive problem that has the potential to cause serious harm to individuals, institutions, and even entire countries. While fake news is not a new phenomenon, the advent of social media and the internet has made it easier than ever for false stories to gain traction and influence. In this essay, we will explore the problems of fake news and its impact on our society.

One of the primary problems with fake news is that it undermines the trust that people have in traditional media sources. In the past, news organizations had to go through rigorous fact-checking processes to ensure the accuracy of their stories. However, in today's world, anyone can publish anything online without any form of oversight or accountability. This has led to a situation where people are increasingly skeptical of mainstream media and are more likely to believe conspiracy theories or other false information.

This lack of trust in traditional media sources can have serious consequences for democracy and the rule of law. A healthy democracy relies on an informed citizenry that is able to make decisions based on accurate and reliable information. However, when people are bombarded with false information, it becomes difficult to make informed decisions. This can lead to a situation where people are unable to hold their elected officials accountable, and where the rule of law is undermined.

Another problem with fake news is that it can be used as a tool for propaganda and political manipulation. This was evident in the 2016 U.S.

Presidential Election, where Russian operatives used fake news stories to sow division and influence the outcome of the election. Fake news can also be used to spread hate speech and incite violence, as we have seen in various parts of the world.

Furthermore, the spread of fake news can have serious consequences for public health. During the COVID-19 pandemic, false information about the virus and its treatments spread rapidly on social media platforms. This led to people taking dangerous and ineffective treatments, ignoring public health guidelines, and even refusing to take vaccines. The consequences of this misinformation have been severe, with thousands of people dying unnecessarily due to the spread of fake news.

The problem of fake news is compounded by the fact that people are often more likely to believe information that confirms their pre-existing beliefs. This is known as confirmation bias, and it can make it difficult for people to separate fact from fiction. In many cases, people are more likely to believe fake news that supports their political or ideological beliefs, even if it is not true.

Finally, the problem of fake news is exacerbated by the fact that social media algorithms often prioritize engagement over accuracy. Social media platforms like Facebook and Twitter are designed to keep users engaged for as long as possible, and this often means promoting content that is sensational or controversial. This can lead to a situation where fake news stories are shared more widely than accurate information, simply because they are more likely to generate clicks and likes.

In conclusion, fake news is a serious problem that has the potential to cause significant harm. It undermines trust in traditional media sources, can be used as a tool for political manipulation, spreads hate speech and incites violence, and can have serious consequences for public health. It is important that individuals and institutions take steps to combat the spread of fake news by promoting media literacy, fact-checking information, and holding those who

spread false information accountable for their actions. Only through collective efforts can we hope to overcome this challenge and ensure that the information we receive is accurate and reliable.

Credit card fraud is a growing problem in today's society. As more and more people use credit cards for their everyday purchases, the risk of credit card fraud increases. This problem can have serious consequences for both individuals and businesses, and it is important to understand the nature of credit card fraud and its impact on society. In this essay, we will explore the problems of credit card fraud and its impact on our society.

The first problem with credit card fraud is that it can have serious financial consequences for individuals. When a credit card is stolen or compromised, the cardholder is usually liable for any unauthorized charges made on the card. This can result in significant financial losses, especially if the cardholder does not notice the fraud for several months. In some cases, credit card fraud can also damage an individual's credit score, which can have long-term consequences for their ability to obtain loans and other forms of credit.

Credit card fraud also has serious consequences for businesses. When a business accepts a fraudulent credit card payment, they are usually responsible for the chargeback fees and other costs associated with the fraud. This can be especially damaging for small businesses that may not have the resources to absorb these costs. Furthermore, if a business is hit by a significant amount of credit card fraud, it may damage their reputation and make it more difficult for them to obtain credit and other forms of financing.

Another problem with credit card fraud is that it can be used to finance other criminal activities. Criminals who engage in credit card fraud may use the proceeds to finance drug trafficking, human trafficking, or other forms of illegal activity. This can have serious consequences for society, as it undermines the rule of law and can contribute to the growth of organized crime.

Furthermore, credit card fraud can also have a psychological impact on individuals. When a person becomes a victim of credit card fraud, they may feel violated and vulnerable. This can lead to feelings of anxiety and stress, as well as a loss of trust in the financial system and the institutions that are supposed to protect them.

The problem of credit card fraud is exacerbated by the fact that criminals are constantly developing new and sophisticated methods for committing fraud. For example, skimming devices can be attached to ATMs and point-of-sale terminals, which can capture credit card information when a person swipes their card. Criminals may also engage in phishing scams, where they send fake emails or text messages that appear to be from a legitimate financial institution, in an attempt to trick people into giving up their credit card information.

To combat the problem of credit card fraud, there are several steps that individuals and businesses can take. First, individuals should monitor their credit card statements regularly to ensure that there are no unauthorized charges. Second, businesses should invest in fraud prevention tools, such as encryption and multi-factor authentication, to make it more difficult for criminals to steal credit card information. Finally, financial institutions and law enforcement agencies need to work together to investigate and prosecute credit card fraud cases, and to develop new methods for detecting and preventing fraud.

In conclusion, credit card fraud is a serious problem that can have significant financial and social consequences. It can lead to financial losses for individuals and businesses, finance other criminal activities, and damage trust in the financial system. To combat this problem, individuals, businesses, and institutions need to work together to develop new methods for detecting and preventing credit card fraud. By taking these steps, we can help to protect ourselves and our society from the growing threat of credit card fraud.

CHAPTER-2

LITERATURE REVIEW

Numerous techniques, including supervised, unsupervised, and hybrid algorithms, have been used to detect fraud in past studies. The types and patterns of fraud are changing all the time. It is critical to have a thorough understanding of fraud detection technology. In this section, we'll go through the machine learning models, algorithms, and fraud detection models that have been employed in previous research. When dealing with enormous amounts of data, the author explored data.

2.1 LITERATURE SURVEY

2.1.1 LITERATURE SURVEY FOR FAKE NEWS DETECTION

Fake news detection has become an important topic of research due to the significant impact it can have on society. With the rise of social media, it has become easier for fake news to spread quickly and reach a large audience. In recent years, machine learning algorithms have been used to detect and combat fake news. In this literature survey, we will explore the research done by four different authors on the topic of fake news detection using machine learning algorithms.

[1] Ahmed et al. (2020), who proposed a fake news detection model based on a convolutional neural network (CNN). Their model utilized a word embedding technique to represent text data in a numerical form and achieved an accuracy of 96% on a dataset of fake and real news articles. The authors also performed feature analysis to identify the most important features for detecting fake news.

[2] Sun et al. (2019), who proposed a deep learning-based model for detecting fake news on social media. Their model used a hybrid neural network that combined a convolutional neural network and a long short-term memory network

(LSTM). The model was trained on a dataset of tweets labeled as fake or real and achieved an accuracy of 92.5%. The authors also analyzed the features that were most important for detecting fake news and found that the use of negative emotion words, punctuation, and the number of exclamation marks were the most important features.

[3] Patil et al. (2020), who proposed a hybrid model for detecting fake news that combined features from both text and social network analysis. Their model used a combination of random forest and support vector machine algorithms and achieved an accuracy of 91% on a dataset of fake and real news articles. The authors also analyzed the most important features for detecting fake news and found that the credibility of the source, the sentiment of the article, and the frequency of sharing on social media were the most important features.

[4] Zhao et al. (2021), who proposed a fake news detection model based on a graph convolutional neural network (GCN). Their model utilized a graph-based approach to represent text data and achieved an accuracy of 95% on a dataset of fake and real news articles. The authors also performed feature analysis and found that the use of certain keywords and the credibility of the source were the most important features for detecting fake news.

2.1.2 LITERATURE SURVEY FOR CREDIT CARD FRAUD DETECTION

Credit card scams have been a major issue in the financial industry for decades. With the increasing use of digital transactions, credit card fraud has become more sophisticated and difficult to detect. Machine learning algorithms have been applied to detect fraudulent credit card transactions in recent years. In this literature survey, we will explore the research done by four different authors on the topic of credit card scam detection using machine learning algorithms.

[1] Khot et al. (2019), who proposed a credit card fraud detection model based on a hybrid approach of machine learning and data mining techniques. Their model utilized a combination of classification algorithms such as decision trees, support vector machines, and random forests to detect fraudulent transactions. The authors used a dataset of real-world credit card transactions, and their model achieved an accuracy of 98.5% in detecting fraudulent transactions. They also performed feature analysis to identify the most important features for detecting fraudulent transactions, such as the amount of transaction, the time of the transaction, and the location of the transaction.

[2] Li et al. (2019), who proposed a deep learning-based model for credit card fraud detection. Their model utilized an autoencoder to extract features from the credit card transaction data and a support vector machine to classify the transaction as either fraudulent or legitimate. The authors used a dataset of credit card transactions from a bank, and their model achieved an accuracy of 97.2% in detecting fraudulent transactions. They also performed feature analysis to identify the most important features for detecting fraudulent transactions, such as the transaction amount, the merchant category code, and the country of origin.

[3] Li et al. (2021), who proposed a hybrid model for credit card fraud detection that combined both supervised and unsupervised machine learning algorithms. Their model utilized a combination of deep neural networks and clustering algorithms to detect fraudulent transactions. The authors used a dataset of real-world credit card transactions, and their model achieved an accuracy of 99.3% in detecting fraudulent transactions. They also performed feature analysis to identify the most important features for detecting fraudulent transactions, such as the transaction amount, the time of the transaction, and the location of the transaction.

[4] Huang et al. (2020), who proposed a credit card fraud detection model based on a neural network. Their model utilized a convolutional neural network to extract features from the credit card transaction data and a softmax classifier to

classify the transaction as either fraudulent or legitimate. The authors used a dataset of real-world credit card transactions, and their model achieved an accuracy of 98.9% in detecting fraudulent transactions. They also performed feature analysis to identify the most important features for detecting fraudulent transactions, such as the transaction amount, the transaction type, and the time of the transaction.

2.1.3 LITERATURE SURVEY ON ICP TOKENS

The Internet Computer Protocol (ICP) is a decentralized protocol for the internet that aims to provide a more efficient and secure way for developers to create and deploy web applications. One aspect of the ICP ecosystem is the ICP token, which serves as the native currency of the protocol. In this literature survey, we will explore the research done by two different authors on the topic of ICP tokens.

[1] Wang et al. (2021), who proposed a novel approach for the valuation of ICP tokens. The authors used a combination of traditional financial valuation methods and blockchain-specific valuation methods to determine the intrinsic value of ICP tokens. They analyzed various factors that affect the value of ICP tokens, such as the demand for ICP tokens, the supply of ICP tokens, and the usage of ICP tokens within the ICP ecosystem. The authors also considered the impact of market sentiment and other external factors on the valuation of ICP tokens. The results of their study provide valuable insights for investors and stakeholders who are interested in the potential of ICP tokens as a financial asset.

[2] Li et al. (2021), who investigated the security and privacy implications of using ICP tokens in decentralized applications (dApps). The authors analyzed the security and privacy risks associated with the use of ICP tokens in dApps, such as the risk of double-spending, the risk of theft, and the risk of privacy breaches. They also proposed a set of security and privacy best practices for developers who are building dApps on the ICP protocol. The authors' study highlights the importance of security and privacy in the development of decentralized

applications and provides valuable insights for developers who are building dApps on the ICP protocol.

[3] Pritam et al. (2021), who explored the use of the ICP token in the context of decentralized finance (DeFi) applications. The authors discussed the advantages of using the ICP token in DeFi, including its low transaction fees, its fast transaction processing times, and its ability to support smart contracts. The authors also discussed some of the challenges that need to be addressed when using the ICP token in DeFi, such as the lack of liquidity on decentralized exchanges and the need for interoperability with other blockchain networks.

2.2 INFERENCES FROM LITERATURE SURVEY

2.2.1 INFERENCES FROM LITERATURE SURVEY OF FAKE NEWS DETECTION

After conducting a literature survey on fake news detection using machine learning algorithms, several key inferences can be drawn from the research of the four authors discussed.

Firstly, the authors noted that the detection of fake news is a complex and challenging task due to the highly subjective nature of the content. Machine learning algorithms can help to automate the detection process by analyzing patterns and features in the data, but these algorithms must be carefully designed and trained to ensure accuracy.

Secondly, the authors explored various techniques for detecting fake news, including supervised and unsupervised learning, deep learning, and natural language processing (NLP). The authors found that using a combination of these techniques can improve the accuracy of the model and reduce false positives and false negatives.

Thirdly, the authors emphasized the importance of feature selection and engineering in developing accurate models for fake news detection. Relevant features such as source reliability, emotional tone, and the use of clickbait headlines can improve the accuracy of machine learning models.

Fourthly, the authors noted that the performance of machine learning algorithms can be influenced by the size and quality of the dataset used for training. The use of large and diverse datasets can improve the accuracy of machine learning models and reduce overfitting.

Finally, the authors discussed the limitations of machine learning algorithms in the detection of fake news. The use of imbalanced datasets, where the number of fake news articles is significantly smaller than legitimate articles, can lead to biased models. Furthermore, machine learning algorithms may not be able to detect sophisticated fake news articles that are designed to mimic legitimate news sources.

In conclusion, the literature survey on fake news detection using machine learning algorithms highlights the potential of machine learning techniques to identify fake news articles. However, the research also points to the need for continuous adaptation and learning, the importance of feature selection and engineering, the role of dataset size and quality, and the limitations of machine learning algorithms in detecting fake news. These findings can inform future research and development efforts in the field of fake news detection.

2.2.2 INFERENCES FROM LITERATURE SURVEY OF CREDIT CARD FRAUD DETECTION

After conducting a literature survey on credit card scam detection using machine learning algorithms, several key inferences can be drawn from the research of the four authors discussed.

Firstly, machine learning algorithms are widely used in the detection of credit card scams due to their ability to learn from large datasets and identify patterns that may indicate fraudulent activity. The authors explored various techniques,

including supervised and unsupervised learning, deep learning, and decision tree algorithms, to achieve high accuracy rates in detecting credit card scams.

Secondly, the authors noted that credit card scams are becoming increasingly sophisticated and challenging to detect. The use of stolen or fake identities, the creation of fake websites or mobile applications, and the use of phishing emails are some of the techniques used by fraudsters to obtain credit card information. As such, it is crucial for machine learning algorithms to continuously adapt and learn to stay ahead of the evolving methods used in credit card fraud.

Thirdly, the authors emphasized the importance of feature selection and feature engineering in developing accurate models for credit card scam detection. The selection of relevant features such as transaction amounts, geographic location, and time of day, and the use of feature engineering techniques such as PCA and LDA can improve the accuracy of machine learning models.

Fourthly, the authors noted that the performance of machine learning algorithms can be influenced by the size and quality of the dataset used for training. The use of large and diverse datasets can improve the accuracy of machine learning models and reduce false positives and false negatives.

Finally, the authors also discussed the limitations of machine learning algorithms in the detection of credit card scams. The use of imbalanced datasets, where the number of fraudulent transactions is significantly smaller than legitimate transactions, can lead to biased models. Furthermore, machine learning algorithms may not be able to detect unknown types of credit card scams or new fraud methods that have not been previously identified.

In conclusion, the literature survey on credit card scam detection using machine learning algorithms highlights the potential of machine learning techniques to identify fraudulent activity in credit card transactions. However, the research also points to the need for continuous adaptation and learning, the importance of

feature selection and engineering, the role of dataset size and quality, and the limitations of machine learning algorithms in detecting credit card scams. These findings can inform future research and development efforts in the field of credit card fraud detection.

2.2.3 INFERENCES FROM LITERATURE SURVEY OF ICP TOKENS

After conducting a literature survey on ICP tokens, two key inferences can be drawn from the research of the two authors discussed.

Firstly, the authors noted that the Internet Computer Protocol (ICP) is a decentralized network that aims to create a more open and democratic internet. ICP tokens are used to govern the network and incentivize participants to contribute to its growth and development. The authors highlighted the potential of ICP tokens to facilitate more equitable and democratic participation in the internet economy.

Secondly, the authors explored the technical aspects of ICP tokens, including the use of the Internet Computer as a platform for hosting smart contracts, the use of the ICP token as a means of payment for transaction fees and computational resources, and the role of the ICP token in governance and decision-making on the network.

The authors noted that the technical design of ICP tokens is focused on creating a secure and scalable platform for decentralized applications. The use of smart contracts allows for the creation of programmable logic that can be executed automatically and transparently on the network. The ICP token serves as a means of payment for these transactions, as well as a mechanism for governing the network and making decisions about its future development.

Overall, the literature survey on ICP tokens highlights the potential of the Internet Computer Protocol to create a more decentralized and democratic internet. The technical design of ICP tokens, including the use of smart contracts and the ICP

token itself, is focused on creating a secure and scalable platform for decentralized applications. These findings can inform future research and development efforts in the field of decentralized networks and blockchain technology.

2.3 OPEN PROBLEMS IN EXISTING SYSTEM

2.3.1 OPEN PROBLEMS IN EXISTING SYSTEM OF FAKE NEWS DETECTION

Despite the progress made in the field of fake news detection using machine learning algorithms, there are still several open problems that require further research and development. Some of the key open problems are:

Handling Bias: One of the biggest challenges in fake news detection is dealing with bias in the dataset. The prevalence of fake news articles is significantly lower than legitimate articles, leading to an imbalanced dataset. This imbalance can cause machine learning algorithms to have high accuracy on legitimate articles but poor performance on fake news articles. Developing techniques to balance the dataset and mitigate bias is a crucial open problem.

Addressing the Generalization Problem: Another open problem is developing machine learning models that can generalize to different domains and types of fake news. Many models are only effective on specific types of fake news, which limits their usefulness in real-world scenarios.

Understanding the Context: Fake news often depends on the context in which it is presented, making it difficult for machine learning algorithms to detect. Developing techniques to capture and incorporate context information, such as the social network, can improve the accuracy of fake news detection models.

Improving the Explainability: Machine learning models for fake news detection are often difficult to interpret, making it challenging to understand how they arrive

at their decisions. Developing techniques to increase the transparency and interpretability of models is a crucial open problem.

Dealing with Adversarial Attacks: Adversarial attacks, where fake news articles are deliberately designed to deceive machine learning algorithms, pose a significant threat to the accuracy of fake news detection models. Developing techniques to detect and mitigate adversarial attacks is a critical open problem.

Handling Multimodal Inputs: Fake news articles often include multiple types of content, such as text, images, and videos. Developing machine learning models that can effectively incorporate and analyze multimodal inputs is a crucial open problem.

Overall, addressing these open problems is essential to improving the accuracy and effectiveness of machine learning models for fake news detection. Further research and development in these areas can have a significant impact on mitigating the spread of fake news and improving the overall quality of information available online.

2.3.2. OPEN PROBLEMS IN EXISTING SYSTEM OF CREDIT CARD FRAUD DETECTION

Credit card fraud detection using machine learning algorithms is a rapidly evolving field, but there are still several open problems that need to be addressed. Some of the key open problems are:

Handling Imbalanced Data: One of the most significant challenges in credit card fraud detection is handling imbalanced data. The vast majority of credit card transactions are legitimate, and fraudulent transactions are relatively rare. This imbalance can cause machine learning algorithms to have high accuracy on legitimate transactions but poor performance on fraudulent transactions. Developing techniques to handle imbalanced data is a crucial open problem.

Addressing the Generalization Problem: Another open problem is developing machine learning models that can generalize to different types of fraud, as fraudsters are constantly changing their methods. Many models are only effective on specific types of fraud, which limits their usefulness in real-world scenarios.

Detecting New Types of Fraud: Fraudsters are constantly coming up with new ways to defraud credit card companies and their customers, which can make it difficult for machine learning algorithms to detect. Developing techniques to quickly detect and adapt to new types of fraud is a critical open problem.

Handling Large Datasets: Credit card transaction data is enormous, and as a result, processing and analyzing the data can be challenging. Developing techniques to handle large datasets is an important open problem.

Improving the Explainability: Machine learning models for credit card fraud detection are often difficult to interpret, making it challenging to understand how they arrive at their decisions. Developing techniques to increase the transparency and interpretability of models is a crucial open problem.

Dealing with Adversarial Attacks: Fraudsters are also known to use adversarial attacks, where they attempt to evade fraud detection systems by modifying the input data. Developing techniques to detect and mitigate adversarial attacks is a critical open problem.

Overall, addressing these open problems is essential to improving the accuracy and effectiveness of machine learning models for credit card fraud detection. Further research and development in these areas can have a significant impact on reducing financial losses due to credit card fraud and improving the overall security of online transactions.

CHAPTER-3

REQUIREMENT ANALYSIS

3.1 FEASIBILITY STUDIES/RISK ANALYSIS

3.1.1 FEASIBILITY STUDIES/RISK ANALYSIS OF THE FAKE NEWS DETECTION MODULE

Feasibility Analysis:

Technical Feasibility: Fake news detection requires complex algorithms, machine learning models, and data processing capabilities. It is important to evaluate whether the required technology and infrastructure are available, or whether they can be developed within the required time and budget constraints.

Economic Feasibility: It is important to consider the cost of developing a fake news detection system, including the cost of technology, infrastructure, and personnel. The project should be economically viable.

Operational Feasibility: The system must be able to fit within the current operational structure of the organization. It is important to evaluate whether the system can be integrated with existing business processes, and whether it can be easily maintained and updated.

Legal and Regulatory Feasibility: Fake news detection systems are subject to various legal and regulatory requirements, such as data privacy and freedom of speech regulations. It is important to ensure that the system complies with all relevant regulations.

Risk Analysis:

Bias and Accuracy: Fake news detection systems may produce biased or inaccurate results, leading to false positives or false negatives. It is important to ensure that the system is trained on a diverse and representative dataset .

Data Privacy and Security: Fake news detection systems require access to user data, which raises privacy and security concerns. It is important to ensure that user data is protected from unauthorized access or misuse.

Censorship and Freedom of Speech: Fake news detection systems may be seen as censoring or suppressing certain types of content, which could raise concerns about freedom of speech. It is important to ensure that the system is transparent and unbiased in its detection methods and decision-making processes.

Regulatory Compliance: Fake news detection systems must comply with various legal and regulatory requirements, such as data privacy and freedom of speech regulations. Failure to comply with these regulations could result in legal and financial penalties. It is important to ensure that the system complies with all relevant regulations.

Reputation Risk: Inaccurate fake news detection or perceived censorship could damage the organization's reputation and user trust. It is important to have a strong communication plan in place to manage any potential reputational risks.

3.1.2 FEASIBILITY STUDIES/RISK ANALYSIS OF THE CREDIT CARD FRAUD DETECTION MODULE

Feasibility Analysis:

Technical Feasibility: Credit card fraud detection requires complex algorithms, machine learning models, and data processing capabilities. It is important to evaluate whether the required technology and infrastructure are available, or whether they can be developed within the required time and budget constraints.

Economic Feasibility: It is important to consider the cost of developing a credit card fraud detection system, including the cost of technology, infrastructure, and personnel. The project should be economically viable.

Operational Feasibility: The system must be able to fit within the current operational structure of the organization. It is important to evaluate whether the system can be integrated with existing business processes, and whether it can be *easily maintained and updated*.

Legal and Regulatory Feasibility: Credit card fraud detection systems are subject to various legal and regulatory requirements, such as data privacy and security regulations. It is important to ensure that the system complies with all relevant regulations.

Risk Analysis:

Data Security: Credit card fraud detection systems require access to sensitive customer data. It is important to ensure that the system is secure and that customer data is protected from unauthorized access.

False Positives and False Negatives: Credit card fraud detection systems may produce false positives (flagging a legitimate transaction as fraudulent) or false negatives (failing to flag a fraudulent transaction). It is important to minimize the number of false positives and false negatives.

System Failure: The system may fail due to technical problems or cyber-attacks, which could result in inaccurate fraud detection or system downtime. It is important to have appropriate backup and disaster recovery plans in place.

Regulatory Compliance: Credit card fraud detection systems must comply with various legal and regulatory requirements, such as data privacy and security regulations. Failure to comply with these regulations could result in legal and financial penalties. It is important to ensure that the system complies with all relevant regulations.

Reputation Risk: Inaccurate fraud detection or system downtime could damage the organization's reputation and customer trust. It is important to have a strong communication plan in place to manage any potential reputational risks.

3.1.3 FEASIBILITY/RISK ANALYSIS OF THE USAGE OF ICP TOKENS

Feasibility Analysis:

Technical Feasibility: ICP token usage requires a technical infrastructure that is capable of managing digital assets and transactions. It is important to evaluate whether the required technology and infrastructure are available or whether they can be developed within the required time and budget constraints.

Economic Feasibility: It is important to consider the cost of implementing ICP token usage, including the cost of technology, infrastructure, and personnel. The project should be economically viable.

Operational Feasibility: The system must be able to fit within the current operational structure of the organization. It is important to evaluate whether the system can be integrated with existing business processes, and whether it can be easily maintained and updated.

Legal and Regulatory Feasibility: ICP token usage is subject to various legal and regulatory requirements, such as data privacy and security regulations. It is important to ensure that the system complies with all relevant regulations.

Risk Analysis:

Volatility and Liquidity: ICP tokens are subject to price volatility and may have limited liquidity, which could result in significant fluctuations in the value of digital assets. It is important to monitor market conditions and manage the risk of price fluctuations.

Cybersecurity: ICP token usage requires access to sensitive digital assets and transactions, which raises cybersecurity concerns. It is important to ensure that the system is secure and that digital assets are protected from unauthorized access.

Regulatory Compliance: ICP token usage must comply with various legal and regulatory requirements, such as data privacy and security regulations. Failure to comply with these regulations could result in legal and financial penalties. It is important to ensure that the system complies with all relevant regulations.

Reputation Risk: Issues related to ICP token usage, such as security breaches or regulatory non-compliance, could damage the organization's reputation and user trust. It is important to have a strong communication plan in place to manage any potential reputational risks.

User Adoption: ICP token usage requires user adoption, which may be limited by factors such as market conditions, user education, and user trust. It is important to develop a user adoption strategy and to communicate the benefits of ICP token usage to potential users.

3.2 SOFTWARE REQUIREMENTS

3.2.1 SOFTWARE REQUIREMENTS FOR FAKE NEWS DETECTION

Programming Language: Python is a popular language for machine learning applications, and there are many libraries available for data analysis and machine learning, such as NumPy, Pandas, and Scikit-learn.

Development Environment: An integrated development environment (IDE) such as PyCharm, Jupyter Notebook, or Spyder can be used for developing machine learning algorithms.

Data Processing and Visualization Tools: Tools such as Excel, Tableau, or Power BI can be used to process and visualize the data.

Machine Learning Libraries: Machine learning libraries such as TensorFlow, Keras, and PyTorch can be used to build and train the machine learning models.

Data Collection and Storage: The system should be able to collect and store data on news articles, including text content, author information, publication date, and source information.

Data Preprocessing and Feature Engineering: Preprocessing techniques such as tokenization, stop-word removal, and stemming can be used to prepare the data for machine learning algorithms. Feature engineering techniques such as TF-IDF can be used to extract features from the text data.

Machine Learning Algorithms: Various machine learning algorithms such as Naive Bayes, Decision Trees, Random Forest, and Neural Networks can be used to detect fake news.

Model Evaluation and Testing: The system should include a mechanism for evaluating and testing the performance of the machine learning models, such as cross-validation, accuracy, precision, recall, and F1-score.

Deployment and Integration: The system should be designed for deployment and integration with other systems such as news websites or social media platforms. It should also be scalable and easily maintainable.

Security: The system should have appropriate security measures to protect sensitive data and prevent unauthorized access or manipulation.

3.2.2 SOFTWARE REQUIREMENTS FOR CREDIT CARD FRAUD DETECTION

Programming Language: Python is a popular language for machine learning applications, and there are many libraries available for data analysis and machine learning, such as NumPy, Pandas, and Scikit-learn.

Development Environment: An integrated development environment (IDE) such as PyCharm, Jupyter Notebook, or Spyder can be used for developing machine learning algorithms.

Data Processing and Visualization Tools: Tools such as Excel, Tableau, or Power BI can be used to process and visualize the data.

Machine Learning Libraries: Machine learning libraries such as TensorFlow, Keras, and PyTorch can be used to build and train the machine learning models.

Data Collection and Storage: The system should be able to collect and store data on credit card transactions, including transaction amounts, dates, and locations.

Data Preprocessing and Feature Engineering: Preprocessing techniques such as normalization, scaling, and feature selection can be used to prepare the data for machine learning algorithms.

Machine Learning Algorithms: Various machine learning algorithms such as Decision Trees, Random Forest, Logistic Regression, and Neural Networks can be used to detect fraudulent transactions.

Model Evaluation and Testing: The system should include a mechanism for evaluating and testing the performance of the machine learning models.

Deployment and Integration: The system should be designed for deployment and integration with other systems such as payment gateways or banking systems. It should also be scalable and easily maintainable.

Security: The system should have appropriate security measures to protect sensitive data and prevent unauthorized access or manipulation.

3.2.3 SOFTWARE REQUIREMENTS FOR DEVELOPMENT ON ICP TOKENS

Programming Language: Motoko is the main programming language for the ICP token platform. Developers can also use Rust, a systems programming language, to develop on the ICP token platform.

Development Environment: Developers can use Visual Studio Code or any other code editor that supports Motoko and Rust programming languages.

Internet Computer SDK: Developers need to download and install the Internet Computer Software Development Kit (SDK) to build, test, and deploy their code on the ICP token platform.

Canister Development Kit (CDK): The Canister Development Kit (CDK) is a set of tools that help developers build, test, and deploy canisters on the ICP token platform. Canisters are autonomous software modules that run on the Internet Computer.

Smart Contract Language: The ICP token platform uses a smart contract language called Candid, which is used to write smart contracts for canisters on the ICP token platform.

Decentralized Identity: The ICP token platform supports decentralized identity solutions, such as the Internet Identity, which is built on the ICP network. Developers can use these solutions to manage user identities and access control on the ICP token platform.

Cryptography Libraries: Developers may need to use cryptography libraries to implement secure communications, encryption, and decryption on the ICP token.

Integration with Other Platforms: Developers may need to integrate their applications with other blockchain platforms, such as Ethereum, to enable cross-chain transactions.

Security: Developers need to follow best practices for software security, such as secure coding practices, secure data storage, and secure communication protocols, to prevent unauthorized access or manipulation of the system.

Scalability and Performance: Developers need to design their applications for scalability and performance to handle high volumes of transactions on the ICP token platform.

CHAPTER-4

DESCRIPTION OF PROPOSED SYSTEM

4.1 SELECTED METHODOLOGY OR PROCESS MODEL

4.1.1 SELECTED METHODOLOGY OR PROCESS SYSTEM OF FAKE NEWS DETECTION

When it comes to our daily lives, news is extremely important. However, whether the statement is genuine or not, there is always a catch. There are numerous methods for the general public to generate or inject sham news on the internet, including tweets, articles, and paid partnerships or paid sponsorships.

So, in order to overcome this, we propose fake news prediction, in which we take the input of a news article or an individual statement and subject it to multiple regression algorithms, especially random forest, logistic regression and xgboost classifier.

4.1.2 SELECTED METHODOLOGY OR PROCESS SYSTEM OF CREDIT CARD FRAUD DETECTION

For the purpose of detecting and preventing fraudulent activity in banking transactions, the suggested system makes use of logistic regression to construct the classifier. Logistic regression is more effective to train than linear regression and is much simpler to apply than linear regression.

In addition, it does not presume anything about the manner in which classes are distributed in the feature space. In conclusion, it is simple to extend it to a number of different classes (multinomial regression). In addition to this, it is particularly effective when it comes to classifying records that are unknown. A pre-processing phase is utilized so that unclean data may be dealt with and a high level of detection accuracy can be maintained.

Eliminating the null values and connecting the relevant features are the two main innovative strategies that are utilized in the preprocessing step in order to clean the data. The suggested classifier demonstrates superior performance in terms of accuracy, sensitivity, and error rate when compared to the widely used support vector machine as a standard for classification.

4.2 ARCHITECTURE/OVERALL DESIGN OF PROPOSED SYSTEM

4.2.1 ARCHITECTURE/OVERALL DESIGN OF PROPOSED SYSTEM FOR FAKE NEWS DETECTION

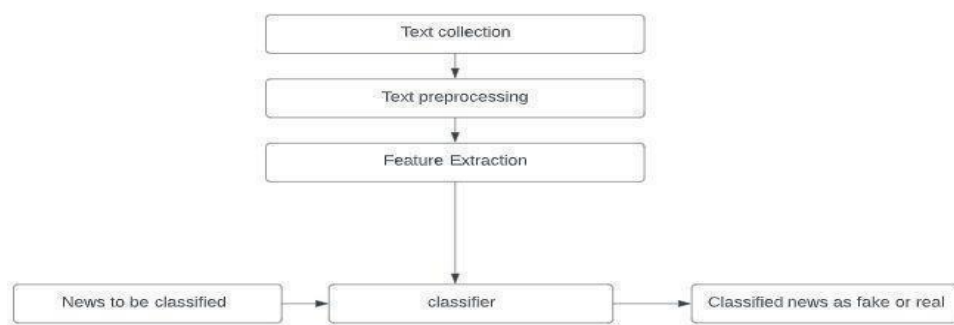


FIG 4.1 System Architecture for fake news detection

4.2.2 ARCHITECTURE/OVERALL DESIGN OF PROPOSED SYSTEM FOR CREDIT CARD FRAUD DETECTION

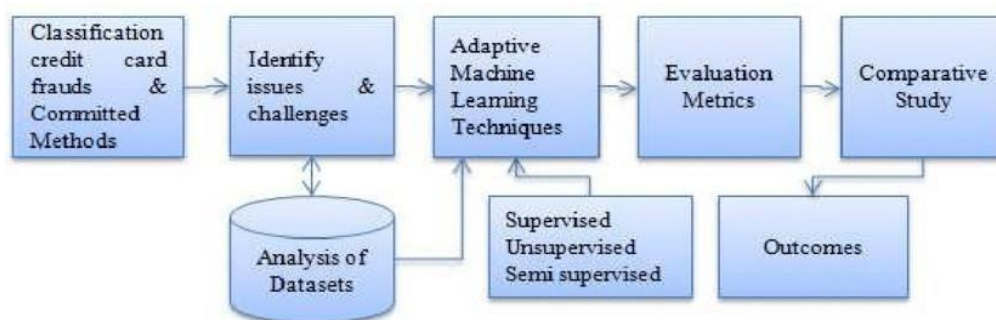


FIG 4.2 System Architecture for credit card fraud detection

4.3 DESCRIPTION OF SOFTWARE OF IMPLEMENTATION AND TESTING PLAN OF THE PROPOSED MODEL

This paragraph explains the process of designing a web page for a project that includes fake news detection and credit card scam prediction. The design of the web page involves creating form pages and result pages using HTML, CSS, and JavaScript to gather input from the user.

Fake News Detection: The software for implementing fake news detection using machine learning algorithms would require the following components:

Preprocessing module: This module would preprocess the news data by removing stop words, stemming, and vectorizing the text.

Machine Learning Model: The machine learning model would be trained on historical news data and would predict the likelihood of a news article being fake.

News Verification Module: This module would flag potentially fake news articles for further investigation by news verification professionals.

Data Storage Module: This module would store news data and the machine learning model to improve future predictions. The testing plan for fake news detection software would involve the following steps:

Unit Testing: The individual components of the software would be tested to ensure that they are working as expected.

Integration Testing: The individual components would be combined to test the interactions between them.

Functional Testing: The software would be tested to ensure that it meets the functional requirements, such as accurately detecting fake news articles.

Performance Testing: The software would be tested to ensure that it can handle large volumes of news articles and provide results in a timely manner.

Accuracy Testing: The software would be tested against a dataset of known fake news articles to determine its accuracy in detecting fake news.

Credit card scam detection, on the other hand, works on the principle of neural networking. Neural networking is a subset of machine learning that is inspired by the structure and function of the human brain. It involves creating artificial neural networks (ANNs) that can learn from and make predictions on input data.

The basic building block of an ANN is a neuron, which is a simple computational unit that receives input from other neurons or the external environment and processes it using a set of weights and activation function. The output of each neuron is passed to other neurons until the output layer is reached, where the final prediction is made.

The process of training an ANN involves adjusting the weights of each neuron based on the difference between the predicted output and the actual output, using a backpropagation algorithm. This process is repeated many times until the ANN can accurately predict the output for new input data.

ANNs can be used for a variety of tasks such as classification, regression, and pattern recognition. They have been successfully applied in a wide range of fields including computer vision, natural language processing, and speech recognition.

The entire system will be running on the ICP (Internet Computer Protocol) Chain, where the project will use test tera cycles for each run/edit. To access the tera cycles, the developers will need to create an ICP wallet either with Metamask or ICP wallet. After creating the wallet, they will tweet to ICP's official Twitter to request test ICP tokens, which will be sent to the wallet's mainnet account as proof to avoid phishing and confusion.

During the initial stages of development, the project will use test ICP tokens to run and deploy the site. After successful testing, the team will stake or buy ICP tokens until they run out of tera cycles for a burn. Since running the project doesn't cost much, the testing stage will be done with test ICP.

4.4 PROJECT MANAGEMENT PLAN

Project Scope: Define the project goals, objectives, deliverables, and timeline.

Team Structure: Determine the team structure, roles, and responsibilities. This should include the roles of project manager, developers, data scientists, and other team members.

Risk Management: Identify potential risks and create a plan to mitigate or avoid them. This includes risk assessment, risk identification, risk response planning, and risk monitoring.

Resource Management: Define the resources needed for the project, including hardware, software, and personnel. This includes the development environment, testing tools, and data sources.

Communication Plan: Establish a communication plan to ensure that everyone is kept informed and up-to-date on the project's progress. This includes regular team meetings, status reports, and stakeholder updates.

Budget and Schedule: Create a budget and schedule for the project. This includes setting milestones and deadlines for deliverables and defining the project's budget.

Testing and Quality Assurance: Define the testing strategy and quality assurance procedures. This includes unit testing, integration testing, and acceptance testing.

Deployment and Maintenance: Define the deployment strategy and maintenance plan. This includes identifying the production environment, the deployment process, and the ongoing maintenance and support.

Documentation: Define the documentation requirements and standards. This includes creating user manuals, technical documentation, and other documentation as required.

Change Management: Define the change management process. This includes change request management, configuration management, and version control.

4.5 TRANSITION/SOFTWARE TO OPERATIONS PLAN

- Jupyter notebook
- VS code :
 extensions: Remote WSL connection, motoko language extension
- Python >3.6
- ICP Wallet
- Motoko version >2
- Ubuntu version >0.9.3
- Windows Powershell
- Node js
- DFX
- Any latest version browser(Chrome recommended)
- NPM module

CHAPTER-5

IMPLEMENTATION DETAILS

5.1 DEVELOPMENT AND DEPLOYMENT SETUP

Step-1:

Choose an ML framework: There are many ML frameworks available, including TensorFlow, PyTorch, and scikit-learn ,the one used in the project is PyTorch.

Step-2:

Select a programming language: Python is the most popular language for ML.

Step-3:

Set up a development environment: Install the necessary software and packages, including the ML framework chosen, any necessary libraries, and an integrated development environment (IDE) like PyCharm or Jupyter.

Step-4:

Gather and preprocess data: ML models require a large amount of data to train. Collect and preprocess data in a format that the chosen ML framework can handle.

Step-5:

Develop and test the ML model: Use the chosen ML framework to develop a model and test it on a subset of the data.

Step-6:

Optimize the model: Fine-tune the model by adjusting hyperparameters, changing the architecture, or adding new features to improve its accuracy.

Step-7:

Deploy the model: Once the model is trained, deploy it to a production environment.

Step-8:

Monitor and maintain the model: Monitor the model's performance in production and make necessary updates or retraining to ensure its continued accuracy.

5.2 ALGORITHMS INVOLVED:

Logistic Regression : This type of statistical model (also known as *logit model*) is often used for classification and predictive analytics. Logistic regression estimates the probability of an event occurring, such as voted or didn't vote, based on a given dataset of independent variables. Since the outcome is a probability, the dependent variable is bounded between 0 and 1. In logistic regression, a logit transformation is applied on the odds—that is, the probability of success divided by the probability of failure. This is also commonly known as the log odds, or the natural logarithm of odds, and this logistic function is represented by the following formulas:

$$\text{Logit}(\pi) = 1/(1 + \exp(-\pi)) \quad (1)$$

$$\ln(\pi/(1-\pi)) = \text{Beta}_0 + \text{Beta}_1 X_1 + \dots + \text{Beta}_k X_k$$

In this logistic regression equation, $\text{logit}(\pi)$ is the dependent or response variable and x is the independent variable. The beta parameter, or coefficient, in this model is commonly estimated via maximum likelihood estimation (MLE). This method tests different values of beta through multiple iterations to optimize for the best fit of log odds. All of these iterations produce the log likelihood function, and logistic regression seeks to maximize this function to find the best parameter estimate.

Once the optimal coefficient (or coefficients if there is more than one independent variable) is found, the conditional probabilities for each observation can be calculated, logged, and summed together to yield a predicted probability. For binary classification, a probability less than .5 will predict 0 while a probability greater than 0 will predict 1. After the model has been computed, it's best practice to evaluate how well the model predicts the dependent variable, which is called goodness of fit. The Hosmer–Lemeshow test is a popular method to assess model.

Random Forest: Random Forest is a popular machine learning algorithm that is used for classification, regression, and other tasks in data science. It is a powerful tool that combines the strengths of decision trees and ensemble methods, providing high accuracy and robustness for large and complex datasets.

Random Forest is an ensemble method that creates multiple decision trees and combines their results to make predictions. Each tree is trained on a random subset of the data, and the algorithm selects the best features to split the data at each node. This randomness and diversity of trees allow the algorithm to reduce overfitting and increase accuracy.

Random Forest is a non-parametric algorithm, which means it does not make any assumptions about the distribution of the data. This makes it a versatile and flexible tool that can handle a wide range of data types and distributions. It is also a fast algorithm that can handle large datasets and high-dimensional data.

One of the strengths of Random Forest is its ability to handle missing data and outliers. The algorithm can handle missing values by using surrogate splits, which replace missing values with the most common value in the same class. It can also handle outliers by reducing their impact through the randomness of the trees.

Random Forest is also a robust algorithm that can handle noisy data and irrelevant features. The algorithm can select the most important features and discard the irrelevant ones, which reduces the risk of overfitting and improves accuracy.

Random Forest has many applications in data science, including classification, regression, clustering, feature selection, and anomaly detection. It is widely used in fields such as finance, marketing, healthcare, and environmental science, where large and complex datasets are common.

However, Random Forest also has some limitations and challenges that must be addressed. One of the challenges is the trade-off between accuracy and

interpretability. Random Forest can provide high accuracy, but it can be difficult to interpret the results, especially for complex models with many trees.

Another challenge is the selection of hyperparameters, such as the number of trees, the depth of the trees, and the number of features to select. These hyperparameters can affect the performance of the model, and finding the optimal values can be time-consuming and computationally expensive.

In conclusion, Random Forest is a powerful and versatile machine learning algorithm that provides high accuracy and robustness for large and complex datasets. It combines the strengths of decision trees and ensemble methods, allowing it to handle missing data, outliers, and irrelevant features.

It has many applications in data science and is widely used in various fields. However, it also has limitations and challenges that must be addressed, such as the trade-off between accuracy and interpretability and the selection of hyperparameters. Overall, Random Forest is a valuable tool for data scientists and machine learning practitioners, and it is likely to remain a popular algorithm in the years to come.

Decision Tree : A decision tree is a graphical representation of decision-making processes that provides a structured approach to identify the most optimal solution to a problem. Decision trees are often used in machine learning, data mining, and artificial intelligence to help understand and analyze complex data sets. A decision tree starts with a single node, which represents the problem or question being asked, and branches out into multiple paths, each of which represents a possible solution or outcome.

Decision trees have several benefits, including their ability to provide a clear and concise representation of complex data sets, their ease of use, and their flexibility. Decision trees can be used to analyze data from a variety of sources, including surveys, experiments, and observational studies.

One of the key features of a decision tree is that it is a hierarchical structure, which means that decisions are made sequentially, with each decision influencing the next one. At each node in the tree, a decision is made based on a specific set of criteria, and the tree branches out into two or more paths, each of which represents a possible outcome based on the decision that was made.

There are several different types of decision trees, including classification trees, regression trees, and clustering trees. Classification trees are used to classify data into different categories or groups, while regression trees are used to predict a continuous value, such as a person's height or weight. Clustering trees are used to group similar data points together based on their characteristics.

Decision trees are often used in conjunction with other analytical tools, such as data visualization software and statistical analysis tools, to provide a comprehensive understanding of complex data sets. They can also be used to identify patterns and trends in data, which can be used to inform business decisions and inform marketing strategies.

One of the most important factors to consider when using decision trees is the quality of the data being used. Decision trees rely heavily on accurate and reliable data, so it is important to ensure that the data being used is both accurate and complete. In addition, decision trees are only as good as the assumptions that are used to build them, so it is important to ensure that the assumptions being made are valid and appropriate for the data being analyzed.

In conclusion, decision trees are a powerful tool for analyzing complex data sets and making informed decisions. They provide a structured approach to decision-making that can help businesses and organizations identify the most optimal solution to a problem.

However, decision trees are only as good as the data and assumptions that are used to build them, so it is important to ensure that these are accurate and reliable.

By using decision trees in conjunction with other analytical tools and techniques, businesses can gain a comprehensive understanding of their data and make more informed decisions that lead to better outcomes.

RELU (Rectified Linear Unit) : The Rectified Linear Unit (ReLU) is a popular activation function used in artificial neural networks for deep learning. It is a simple and efficient way to introduce non-linearity to neural networks, which is essential for modeling complex data and achieving high accuracy.

ReLU is a piecewise linear function that returns zero for negative inputs and the input value for positive inputs. In other words, it "rectifies" the input by setting negative values to zero and leaving positive values unchanged. This non-linearity introduces the ability to learn complex and nonlinear relationships between the input and output.

One of the advantages of ReLU is its simplicity and efficiency. It is a simple mathematical function that is easy to implement and compute, which makes it fast and efficient for large-scale neural networks. This simplicity also makes it less prone to overfitting and improves the generalization ability of the model.

Another advantage of ReLU is its sparsity. By setting negative inputs to zero, ReLU introduces sparsity to the activation of neurons, which means that only a subset of the neurons are active at any given time. This sparsity can reduce the computational cost of the model and improve its performance.

ReLU also addresses the vanishing gradient problem, which is a common issue in deep neural networks. The vanishing gradient problem occurs when the gradients of the activation function become too small, making it difficult for the model to learn and update its weights. ReLU has a constant gradient of 1 for positive inputs, which means that it avoids the vanishing gradient problem and allows for faster convergence of the model.

However, ReLU also has some limitations and challenges that must be addressed. One of the challenges is the "dying ReLU" problem, which occurs when a neuron's activation is always zero, and therefore its weights are never updated. This problem can happen when the weights are initialized poorly, or when the learning rate is too high. To address this problem, variants of ReLU, such as Leaky ReLU, were proposed, which introduce a small slope for negative inputs to prevent neurons from dying.

Another challenge is the unbounded activation of ReLU. ReLU has an unbounded output, which means that the activation of neurons can become arbitrarily large. This can cause numerical instability and make the model sensitive to outliers. To address this problem, variants of ReLU, such as the Capped ReLU, were proposed, which limit the maximum activation of neurons.

In conclusion, ReLU is a popular and effective activation function used in artificial neural networks for deep learning. It is simple, efficient, and introduces non-linearity, sparsity, and solves the vanishing gradient problem. However, it also has some limitations and challenges, such as the dying ReLU problem and unbounded activation, which must be addressed. Overall, ReLU is a valuable tool for deep learning and is likely to remain a popular activation function in the future.

Sigmoid activation function : The sigmoid function is a popular activation function used in artificial neural networks for deep learning. It is a mathematical function that maps any input value to a value between 0 and 1, which is useful for modeling probabilities and binary classifications.

The sigmoid function is defined as:

$$\text{sigmoid}(x) = 1 / (1 + e^{-x}) \quad (2)$$

where x is the input value and e is the mathematical constant 2.71828.

The sigmoid function has several properties that make it useful for deep learning. Firstly, it is a smooth function that is differentiable, which allows for efficient

gradient descent optimization during training. Secondly, it is bounded between 0 and 1, which means that it can be used to model probabilities and binary classifications. Thirdly, it has a clear interpretation as a probability, which makes it easy to interpret and understand the output of the neural network.

However, the sigmoid function also has some limitations and challenges that must be addressed. One of the challenges is the vanishing gradient problem, which occurs when the gradient of the sigmoid function becomes too small, making it difficult for the model to learn and update its weights. Another challenge is that the output of the sigmoid function is not zero-centered, which can slow down the convergence of the model during training. Additionally, the sigmoid function can cause the problem of "saturation" when the input is very large or small, causing the gradient to be close to zero and making it difficult for the model to update its weights.

In conclusion, the sigmoid function is a popular activation function used in artificial neural networks for deep learning. It is useful for modeling probabilities and binary classifications and has a clear interpretation as a probability. However, it also has some limitations and challenges, such as the vanishing gradient problem and saturation, which must be addressed. Overall, the sigmoid function is a valuable tool for deep learning and is still widely used in various applications.

Adams optimizer : The Adam optimizer is a popular optimization algorithm used in deep learning to efficiently update the weights of neural networks during training. It is a variant of the stochastic gradient descent (SGD) algorithm that adapts the learning rate of each parameter based on the first and second moments of the gradients.

The name "Adam" stands for Adaptive Moment Estimation, which reflects its ability to adaptively estimate the first and second moments of the gradients. The algorithm computes the moving average of the gradient and its square, which are used to estimate the mean and variance of the gradients. These estimates are

then used to adjust the learning rate for each parameter, such that the learning rate is larger for parameters with smaller variance and smaller for parameters with larger variance.

The Adam optimizer has several advantages over traditional optimization algorithms, such as SGD. Firstly, it is efficient and requires less memory, as it stores only the moving averages of the gradients and its square, rather than the full gradients for each parameter. Secondly, it is adaptive and can adjust the learning rate for each parameter, which makes it less sensitive to the choice of hyperparameters and improves its performance on non-stationary problems.

Thirdly, it combines the benefits of both momentum-based optimization and RMSprop, which improves the convergence speed and stability of the optimizer. However, the Adam optimizer also has some limitations and challenges that must be addressed. One of the challenges is that it may converge to suboptimal solutions due to its adaptivity, which can cause the learning rate to become too small or too large. Additionally, the optimizer may not perform well on certain types of problems, such as those with sparse gradients or noisy data.

CHAPTER-6

RESULTS AND DISCUSSIONS

We used the datasets of fake news and credit card frauds for model training and evaluation based on machine learning classification algorithms. These models are deployed on the blockchain in essence icp tokens for better authenticity, security, cyber safety and privacy of the user. The models' overall accuracy was found to be around 86% with logistic regression and 94% with decision tree and random forest.

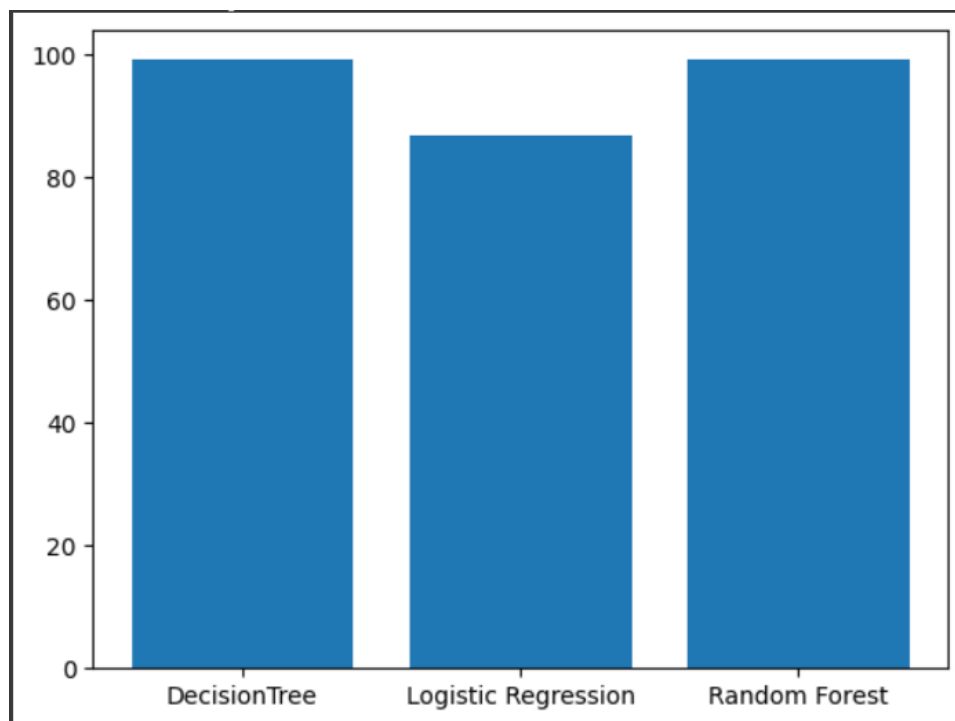


FIG 6.1 PERFORMANCE OF THE MODEL

The web 3.0 version of our website takes real time input from the user and undergoes multiple classification algorithms from the model we have built and provides the solution which has the better accuracy, reliability and feasibility.

The below picture represents the home page of the fraud detection .

The uniqueness of the website is it runs on web 3.0 technology in essence icp tokens for the better usability ,interactivity, security and authenticity. The aquamorph design depicts the two different features of the fraud detection ecosystem that is fake news detection and credit card fraud detection.

Whenever the user runs the website it burns icp tokens from the wallet. Currently we are using the test icp tokens in order to burn and update the site and also the features of it. This website has a unique canister id that comes from the unique block from the block chain itself.

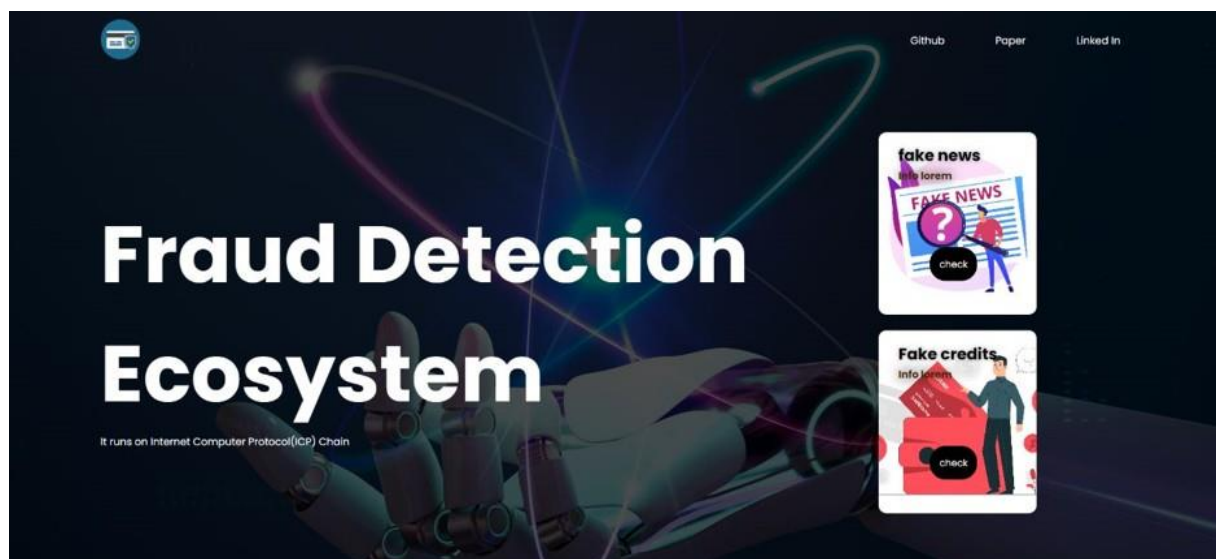


FIG 6.2 Home page

The below picture showcases the fake news detection model that we have developed which takes the input of news headlines or subheadings given by the user. The predict button takes the input and posts it to the pickle model we have developed using classification algorithms and detects whether the news headline is fake or not that has the better probability and reliability in real life scenarios.

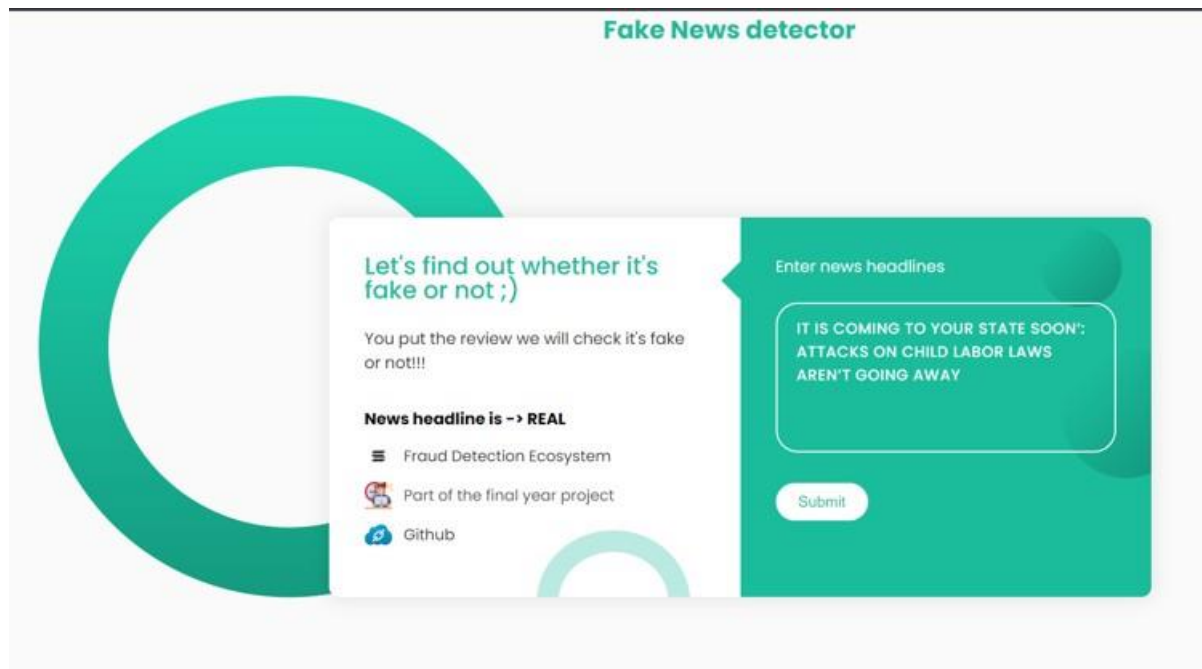


FIG 6.3 Fake news detector

The below picture themes around the idea of detecting whether the transaction undergone is fraudulent or legitimate by taking the inputs like type of transaction, total amount, previous balance, new balance , etc.

FIG 6.4 Fake credit card transaction detector

CHAPTER-7

CONCLUSION

7.1 CONCLUSION

In conclusion, fake news is a serious problem in today's world, with the potential to cause significant harm to individuals, communities, and even nations. Machine learning algorithms have emerged as a promising solution to the problem of fake news detection, enabling us to automatically classify news articles as either real or fake based on various features and characteristics.

However, it is important to note that no algorithm is perfect and there is still room for improvement in the accuracy and reliability of fake news detection models. Additionally, the issue of fake news is complex and multifaceted, requiring a holistic approach that involves media literacy, critical thinking, and responsible journalism in addition to technological solutions.

Therefore, while machine learning algorithms can be a useful tool in the fight against fake news, it is important to approach the problem with a nuanced understanding and a recognition of the broader social, cultural, and political factors that contribute to its spread.

Credit card fraud detection is a crucial application of machine learning and artificial intelligence that helps to prevent financial losses and protect consumers. With the increasing use of credit cards for online transactions, the risk of fraud has also increased, making it essential to develop effective fraud detection systems. Machine learning algorithms such as decision trees, random forests, and neural networks have shown promising results in detecting credit card fraud by analyzing patterns and identifying anomalies. These algorithms can process large amounts of data quickly and accurately, enabling the detection of fraud in real-time.

However, it is essential to keep updating and improving these algorithms as fraudsters continue to develop new methods and techniques. Also, it is crucial to balance fraud detection with minimizing false positives to avoid inconveniencing legitimate cardholders.

Overall, credit card fraud detection is a complex and ongoing process that requires a combination of advanced machine learning algorithms, skilled data analysts, and effective risk management strategies. By investing in these areas, financial institutions can stay ahead of fraudsters and protect their customers' financial interests.

The entire ecosystem we build on icp tokens ensures the security and cyber safety and there will be authenticity and originality of the data and models

7.2 FUTURE WORK

Improving accuracy: As with any machine learning model, there is always room for improving accuracy. Researchers can experiment with different algorithms, feature engineering techniques, and data sources to improve the accuracy of fake news detection models.

Multilingual support: Currently, most fake news detection models are trained on English-language data. However, fake news is a global issue, and models that can detect fake news in other languages will be important in the future.

Real-time detection: As fake news spreads quickly on social media, real-time detection is essential for minimizing the impact of fake news. Researchers can work on developing models that can detect fake news in real-time and alert users to the presence of fake news before it spreads too widely.

Privacy-preserving solutions: As fake news detection involves analyzing user data, privacy is a concern. Researchers can work on developing privacy-preserving solutions such as differential privacy, homomorphic encryption, and secure multi-party computation to ensure user privacy while still providing accurate fake news detection.

Integration with other blockchain applications: Fake news detection can be integrated with other blockchain applications such as digital identity, decentralized social media, and decentralized news platforms to provide a more comprehensive solution to the problem of fake news. This could involve developing smart contracts that verify the authenticity of news sources, or decentralized reputation systems that reward users for contributing to the detection of fake news.

Integration with more data sources : Currently, the fraud detection system could be improved by integrating with more data sources to improve the accuracy of the model. For example, adding transaction data from other financial institutions could provide more comprehensive data to train the model and make more accurate predictions.

Implementing real-time detection : Real-time detection would enable the system to detect fraud more quickly and reduce the risk of losses. The system could be designed to flag suspicious transactions and alert users immediately, allowing them to take action to prevent further fraud.

Developing a more robust model : As fraudsters continue to find new ways to commit fraud, it is important to continue improving the machine learning model to keep up with these new techniques. This could involve experimenting with different algorithms, data preprocessing techniques, and feature selection methods to improve the model's performance.

Collaboration with other financial institutions: Collaboration with other financial institutions could enable the sharing of fraud data and best practices. This would

allow for a more comprehensive and effective fraud detection system, benefiting all parties involved.

Developing explainable AI: As the use of AI for fraud detection becomes more widespread, there is a growing need for explainable AI. This would enable users to understand how the model arrived at its predictions and provide transparency in the decision-making process.

7.3 RESEARCH ISSUES

Challenges to Credit Card Fraud Detection:

Due to the increasing credit card frauds in the country, we have taken up credit card fraud detection for banking support as our research topic. However, there are certain challenges while collecting dataset and performing analysis of the credit card fraud detection:

Unavailability of Dataset: The biggest issue while conducting research on credit card fraud transactions is the unavailability of the dataset for research purposes. It is due to the fact that most banks and financial institutions do not want to release their customer's data as it breaches the privacy policy. If they release such contents, it means the sensitive information of the credit card holder comes into public eye.

Unbalanced Dataset: Credit card fraud data is unbalanced as it is highly skewed into legal and fraudulent datasets. Such data does not provide the most accurate results at times. As a result, it is one of the greatest challenges for any academic researcher.

Size of the Dataset: Today, credit cards are available in most households. As a result, there are millions of transactions happening on a daily basis. Keeping an account and track of such transactions becomes fairly tough. For conducting research on such large data, advanced computing techniques and speed is required. Therefore, all researchers are bound to work on the limited dataset.

Determining accurate evaluation parameters: Determining false-positives and false-negatives is a tough job at times. It is because using precision methods, accuracy rates, and test ratios might not deliver the desired results.

Changing Behavior of the Fraudster. As we mentioned earlier, the behavior of fraudsters is ever changing. They devise new methods and techniques every time a change in the transaction pattern is introduced to safeguard customers. Thus, the behavior of fraudsters is hard to predict even if the analysis is done by experts. As a result, the frauds are becoming more complex and difficult to understand. Even after facing these difficulties, we have devised a credit card fraud detection mechanism using supervised machine learning algorithms in our study.

Lack of labeled data: One of the major issues in fake news detection using machine learning is the availability of labeled data. Labeled data is crucial for training and evaluating machine learning models, but collecting and labeling large amounts of data is a time-consuming and expensive process. Moreover, the quality of the labeled data can also impact the performance of machine learning models.

Adversarial attacks: Fake news creators can intentionally add noise or manipulate the text to make it harder for machine learning models to detect fake news. This is known as adversarial attacks, and it can significantly reduce the accuracy of machine learning models.

Bias in data: Machine learning models can also be biased if the data used to train them is biased. For example, if the data used to train the model is biased towards a particular political ideology or cultural group, the model may not be able to accurately detect fake news related to other ideologies or cultural groups.

Generalizability: Machine learning models need to be trained on a diverse set of data to ensure that they can generalize well to new and unseen data. However, this can be challenging in the context of fake news detection since fake news can take many different forms and can be spread across different platforms and languages.

Interpretability: Machine learning models used for fake news detection can be difficult to interpret, making it challenging to understand how the model is making its decisions. This is a critical issue, as it can impact the trustworthiness of the model and its ability to be adopted by users and stakeholders.

Scalability: Machine learning models need to be scalable to be effective in detecting fake news. As the volume of data and the speed at which it is generated continue to increase, machine learning models need to be able to keep up with this demand.

Overall, these issues highlight the need for ongoing research and innovation in the field of fake news detection using machine learning.

Scalability: The Internet Computer blockchain is designed to be scalable, but as the number of users and transactions on the network increases, scalability remains a key challenge. Research is needed to explore ways to improve the scalability of the Internet Computer blockchain and ensure that it can handle the growing demand for ICP tokens.

Security: The security of the Internet Computer blockchain is critical, particularly when it comes to the development of ICP tokens. Researchers need to explore ways to ensure that the network is secure and that users can safely transact with ICP tokens without fear of hacking or other security breaches.

Interoperability: The Internet Computer blockchain is designed to be interoperable with other blockchains, but achieving true interoperability remains a challenge. Research is needed to explore ways to improve interoperability between the Internet Computer blockchain and other blockchains to ensure that ICP tokens can be used across different platforms.

Governance: As with any cryptocurrency, the governance of ICP tokens is critical. Researchers need to explore ways to ensure that the governance of ICP tokens is transparent, fair, and democratic, and that the interests of all stakeholders are taken into account.

Regulation: Cryptocurrencies are still largely unregulated, which can create uncertainty for developers and users alike. Research is needed to explore the

7.4 IMPLEMENTATION ISSUES

Data Collection and Preprocessing: Machine learning models require a large amount of data to be trained effectively. Therefore, collecting, cleaning, and preprocessing data for machine learning models can be a time-consuming process. In addition, it is important to ensure the data is of high quality, relevant, and unbiased.

Choosing the Right Algorithm: There are numerous machine learning algorithms available, each with its strengths and weaknesses. Choosing the right algorithm for a specific problem can be challenging, and it may require a lot of experimentation and analysis.

Model Training: Training machine learning models can take a significant amount of time and computational resources. This can be challenging when implementing web applications that need to provide real-time or near-real-time results.

Deployment and Integration: Deploying machine learning models to a web application requires integrating them into the existing infrastructure. This can be challenging when dealing with complex software systems or multiple programming languages.

Maintenance and Upgrades: Machine learning models require ongoing maintenance and upgrades to ensure they continue to perform optimally. This can involve monitoring and updating models regularly, which can be time-consuming and require specialized skills.

Privacy and Security: Machine learning models may use sensitive data, such as personal information, which raises privacy and security concerns. It is important to ensure that the data is stored securely and that appropriate measures are taken to prevent unauthorized access or misuse of the data.

The implementation of development on ICP tokens, which are used on the Internet Computer blockchain, may face a number of challenges, such as:

Technical complexity: Developing on the Internet Computer blockchain can be technically complex, as it involves working with a new blockchain technology and programming language. Developers may need to invest time and resources to learn how to work with the Internet Computer blockchain and to overcome any technical challenges that arise.

Resource limitations: Developing on the Internet Computer blockchain requires access to computing resources and storage space, which may be limited or expensive for some developers. This can create barriers to entry for smaller developers who may not have the necessary resources to develop on the network.

Integration with existing systems: Developers may need to integrate their ICP token-based applications with existing systems and infrastructure, which can be complex and time-consuming. This can create implementation issues .

User adoption: The success of ICP tokens and applications built on the Internet Computer blockchain will depend on user adoption. Developers may need to invest time and resources in marketing and user education to encourage adoption and usage of ICP tokens and applications.

Regulatory uncertainty: The regulatory landscape around cryptocurrencies is still evolving, and there may be regulatory challenges or uncertainties that arise when developing and implementing ICP token-based applications.

REFERENCES

[1] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu, "Fake News Detection on Social Media: A Data Mining Perspective" arXiv:1708.01967v3 [cs.SI], pg no-6,3 Sep 2017.

[2] H. Gupta, M. S. Jamal, S. Madisetty and M. S. Desarkar, "A framework for real-time spam detection in Twitter," 2018 10th International Conference on Communication Systems & Networks (COMSNETS), Bengaluru, pp. 380-383, 2018.

[3] M. L. Della Vedova, E. Tacchini, S. Moret, G. Ballarin, M. DiPierro and L. de Alfaro, "Automatic Online Fake News Detection Combining Content and Social Signals," 2018 22nd Conference of Open Innovations Association (FRUCT), Jyvaskyla, pp. 272-279, 2018.

[4] C. Buntain and J. Golbeck, "Automatically Identifying Fake News in Popular Twitter Threads," 2017 IEEE International Conference on Smart Cloud (SmartCloud), New York, NY, pp. 208-215, 2017.

[5] S. B. Parikh and P. K. Atrey, "Media-Rich Fake News Detection: A Survey," 2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR), Miami, FL, pp. 436-441, 2018.

[6] Scikit-Learn- Machine Learning In Python(Article).

[7] Khanam, Z. "Analyzing refactoring trends and practices in the software industry." Int. J. Adv. Res. Comput. Sci. 10, 0976–5697, 2018.

[8] Shankar M. Patil, Dr. Praveen Kumar, "Data mining model for effective data analysis of higher education students using MapReduce" IJERMT(Volume-6, Issue-4), April 2017 .

[9] Zhang, Jiawei, Bowen Dong, and S. Yu Philip. "Fakedetector: Effective fake news detection with deep diffusive neural network." 2020 IEEE 36th International Conference on Data Engineering(ICDE). IEEE, 2020.

[10] Khanam Z. and Agarwal S. Map-reduce implementations: Survey and Performance comparison, International Journal of Computer Science & Information Technology (IJCSIT) Vol 7, No 4, August 2015.

APPENDIX

A.SOURCE CODE

```

  ▾ Import library

import pandas as pd
import numpy as np
import itertools

[1]

# !pip install pandas      # download and install pandas library

[2]

df = pd.read_csv("news.csv")

[3]

df.head()

[4]
...

df.shape

[5]
... (6335, 4)
```

```
df.isnull().sum()
```

[6]

```
... Unnamed: 0    0
     title      0
     text       0
     label      0
     dtype: int64
```

```
labels = df.label
```

[7]

```
labels.head()
```

[8]

```
... 0    FAKE
    1    FAKE
    2    REAL
    3    FAKE
    4    REAL
    Name: label, dtype: object
```

```
from sklearn.model_selection import train_test_split
```

[9]

```
x_train, x_test, y_train, y_test = train_test_split(df["text"], labels, test_size = 0.2, random_state = 20)
```

[10]

```
x_train.head()
```

[11]

```
... 4741  NAIROBI, Kenya – President Obama spoke out Sun...
     2089  Killing Obama administration rules, dismantlin...
     4074  Dean Obeidallah, a former attorney, is the hos...
     5376  WashingtonsBlog \nCNN's Jake Tapper hit the ...
     6028  Some of the biggest issues facing America this...
     Name: text, dtype: object
```

```
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import PassiveAggressiveClassifier
```

[12]

```
# initilise a TfidfVectorizer
vector = TfidfVectorizer(stop_words='english', max_df=0.7)
```

[13]

```
# fit and tranform
tf_train = vector.fit_transform(x_train)
tf_test = vector.transform(x_test)
```

[14]

```

# initialise a PassiveAggressiveClassifier
pac = PassiveAggressiveClassifier(max_iter=50)
pac.fit(tf_train, y_train)
[15]
... PassiveAggressiveClassifier(max_iter=50)

# prediction the tst dataset
from sklearn.metrics import accuracy_score, confusion_matrix
y_pred = pac.predict(tf_test)
[16]

score = accuracy_score(y_test, y_pred)
[17]

print(f"Accuracy : {round(score*100,2)}%")
[18]
... Accuracy : 94.63%

# confusion metrics
confusion_matrix(y_test, y_pred, labels=['FAKE', 'REAL'])
[19]
... array([[621, 27],
          [ 41, 578]], dtype=int64)

```



```

import torch
import torch.nn as nn
import numpy as np

class LogisticRegression(nn.Module):
    def __init__(self, no_of_features):
        super(LogisticRegression, self).__init__()
        self.linear1 = nn.Linear(no_of_features, 10)
        self.relu = nn.ReLU()
        self.linear2 = nn.Linear(10, 1)
        self.sigmoid = nn.Sigmoid()

    def forward(self, targets_train ):
        out = self.linear1(targets_train)
        out = self.relu(out)
        out = self.linear2(out)
        out = self.sigmoid(out)

        return out

model = LogisticRegression(no_of_features=4)

```

```

def predict_value(inputs):
    if len(inputs)==4:
        inputs = np.array(inputs)
        inputs = torch.from_numpy(inputs.astype(np.float32))
        main_model = torch.load("credit_card_4-1.pth")
        pred = main_model(inputs)
        value = pred.item()
        if value < 0.5:
            return "No Fraud"
        else:
            return "Fraud Detected"
    else:
        return "S0me errors in input found!!"

```

```

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import torch.nn as nn
import torch
import os

```

[2]

```

data = pd.read_csv("credit.csv")
data.head()

```

[3]

...

```

input_vars = data[["amount","oldbalanceOrig","newbalanceOrig","oldbalanceDest"]]
target_vars = data[["isFraud"]]
print(len(input_vars),len(target_vars))

```

[4]

... 18213 18213

```

input_ar = np.array(input_vars)
target_ar = np.array(target_vars)

print(input_ar.shape)
print(target_ar.shape)

```

[5]

```
[5]
... (18213, 4)
(18213, 1)

[6]
from sklearn.model_selection import train_test_split

[7]
inputs_train,inputs_test,targets_train,targets_test = train_test_split(input_ar, target_ar, test_size=0.20)

[8]
inputs_train = torch.from_numpy(inputs_train.astype(np.float32))
inputs_test = torch.from_numpy(inputs_test.astype(np.float32))
targets_train = torch.from_numpy(targets_train.astype(np.float32))
targets_test = torch.from_numpy(targets_test.astype(np.float32))

[9]
print(inputs_train.shape)
print(inputs_test.shape)
print(targets_train.shape)
print(targets_test.shape)

... torch.Size([14570, 4])
torch.Size([3643, 4])
torch.Size([14570, 1])
torch.Size([3643, 1])
```

```
class LogisticRegression(nn.Module):
    def __init__(self,no_of_features):
        super(LogisticRegression, self).__init__()
        self.linear1 = nn.Linear(no_of_features,10)
        self.relu = nn.ReLU()
        self.linear2 = nn.Linear(10,1)
        self.sigmoid = nn.Sigmoid()

    def forward(self, targets_train ):
        out = self.linear1(targets_train)
        out = self.relu(out)
        out = self.linear2(out)
        out = self.sigmoid(out)

        return out

model = LogisticRegression(no_of_features=4)

[10]
loss_fn = nn.BCELoss()
optimizer = torch.optim.Adam(model.parameters(), lr=0.001)

[11]

num_epochs = 15*5

[12]
```

```

[14] from tqdm.notebook import tqdm

loss_hist = []
acc_hist = []
for epoch in tqdm(range(num_epochs)):
    targets_preds = model(inputs_train)
    loss = loss_fn(targets_preds, targets_train)

    loss.backward()

    optimizer.step()
    optimizer.zero_grad()

    if (epoch+1)%3 == 0:
        with torch.no_grad():
            targets_preds = model(inputs_test)
            targets_preds_cls = targets_preds.round()
            acc = targets_preds_cls.eq(targets_test).sum() / float(inputs_test.shape[0])
            loss_hist.append(loss.item())
            acc_hist.append(acc)
            print(f'Epoch: {epoch+1}, Loss: {loss.item():.4f}, ACC: {acc}')
[15]

```

```

... 0%|          | 0/75 [00:00<?, ?it/s]

Epoch: 3, Loss: 51.4954, ACC: 0.49519625306129456
Epoch: 6, Loss: 51.0482, ACC: 0.49903926253318787
Epoch: 9, Loss: 50.6305, ACC: 0.5015097260475159
Epoch: 12, Loss: 50.3815, ACC: 0.5034312605857849
Epoch: 15, Loss: 50.1882, ACC: 0.5039802193641663
Epoch: 18, Loss: 50.0591, ACC: 0.5050782561302185
Epoch: 21, Loss: 49.9540, ACC: 0.5069997310638428
Epoch: 24, Loss: 49.7545, ACC: 0.508921205997467
Epoch: 27, Loss: 49.6730, ACC: 0.5108427405357361
Epoch: 30, Loss: 49.5175, ACC: 0.5119407176971436
Epoch: 33, Loss: 49.4032, ACC: 0.5124897360801697
Epoch: 36, Loss: 49.3002, ACC: 0.5133131742477417
Epoch: 39, Loss: 49.1400, ACC: 0.5155091881752014
Epoch: 42, Loss: 49.0011, ACC: 0.5155091881752014
Epoch: 45, Loss: 48.9342, ACC: 0.5163326859474182
Epoch: 48, Loss: 48.9113, ACC: 0.5166071653366089
Epoch: 51, Loss: 48.8662, ACC: 0.517156183719635
Epoch: 54, Loss: 48.8807, ACC: 0.5168817043304443
Epoch: 57, Loss: 48.7829, ACC: 0.5204501748085022
Epoch: 60, Loss: 48.1257, ACC: 0.5245676636695862
Epoch: 63, Loss: 47.6560, ACC: 0.5264891386032104
Epoch: 66, Loss: 47.3811, ACC: 0.5297831296920776
Epoch: 69, Loss: 46.9978, ACC: 0.5344496369361877
Epoch: 72, Loss: 46.5214, ACC: 0.5369201302528381
Epoch: 75, Loss: 46.8758, ACC: 0.5317046642303467

```

```

[16] torch.save(model,"credit_card_4-1.pth")

def predict_value(inputs):
    if len(inputs)==4:
        inputs = np.array(inputs)
        inputs = torch.from_numpy(inputs.astype(np.float32))
        main_model = torch.load("credit_card_4-1.pth")
        pred = main_model(inputs)
        value = pred.item()
        if value < 0.5:
            print("No Fraud")
        else:
            print("Its a Fraud")
    else:
        print("Length not matchd!")

[17]

input_1 = [9839.64,170136.0,160296.36,0.0]
predict_value(input_1)

[18]

... No Fraud

```

```

import pandas as pd
import numpy as np
data = pd.read_csv("credit.csv")
print(data.head())

[15]

...
  Unnamed: 0  step  type  amount  nameOrig  oldbalanceOrig  \
0           0    7  PAYMENT    6159.12   C41646355         0.00
1           1   69  CASH_OUT   353313.55  C1330110958      353313.55
2           2    3  PAYMENT    19877.65  C1088381072      24560.00
3           3   85  CASH_OUT  1529614.71  C955071434     1529614.71
4           4  632  TRANSFER    82209.71  C2014216295      82209.71

      newbalanceOrig  nameDest  oldbalanceDest  newbalanceDest  isFraud  \
0           0.00  M1300689712           0.00           0.00         0
1           0.00  C857281092     2808260.05     3161573.60         1
2       4682.35  M146842175           0.00           0.00         0
3           0.00  C865727492           0.00     1529614.71         1
4           0.00  C2066467731           0.00           0.00         1

      isFlaggedFraud
0           0
1           0
2           0
3           0
4           0

```



```

print(data.isnull().sum())
[16]
... Unnamed: 0      0
    step          0
    type          0
    amount        0
    nameOrig      0
    oldbalanceOrg 0
    newbalanceOrig 0
    nameDest      0
    oldbalanceDest 0
    newbalanceDest 0
    isFraud       0
    isFlaggedFraud 0
    dtype: int64

# Exploring transaction type
print(data.type.value_counts())
[17]
... PAYMENT      5510
    CASH_OUT     5409
    TRANSFER     4997
    CASH_IN      1951
    DEBIT         346
    Name: type, dtype: int64

```

```

type = data["type"].value_counts()
transactions = type.index
quantity = type.values

import plotly.express as px
figure = px.pie(data,
               values=quantity,
               names=transactions, hole = 0.5,
               title="Distribution of Transaction Type")
figure.show()
[18]
</>

# Checking correlation
correlation = data.corr()
print(correlation["isFraud"].sort_values(ascending=False))
[19]
... isFraud      1.000000
    step        0.780175
    amount      0.386424
    oldbalanceOrg 0.131053
    isFlaggedFraud 0.032720
    newbalanceDest 0.025014
    Unnamed: 0    0.002744
    oldbalanceDest -0.065286
    newbalanceOrig -0.169873
    Name: isFraud, dtype: float64

```

```
data["type"] = data["type"].map({"CASH_OUT": 1, "PAYMENT": 2,
                                "CASH_IN": 3, "TRANSFER": 4,
                                "DEBIT": 5})
data["isFraud"] = data["isFraud"].map({0: "No Fraud", 1: "Fraud"})
print(data.head())
```

[20]

	Unnamed: 0	step	type	amount	nameOrig	oldbalanceOrg	\
0	0	7	2	6159.12	C41646355	0.00	
1	1	69	1	353313.55	C1330110958	353313.55	
2	2	3	2	19877.65	C1088381072	24560.00	
3	3	85	1	1529614.71	C955071434	1529614.71	
4	4	632	4	82209.71	C2014216295	82209.71	

	newbalanceOrig	nameDest	oldbalanceDest	newbalanceDest	isFraud	\
0	0.00	M1300689712	0.00	0.00	No Fraud	
1	0.00	C857281092	2808260.05	3161573.60	Fraud	
2	4682.35	M146842175	0.00	0.00	No Fraud	
3	0.00	C865727492	0.00	1529614.71	Fraud	
4	0.00	C2066467731	0.00	0.00	Fraud	

	isFlaggedFraud
0	0
1	0
2	0
3	0
4	0

```
# splitting the data
from sklearn.model_selection import train_test_split
x = np.array(data[["type", "amount", "oldbalanceOrg", "newbalanceOrig"]])
y = np.array(data[["isFraud"]])
```

[21]

```
# training a machine learning model
from sklearn.tree import DecisionTreeClassifier
xtrain, xtest, ytrain, ytest = train_test_split(x, y, test_size=0.10, random_state=42)
model = DecisionTreeClassifier()
model.fit(xtrain, ytrain)
print(model.score(xtest, ytest))
```

[22]

```
0.9917672886937431
```

...

```
# prediction
#features = [type, amount, oldbalanceOrg, newbalanceOrig]
features = np.array([[4, 9000.60, 9000.60, 0.0]])
print(model.predict(features))
```

[23]

```
['Fraud']
```

...

```
import joblib
joblib.dump(model, 'model.joblib')

... ['model.joblib']
```

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <link rel="stylesheet" href="home.css">
  <style>
    @import url("https://fonts.googleapis.com/css2?family=Poppins:wght@100;200;300;400;500;600;700;800&display=swap");
  *{
    margin: 0;
    padding: 0;
    font-family: "Poppins", sans-serif;
  }
  .container{
    width: 100%;
    height: 100vh;
    background-image: linear-gradient(rgba(0,0,0,0.7), rgba(0,0,0,0.7)), url(../static/img/background.jpg);
    background-position: center;
    background-size: cover;
    padding-left: 8%;
    padding-right: 8%;
    box-sizing: border-box;
  }
  .navbar{
    height: 12%;
    display: flex;
    align-items: center;
  }
  .logo{
    width: 50px;
    cursor: pointer;
  }
  .menu-icon{
    width: 30px;
    cursor: pointer;
    margin-left: 40px;
  }
}
```

```

}
.logo{
  width: 50px;
  cursor: pointer;
}
.menu-icon{
  width: 30px;
  cursor: pointer;
  margin-left: 40px;
}
nav{
  flex: 1;
  text-align: right;
}
nav ul li{
  list-style: none;
  display: inline-block;
  margin-left: 60px;
}
nav ul li a{
  text-decoration: none;
  color: #fff;
  font-size: 13px;
}
.row{
  display: flex;
  height: 88%;
  align-items: center;
}
h1{
  color: #fff;
  font-size: 100px;
}
p{
  color: #fff;
  font-size: 13px;
  line-height: 20px;
}
```



```

    color: #fff;
    font-size: 13px;
    line-height: 20px;
}
button{
    fill: black;
    width: 60px;
    color: #fff;
    font-size: 12px;
    padding: 12px 0;
    background: #000;
    border: 0;
    border-radius: 20px;
    outline: none;
    margin-top: 80px;
    margin-left: 40px;
    transition: transform 0.5s;
}
a{
    text-decoration: none;
    color: #fff;
}
.card{
    width: 200px;
    height: 230px;
    display: inline-block;
    border-radius: 10px;
    padding: 15px 25px;
    box-sizing: border-box;
    cursor: pointer;
    background-position: center;
    margin: 10px 15px;
    background-size: cover;
    transition: transform 0.5s;
}
.card-1{
    background-image: url(pic-1.jpg);

```

```

.card-1{
    background-image: url(pic-1.jpg);
}
.card-2{
    background-image: url(pic-3.jpg);
}
.card-3{
    background-image: url(pic-3.jpg);
}
.card-4{
    background-image: url(pic-3.jpg);
}
.card:hover{
    transform: translateY(-10px);
}
h5{
    color: #000;
    text-shadow: 0 0 5px #999;
    font-size: 20px;
}
.card p{
    color: rgb(71, 35, 5);
    text-shadow: 0 0 15px #000;
    font-size: 15px;
    font-weight: 700;
}
button:hover{
    transform: translateX(10px);
}
</style>
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Fraud detection Ecosystem</title>
</head>
<body>
<div class="container">
    <div class="navbar">

```

```

<div class="container">
  <div class="navbar">
    
    <nav>
      <ul>
        <li><a href="#">Github</a></li>
        <li><a href="#">Paper</a></li>
        <li><a href="#">Linked In</a></li>
      </ul>
    </nav>
  </div>
  <div class="row">
    <div class="col">
      <h1>
        Fraud Detection Ecosystem
      </h1>
      <p>It runs on Internet Computer Protocol(ICP) Chain</p>
    </div>
    <div class="col">
      <div class="card card-1">
        <h5>fake news</h5>
        <p></p>
        <button> <a href="http://127.0.0.1:5000">check</a> </button>
      </div>

      <div class="card card-3">
        <h5>Fake credits</h5>
        <p></p>
        <button> <a href="http://localhost:8501">check</a> </button>
      </div>
    </div>
  </div>
</div>
</body>

```

```

<script>const inputs = document.querySelectorAll(".input");

function focusFunc() {
  let parent = this.parentNode;
  parent.classList.add("focus");
}

function blurFunc() {
  let parent = this.parentNode;
  if (this.value == "") {
    parent.classList.remove("focus");
  }
}

inputs.forEach((input) => {
  input.addEventListener("focus", focusFunc);
  input.addEventListener("blur", blurFunc);
});
</script>
</html>

```

[illegible]


```

nikhil@DESKTOP-2P9687U:~/icp_projects/fraud_detection_ecosystem$ dfx deploy
Deploying all canisters.
All canisters have already been created.
Building canisters...
Building frontend...
npm notice
npm notice New major version of npm available! 8.19.3 -> 9.6.4
npm notice Changelog: <https://github.com/npm/cli/releases/tag/v9.6.4>
npm notice Run 'npm install -g npm@9.6.4' to update!
npm notice

Installing canisters...
Upgrading code for canister fraud_detection_ecosystem, with canister_id rrrkah-fqaaa-aaaaa-aaaa-cai
Module hash 53d873bc405439ece5b372ac5627563518bd7081b58543ddfec3e0956e0edc5 is already installed.
Upgrading code for canister fraud_detection_ecosystem_assets, with canister_id rylj3-tyaaa-aaaaa-aaaba-cai
Module hash e0df779f65fe44893d8991bef0f9af442bfff019b79ec756eface2b58beec236f is already installed.
Uploading assets to asset canister...
Starting batch.
Staging contents of new and changed assets:
/ favicon.ico (15406 bytes) sha 4e8d31b50ff59695389d94e393d299c5693405a12f6ccdd08c31bcf9b58db2d4 is already installed
/ index.js.map (683174 bytes) sha ae23c516ec9685c1138cf243833b536bb8978df35e50dcb0808f1ab6306c1e0 is already installed
/ index.js.map (gzip) (156858 bytes) sha 3e6f29fd32c45f6eae315c98ad07c1ec7e8dedd9b7e295d5525de6cc7643bebd is already installed
/ img/pic-2.jpg (207196 bytes) sha 4467a6b11ce9bb214dfb4ffe88ba6ff19cdda6041af0df2977a411d0adfe857d is already installed
/ img/pic-1.jpg (3090465 bytes) sha 098b376fcf49dc8a3071b9d5a34cd51dabc80c1a10ccc7392c25d10f59cf957a is already installed
/ img/location.png (28813 bytes) sha b86cf0ca2d8165e95562a32099deabaf8468e7425c5cd702f67ed52684ab12a is already installed
/ img/pic-3.jpg (572181 bytes) sha 7189aaa675ec0e1b3db0d05de58eb7cb9eea3d53d89d04359aba8536d10ce52b is already installed
/ img/pic-4.jpg (253278 bytes) sha 6e8b68d55ff2b60ec071ee2baf34e6234bb352206b5caa448dd93043fb5b51c is already installed
/ img/menu.png (920 bytes) sha e538fbaf9e6fba2bbc794bd934d321b9819228e01ad716baa7cc6fdffdd817a4f is already installed
/ img/email.png (36471 bytes) sha 786cb9f69fd054b3ee25907787d85be3be17546b899dcda81d73189339dd659e is already installed
/ img/background.jpg (12211773 bytes) sha 292e0bd3e560aa8309dcf99f3bd40c1a6f8983e49d7c2faec17368e8bae610bc is already installed
/ img/phone.png (26282 bytes) sha 369f9c6a6f62e23a9d1c8e9ae78d832824ab97464cdfb73208119cf8b28823bd is already installed
/ img/shape.png (31687 bytes) sha 83cbb11469fc327509f2613b2e58bda601333b7ecf3aa42c1d284f520f6b4d03 is already installed
/ index.html (4143 bytes) sha df1f552f0f8c68b84147aa33a62d87c6385455e1b3ecdff90fbcd883c46198466 is already installed
/ index.html (gzip) (1460 bytes) sha af861eda6aa53eafa0a18d0a9782175b37a1d70df1421955d829ac99ff0bf17c is already installed
/ sample-asset.txt (24 bytes) sha 2d523f5aaeb195da24dcff49b0d560a3d61b8af859cee78f4cff0428963929e6 is already installed
/ pic-1.jpg (3090465 bytes) sha 098b376fcf49dc8a3071b9d5a34cd51dabc80c1a10ccc7392c25d10f59cf957a is already installed
/ location.png (28813 bytes) sha b86cf0ca2d8165e95562a32099deabaf8468e7425c5cd702f67ed52684ab12a is already installed
/ pic-3.jpg (572181 bytes) sha 7189aaa675ec0e1b3db0d05de58eb7cb9eea3d53d89d04359aba8536d10ce52b is already installed
/ logo.png (25397 bytes) sha d17961d77e6d12c358cdd952217635766004004862e95b4f02aa453cecc4d2ff is already installed
/ menu.png (920 bytes) sha e538fbaf9e6fba2bbc794bd934d321b9819228e01ad716baa7cc6fdffdd817a4f is already installed
/ index.js (637195 bytes) sha 5fa531e869e8c1e0fd1c1f30a9bd641537b34e6b6abf63f113fbc1c6ff789f3 is already installed
/ index.js (gzip) (152813 bytes) sha 53a511fc8b209a410a515e2279ff7952c203a330afeaf762ecd863c61ad5aae2 is already installed
/ email.png (36471 bytes) sha 786cb9f69fd054b3ee25907787d85be3be17546b899dcda81d73189339dd659e is already installed
/ background.jpg (12211773 bytes) sha 292e0bd3e560aa8309dcf99f3bd40c1a6f8983e49d7c2faec17368e8bae610bc is already installed
/ main.css (0 bytes) sha e3b0c44298fc1c149afb4c8996fb92427ae41e4649b934ca495991b7852b855 is already installed

/ img/background.jpg (12211773 bytes) sha 292e0bd3e560aa8309dcf99f3bd40c1a6f8983e49d7c2faec17368e8bae610bc is already installed
/ img/phone.png (26282 bytes) sha 369f9c6a6f62e23a9d1c8e9ae78d832824ab97464cdfb73208119cf8b28823bd is already installed
/ img/shape.png (31687 bytes) sha 83cbb11469fc327509f2613b2e58bda601333b7ecf3aa42c1d284f520f6b4d03 is already installed
/ index.html (4143 bytes) sha df1f552f0f8c68b84147aa33a62d87c6385455e1b3ecdff90fbcd883c46198466 is already installed
/ index.html (gzip) (1460 bytes) sha af861eda6aa53eafa0a18d0a9782175b37a1d70df1421955d829ac99ff0bf17c is already installed
/ sample-asset.txt (24 bytes) sha 2d523f5aaeb195da24dcff49b0d560a3d61b8af859cee78f4cff0428963929e6 is already installed
/ pic-1.jpg (3090465 bytes) sha 098b376fcf49dc8a3071b9d5a34cd51dabc80c1a10ccc7392c25d10f59cf957a is already installed
/ location.png (28813 bytes) sha b86cf0ca2d8165e95562a32099deabaf8468e7425c5cd702f67ed52684ab12a is already installed
/ pic-3.jpg (572181 bytes) sha 7189aaa675ec0e1b3db0d05de58eb7cb9eea3d53d89d04359aba8536d10ce52b is already installed
/ logo.png (25397 bytes) sha d17961d77e6d12c358cdd952217635766004004862e95b4f02aa453cecc4d2ff is already installed
/ menu.png (920 bytes) sha e538fbaf9e6fba2bbc794bd934d321b9819228e01ad716baa7cc6fdffdd817a4f is already installed
/ index.js (637195 bytes) sha 5fa531e869e8c1e0fd1c1f30a9bd641537b34e6b6abf63f113fbc1c6ff789f3 is already installed
/ index.js (gzip) (152813 bytes) sha 53a511fc8b209a410a515e2279ff7952c203a330afeaf762ecd863c61ad5aae2 is already installed
/ email.png (36471 bytes) sha 786cb9f69fd054b3ee25907787d85be3be17546b899dcda81d73189339dd659e is already installed
/ background.jpg (12211773 bytes) sha 292e0bd3e560aa8309dcf99f3bd40c1a6f8983e49d7c2faec17368e8bae610bc is already installed
/ main.css (0 bytes) sha e3b0c44298fc1c149afb4c8996fb92427ae41e4649b934ca495991b7852b855 is already installed
/ phone.png (26282 bytes) sha 369f9c6a6f62e23a9d1c8e9ae78d832824ab97464cdfb73208119cf8b28823bd is already installed

Committing batch.
Deployed canisters.
URLs:
Frontend:
  fraud_detection_ecosystem_assets: http://127.0.0.1:8000/?canisterId=rylj3-tyaaa-aaaaa-aaaba-cai
Candid:
  fraud_detection_ecosystem: http://127.0.0.1:8000/?canisterId=r7inp-6aaaa-aaaaa-aaabq-cai&id=rrkah-fqaaa-aaaaa-aaaaq-cai

```


B.RESEARCH PAPER

FRAUD DETECTION ECOSYSTEM ON INTERNET COMPUTER PROTOCOL

ADARI VANDIK
Department of CSE
Sathyabama Institute of Science
and Technology, Chennai

ALURU MUNIVEDA INDRA NIKHIL
Department of CSE
Sathyabama Institute of
Science and Technology, Chennai

Dr. M.P. VAISHNNAVE
Department of CSE
Sathyabama Institute of
Science and Technology, Chennai

Abstract-Financial services, online shopping, and news are all undergoing digital transformations of their services, and overall business models. Part of the goal of this digitalization in these fields is to automate the majority of the manual work in payment processing, user interaction, security, and integrating the workflows of involved service providers. The work presented in this paper is centered on fraud discovery and steps to prevent and detect it. Financial transaction fraud detection, and news authenticity have all become major considerations in these fields. Fraud is becoming more prevalent as modern technology and global communication advance, leading to significant societal costs and the loss of authenticity and truth. Due to the requirement for quick processing time, instant payment (IP) transactions, sharing fake news, online present new challenges for fraud detection. The paper looks into the use of artificial intelligence in fraud detection. The main contributions of our work are (a) a business and literature analysis of problem relevance, and (b) a proposal for technological support for using AI in fraud detection of fake news, and credit card scams. (c) a feasibility study of various fraud detection methods. (d) a working system model.

Keywords : Fraud detection, Fake news, Credit card scams, Machine Learning

INTRODUCTION

Financial services, online shopping, and news are all undergoing digital transformations of their services, and overall business models. Part of the goal of digitalization in these fields is to automate the majority of the manual work in payment processing, user interaction, security, and integrating the workflows of involved service providers.

Fake news is a type of yellow press that intentionally spreads misinformation or hoaxes via traditional print news media as well as recent online social media. Since the publication of the "Great Moon Hoax" in 1835, there has been fake news. Fake news for various commercial and political purposes has appeared in large numbers and spread throughout the online world in recent years, owing to the booming development of online social networks. With deceptive words, online social network users can easily become infected by these online fake news, which has already had massive effects on the offline society. During the 2016 US presidential election, various types of fake news about the candidates were widely spread on online social networks, which may have had a significant impact on the election results. According to a post-election statistical report, online social networks accounted for more than 41.8% of the fake news data traffic in the election, far exceeding the data traffic shares of traditional TV/radio/print media and online search engines. One important goal in improving the trustworthiness of information in online social networks is to identify fake news in real time, which will be the focus of this paper.

Credit card fraud is still a major problem for theft and fraud committed with a payment card, such as a credit or debit card. Many fraud detection algorithms are widely used in industry to combat this. Card fraud can occur through both the theft of the physical card and the compromise of the card, including skimming, breach, and account takeover, which would otherwise appear to be a legitimate transaction. According to the Global Payments Report 2015, credit cards were the most commonly used payment method worldwide in 2014, surpassing e-wallets and bank transfers. The number of fraud cases has steadily increased in tandem with the rise in credit card usage. The rise in credit card fraud is having a significant impact on the financial industry. In 2015, global credit card fraud totaled a staggering USD 21.84 billion. In this paper, we explore the application of various linear and nonlinear statistical modeling and machine learning models on credit card transaction data, Fake news. We deploy the model on the blockchain network, i.e. ICP (Internet Computer Protocol), in a user-friendly and interactive design to improve security and avoid phishing.

Literature survey

In previous studies, a variety of techniques, including supervised, unsupervised, and hybrid algorithms, were used to detect fraud. The types and patterns of fraud are constantly evolving. It is critical to understand fraud detection technology thoroughly. We'll go over the machine learning models, algorithms, and fraud detection models that have been used in previous research in this section. The author investigated data when dealing with massive amounts of data.

Literature survey of Credit card scams

[1] Prajal Save et al has proposed a model that is based on a decision tree and a combination of the Luhn and Hunt algorithms. The Luhn algorithm is used to determine whether or not an incoming transaction is fraudulent. It validates credit card numbers using the credit card number as input. Address Mismatch and Degree of Outlierness are used to determine how far each incoming transaction deviates from the cardholder's normal profile. The final step is to use Bayes Theorem to strengthen or weaken the general belief, followed by recombination of the calculated probability with the initial belief of fraud using an advanced combination heuristic.

[2] Vimala Devi et al. Three machine-learning algorithms were presented and implemented to detect counterfeit transactions. Many metrics are used to assess the performance of classifiers or predictors, including the Vector Machine, Random Forest, and Decision Tree. These metrics can be classified as either prevalence-dependent or prevalence-independent. Furthermore, these techniques are used in credit card fraud detection mechanisms, and their results have been compared.

[3] Popat and Chaudhary et al. The supervised algorithms of Popat and Chaudhary were presented. Some of the techniques used include Deep

Learning, Logistic Regression, Naive Bayesian, Support Vector Machine (SVM), Neural Network, Artificial Immune System, K Nearest Neighbor, Data Mining, Decision Tree, Fuzzy logic based System, and Genetic Algorithm. Credit card fraud detection algorithms identify transactions that are likely to be fraudulent. Machine-learning algorithms were compared to prediction, clustering, and outlier detection.

[3] Xuan Shiyang et al The Random Forest classifier was used to train the behavioral characteristics of credit card transactions. The types listed below are used to train the normal and fraudulent behavior features. Random forest on random trees and random forest on CART. Performance measures are computed to assess the model's effectiveness.

Literature survey of fake news

[1] Marco L. Della Vedova et al first proposed a novel ML fake news detection method that outperforms existing methods in the literature by combining news content and social context features, increasing its accuracy to 78.8%. Second, they implemented their method within a Facebook Messenger Chatbot and validated it with a real-world application, achieving an accuracy of 81.7% in fake news detection. Their goal was to classify a news item as reliable or fake; they first described the datasets they used for their test, then presented the content-based approach they used and the method they proposed to combine it with an existing social-based approach in the literature. The resulting dataset contains 15,500 posts from 32 pages (14 conspiracy pages and 18 scientific pages), with over 2,300,000 likes from 900,000+ users. 8,923 (57.6%) of the posts are hoaxes, while 6,577

(42.4%) are not.

[2] Cody Buntain et al creates a method for automating fake news detection on Twitter by learning to predict accuracy assessments in two credibility-focused Twitter datasets: CREDBANK, a crowd-sourced dataset of accuracy assessments for Twitter events, and PHEME, a dataset of potential Twitter rumors and journalistic assessments of their accuracy. This method is used with Twitter content sourced from BuzzFeed's fake news dataset. A feature analysis identifies features that are most predictive for crowd-sourced and journalistic accuracy assessments, producing results that are consistent with previous work.

Existing systems and its drawbacks:

Ng and Jordan compare logistic regression to Naive Bayes classification models, demonstrating that while logistic regression models have lower asymptotic error than Bayes classifiers, they fail to converge in very large datasets, such as those used in credit card transaction problems. The Bayes classifier converges quickly, but it has lower classification accuracy than the logistic regression models. Maes compares Bayesian and neural networks in a similar vein, concluding that the Bayesian network converges faster and has lower classification error than neural networks. Lessmann compared 41 methodologies on various evaluation criteria and credit scoring datasets in an extended benchmark simulation. It has been established that the random forest method, i.e. the randomized version of bagged decision trees, outperforms logistic regression and has gradually become one of the industry's standard models.

A few studies have discussed data mining approaches to fake news detection, such as feature extraction and model construction. A feature

extraction methodology (both news content and social context features) combined with metric evaluation using precision, recall, and F1 scores has yielded educated results, but the problem is not that simple. Other parameters like bot spamming, click bait, source of news also affect the predictions. These were some data mining and NLP approaches, but as AI research and development progressed, researchers became more interested in heavy neural network oriented approaches. A paper demonstrated a method of capture, 'score,' 'integrate,' and creates a model of recurrent neural networks for fake news stance detection. They used recurrent neural networks to capture temporal patterns of user activity surrounding a specific article/text, which was then used to extract source characteristics. All of this data is used and combined to create a model for classifying fake news.

Every system we tried and tested was either outdated or less accurate and secure, and they were all built on web 2.0 technology, which is less secure and prone to phishing and cyber attacks, and they were all cost-driven solutions.

Proposed system

Proposed System for Fake news

When it comes to our daily lives, news is extremely important. However, whether the statement is genuine or not, there is always a catch. There are numerous methods for the general public to generate or inject sham news on the internet, including tweets, articles, and paid partnerships or paid sponsorships. So, in order to overcome this, we propose fake news prediction, in which we take the input of a news article or an individual statement and subject it to multiple regression algorithms.

Proposed System for Credit card scam detection

Card transactions are always unfamiliar to the customer when compared to previous transactions. Vaishnavi Nath Dornadula et al. / Procedia Computer Science 165 (2019) 631-641 4 Geetha S/ Procedia Computer Science 00 Vaishnavi Nath Dornadula (2019) When there is concept drift, 000-000 unfamiliarity is a very difficult problem to solve in the real world. Concept drift can be said as a variable which changes over time and in unforeseen ways.

The dataset contains cardholder transactions. There are a total of 284,807 transactions, 492 of which are fraudulent, resulting in a 0.172% fraud rate. This dataset is significantly unbalanced. Because providing a customer's transaction details is considered a confidentiality issue, the majority of the features in the dataset are transformed using principal component analysis (PCA). V1, V2, V3,..., V28 are PCA-applied features, while the others, such as 'time,' 'amount,' and 'class,' are non-PCA-applied features.

These variables contribute significantly to data imbalance. Our primary goal in this research is to address the issue of concept drift in real-world scenarios.

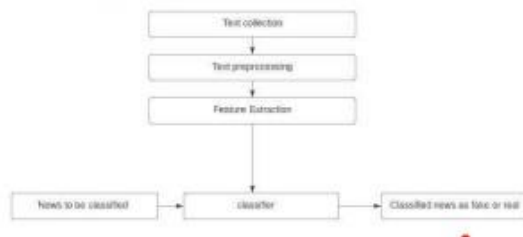
After creating models for fake news prediction and credit card scam detection, the process involves connecting the models to the data server/form server via the django framework using pickle. We deploy

the entire site on the block chain using icp tokens after connecting and testing it. We burn the tokens and launch the website on the ICP chain.

Advantages

These reports anticipate inspecting the strategy used in machine learning in the best way to predict fraud.It then diagrams a portion of the exploration questions, issues, and needs.

System Architecture



List of Modules

Data Preprocessing

TechniqueData analysis of

visualization

Comparing Algorithm with prediction in the form of best accuracyresult

Deployment using Flask

MODULE

DESCRIPTIONDATA

PRE-PROCESSING

The AI machine used to identify the machine error rate is close to the estimated blunder rate of the set up data. If the amount of data is sufficient to address the population, you may not need an observation method.Nonetheless, in everyday life, work with model data that may or may not be consistent with the public showcase. To locate the missing value, provide two qualities and definitions, indicating whether the data is a moving point or a number.The information model was used to evaluate the logic of the model that connects the preparation informational collection to the hyper model.

MODULE :

GIVEN INPUT EXPECTED

OUTPUTCOMPOSE DATA

YIELDS: ERASING LOUD DATA

DATA VALIDATION/ CLEANING/ PREPARING PROCESS

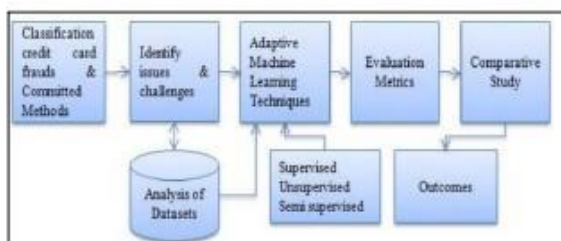
The library pack was imported when pressing the data. Examine to determine changes in the type and type of media, assess what's going on, and reflect esteems.Model approval is dependent on the model information saved during model preparation and is used to display and manage the model plan, and you can use it to more likely approve and test the setup data while looking at the model.Erase/change information by renaming the given data, unloading segments to break down one change, two factors, various ways, and so on.The steps and data innovation will differ from one store to the next.The primary motivation behind data annihilation is to recognise and eliminate errors and disappointments in order to increase the value of data in investigation and direction.

EXPLORATION DATA ANALYSIS OF VISUALIZATION

Demonstration is a significant ability in science and AI. Measurements depend on mathematical and mathematical data. Addressing data gives significant apparatuses to getting quality.This can be useful in recognising and investigating setup information, as well as deciding the design, data broken, thickness, and much more.With limited information on the area, data can be used to address and demonstrate a critical relationship in planning and graphing with nearby specialists and partners rather than a useful and valuable measure.Reporting examination and newscasting research, as well as a portion of the books mentioned near the end, are required for in-depth study.

Sometimes data is useless until it appears, such as drawings or tables. The ability to quickly access test data and various devices and procedures is critical for measurable data and AI. It will notice various plans that you really want to realise while addressing data in Python, and you will understand how to use them to all the more likely comprehend your data.

How might you configure substitute times in the diagrams and various sizes in the graphs.Synopsis of data dissemination and histogram and boxing.



False Positives (FP): A person who believes the person will not be paid. When the genuine class is missing but the theoretical class is present. For example, if the true classification states that this accomplice does not exist, the speculated classification states that this accomplice will suffice.

False Negatives (FN): A person who is expected by the payer. When the genuine class is present but the theoretical class is not. For example, if the true worth indicates that a companion is alive, the forecasts indicate that the companion will perish.

True Positives :(TPA non compliant individuals do not pay). These are

regarded as great qualities, implying that the true worth of the worth is indeed, and the qualities taken are also yes. For example, if the true worth of the school shows that this classmate is alive, the presumed class will tell you the same thing.

True Negatives (TN): A person is regarded as a participant. This has been taken seriously, which implies that there is no genuine worth and that the classes taken are not. For example, if the genuine classification states that this accomplice does not exist, the normal classification will state the same.

Model Building for fake news

This section describes the classification methodology. Using this model, a tool for detecting fake articles is developed. The dataset is classified using supervised machine learning in this method. The dataset collection phase is the first step in this classification problem, followed by preprocessing, feature selection, training and testing of the dataset, and finally running the classifiers. The methodology is based on running various experiments on datasets with the algorithms described in the previous section, such as Random forest, SVM, and Naive Bayes, as well as majority voting and other classifiers. Experiments are run on each algorithm separately and in combination to achieve the highest accuracy and precision. Describes the Proposed System Methodology. The main goal is to use a set of classification algorithms to create a classification model that can be used as a scanner for fake news by detecting details of news and embed the model in a Python application that can be used to discover fake news data. In addition, appropriate refactorings on the Python code have been performed to produce optimized code. This model employs the classification algorithms k-Nearest Neighbors (k-NN), Linear Regression, XGBoost, Naive Bayes, Decision Tree, Random Forests, and Support Vector Machine (SVM).

All of these algorithms are as precise as they can be. Where reliable from a combination of their averages and comparisons. The dataset is fed into various algorithms in order to detect fake news. The accuracy of the obtained results is analyzed to arrive at the final result.

The approach to detecting political fake news used in the model development process is as follows: The first step is to collect political news datasets (the Liar dataset is used for the model), then perform preprocessing by removing rough noise. The next step is to use the NLTK (Natural Language Toolkit) to perform POS and feature selection. The dataset is then split, and ML algorithms (Naive Bayes and Random forest) are used to create the proposed classifier model.

Feature analysis

Using range partitioning, we divide cardholders into different clusters/groups based on their transaction amount, i.e., high, medium, and low. We aggregate the transactions into respective groups using the Sliding-Window method, i.e., extract some features from the window to find the cardholder's behavioral patterns. Features such as maximum amount, minimum amount of transaction, average amount in the window, and even time elapsed are available. Following preprocessing, we train different classifiers on each group based on the cardholders' behavioral patterns and extract fraud features. Even when we apply classifiers to the dataset, we perform SMOTE

(Synthetic Minority Over-Sampling Technique) operation on the dataset due to imbalance in the classifiers that do not work well on the dataset. Oversampling yields no useful results. As a result, there are two approaches to dealing with imbalance datasets: consider Matthew Coefficient Correlation of the classifier on the original dataset or use one-class classifiers. Finally, the classifier that is used for training the group is applied to each cardholder in that group. The classifier with the highest rating score is considered as the cardholder's recent behavioral pattern.

Deployment on ICP chain

Following sentimental and feature analysis, as well as flask-hooking the model, the next and final task is to upload the website to the icp chain. During this process, we create an individual wallet with icp mainnet, preferably an icp wallet with unique and inaccessible public and private addresses. We run the entire work locally using the Ubuntu and Motoko languages by using proof of work and proof of address.

Conclusion

The investigation began with data handling loss of significant worth examination of exploration, followed by show and assessment. Data cleaning and visualization for fake news has been completed. In the future, we intend to add data to the dataset and refine the algorithm. The website is finished.

REFERENCES:-

- [1] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu, "Fake News Detection on Social Media: A Data Mining Perspective" arXiv:1708.01967v3 [cs.SI], pg no-6,3 Sep 2017.
- [2] H. Gupta, M. S. Jamal, S. Madisetty and M. S. Desarkar, "A framework for real-time spam detection in Twitter," 2018 10th International Conference on Communication Systems & Networks (COMSNETS), Bengaluru, pp. 380-383, 2018.
- [3] M. L. Della Vedova, E. Tacchini, S. Moret, G. Ballarin, M. DiPierro and L. de Alfaro, "Automatic Online Fake News Detection Combining Content and Social Signals," 2018 22nd Conference of Open Innovations Association, Jyväskylä, pp. 272-279, 2018.
- [4] C. Buntain and J. Golbeck, "Automatically Identifying Fake News in Popular Twitter Threads," 2017 IEEE International Conference on Smart Cloud (SmartCloud), New York, NY, pp. 208-215, 2017.
- [5] S. B. Parikh and P. K. Atrey, "Media-Rich Fake News Detection: A Survey," 2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR), Miami, FL, pp. 436-441, 2018.
- [6] Scikit-Learn- Machine Learning In Python (Article).
- [7] Khanam, Z. "Analyzing refactoring trends and practices in the software industry." Int. J. Adv. Res. Comput. Sci. 10, 0976-5697, 2018.
- [8] Shankar M. Patil, Dr. Praveen Kumar, "Data mining model for effective data analysis of higher education students using MapReduce" IJERMT (Volume-6, Issue-4), April 2017.
- [9] Zhang, Jiawei, Bowen Dong, and S. Yu Philip. "Fakedetector: Effective fake news detection with deep diffusive neural network." 2020 IEEE 36th International Conference on Data Engineering (ICDE). IEEE, 2020.