COLLEGE COMPUTING AND INFORMATICS OF UNIVERSITY TENAGA NASIONAL (UNITEN)

Harnessing Big Data to Identify Scam Vulnerabilities and Threats

MUHAMMAD AZIF FARHAN BIN ROSAINI

IS01082210

JUNE 2025

Harnessing Big Data to Identify Scam Vulnerabilities and Threats

by

MUHAMMAD AZIF FARHAN BIN ROSAINI

Project Supervisor: Ts Dr Zaihisma Binti Che Cob

PROJECT REPORT SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENT FOR BACHELOR IN INFORMATION TECHNOLOGY (INFORMATION SYSTEM) (Hons.) COLLEGE OF COMPUTING AND INFORMATICS UNIVERSITI TENAGA NASIONAL

JUNE 2025

DECLARATION

I, hereby declare that my whole Final Year Project solely mine except for the quotations and citations that have been given credits for as acknowledgements. I also proclaim that the project has not been submitted previously and/or is not concurrently being done by anyone in the degree program of Universiti Tenaga Nasional (UNITEN) or from any other universities. I allow my project to be made available within the university library and can be borrowed, consulted, copied or reproduced in accordance with the provision of the UNITEN Library Regulations from time to time made by the Library Committee.

MUHAMMAD AZIF FARHAN BIN ROSAINI

IS01082210

2024

APPROVAL PAGE

This title of this report is ‘Harnessing Big Data to Identify Scam Vulnerabilities and Threats’

Author: Muhammad Azif Farhan Bin Rosaini (IS01082210)

The undersigned has certified that candidate above fulfilled the conditions set for Final Year Project in requirement of Bachelor’s in Information Technology (Information System) (Hons.) Universiti Tenaga Nasional

Supervisor:

Signature: ……

Name: Ts Dr Zaihisma Binti Che Cob

Date: 15 November 2024

Acknowledgement

I would like to express my sincere gratitude to all those who guided and supported me throughout the project's completion. Firstly, I would like to thank my project advisor, Dr Zaihisma for their continuous support, feedback and expertise, which contributed a lot to the development of this project. I also would like to extend my appreciation to my family and friends for their constant support and patience which motivated me the entire journey.

Executive Summary

Scams are a global issue, and Malaysia is no exception. Scammers deploy various tactics and deceptive strategies, making it difficult to trace their activities and which allows them to operate their shady crime without ever getting caught. Technology is their strongest ally, as it became more and more advanced and continuously evolving making room for scammers to improve their strategy, targeting more innocent victims and maximizing their financial gains.

The dataset chosen for this project is from a website called ScamWatch which is an Australian Government website, and it primarily focuses on scam cases that were reported in the year 2024. Although the data is from Australia, it could still beneficial here in Malaysia as it will serve as a reference point to match with the scam trends occurring in the country.

The CRISP-DM methodology serves as the foundation of this project, ensuring a well-structured and organized workflow. In project 1, only the initial phases of the methodology will be implemented, namely business understanding, data understanding and data preparation while the remaining phases, modelling, evaluation and deployment will be continued in Project 2.

The completion of this project will allow all information related to scams to be compiled into an interactive and comprehensive dashboard. The dashboard will provide an eye-opening and actionable insights that benefits both the Malaysian public by raising awareness and promoting education, as well as the country's authorities, such as Cybersecurity Malaysia or the ministry of digital and communication in their effort to fight scams.

Table of Contents

List of Figures

List of tables

List of Abbreviations

SDG	Sustainable Development Goals
FaaS	Fraud as a service
US	United States
NSW	New South Wales
FBI	Federal Bureau of Investigation
SMS	Short Message Service
SHAP	SHapley Additive exPlanations
GMM	Gaussian Mixture Models
NMF	Non-negative Matrix Factorization Recurrent
RNN	Neural Network
LSTM	Long Short-Term Memory
DT	Decision Tree
RF	Random Forest
SVM	Support Vector Machine
ANN	Artificial Neural Network

CHAPTER 1

INTRODUCTION

1.0 Overview

Chapter 1 will introduce the project in many different subtopics which are the problem statement with journals released to support my justifications, the purpose of the project, who would be benefiting from the findings, expected outcomes and the project timeline.

1.1 Project background

Scam refers to a dishonest plan for making money or getting an advantage, especially one that involves tricking people. In other words, scam is a form of trickery that takes advantage of people by stealing something valuable like money. Scams have long existed in society (The History and Evolution of Fraud, 2024). They prey on individuals' vulnerabilities and trust. Scams occur in various methods like online shopping scams, love scam, phishing emails and fake investment opportunities (ScamWatch, 2024). Scammers usually target victims by exploiting their emotions such as fear or urgency. This will make it harder for individuals to think whether the call or link is legitimate. With the booming world of digitalization, scams have evolved and widespread, affecting people of all ages. Scams not only impact financially but also emotionally since it often leaves victims feeling helpless and manipulated. To combat this issue, governments and finance agencies around the world are working tirelessly by spreading awareness to the public via various campaigns and establishing stricter laws. However, scammers adapt very quickly by using advanced technology to outsmart the victims. Education is crucial to protect oneself against scams. Understanding common scam tactics and being cautious can significantly reduce the risk of becoming a victim (MyGovernment, 2024).

This project aligns with several Sustainable Development Goals. The first one is Decent Work and Economic Growth. This goal promotes a sustainable economic growth that encourages a positive and productive work environment for all. With scams happening on a dangerous rate daily, it is important to quickly battle this issue to prevent more new scam victims. Law enforcement and financial institution need to step up their security measure to curb this serious problem. Another SDG that properly align with this project is Peace, Justice and Strong Institution. One of the key highlights of this sustainable goal is to strengthen the capacity of institution and ensure accountability. Two main institution that closely relate with this project are law enforcement and finance. They are responsible to help scam victims to trace back the scammer and to prevent the incident from happening in the first place.

1.2 Problem Statement

Scams are among the popular crime that occurs in many countries. Scammers nowadays targets various kinds of victims from individual to businesses. With the world rapidly shifting to digitalization, scammers have also found creative new ways to trick people to exploit their vulnerabilities. While scams caused valuable assets loss, it also impacts people trust in online banking transaction since it is exposed to various threats. Despite numerous awareness campaigns and preventive measures implemented to fight this crime, scammers always find new tricks to successfully trick their victims. This serves a continuous challenge to law enforcement to catch these scammers. Scammers have a very specific target that they like to prey on. They usually target victims with high income, older groups of people and individuals who are easily convinced by what they were instructed. These people are most likely to fall for the scam because they are vulnerable and easy to manipulate. Scammers always start the conversation by imposing themselves as people of authority like bank personnel or a police officer. This is a scare tactic to get the victims attention and do as they told.

1.3 Objective

The aim of this report is to provide insight regarding scams. Although the data originated from a source in Australia, some of the scam activities also occur in Malaysia. The expected output of this project is hoped to be useful for authorities in Malaysia to mitigate the issues of scam

The objectives for this project are:

1. To explore the relationship between various attributes in the dataset
2. To develop a predictive model that can predict the number of scam cases
3. To create an interactive dashboard that visualizes scam trends and the number of scam cases based on selected features.

1.4 Scope

The scope of this project mainly focuses on scam cases happening across Australia however it will then relate with scam related issues that occurred here in Malaysia. The dataset used for this project was taken from the Australian government called ScamWatch and it contains information on scam reported cases in the country in the year 2024. The dataset contains various kinds of data types namely numerical, categorical and temporal. Number of reports, aggregated amount lost, and a newly added column called average state income are numerical data types while date belongs to temporal data types. The dataset mostly consists of categorical data such as scam category, scam type, gender, age group and states.

1.5 Expected Outcome

The expected outcome of this project is the development of two key products which are an information visualization dashboard and a predictive model to predict scam likelihood based on selected features and monthly scam case trends, using a chosen model that showcase strong accuracy and precision in its prediction. The former contains powerful and meaningful information that serves as an understanding of the overall data. Some insights include which state has the highest number of reported scam cases, which gender and age group are more likely to fall victim to scams and more. Although the data is from Australia and about scam cases in that country, ultimately, the project aims to spread awareness and serves as a reference to provide data-driven insights to Malaysia's government and authorities, to support their decision-making processes for further actions to curb scams in our country.

1.6 Timeline

Table 1: Project Timeline

Planned Activities	Weeks															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Research Project Title, Related SDG Goals, Exploring Available Dataset																
Supervisor Meeting #1																
Project Briefing																
Milestone 1: Proposal																
Finalized Proposal Submission																
Supervisor Meeting #2																
Milestone 2A: Oral Presentation																
Chapter 1 & 2 Draft Report																
Milestone 2B: Draft Report Submission																
Supervisor Meeting #3																
Data Understanding and Preparation																
Supervisor Meeting #4																
Selection of Predictive Model and Dashboard Design																
Milestone 3A: Final Project Presentation																
Chapter 4 and 5 Documentation Report																
Milestone 3B: Final Report Submission																

1.6.1 Project Timeline Explanation

The project timeline above breaks down the tasks and activities throughout the completion of Project 1 starting from the very first week to week 14 which is the submission of the final report, logbook and dashboard demo. The following are the important datelines for Project 1 Timeline:

Table 2: Key Dates of Milestones

Milestone	Week	Due Date
Proposal	Week 2	
Oral Presentation	Week 5	
Draft Report Submission	Week 9	
Final Project Presentation	Week 12	
Final Report Submission	Week 14	

1.7 Chapter Summary

The project background talks about the definition of scams, identified their popular methods and how they evolved over time and addresses the impact of scam both financially and emotionally. On top of that, Project Background also discusses about the project alignments with Sustainable Development Goals. Next, in Problem Statement section, it identifies the evolving tactics of scams that makes it difficult to stop them alongside the broken trust in online transactions because of scam. Objectives outlines the project goals which is to analyse the relationship between the types of scams and location, to develop a predictive model for scam likelihood and to create an interactive dashboard to visualize key insights. Then, in the Project Scope highlights the project focus on scam cases reported in Australia utilizing a dataset from a website called ScamWatch and relevance with scam cases in Malaysia. Finally, the expected outcomes outline two key deliverables, an interactive dashboard and a predictive model with hopes to spread awareness and provide data-driven insights to Malaysian's government. Chapter 2 will be discussing about literature review that may support the project's main objective.

CHAPTER 2

PRELIMINARY STUDY

2.0 Introduction

Chapter 2 will focus on studies, research and articles related to scam that provide a solid foundation for my topic. It talks about various scam types, how scams have evolved to suit modern day technology, the impact of scams and more.

2.1 Overview of cybercrime

The definition of cybercrime is the illegal usage of any communication device to commit or facilitate in committing any illegal act, according to Dennis (2024). A cybercrime is a type of crime that uses a computer or a network of computers to cause harm. These crimes often target specific individuals, businesses or sometimes even organizations. A cybercriminal is a person who has good technical skills on technology but apply the knowledge to perform malicious acts and illegal activities like cybercrimes. Scammers fall perfectly under cybercrimes because they use a communication device such as a phone to deceive people into giving their money.

2.1.1 The history of fraud

Fraud or scams often associated with modern day crimes due to its usage of technology in order to execute. Actually, the first ever cases of fraud started in the third century (“The history and evolution of fraud,” 2024). In that era, two sea merchants wanted to enrich themselves by outsmarting their insurance policy. The policy states that they have to repay their loan plus interest after successfully selling the merchandise. If the loan cannot be paid, the lender would confiscate the ship as well as its cargo. The two sea merchants realize that they could not afford to pay the loan so they decided to sink the ship so they could receive the loaned money.

However, the attempt to sink the ship was unsuccessful because of them died trying to flee whereas the other was caught.

2.1.2 Types of scams

Figure 1: Types of Scams according to a survey by Ipsos Malaysia

Scams can come in various ways, each targeting individuals through different methods to exploit specific vulnerabilities. It is vital to classify these types of scams to raise awareness and deploy a set of effective preventive strategies. One of the common types of scams according to Ipsos Malaysia (2021) is Phone scams. This is similar to phishing, where scammers deceive their victims into giving out personal information such as bank account numbers or credit card details by impersonating as people of authority like bank personnel. According to the survey conducted by Ipsos, phone scams account for 40% in November 2023. Online scams are another popular type, where scammers send out spam messages in Telegram or Whatsapp to deceive victims into clicking the link. The message typically contains promotional ads such as a chance to win a large amount of money just by clicking a link. According to the survey, Whatsapp and Telegram are among the popular platforms used by scammers with each account for 54% and 30% respectively. Investment scams are also a common type of scams based on the survey which takes 24% of the overall survey. This scam operates by attracting victims with big, high and instant return when they “invest” some money which will then lead to serious financial

loss. According to Ministry of Finance (2022) via Malaysia's Official Government Portal, investment scams typically offer potential investors, the victims, an investment opportunity that is too good to be true. The portal also shares the signs to look out for like unrealistic high returns within a short period of time or the investment promises minimal risk. Additionally, the individual who offered the opportunity in the first place seldom gives out detailed information about the investment such as the company background or who the directors and shareholders are. Another sign to pay close attention to is urgency. Scammers always pressure their victims by making them decide at the moment and saying that the offer only stands for a limited time.

2.1.3 Modern Day Scam approach

While traditional scams still being use, modern day scams start to emerge. According to a website called Comply Advantage, there are a few trends in 2024 ("Top fraud trends," 2024). The first one is synthetic identity fraud. This fraud occurs when someone combines a genuine and fake information to create a false identity and commit fraud. Some of the examples for this type of fraud is to create fake accounts. These individuals open and build the accounts for a while to make it seem legit and then apply for maximum credit. Once they have collected a significant amount of credit, they will spend the money and disappear. Another modern-day scam approach is Fraud as a service (FaaS) where the criminals offer fraud-related tools to someone who do not have the technical knowledge to commit fraud themselves. Scammers often rely on apps like Telegram to hide their identity. Contactless fraud is also said to be emerging in 2024 since the number of people performing contactless payment are estimated to reach 1 billion ("Contactless mobile payment to surpass 1 billion," 2024). Contactless payments includes tapping cards, smartphones and digital wallets by providers like Tng E-Wallet, GrabPay, SamsungPay and ApplePay (SoyaCincau, 2024). This feature is enabled with the use of NFC technology making it easy for vendors to accept payments will continue to grow in the future. However, this technology also creates an opportunity for criminals to intersect the transaction or steal the information from the device to perform payment elsewhere without the victim's knowledge.

2.1.4 Impacts of Scams

Scams have a significant impact not only on financial losses but also on emotional distress and loss of trust. According to “Psychological impact of scam” (2024), being a victim of a scam can be a very traumatic experience with long-lasting effects. The impact of scams is broad. For example, if someone gets scammed, they lose their money. This is already a financial impact. Losing money they worked hard to earn affects their emotions because that money represents their efforts and sacrifices. Additionally, being scammed can lead them to doubt the system. Additionally, being scammed can break the trust in various systems, such as financial institutions or law enforcement. Victims may feel that these entities failed to protect them or prevent such incidents from occurring in the first place.

When it comes to emotional distress, this issue often targets older individuals, such as retirees. They might lose their retirement savings to a scam, and all that money could be gone in an instant. This can severely affect their mental state because the money represents both a large sum and the result of years of hard work. They may feel regret, and the scam can haunt them indefinitely. Their children may also feel the impact because they are unable to help their parents recover the lost money. Ultimately, scams can affect entire families, especially those already struggling financially.

Some people also blame certain parties, such as the police or banks, for not doing enough to prevent scams. Many victims expect these entities to handle issues like scams and even help them recover their money, especially if the amount is substantial. Additionally, some people lose trust in using online banking or any form of online payment due to fear. They feel unsafe using these platforms or methods because of the risk involved.

2.1.5 Factors influencing Scam

The Target

Why are some individuals more susceptible to scams?

Age is generally acknowledged to have a positive correlation. The US Federal Bureau of Investigation issued a public warning in November 2023 about scammers specifically targeting senior citizens. Increasingly, however, the young and technology-savvy are not spared, possibly due to over-confidence and impulsiveness.

Some evidence suggests that gender may be a factor. In Japan, for instance, older women who live alone and have smaller social networks tend to have higher vulnerability scores in relation to scams.

In general, the target's experience and disposition influence the success rate of a scam.

Experience

There is general consensus that individuals with higher levels of education, greater financial literacy, enhanced technological savviness, extensive experience using the Internet, heightened awareness of Internet safety, and staying abreast of developments in Internet security and scams are better equipped to guard against falling victim to scams.

Consequently, there are substantial benefits for individuals to pursue additional education and training in online usage. Complacency and pride, however, often manifest in the belief that "it-will-not-happen-to-me", leading to an individual's downfall.

Interestingly, the debate over the amount of time spent online persists. While a longer duration of online activity may expose individuals to a higher risk of scams, it is also acknowledged that experience is gained over time.

Conversely, too little time spent online may result in insufficient experience to identify potential scams.

Disposition

There is no cause-and-effect relationship between one's inherent temperament and the likelihood of being a scam victim because mitigating circumstances always play a role. Even individuals with similar dispositions may react differently to a phishing email. However, there are documented propensities towards scam compliance for individuals with certain inherent temperaments.

Individual traits that are more prone to falling for scams are depicted in the diagram below.

As can be seen, the traits can be either positive or negative. An open and trusting person, especially one insistent on honoúring commitments and who is generally agreeable, may also be susceptible to scams.

Figure 2: Factors influencing scams

Some people are more likely to fall victim to scams than others. According to (The psychology of scams) common factors influencing this include age, experience, and personality traits. Age has a positive correlation with scam vulnerability, meaning that the older someone is, the more likely they are to be scammed. For example, in the U.S., the FBI has warned the public that scammers often target senior citizens. However, younger people aren't completely safe either. Despite growing up with technology, they might be overconfident or easily manipulated.

There's also evidence suggesting that gender plays a role; for instance, elderly women living alone tend to be more vulnerable due to their circumstances.

Experience can be a protective factor. People with higher education, better financial literacy, and knowledge about technology are generally more cautious. They're often aware of online threats and stay updated on internet security. Additionally, those who have spent more time on the internet are likely to learn about scam tactics over time, making them less susceptible. On the other hand, individuals with limited online exposure or knowledge are at greater risk because they may not recognize potential scams. Their lack of familiarity makes it harder for them to identify red flags.

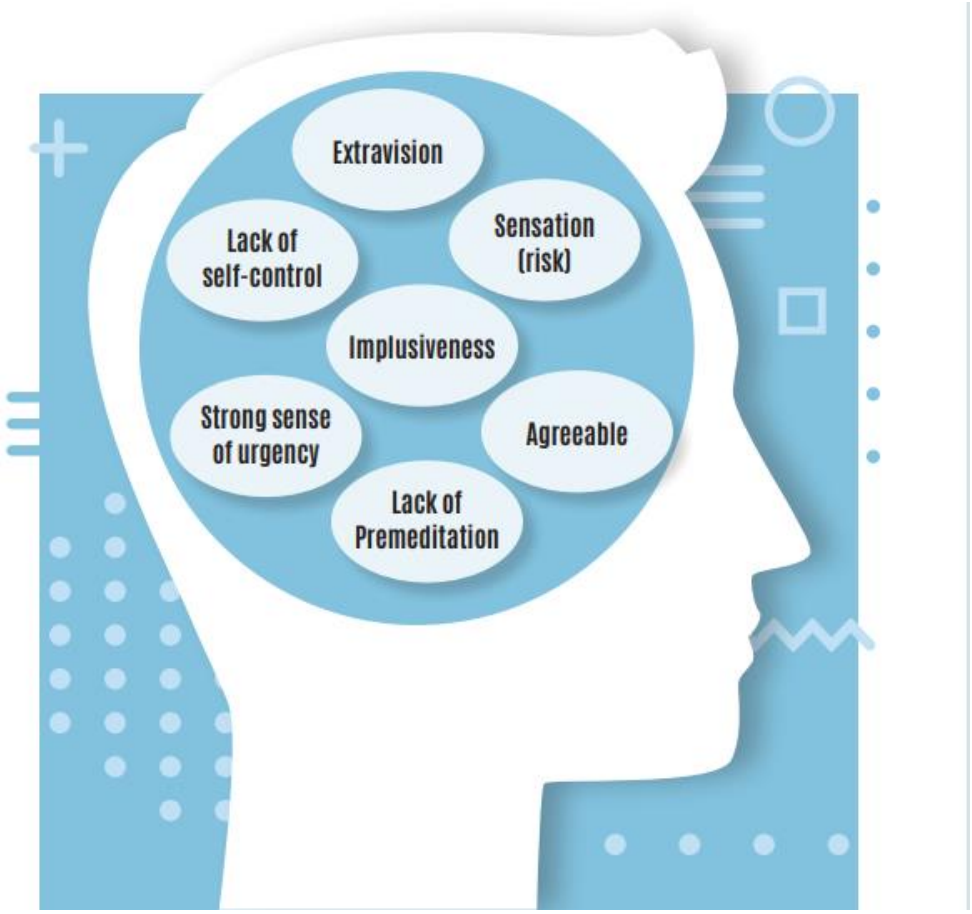


Figure 3: Victim traits that are more prone to fall for scam

Personality traits also influence how someone reacts to scams, even though two people with similar personalities might respond differently. Some traits that increase scam vulnerability are lack of self-control, agreeableness and impulsiveness. People who lack self control might click on a suspicious link offering large sums of money without thinking it through. Their inability

to pause and evaluate the situation makes them an easy target. Agreeable individuals means that they tend to go along with what others say. For instance, if a scammer asks them over a phone call to transfer money to an unknown account, they might comply without questioning the request. This blind trust may prompt the scammer to target the victims again. Thirdly, impulsive people act quickly without verifying the legitimacy of a request. Scammers often use scare tactics or threats to pressure their victims into responding immediately, leaving little room for critical thinking.

2.2 Review of Related Work

The first research was done by Afriyie et al. (2020), which talks about detecting and predicting fraud in credit card transactions using machine learning algorithms. In the study, they mainly used three classification machine learning models which are logistics regression, decision tree (DT) and random forest (RF) to detect whether a credit card transaction is fraudulent or not. The dataset used was a simulation of credit card transactions in the western side of the US in the year 2020 and is available from Kaggle. The data includes purchase history, customers name, merchant, type of purchase and whether or not the transaction is fraudulent. Accuracy, precision, recall, specificity and F1-score were used in order to compare the performance of the three models. Out of 427 total transactions of the testing data, RF correctly predicts 409 transactions as fraudulent and falsely identified 18 fraudulent transactions as not fraud. The final result showed that RF performed better than others by achieving an accuracy rate of 96%. The research proves the capability of machine learning in predicting fraudulent activities, specifically in credit card transactions to help banks and cybercrime agencies to detect fraudsters in the ever-changing world of digitalization.

Atikah Hanisah Mohd Hanif et al. (2024) explores how to predict fraudulent job advertisement using machine learning. The main purpose of this research is to identify the most effective model for classifying fraudulent job advertisement and ultimately preventing individuals from becoming victims to job scams. The study uses Employment Scam Aegean Dataset to conduct the predictions and used Logistic Regression, Support Vector Machine (SVM), DT, and Naïve Bayes algorithms. The dataset comprises of over 17,000 job listings and 18 attributes. Since the dataset was originally imbalanced, meaning that one class has significantly lesser sample than the other, can result in biased model performance. There are three different methods to handle imbalanced dataset like data-level method, algorithm-level method and hybrid approach. Data level-method method modifies the dataset sample to balance the distribution either with undersampling or oversampling. Algorithm-level method, on the other hand, emphasizes on the importance of the positive class without needing to adjust the dataset distributions. Hybrid approach combines the other two methods to obtain a more accurate model. In this study, they performed the machine learning prediction for both imbalanced and balanced dataset and found that the most suitable model is DT with result for imbalanced of accuracy (0.948), precision (0.23), recall (0.37) and F1-score (0.31) and balanced dataset achieving accuracy (0.705), precision (0.73), recall (0.7) and F1-score (0.72).

According to Asha et al. (2021), the suitable machine learning models to predict fraudulent credit card transactions are SVM, K-nearest algorithm and artificial neural network (ANN). In their study, they used these three models to determine which one is the most accurate in predicting credit card frauds. The study also listed around 10 classification of credit card frauds such as account takeover, card id theft, counterfeit card fraud and more. For the fraudulent prediction, Asha et al. (2021) used European Bank Customer Transaction Dataset which has 31 columns including the class where the models predict fraudulent or non-fraudulent transactions. To compare the performance of the three models used in the study, researchers evaluate based on the accuracy, precision and recall for each model where SVM achieved 93%, 97% and 89% respectively.

In this study, Mazorra et al. (2022) examines the use of machine learning to detect rug pull scams in cryptocurrency tokens, aiming to protect investors from financial losses. Using a dataset of 27,588 labeled tokens comprises of 631 non-malicious token and 26,957 malicious tokens. The study evaluates two approaches which are Activity-Based Method, that analyzes token activity before malicious events, and a 24-Hour Early Method, focusing on token activity within the first 24 hours of pool creation. Gradient Boosting Decision Tree (XGBoost) and FT-Transformer machine learning models were applied, with XGBoost outperforming due to optimized hyperparameters. Due to imbalance dataset, data augmentation strategy was used to select more evaluation points for non-malicious tokens. SHapley Additive exPlanations (SHAP) values revealed key features influencing predictions, like transaction count and the time between token and pool creation. Results show that XGBoost achieved high accuracy, recall, and F1 scores, indicating the ability to detect malicious tokens early. This research highlights the importance of addressing class imbalance and optimizing models to enhance detection in real-world scenarios.

Anjali et al. (2024) explores detecting smishing, phishing in SMS messages, using machine learning techniques. The research aims to find the best model for classifying spam SMS and minimizing the impact of smishing scams. Using the UCI SMS Spam Dataset, which includes over 5,000 messages, the study tests several models, including K-Means, Gaussian Mixture Models (GMM), Non-negative Matrix Factorization (NMF), and deep learning models like Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), and Bidirectional LSTM. To address the imbalance in the dataset, which has more non-spam messages, the study applies undersampling and oversampling techniques, alongside semi-supervised learning methods. The results show that the RNN-Flatten model outperforms others, achieving 94.13%

accuracy on training data and 91% on unseen data. The Bi-LSTM and LSTM models follow with accuracies of 92.78% and 92.09%, respectively. This research concludes that deep learning, especially RNN-Flatten, provides the best performance for detecting smishing messages.

In summary, this section of the report explores various research studies related to both financial and non-financial scams, from fraudulent credit card transaction, fake job posting, cryptocurrency scams and phishing in SMS scams. While these studies share a common objective, which is to develop a predictive model that predicts scam likelihood based on specific features, they used different approach in their study particularly in selecting the ideal model and the dataset chosen. As a result, each model performs differently. This review section hopes to provide a general understanding of the research to examine whether their objectives align with this project and to look for similarities or discrepancies in their findings.

2.2.1 Summarized Table for Literature Review

Table 3: Summarized Table for Literature Review

Citation & Title	Purpose of Research	Data Source	ML Model/ Technique	Findings
(Afriyie et al., 2023) A supervised machine learning algorithm for detecting and predicting fraud in credit card transactions	Classify credit card transactions as fraud or non fraud	Credit Card Transaction Fraud Detection Dataset	Logistics Regression, Decision Tree, Random Forest	Random forest correctly identified 409 fraudulent transactions out of the 429 total transactions from the training data. Only 18 transactions of fraudulent transact ion were falsely identified as legit transaction
(Atikah Hanisah Mohd Hanif et al., 2024) Machine Learning Approach in Predicting Fraudulent Job Advertisement	Predict and identify fraudulent job advertisement	The Employment Scam Aegean Dataset	Linear and Logistic Regression (LR), Support Vector Machine (SVM) Naïve Bayes (NB), Decision Tree (DT)	Decision Tree outperformed other models by achieving an accuracy of 70% in the balanced dataset
(Asha et al., 2021) Credit card fraud detection using artificial neural network	Identifying fraud in credit card transactions using Machine Learning	European Bank Customer Transaction Dataset (2013-2014)	Support Vector Machine (SVM) K-nearest algorithm Artificial Neural Network (ANN)	The accuracy value for SVM is 93%, followed by KNN with 99.8% and the highest accuracy value falls to ANN with 99.9%
(Mazorra et al., 2022) Leveraging	Using machine learning to detect rug	Cryptocurrency token Dataset	XGBoost and FT-Transformer	XGBoost performed better with higher

Machine Learning Techniques for Automated Scam Detection	pull scams in cryptocurrency	containing malicious and non malicious token		accuracy and recall than FT-Transformer
(Anjali et al., 2024) SMS Scam Detection Application Based on Optical Character Recognition using Machine Learning	Focuses on detecting smishing using machine learning to identify spam messages	UCI Spam Messages Dataset	K-Means, Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), and Bidirectional LSTM.	The RNN-Flatten model outperformed other models, achieving 94.13% accuracy on training data and 91% on testing data

Based on the literature review, it is discovered that there are no commonly used machine learning models. All research studies use different model to better suit their own topic, dataset and end goal. Despite having a slight difference in topics discussed and objectives, these studies share one common strategy, that is to compare the performance between models. That being said, the overall concept of all the studies is largely similar, that is to predict between scam (1) or not scam (0). On this note, it is perfectly aligned with the project's goal which is to predict scam likelihood based on selected feature. For example, in Atikah Hanisah Mohd Hanif et al. (2024) study, they investigate tactics used by scammers to create fake job openings and who are their main target. Similarly, in Anjali et al. (2024) study, they explore the most common phrases in text used by scammers to lure in and trick the victims.

2.3 Review and comparison of machine learning techniques

2.3.1 Linear Regression

Linear regression is a form of supervised machine learning algorithm that predicts the probability of an outcome using binary classification. Meaning that the outcome that this model predict will either be in the form of true/false, yes/no or 0/1. Linear regression is perfect to determine whether an instance belong to which category. It analyses the relationship between one or more variables and classify them into discrete category. Linear regression was used in Afriyie et al. (2023) and Atikah Hanisah Mohd Hanis et al. (2024) study. In the context of predicting fraudulent activities, linear regression can come in handy to classify between legit or fake activity.

Strength

1. Easy to implement
2. Less complex compared to other if relationship between dependent and independent variables is known
3. Easy to avoid overfitting by using dimensionality reduction techniques

Weaknesses

1. Outliers can affect regression
2. Regression tends to assume straight line relationship between dependent and independent variables
3. Relies on the mean for both dependent and independent variables

Formula

$$y = \frac{e^{(b_0 + b_1 X)}}{1 + e^{(b_0 + b_1 X)}}$$

Based on the formula y represents the outcome predicted. X on the other hand is input value. b0 and b1 reflects intercept terms and coefficient for x respectively.

2.3.2 Decision Tree

Decision tree is a representation of decisions which consists of nodes, branches, root nodes and leaf nodes that includes the potential consequences of the decisions. Root nodes symbolizes the first ever decision made, whereas node represents the next decision. Branches shows the potential result of a decision and leaf node is the final output. According to Hrvoje Smolic (2024), decision tree works in two ways which are splitting criteria and pruning techniques. The former works by determining how decision tree divides the data and the former works by preventing the model from overfitting and become overly complex. Decision tree was used by Afriyie et al. (2023) and Atikah Hanisah Mohd Hanis et al. (2024) for their study to predict fraudulent behaviour in credit card transaction and job advertisement respectively.

Strength

1. Works with categorical and numerical data
2. Able to accommodate missing values and outliers
3. Handles classification and regression tasks

Weaknesses

1. Prone to overfit
2. Struggles when dealing with complex data
3. Does not perform well with inseparable variables

Formula

$$Entropy = -\sum_{i=1}^{c} p_i \times log_2(p_i)$$

Based on the formula, *pi* is the proportion of which belongs to i class and c is the number of classes. Usually, a higher entropy value indicates uncertainty within the dataset while conversely, indicates homogenous dataset. Entropy measures the impurity of an attribute to determine which is the best to split at each node.

2.3.3 Random Forest

Random Forest (RF) is a machine learning technique that creates series of Decision Tree (DT) during training phase. The trees are randomly constructed using the dataset provided, reducing risks of overfitting and to improve the performance of the model in prediction. Results of the tress are aggregated in prediction phase for classification tasks or regression tasks.

The first step on how to work on RF is to choose random sample from the dataset to create multiple bootstrap samples and for each of them, decision tree will be constructed. Since RF is known for its randomness, each node of trees will have random subset, which is considered as splitting. Once decision trees have successfully constructed, RF could make predictions.

Strengths:

1. With the several decision trees made, Random Forest tend to give a more accurate for the predictions
2. It can handle data with missing information and outliers which are unlikely to affect forecasts
3. Has the capability to handle both numerical and categorical data without being biased due to the random subsets

Weaknesses:

1. Model may be too complex
2. Limitations in memory of hardware systems as it works on big datasets
3. Model developed could take long period of time to be done for prediction

2.3.4 Support Vector Machine

A support vector machine (SVM) is supervised machine learning algorithm that classifies data by searching for an optimal line, also known as the hyperplane. It was developed by a man named Vladimir N. Vapnik and his colleagues in the year 1990. How SVM work is by maximizing the distance between each class in a number of dimensional space. SVMs are often used with classifications because it distinguishes between two or more classes by creating an optimal line that maximizes the space between the closest data point of the other class on the opposite side.

Strength

1. Uses memory effectively
2. Require less memory because they only use a portion of the training data
3. Performs reasonably well when there is a large gap between classes

Weaknesses

1. Requires a long training period
2. Inability to handle overlapping classes
3. Large data sets are not a good fit for the SVM algorithm

2.3.5 Comparison Table of Machine Learning Methods

Table 4: Summarized Table of Machine Learning Methods

Technique	Brief Explanation	Strengths	Weaknesses
Linear Regression	Predicts probability of an outcome using binary classification	Easy to implement, less complex	Presence of outliers can impact performance
Decision Tree	Splits data into branches based on feature values to achieve final outcomes	Handles both categorical and numerical data	Prone to overfitting, does not perform well with complex data
Random Forest	Series of decision tree will be randomly split for better performance.	Accurate prediction, handles both numerical and categorical data	Memory usage can affect hardware in use
Support Vector Machine	A model that sets boundaries between data points to classify them into categories.	SVM works well in high dimensional space with good accuracy	Unable to handle large datasets, training period takes a long time.

2.3.6 Ideal Model Selection

From the literature review section, there are a lot of different models used in a variety of studies and research with each of their own end goal. Each model has its own pros and cons depending on the situation, complexity of the data and the outcome that we want to achieve. However, given that the objective of this project is to predict scam likelihood based on features, the optimum model for that would be Random Forest. In one of the studies from the literature, the research wanted to classify whether or not a credit card transaction was genuine or fraudulent and they used a wide selection of model including RF. Since the dataset for this project contains a lot of categorical features such as scam category and scam types, RF would be a suitable choice to predict scam likelihood in the model development phase during Final Year Project 2.

2.4 Chapter Summary

In a nutshell, chapter 2 discusses about the introduction of cybercrime such as what counts as a cybercrime and what does not or what differentiates cybercrime than a traditional crime. Then we talk about the origins of scams, how it started and when. Apart from that, we explored various scam types that occur in Malaysia from email phishing, online dating to investment scams. Furthermore, the chapter dives into the new scam types that emerge in today's day and age as well as the impact and contributing factors of scams. Other than that, chapter 2 explores five different research studies that touches on various kinds of scams happening all across the world like fraudulent credit card transactions and fake job openings. Finally, we ended the chapter by comparing the performance between different models used in the studies and eventually choose the best model to be implemented in the project, which is Random Forest.

CHAPTER 3

DATA COLLECTION, PREPARATION AND DASHBOARD

3.0 Overview

Chapter 3 discusses about data collection and preparation for this project which shows the dataset to be utilized. This chapter will breakdown in details two phases in the CRISP-DM methodology, which are data understanding and data preparation. It also demonstrates the necessary action during data preparation using some techniques in Microsoft Excel that will help during data exploration phase for further analysis.

3.1 CRISP-DM Methodology

This project will follow closely with CRISP-DM methodology as it will help guide the flow of the project in a clear and concise manner. The first phase is business understanding. From the project, we have established that scam cases have been on the rise in the past couple of years. This is due to the advancement of technology and reliance of mobile devices which makes everyone susceptible to fall victim to scams. Scammers use industry standard technology to deceive victims into thinking that the call or demand is legit. In Project 1, only the initial phases of CRISP-DM methodology will be discussed as modelling and deployment will be done in Project 2. The next subsection will explain and explore more about the next phases in CRISP-DM which is data understanding and data preparation.

3.2 Data Sources

The dataset that will be used in this project is taken from a website called ScamWatch under the Australian Government. The dataset contains around 31,000 number of rows and 9 columns. This dataset can be useful for both descriptive and predictive analysis analytics which tailors with the project objective to predict scam occurrence probability. The dataset is accessible to anyone that visits the website and the CSV file can be downloaded via an export function.

3.3 Data Understanding

The contents of the dataset are explained below:

Table 5: Tabulated scam cases by columns

Features	Column Name	Detail
Temporal Feature	Date	Represents when the report is made using YYYY-MM-DD format
Categorical Feature	State	Represents which state the report is from
	Contact method	Represents the methods in which the scammer approaches the victims
	Age group	Represents the age of the victims
	Gender	Represents the gender of the victims
	Scam Category	Represents the category of scam (General)
	Scam type	Represents the type of scam (Detailed)
Numerical Feature	Aggregated amount loss	Represents how much money loss per report
	Number of report	Represents the number of report in a state for a specific date

Table above explores the various data types found in the dataset chosen for the project. Data types include temporal, categorical and numerical. Based on the table, the dataset comprises of mainly categorical features with six attributes like state, contact method, age group, gender, scam category and scam type. There is only one temporal attribute which is date while numerical data type has two, they are aggregated amount loss and number of reports.

3.4 Data Preparation

Firstly, the dataset was taken from Australian Government Website Called ScamWatch. and downloaded the CSV file. The dataset was cleaned and reformatted using Excel to produce a detailed visualizations for descriptive analysis purposes.

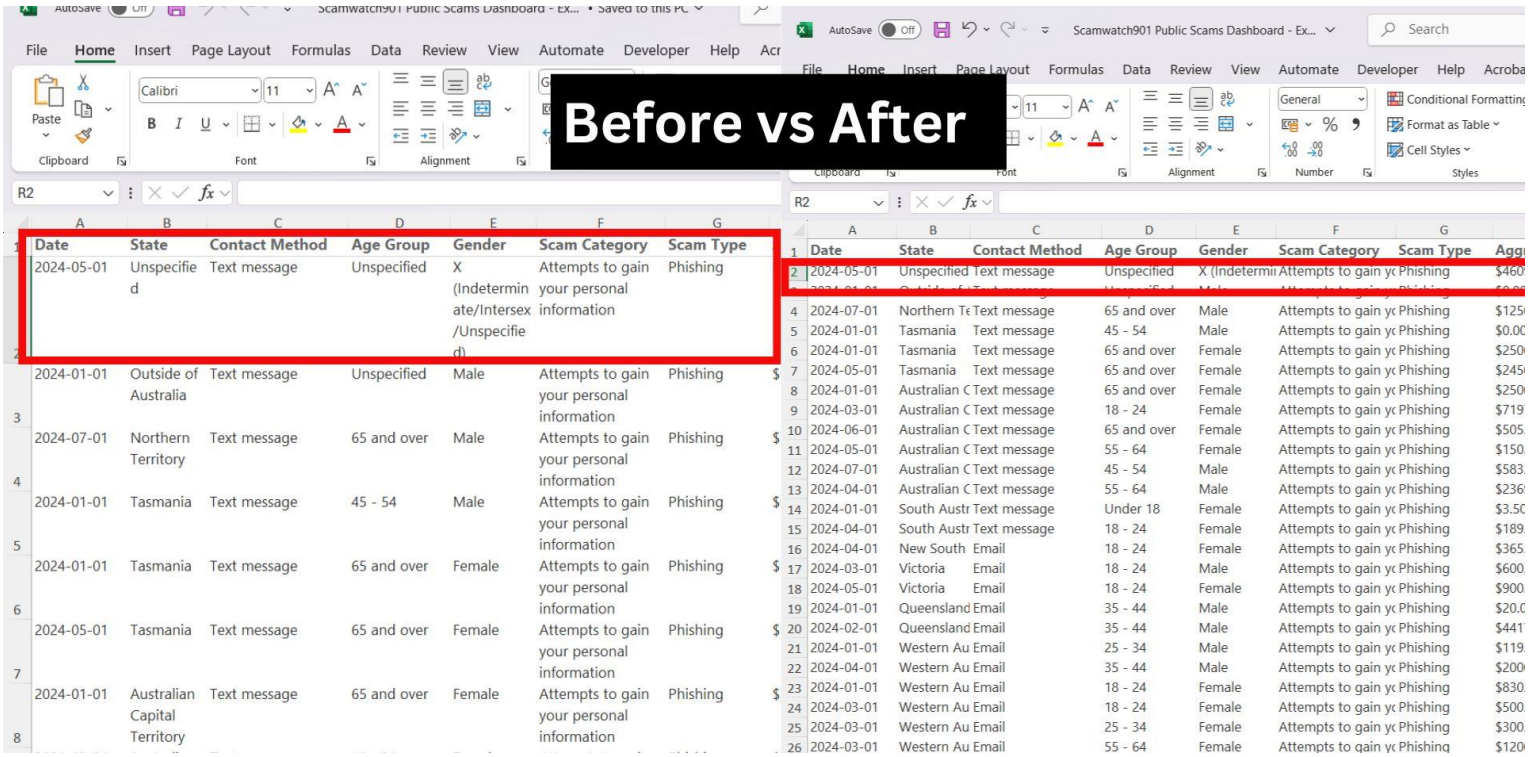


Figure 4: Wrap text feature in Excel to remove multiple lines in a cell

The first cleaning process is called wrap text. The original dataset has multiple lines for long text in one cell. Wrap text feature in Excel can make the text appear in one line to create a more compact-looking view. The next process is Auto Fit Column Width. This is done to improve readability for each column. This was done in Excel under Home Tab, in Cells then format.

Scam Category	Scam Type
Attempts to gain your personal information	Phishing
Attempts to gain your personal information	Identity theft
Attempts to gain your personal information	Hacking
Attempts to gain your personal information	Remote accesss
Buying or selling	Overpayments
Buying or selling	Health and medical products
Buying or selling	Mobile premium services
Buying or selling	Psychic and clairvoyant
Buying or selling	Classifieds
Buying or selling	Online shoppings
Buying or selling	False billing
Threats and extortion	Threats to life, arrest or other
Investment scams	Investments
Dating and romance	Dating and romances
Unexpected winnings	Travel, prizes and lotterys
Jobs and employment	Jobs and employments
Unexpected money	inheritance and unexpected money
Investment scams	Betting and sports investments
Unexpected money	Rebates
Fake charities	Fake charitys
Jobs and employment	Pyramid schemes
Threats and extortion	Ransomware and malware
Other	Others

Row Labels	Count of Scam Type
Betting and sports investment scams	214
Classified scams	2181
Dating and romance scams	1090
Fake charity scams	342
False billing	2824
Hacking	1944
Health and medical products	755
Identity theft	2641
inheritance and unexpected money	785
Investment scams	1932
Jobs and employment scams	1040
Mobile premium services	525
Online shopping scams	2976
Other scams	3045
Overpayment scams	863
Phishing	3201
Psychic and clairvoyant	75
Pyramid schemes	94
Ransomware and malware	532
Rebate scams	1040
Remote access scams	1407
Threats to life, arrest or other	1040
Travel, prizes and lottery scams	921
Grand Total	**31467**

Row Labels	Count of Scam Category
Attempts to gain your personal information	9193
Buying or selling	10199
Dating and romance	1090
Fake charities	342
Investment scams	2146
Jobs and employment	1134
Other	3045
Threats and extortion	1572
Unexpected money	1825
Unexpected winnings	921
Grand Total	**31467**

Figure 5: Count function in Pivot Table Excel

Since the dataset contains over 30,000 rows of data, it is difficult to identify the unique values for certain columns. So, to get the unique values, remove duplicates was done. This part is applied to 3 columns which are state column, scam category column and scam type column. Next, a pivot table is created to obtain the count function. There are a lot of repetitive values in the dataset, pivot table can automatically count all the unique values for a specific column and sum them up.

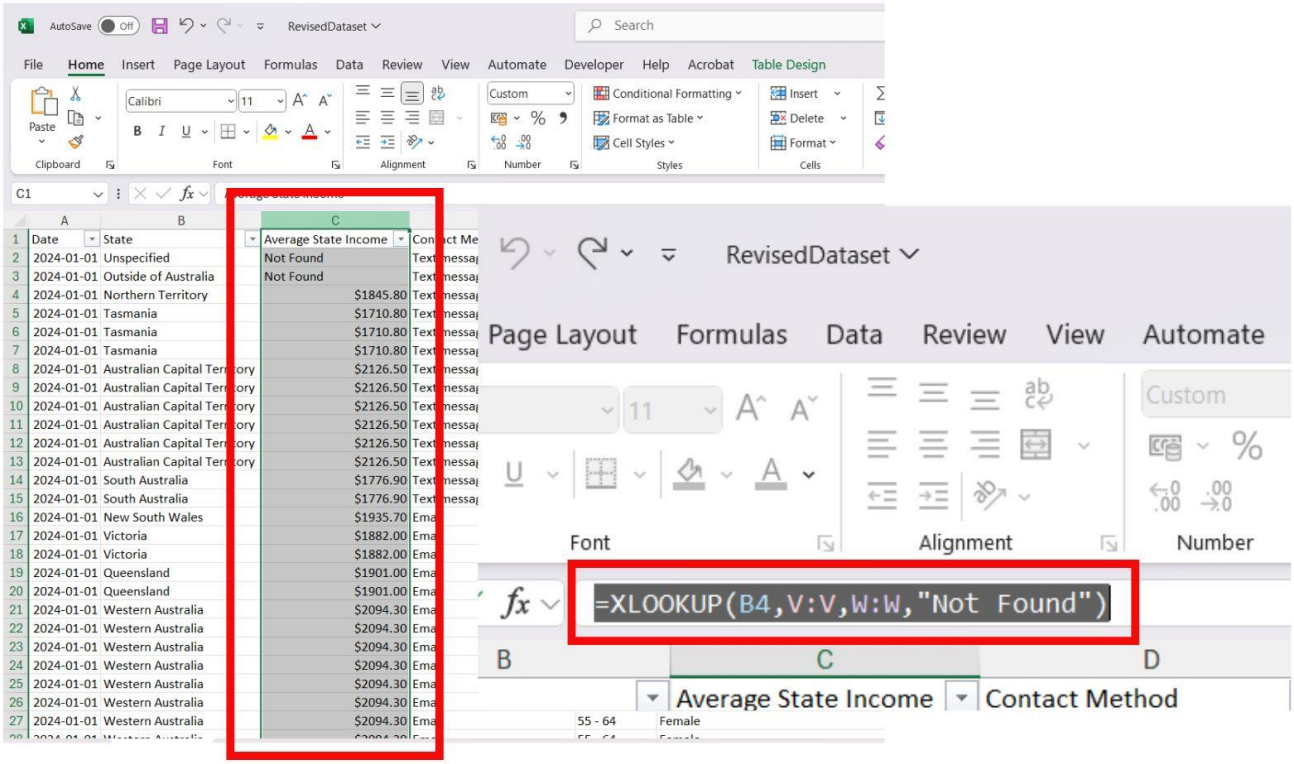


Figure 6: XLOOKUP function in Excel to add new column

Then, to further aid the process of analysis and data visualization, a new column is created using Excel's XLOOKUP function. The new column's name is State Average Income. It is done by referencing the state and their respective average income. For state values Unspecified and Outside of Australia, Excel returns "Not Found".

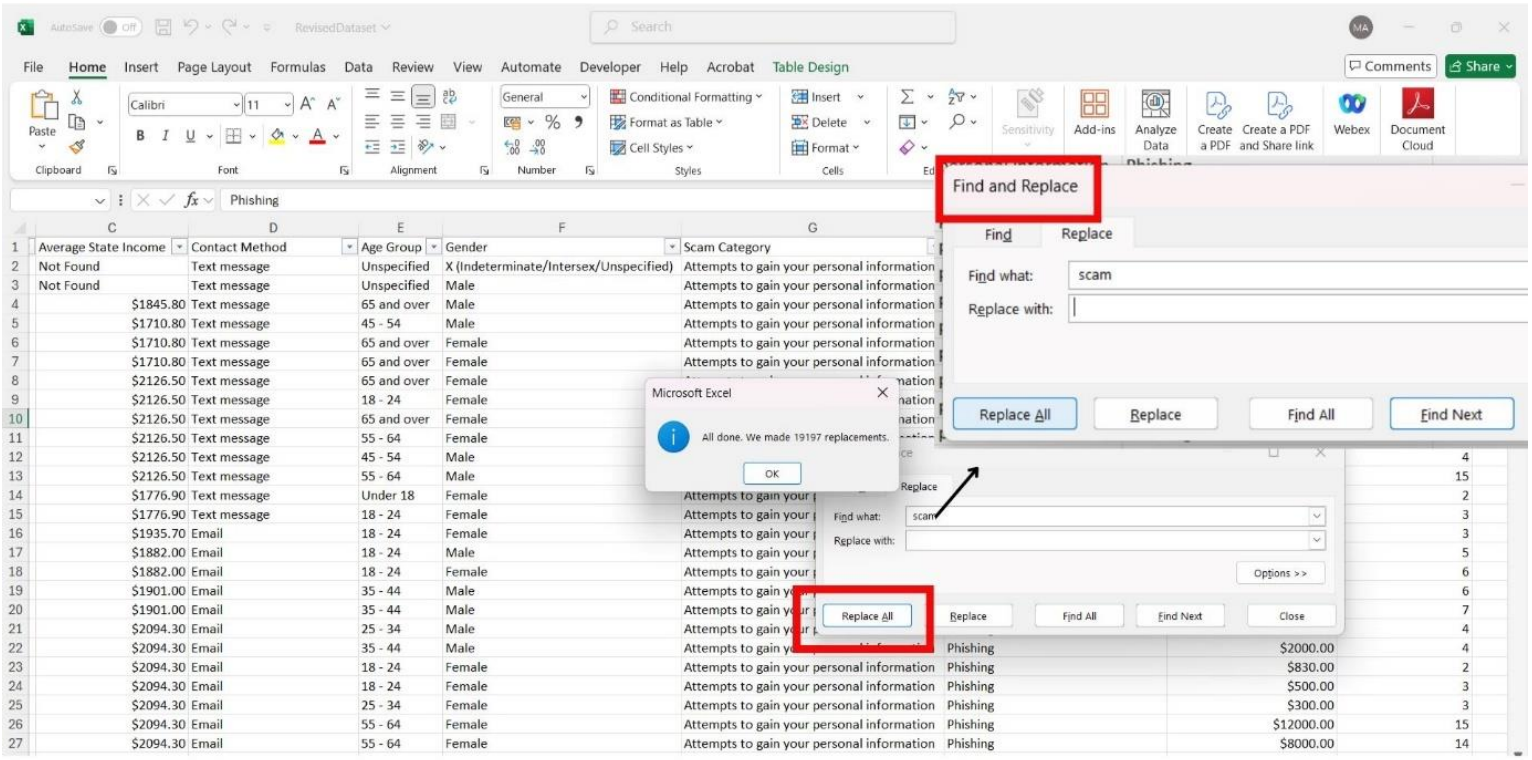


Figure 7: Find and Replace in Excel

Lastly, for every scam type, there is a word scam at the very end of the sentence. This creates redundancy and make reading harder because the sentence is very long. To counter this problem, using Excel's find and remove feature was used to delete the word scam in Scam Type Column. This shortens the length for each type and eliminates redundancies.

3.5 Dashboard Insight

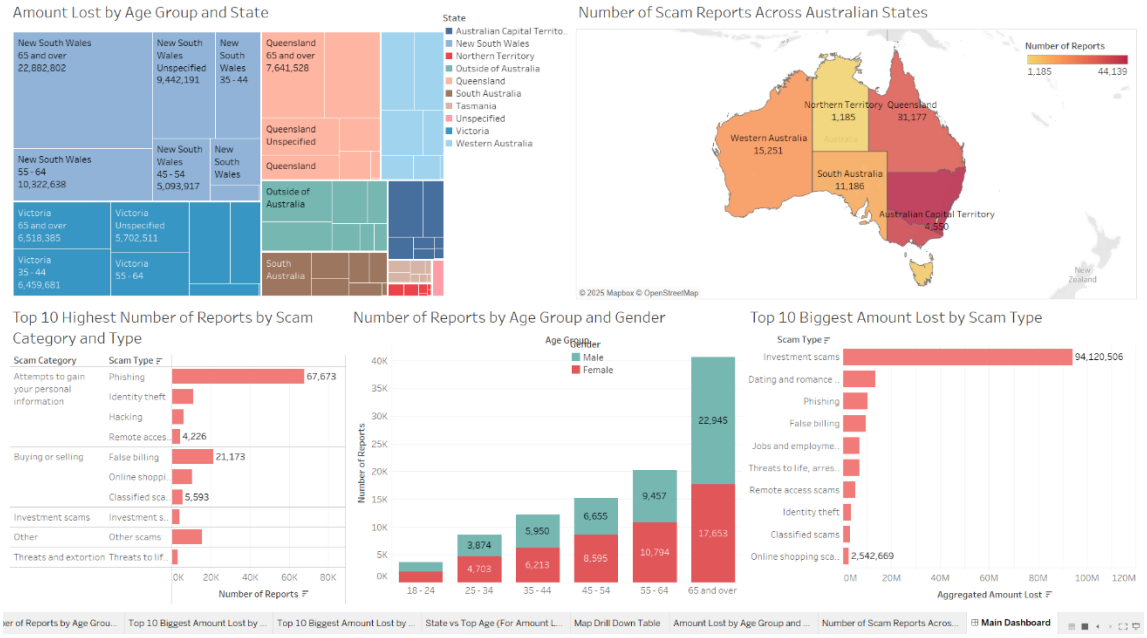


Figure 8: Overview of 5 key insights from the dashboard

This dashboard presents five key insights derived from the dataset which offer useful information to the public for educational purposes. Moreover, it serves as a platform for Malaysia's government to highlight the severity and impact of scams to help its decision-making processes in addressing this issue in the country. The visualizations are extracted from the raw data to make the information presentable and useful to a wider audience like the public and policymakers. By showcasing which states have the highest reported cases, the total financial loss for each scam type and the most affected demographic groups, this dashboard covers a detailed overview of the scam universe. Not just that, the dashboard also emphasizes the importance and the power of using data-driven approaches to address an issue, particularly scams in this situation. While the dataset is based on Australian scam reports, Malaysians can use the insights as a reference about how scam cases occur locally. Thus, this dashboard clearly serves as a tool for analysis to fight scams.

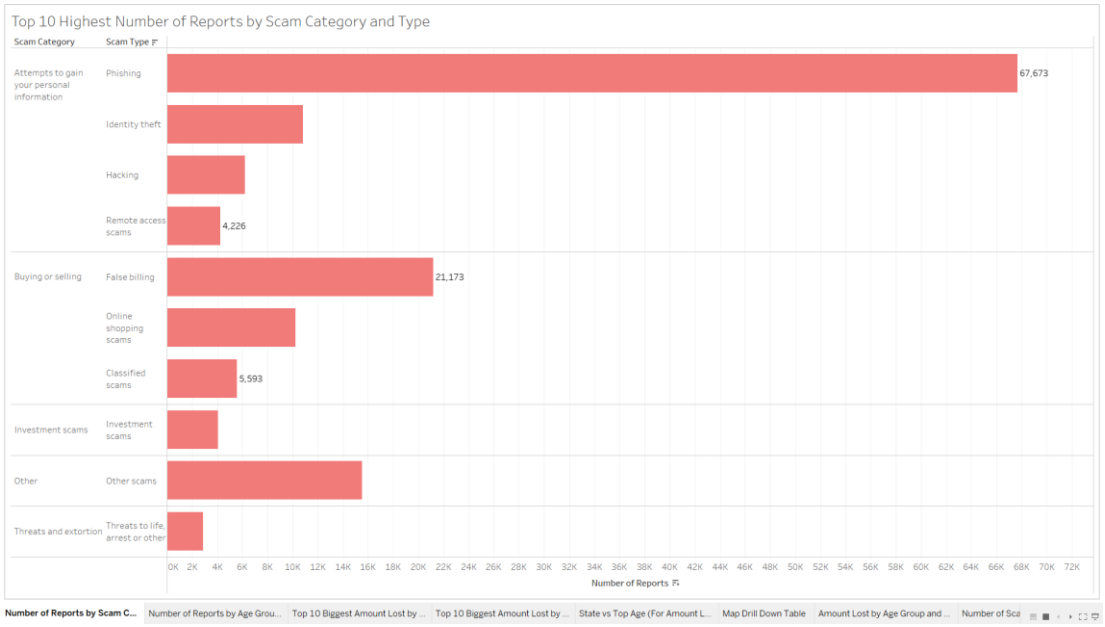


Figure 9: Top 10 Highest Number of Reports by Scam Category and Type

The bar chart in Figure 1 illustrates top 10 highest number of reports grouped by scam category and their type. Each scam type in its category is sorted in descending order to provide a natural way of reading from top to bottom and left to right. The chart shows that phishing dominates a total of 67,673 reported cases, significantly surpassing all other scam types. Phishing is easy to fall victim to because of our frequent daily use of mobile devices leading to a higher tendency to click suspicious links by accident. The vast difference between phishing and other scams highlights the need for further analysis why it is so successful. Phishing's ability to exploit online behaviour, such as responding to emails and clicking on links, may contribute to its high number of reported cases. This chart aims to spread awareness on the various types of scams happening around us and to implement preventive measures to tackle them in the future.

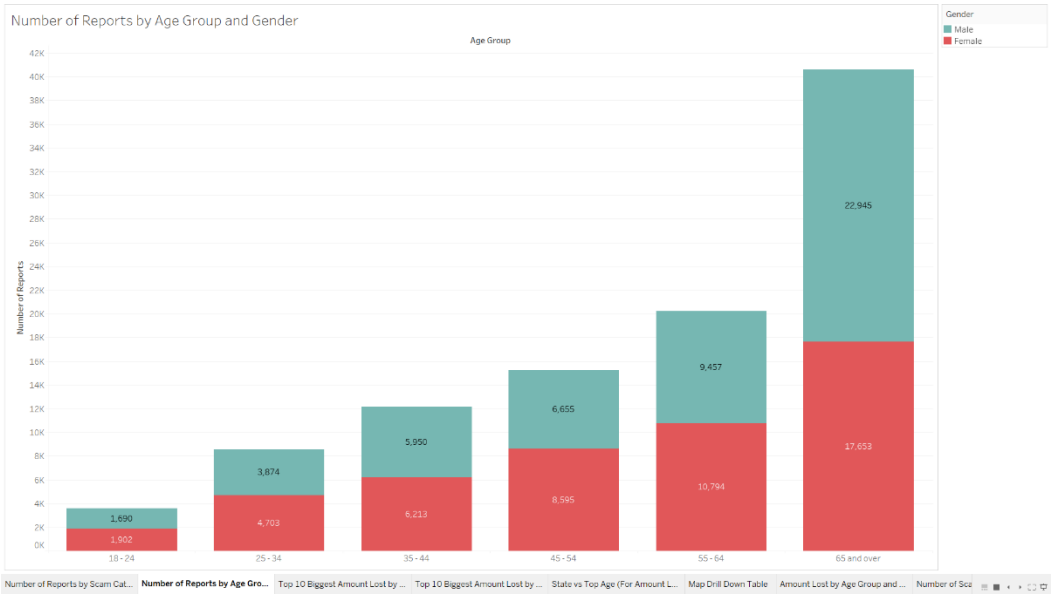


Figure 10: Number of Reports by Age Group and Gender

Figure 2 shows a stacked column chart that represents the number of reports for six different age groups and their respective gender. At first glance, the most obvious age group at risk is individuals aged 65 and over with over 42,000 reports in 2024. This may be due to the fact that older groups of people are easier to manipulate and are vulnerable which makes them an easy target for scammers. Interestingly, only this age group has more male victims than females, whereas in the other groups, the number of reports for females is higher than for males. This could potentially be due to two reasons, the first being younger women are more trusting of scammers and often panic during such situation, while men tend to perform fact-checking first before giving out personal details to the scammers. And secondly, men of older age usually handle their own wealth hence they have the money in their own accounts which is why scammers make them the primary target.

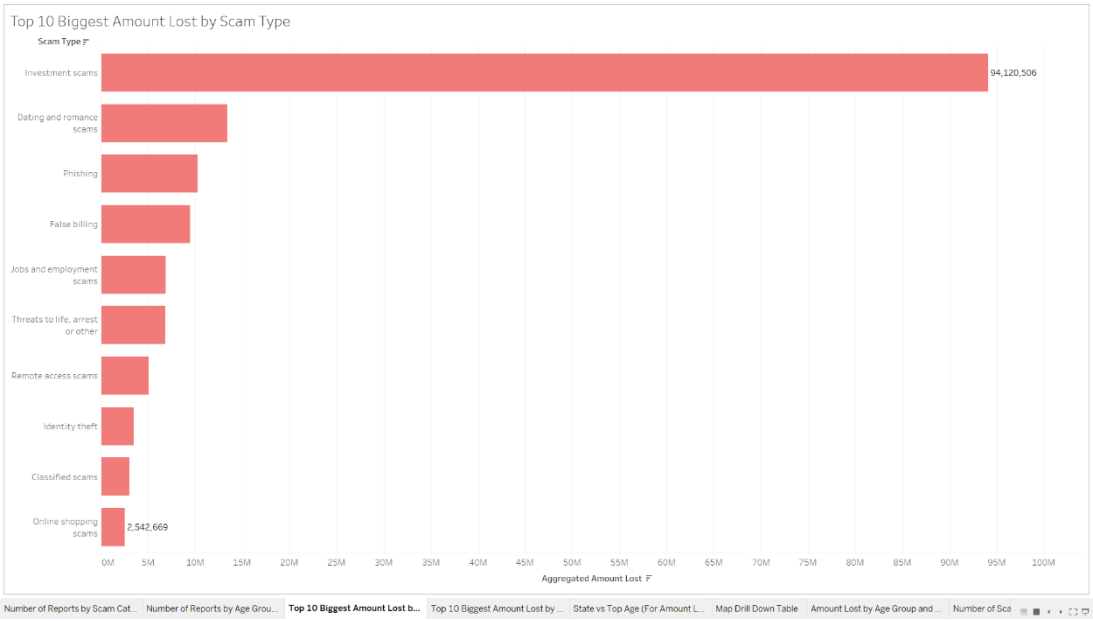


Figure 11: Top 10 Biggest Amount Lost by Scam Type

The chart above represents the top ten highest amounts lost for each scam type. Investment scams take the lead with figures reaching over 94 million dollars while the least impactful scam in the chart is online shopping with a little over 2.5 million dollars in losses. To be fair, the rest of the scam type appear insignificant because the scale of the entire chart is skewed or is catering for the big number from investment scams. If we were to exclude this scam type, then the remaining bar could extend much further and the overall message would be different. However, the chart is to focus on the top 10 highest, therefore there is no other way to present it. Focusing on investment scams, it is understandable why the number is so much higher compared to the rest because investing costs a ton of money and once the investors realize they are being scammed, there have already incurred a substantial loss. Although the other types of scams seem like a non big deal, it is worth highlighting that dating scams contribute significantly with it being the number two in the chart with more than 10 million dollars in losses.

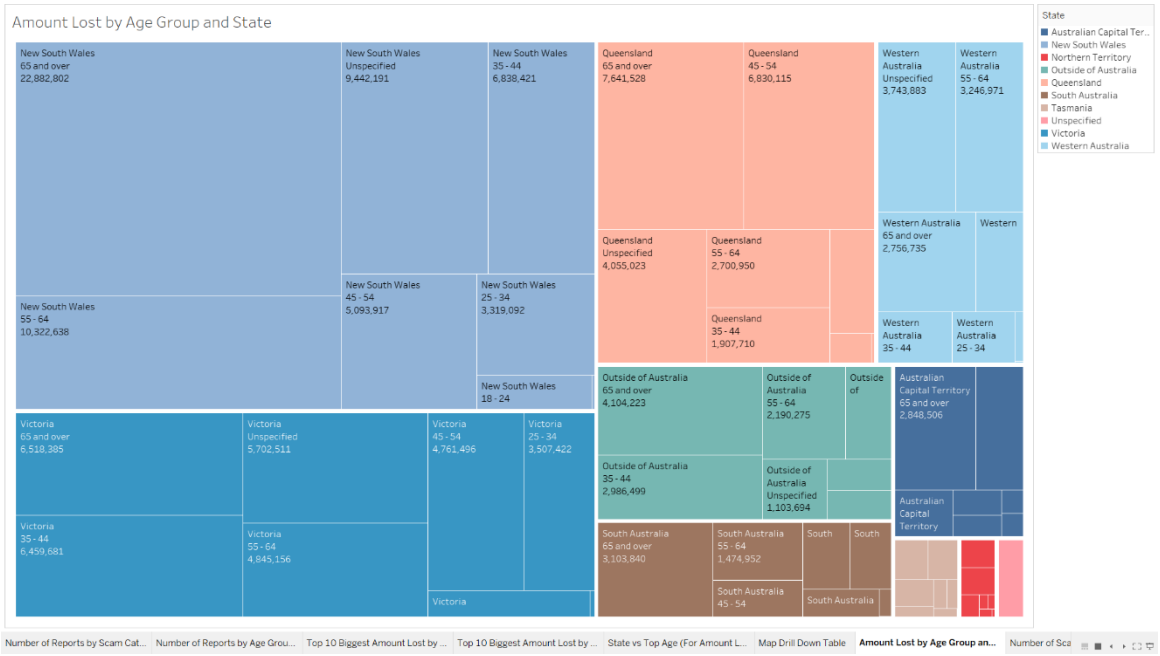


Figure 12: TreeMap of Amount Lost by Age Group and State

Figure 3 shows a treemap diagram that represents the amount lost grouped by victim's age group and their respective state. The treemap is differentiated using different colors that represents each state. A legend on the side of the diagram is provided as a reference to identify colors for the states. The size of the tile highlights the amount lost for each state, a bigger size indicates a higher loss. Additionally, for each tile, there are multiple smaller ones that break down the losses by age groups in that state. Inside each small tile is a text labeling the name of the state, the age group and the respective amount loss. This treemap helps identify two main information. The first being which state has the highest amount of losses, and second, which age group is most affected from scams.

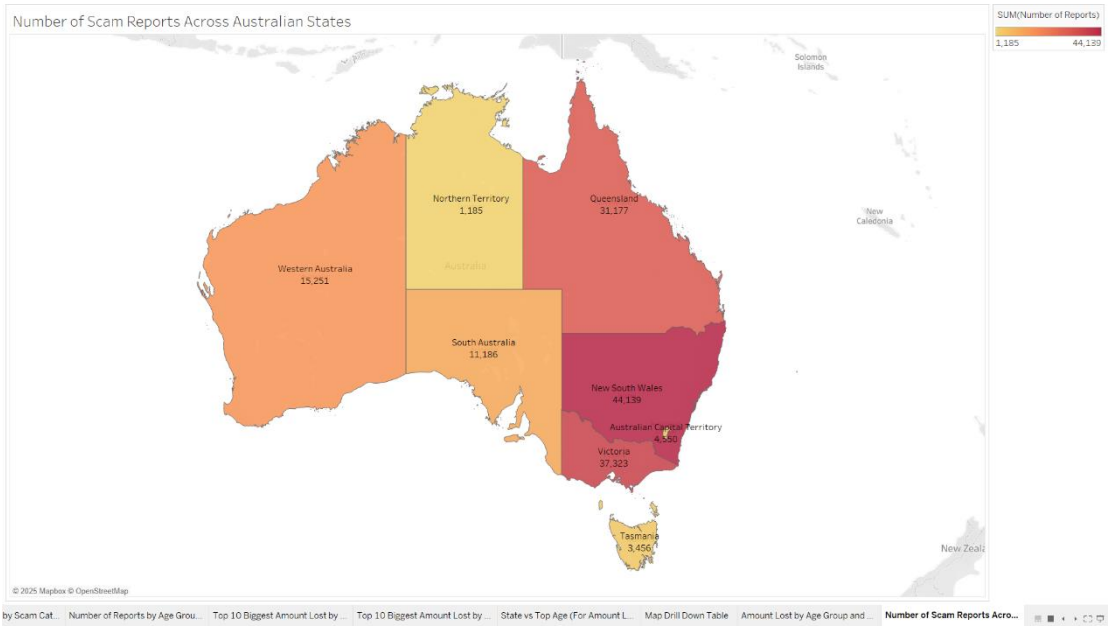


Figure 13: Number of Scam Reports Across Australian States

The last chart is a map of every major state in Australia and is used to represents the number of reports for each state. The intensity of the color indicates high or low number of reports in a particular state with lighter color resembling lower report rates and darker colors indicating the opposite. The state with the highest reported cases is New South Wales (NSW) with over 44,000 cases followed by Queensland 31,000. The illustration of this chart is essential to visualize the geographical distribution of scam activities across the country. By identifying the states with the most reports, authorities can strategize ways to tackle scams in highly affected areas such as NSW. Apart from that, the chart helps raise awareness to the public as they can identify the scam rates in their states.

Chapter 4

MODEL DEVELOPMENT

4.0 Introduction

Chapter 5 explains the whole process of developing a prediction model that learns from existing data that will then be tested using a set of testing data. This chapter will breakdown step by step process to develop the model from data processing, data manipulation as well data analysis and also model evaluation to assess the model's performance.

4.1 Model Development

To create the model, I will be exploring Jupyter Notebook which uses Python and Pandas libraries which is useful for data related processes such as data transformation, data analysis, data manipulation and also a bunch of options for machine learning models. This makes Jupyter Notebook an ideal choice to develop the model which will be explain in detail in the next chapter. In chapter 2, we decided to use Random Forest as the most suitable model for the project to predict the number of scam cases in Australia.

4.2 Full Code for Model Development

```
import pandas as pd


# Load the dataset
df = pd.read_csv("Jan-Dec Dataset.csv")


from sklearn.model_selection import train_test_split
from sklearn.preprocessing import OrdinalEncoder, StandardScaler
from sklearn.ensemble import RandomForestRegressor


# Drop unused column
```

```
df = df.drop(columns=["Season"])

# One-hot encode 'State'
df = pd.get_dummies(df, columns=["State"], drop_first=True)

# Ordinal encode Age Group
age_order = [["Under 18", "18 - 24", "25 - 34", "35 - 44", "45 - 54", "55 - 64", "65 and
over"]]
encoder = OrdinalEncoder(categories=age_order)
df["Age Group"] = encoder.fit_transform(df[["Age Group"]])

# Normalize categorical columns
df["Scam Category"] = df["Scam Category"].map(df["Scam
Category"].value_counts(normalize=True))
df["Contact Method"] = df["Contact Method"].map(df["Contact
Method"].value_counts(normalize=True))
df["Scam Type"] = df["Scam Type"].map(df["Scam Type"].value_counts(normalize=True))

# Encode Gender
df["Gender"] = df["Gender"].map({"Male": 0, "Female": 1})

# Convert boolean to int
df = df.astype({col: int for col in df.select_dtypes(include=["bool"]).columns})

# Clean Amount_lost
df["Amount_lost"] = df["Amount_lost"].replace(r'[\$,]', '', regex=True).astype(float)

# Define X and y
X = df.drop(columns=["Number_of_reports", "StartOfMonth"])
y = df["Number_of_reports"]

# Split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

X_test_original = X_test.copy()
```

```
# Scale
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)

# Train model
model = RandomForestRegressor(n_estimators=100, random_state=42)
model.fit(X_train, y_train)

# Predict
y_pred = model.predict(X_test)

X_test_original["Actual_Number_of_Reports"] = y_test.values
X_test_original["Predicted_Number_of_Reports"] = y_pred

X_test_original.to_csv("NEW_RF_Predictedt.csv", index=False)

print("Sukses")
```

The code above is for the Random Forest model development and I will be explaining thoroughly what they do. The first few lines of code is what we call preprocessing. These are the kind of preparation that needs to be done before feeding the data into the model. These preprocessing steps are crucial towards the model's performance as well as achieving the desired output. However, before we perform any of that, we need to import several packages from python first.

```
import pandas as pd

# Load the dataset
df = pd.read_csv("Jan-Dec Dataset.csv")
```

First, we load the CSV file dataset into a DataFrame for preprocessing steps and for the model development later.

```
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import OrdinalEncoder, StandardScaler
from sklearn.ensemble import RandomForestRegressor
```

train-test_split is a method that will be used later in the codes and as the name implies, they train the model and then test to see how they perform. Normally the split ratio for training and testing is 70:30 or 80:20. Next is OrdinalEncoder and StandardScaler. These functions are used for the much bigger and complicated columns such as Scam Category, Scam Type, Contact Method, Age Group and State. Essentially what they do is either assign each value in a column with a number or scale the numbers to give them a spread value. And lastly, we have the model itself, which its full name is Random Forest Regressor.

```
# Drop unused column
df = df.drop(columns=["Season"])


# One-hot encode 'State'
df = pd.get_dummies(df, columns=["State"], drop_first=True)
```

The first step is to drop any columns whether it has no significance for the model or we need to drop it because it is the target column. The former is Season state and the latter is Number_of_reports. State column gets converted via One-hot encoder because its value needs to be retained as it is. So one way to achieve that is by this method. What this does is that for each unique value in the column, a new column will be created. And to indicate what state a particular row has is by 1 or 0.

```
# Ordinal encode Age Group
age_order = [["Under 18", "18 - 24", "25 - 34", "35 - 44", "45 - 54", "55 - 64", "65 and over"]]
encoder = OrdinalEncoder(categories=age_order)
df["Age Group"] = encoder.fit_transform(df[["Age Group"]])
```

Age group column receives a similar treatment like State column however this time around each value in the column gets assigned a number instead of a column. The word ordinal means that the order in which the group appears in the code is reflected in the number its assigned to. For example "Under 18" is first on the list, so it will be assigned with 0 and "18-24" will be getting number 1 and so on.

```
# Normalize categorical columns
df["Scam Category"] = df["Scam Category"].map(df["Scam
Category"].value_counts(normalize=True))
df["Contact Method"] = df["Contact Method"].map(df["Contact
Method"].value_counts(normalize=True))
df["Scam Type"] = df["Scam Type"].map(df["Scam Type"].value_counts(normalize=True))

# Encode Gender
df["Gender"] = df["Gender"].map({"Male": 0, "Female": 1})
```

Three columns, Scam Category, Contact Method and Scam Type are columns with the most unique values in them and will undergo a normalization process. How it works is rather simple. The frequency of occurrence of the values in the entire dataset will be used as a new representation of the value. For instance, Phishing, in Scam Type column, appears 9% in the entire dataset. Hence it will now be converted to 0.09. The exact same process is done for the other two columns. Gender column only has two values, Male and Female so the conversion process is quite easy where Male becomes 1 and Female becomes 0.

```
# Define X and y
X = df.drop(columns=["Number_of_reports", "StartOfMonth"])
y = df["Number_of_reports"]
# Split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

X_test_original = X_test.copy()

# Scale
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)

# Train model
model = RandomForestRegressor(n_estimators=100, random_state=42)
model.fit(X_train, y_train)

# Predict
y_pred = model.predict(X_test)

X_test_original["Actual_Number_of_Reports"] = y_test.values
X_test_original["Predicted_Number_of_Reports"] = y_pred

X_test_original.to_csv("NEW_RF_Predictedt.csv", index=False)

print("Sukses")
```

Once these important steps are accomplished, we can start developing the model. We first start by specifying the test size of the dataset. In this case, 20% will be for testing and the remaining 80% will be used for training purposes. This is to increase the model's performance by allowing it to learn more of the data. Next, we apply the data into the model and compare the actual value with the model's prediction. We can see the prediction better in CSV format and allow for further analysis.

```
import pandas as pd
df = pd.read_csv("NEW_RF_Predicted.csv")
```

The code above is to upload the new dataset that we have predicted using Random Forest. The dataset now contains a new column, predicted cases and many expanded new columns which will need to be converted back to its original value. The steps to convert the columns will be explained down below.

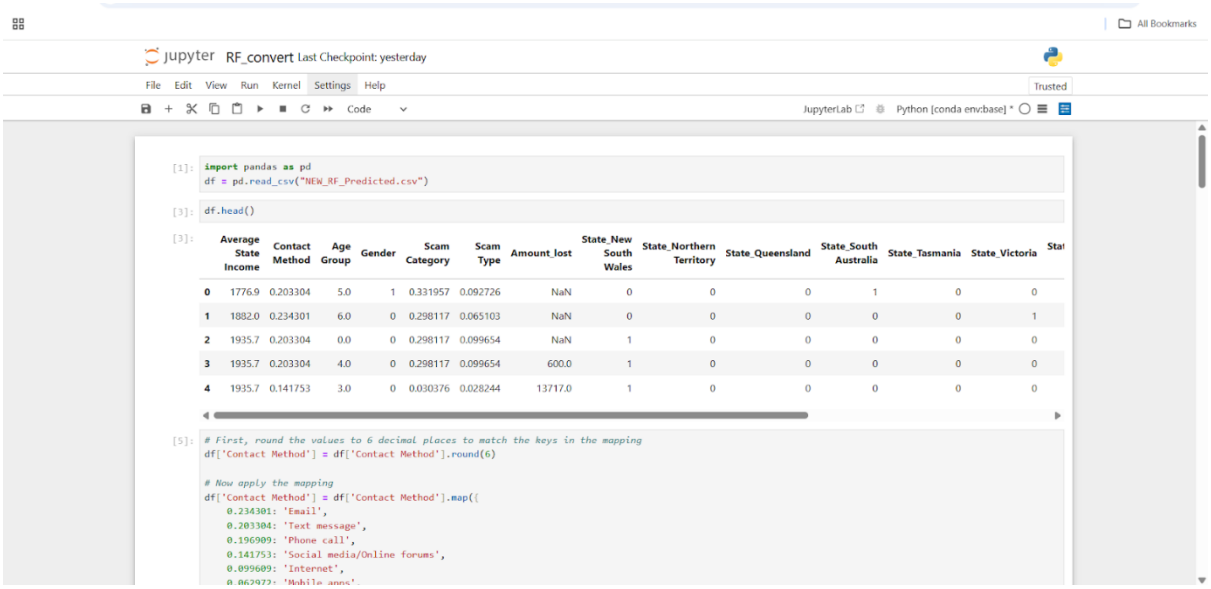


Figure 14: Loading NEW_RF_Predicted.csv into DataFrame

```
# Round the values to 6 decimal places to match the keys in the mapping
df['Contact Method'] = df['Contact Method'].round(6)

# Now apply the mapping
df['Contact Method'] = df['Contact Method'].map({
    0.234301: 'Email',
    0.203304: 'Text message',
    0.196909: 'Phone call',
    0.141753: 'Social media/Online forums',
    0.099609: 'Internet',
    0.062972: 'Mobile apps',
    0.033085: 'Mail',
    0.026024: 'In person',
    0.001066: 'unspecified',
    0.000977: 'Fax'
```

```
})
```

Averag Stat Incom	Contact Method	Age Group	Gender	Scam Category	Scam Type	Amount_lost	State_New South Wales
1776.	0.203304	5.0	1	0.331957	0.092726	NaN	0
1882.	0.234301	6.0	0	0.298117	0.065103	NaN	0
1935.	0.203304	0.0	0	0.298117	0.099654	NaN	1
1935.	0.203304	4.0	0	0.298117	0.099654	600.0	1
1935.	0.141753	3.0	0	0.030376	0.028244	13717.0	1

Average State Income	Contact Method	Age Group	Gender	Scam Category	Scam Type	Amount_lost	State_New South Wales
1776.9	Text message	5.0	1	0.331957	False billing	NaN	0
1882.0	Email	6.0	0	0.298117	Hacking	NaN	0
1935.7	Text message	0.0	0	0.298117	Phishing	NaN	1
1935.7	Text message	4.0	0	0.298117	Phishing	600.0	1
1935.7	Social media/Online forums	3.0	0	0.030376	Jobs and employment	13717.0	1

Figure 15: Comparison before and after for Contact Method column

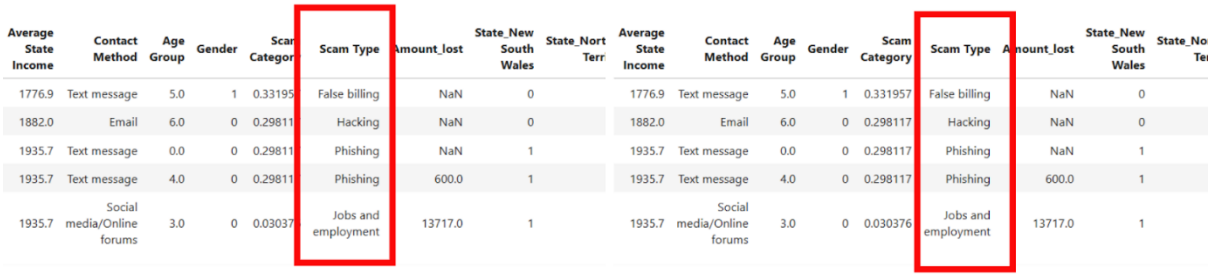
The code lists each unique number in column “Contact Method” and converts them into their original text value from the original dataset for easy interpretation. The image above shows that “Contact Method” column is now in text values which is essential for analysis and dashboard development.

```
# Round the Scam Type values to 6 decimal places
df['Scam Type'] = df['Scam Type'].round(6)

# Mapping dictionary for Scam Type
scam_type_mapping = {
    0.099654: 'Phishing',
    0.098499: 'Other',
    0.097611: 'Online shopping',
    0.092726: 'False billing',
    0.087752: 'Identity theft',
    0.072875: 'Classified',
    0.065103: 'Hacking',
    0.057332: 'Investment',
    0.045608: 'Remote access',
    0.035438: 'Dating and romance',
    0.033085: 'Threats to life, arrest or other',
    0.032507: 'Rebate',
    0.028244: 'Jobs and employment',
    0.027267: 'Travel, prizes and lottery',
    0.026557: 'Overpayment',
```

```
    0.024203: 'inheritance and unexpected money',
    0.023759: 'Health and medical products',
    0.016653: 'Ransomware and malware',
    0.016520: 'Mobile premium services',
    0.008837: 'Fake charity',
    0.005729: 'Betting and sports investment',
    0.002132: 'Pyramid schemes',
    0.001910: 'Psychic and clairvoyant'
}

# Apply the mapping
df['Scam Type'] = df['Scam Type'].map(scam_type_mapping)
```



Average State Income	Contact Method	Age Group	Gender	Scam Category	Scam Type	Amount_lost	State_New South Wales	State_Nort Terr
1776.9	Text message	5.0	1	0.33195	False billing	NaN	0	
1882.0	Email	6.0	0	0.29811	Hacking	NaN	0	
1935.7	Text message	0.0	0	0.29811	Phishing	NaN	1	
1935.7	Text message	4.0	0	0.29811	Phishing	600.0	1	
1935.7	Social media/Online forums	3.0	0	0.03037	Jobs and employment	13717.0	1	

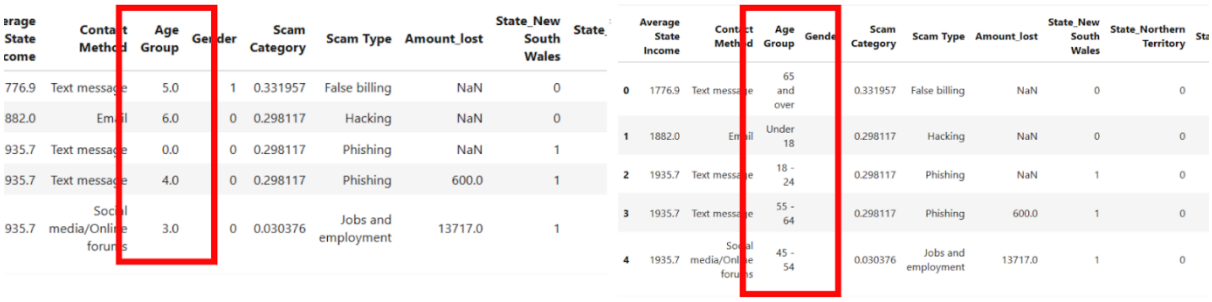
Average State Income	Contact Method	Age Group	Gender	Scam Category	Scam Type	Amount_lost	State_New South Wales	State_No Te
1776.9	Text message	5.0	1	0.331957	False billing	NaN	0	
1882.0	Email	6.0	0	0.298117	Hacking	NaN	0	
1935.7	Text message	0.0	0	0.298117	Phishing	NaN	1	
1935.7	Text message	4.0	0	0.298117	Phishing	600.0	1	
1935.7	Social media/Online forums	3.0	0	0.030376	Jobs and employment	13717.0	1	

Figure 16: Comparison before and after for Scam Type column

The same process goes to the “Scam Type” column where every value was previously converted into numerical values based on the frequency it appears in the dataset. It now needs to be converted back to text to be able to read and be used in developing the dashboard.

```
# Mapping from number (as float) back to age group labels
age_group_reverse_mapping = {
    0.0: “18 – 24”,
    1.0: “25 – 34”,
    2.0: “35 – 44”,
    3.0: “45 – 54”,
    4.0: “55 – 64”,
    5.0: “65 and over”,
    6.0: “Under 18”
}
```

```
# Apply the reverse mapping
df[“Age Group”] = df[“Age Group”].map(age_group_reverse_mapping)
```



erage State come	Contact Method	Age Group	Gender	Scam Category	Scam Type	Amount_lost	State_New South Wales	State_
776.9	Text message	5.0	1	0.331957	False billing	NaN	0	
882.0	Email	6.0	0	0.298117	Hacking	NaN	0	
935.7	Text message	0.0	0	0.298117	Phishing	NaN	1	
935.7	Text message	4.0	0	0.298117	Phishing	600.0	1	
935.7	Social media/Online forums	3.0	0	0.030376	Jobs and employment	13717.0	1	

	Average State Income	Contact Method	Age Group	Gende	Scam Category	Scam Type	Amount_lost	State_New South Wales	State_Northern Territory	Sta
0	1776.9	Text message	65 and over		0.331957	False billing	NaN	0	0	
1	1882.0	Email	Under 18		0.298117	Hacking	NaN	0	0	
2	1935.7	Text message	18 - 24		0.298117	Phishing	NaN	1	0	
3	1935.7	Text message	55 - 64		0.298117	Phishing	600.0	1	0	
4	1935.7	Social media/Online forums	45 - 54		0.030376	Jobs and employment	13717.0	1	0	

Figure 17: Comparison before and after for Age Group column

“Age Group” column went to a slightly different approach than the other two previously mentioned columns. There are six groups altogether in the column and each has been assigned with a number, precisely between zero and six. This column is also required to be in its original state.

```
# List of one-hot encoded state columns
state_columns = [
    'State_New South Wales',
    'State_Northern Territory',
    'State_Queensland',
    'State_South Australia',
    'State_Tasmania',
    'State_Victoria',
    'State_Western Australia'
]

# Recreate 'State' column from one-hot encoding
df['State'] = df[state_columns].idxmax(axis=1)

# Remove the 'State_' prefix to get clean state names
df['State'] = df['State'].str.replace('State_', '', regex=False)

# Drop the one-hot columns
df.drop(columns=state_columns, inplace=True)
```

State_New South Wales	State_Northern Territory	State_Queensland	State_South Australia	State_Tasmania	State_V	nder	Scam Category	Scam Type	Amount_lost	Actual_Number_of_Reports	Predicted_Number_of_Reports	State
0	0	0	1	0		1	0.331957	False billing	NaN	4	5.750036	South Australia
0	0	0	0	0		0	0.298117	Hacking	NaN	6	8.949000	Victoria
1	0	0	0	0		0	0.298117	Phishing	NaN	4	1.249042	New South Wales
1	0	0	0	0		0	0.298117	Phishing	600.0	73	79.203219	New South Wales
1	0	0	0	0		0	0.030376	Jobs and employment	13717.0	2	2.080000	New South Wales

Figure 18: Comparison before and after conversion for State column

Based on the image above, we can see that each state gets its own separate column. The conversion was critical in order to develop model since the dataset must be in numerical data type. Now, what the code does it combine all columns into one single column called “State”. If the row contains the number “1” in it, then the State will be equal to that specific column. For example, across seven separate state columns, a row has only one column containing the number “1” which is Victoria while the rest is “0”. Therefore the new combined column “State” will set that row to be “Victoria”.

```
new_order = [
    'Age Group',
    'State',
    'Average State Income',
    'Scam Type',
    'Contact Method',
    'Amount_lost',
    'Actual_Number_of_Reports',
    'Predicted_Number_of_Reports'
]

df = df[new_order]
```

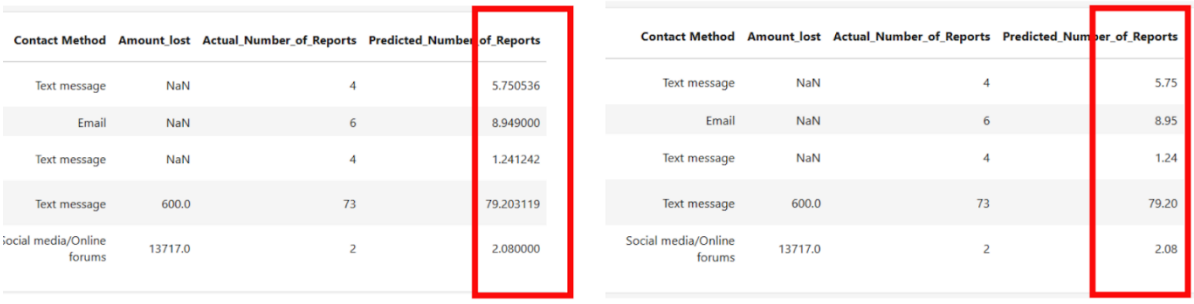
der	Scam Category	Scam Type	Amount_lost	Actual_Number_of_Reports	Predicted_Number_of_Reports	State
1	0.331957	False billing	NaN	4	5.750536	South Australia
0	0.298117	Hacking	NaN	6	8.949000	Victoria
0	0.298117	Phishing	NaN	4	1.241242	New South Wales
0	0.298117	Phishing	600.0	73	79.203119	New South Wales
0	0.030376	Jobs and employment	13717.0	2	2.080000	New South Wales

Age Group	State	Average State Income	Scam Type	Contact Method	Amount_lost	Actual_Number_of_Reports	Pre
65 and over	South Australia	1776.9	False billing	Text message	NaN	4	
Under 18	Victoria	1882.0	Hacking	Email	NaN	6	
18 - 24	New South Wales	1935.7	Phishing	Text message	NaN	4	
55 - 64	New South Wales	1935.7	Phishing	Text message	600.0	73	
45 - 54	New South Wales	1935.7	Jobs and employment	Social media/Online forums	13717.0	2	

Figure 19: Comparison of column order after conversion

At the beginning of the process, all the columns are mixed up and not in a desired order. What this code does is that it rearranges all of the columns in a way that is logical and readable from left to right. It also ensures that the predicted column is located at the very end and right beside the actual value for easier comparison.

```
df['Predicted_Number_of_Reports'] = df['Predicted_Number_of_Reports'].round(2)
```



Contact Method	Amount_lost	Actual_Number_of_Reports	Predicted_Number_of_Reports
Text message	NaN	4	5.750536
Email	NaN	6	8.949000
Text message	NaN	4	1.241242
Text message	600.0	73	79.203119
ocial media/Online forums	13717.0	2	2.080000

Contact Method	Amount_lost	Actual_Number_of_Reports	Predicted_Number_of_Reports
Text message	NaN	4	5.75
Email	NaN	6	8.95
Text message	NaN	4	1.24
Text message	600.0	73	79.20
Social media/Online forums	13717.0	2	2.08

Figure 20: Decimal points for Predicted values column

During prediction, the model came up with very long, accurate but unnecessary decimal points. To counter that situation, the code above eliminates or specifies the predicted column only to be two decimal points at most. This is to give a very clean and somewhat standardize across the dataset.

```
Df['Predicted_Number_of_Reports'] = df['Predicted_Number_of_Reports'].clip(lower=0)


df.to_csv("Updated_RF.csv", index=False)
```

Also during prediction, sometimes the predicted values can be way off such as negative values or close to zero. To change that, the code above helps to take negative and close to zeros values

to become zero altogether. This gives the dataset a more unified and consistent look. Lastly all these changes will then be saved in a CSV file to help create the predictive dashboard.

Chapter 5

DASHBOARD DEVELOPMENT

5.0 Introduction

Chapter 7 is about building the dashboard for both descriptive and predictive analysis using Python Streamlit. Streamlit is a Python-based open-source framework that can build interactive apps such as dashboards. It allows us to integrate the outcome of our model and create visualizations. Streamlit is a very powerful tool since it can build anything that we want from interactive applications such as dashboards to even a full-on game that is user friendly. The outcome of the model mentioned in the previous chapter will be used and integrated into this dashboard completely using python codes.

5.1 Descriptive Dashboard Development

On this part, I will be explaining the python codes that were used to build this dashboard, from its layout, design and colors including all the charts chosen for visualizations.

```
import streamlit as st
import pandas as pd
import plotly.express as px
```

These python codes imports libraries that will be used to develop the dashboard. The first library is Streamlit. This is the backbone of the entire dashboard since it suitable for machine learning development. Pandas is another library critical for this project as it allows for data analysis and manipulation. Lastly, to visualize our data, we use plotly to create various interactive charts.

```
# Load dataset
df = pd.read_csv("Jan-Dec Dataset.csv")
```

```
df.columns = df.columns.str.strip().str.lower().str.replace(‘ ‘, ‘_’)

# Clean ‘Amount_lost’ column to numeric
df[‘amount_lost’] = (
    df[‘amount_lost’]
    .astype(str)
    .str.replace(r’[$,]’, ‘’, regex=True)
    .astype(float)
)
```

Firstly, we load the dataset into the DataFrame and convert every column to lower case for later processes. Then for column “Amount loss”, we remove the symbol “$” because it is not a numerical character since this is important to create the graphs later on.

```
# Streamlit setup
st.set_page_config(page_title=”Azif’s FYP2”, layout=”wide”)
st.title(“Australia Scam Cases Dashboard (Jan-July 2024)”)
```

Next, we set up the main page, the title, and its layout ready to create the dashboard. The image below shows the overall layout of the dashboard.

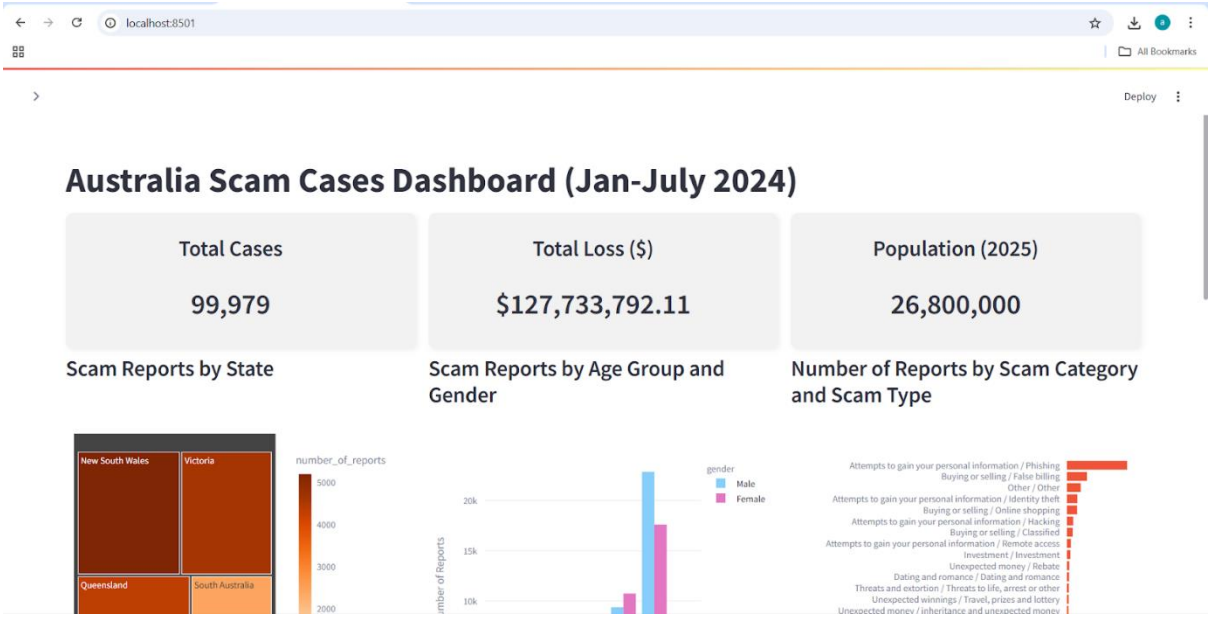


Figure 21: Dashboard’s main page and layout view

```
total_cases = df[‘number_of_reports’].sum()
total_loss = df[‘amount_lost’].sum()
64ustralia_population_2025 = 26800000

kpi1, kpi2, kpi3 = st.columns(3)

with kpi1:
   st.markdown(f””” 
      <div style=”background-color:#f2f2f2; padding:20px; border-radius:10px; box-shadow: 2px 2px 6px rgba(0,0,0,0.05); text-align:center;”>
         <h3> Total Cases</h3>
         <h2>{total_cases:,.0f}</h2>
      </div>
   “””, unsafe_allow_html=True)

with kpi2:
   st.markdown(f””” 
      <div style=”background-color:#f2f2f2; padding:20px; border-radius:10px; box-shadow: 2px 2px 6px rgba(0,0,0,0.05); text-align:center;”>
         <h3> Total Loss ($)</h3>
         <h2>${total_loss:,.2f}</h2>
      </div>
   “””, unsafe_allow_html=True)

with kpi3:
   st.markdown(f””” 
      <div style=”background-color:#f2f2f2; padding:20px; border-radius:10px; box-shadow: 2px 2px 6px rgba(0,0,0,0.05); text-align:center;”>
         <h3> Population (2025)</h3>
         <h2>{64ustralia_population_2025:,}</h2>
      </div>
   “””, unsafe_allow_html=True)
```

These three boxes are called KPI’s. They are used to get a quick glimpse of the overall information of the data. In this case, the total number of cases in Australia from January to July

in the year 2024. In addition to that, total amount lost from all types of scams that occurred during the period as well as Australia's current population.

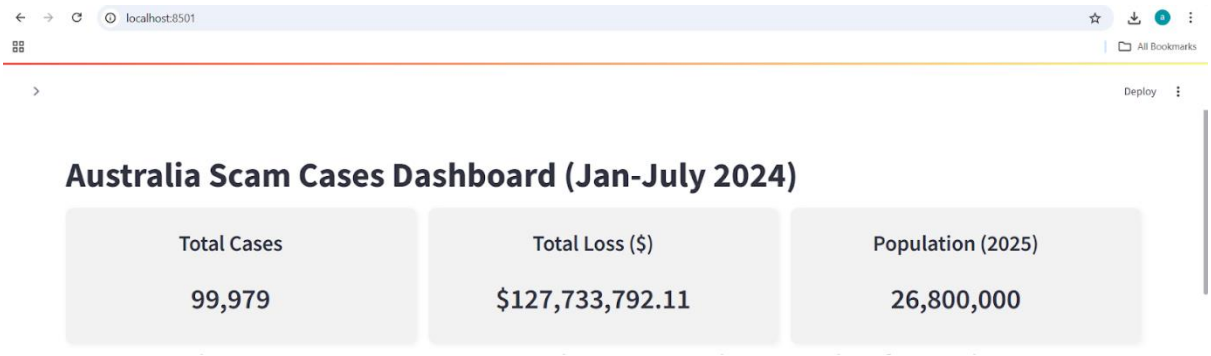


Figure 22: 3 KPIs for main dashboard

```
# Sidebar filters
st.sidebar.header("Filters")

# Scam Category filter (for VIZ 3)
unique_categories = sorted(df['scam_category'].dropna().unique())
select_all = st.sidebar.checkbox("Select All Scam Categories", value=True)
if select_all:
    selected_categories = st.sidebar.multiselect("Filter by Scam Category:", unique_categories, default=unique_categories)
else:
    selected_categories = st.sidebar.multiselect("Filter by Scam Category:", unique_categories)

# Age Group filter (for VIZ 2 & VIZ 4)
unique_age_groups = sorted(df['age_group'].dropna().unique())
selected_age_groups = st.sidebar.multiselect("Filter by Age Group:", unique_age_groups, default=unique_age_groups)

# Global State filter (for all graph)
unique_states = sorted(df['state'].dropna().unique())
selected_states = st.sidebar.multiselect("Filter by State:", unique_states, default=unique_states)

# Filter dataframe globally based on selected states
df = df[df['state'].isin(selected_states)]
```

These filters sit on the side panel of the dashboard. It can be used to apply to either all graphs or specific ones. For example, the first filter applies to all graphs, known as the anchor filter. Once selected, all graphs will change corresponding to the selection. The second and third filter only applies to a linked graph which only changes if the filter is being used. These two filters will not affect the rest of the graph.

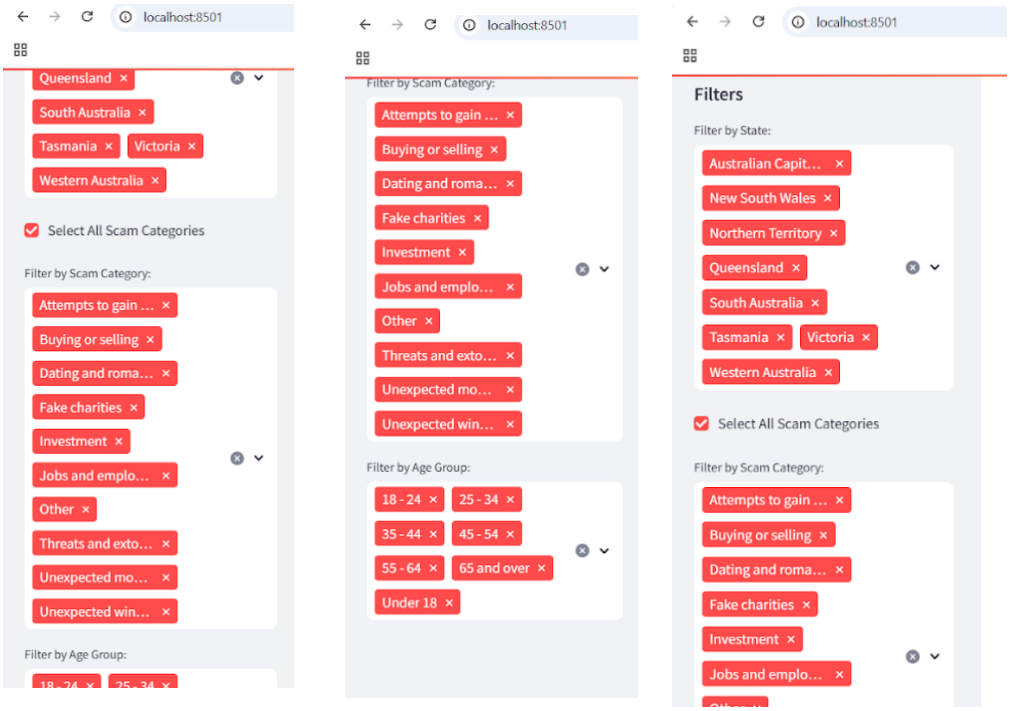


Figure 23: Filters located at sidebar of main dashboard

```
# ----------------------------------------
# VIZ 1: Treemap of Reports by State
# ----------------------------------------
report_counts = df['state'].value_counts().reset_index()
report_counts.columns = ['state', 'number_of_reports']
report_counts = report_counts.sort_values(by='number_of_reports', ascending=False)
fig1 = px.treemap(
    report_counts,
    path=['state'],
    values='number_of_reports',
    color='number_of_reports',
    color_continuous_scale='Oranges',
    title="Scam Reports by State"
)
fig1.update_layout(margin=dict(t=50, l=10, r=10, b=10))
```

The code above helps to create a treemap chart for the number of reports by each state. It specifies what attributes to be in the boxes and what attribute to be set as colors. The tone of the color resembles the number of reports. The higher the number, the darker the color, and vice versa.

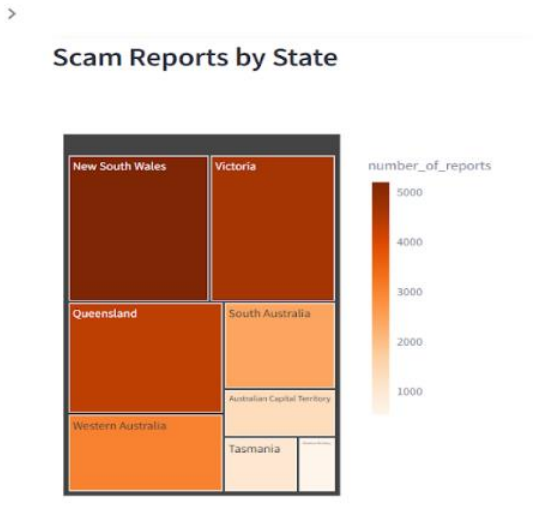


Figure 24: TreeMap diagram of Number of scam reports by each state

```
# ----------------------------------------
# VIZ 2: Bar Chart - Reports by Age Group & Gender (with age filter)
# ----------------------------------------
age_gender_counts = df[df['age_group'].isin(selected_age_groups)].groupby(['age_group', 'gender'])['number_of_reports'].sum().reset_index()

gender_order = ['Male', 'Female']

fig2 = px.bar(
    age_gender_counts,
    x='age_group',
    y='number_of_reports',
    color='gender',
    barmode='group',
    title="",
    category_orders={'gender': gender_order},
    color_discrete_map={
        'Male': '#87CEFA',   # Blue
        'Female': '#e377c2'  # Pink
    }
)
fig2.update_layout(xaxis_title="Age Group", yaxis_title="Number of Reports", margin=dict(t=50, l=10, r=10, b=10))
```

The graph shows the number of reports by Age Group and Gender. The gender is distinguished using colors, blue and pink for male and female respectively.

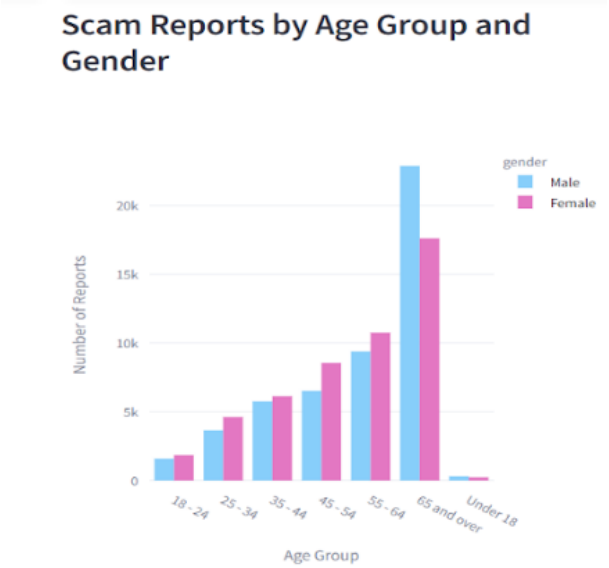


Figure 25: Grouped column chart of Number of scam reports by Age Group and Gender

```
# VIZ 3: Bar Chart – Reports by Scam Category and Scam Type
filtered_df3 = df[df['scam_category'].isin(selected_categories)].copy()
filtered_df3['scam_category_type'] = filtered_df3['scam_category'].str.strip() + ' / ' + filtered_df3['scam_type'].str.strip()
scam_counts = filtered_df3.groupby('scam_category_type')['number_of_reports'].sum().reset_index()
scam_counts = scam_counts.sort_values(by='number_of_reports', ascending=True)

fig3 = px.bar(
    scam_counts,
    x='number_of_reports',
    y='scam_category_type',
    orientation='h',
    title="",
    color_discrete_sequence=['#EF553B']
)
fig3.update_layout(
    xaxis_title="Number of Reports",
    yaxis_title="",
    showlegend=False,
    margin=dict(l=40, r=10, t=50, b=10)
)
```

The graph below shows the number of reports grouped by scam category and scam type. The python code specifies the graph title, its layout, the color and axis. The graph is linked to a filter which will change the outcome of its content depending on the user's selection.

Figure 26: Bar chart showing number of reports per scam category and type

```
# Layout with columns: VIZ 1, 2, 3 side by side
col1, col2, col3 = st.columns([1,1,1])

with col1:
    st.subheader("Scam Reports by State")
    st.plotly_chart(fig1, use_container_width=True)

with col2:
    st.subheader("Scam Reports by Age Group and Gender")
    st.plotly_chart(fig2, use_container_width=True)

with col3:
    st.subheader("Number of Reports by Scam Category and Scam Type")
    st.plotly_chart(fig3, use_container_width=True)
```

The code above ensures that all three graphs, Graph 1, Graph 2 and Graph 3 are positioned in a straight line, side by side instead of top and bottom if not specified. This is to create a more professional look and is consistent throughout the dashboard. Streamlit contains an imaginary column that can place an object in that column. For example "col1" means the first column for Graph 1 followed by the rest.

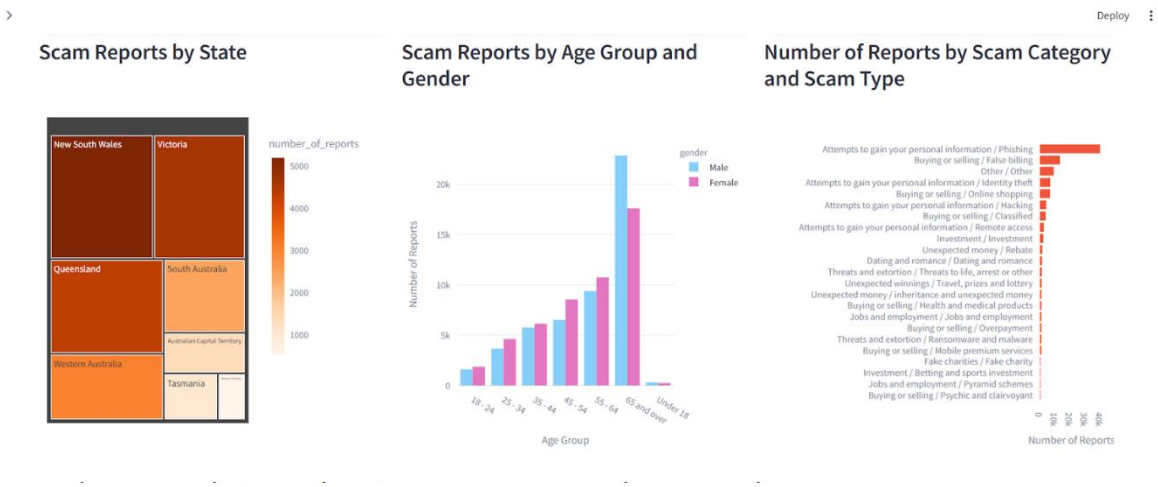


Figure 27: Graphs positioned side by side in a container

```
# ----------------------------------------
# VIZ 4 and VIZ 5
# ----------------------------------------
col4, col5 = st.columns(2)
```

The above python code is to specify the position of Graph 3 and Graph 4 in the same row as shown in the image below.

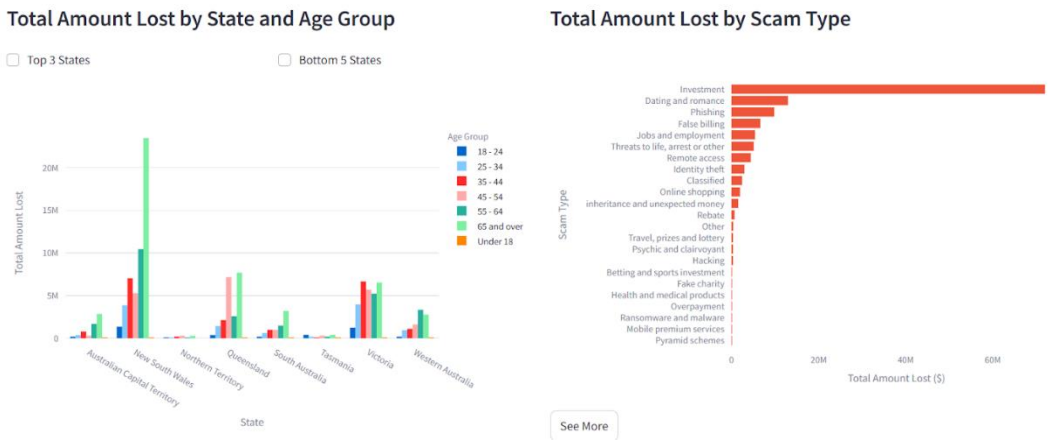


Figure 28: Graphs positioned next to each other

```
with col4:
    st.subheader("Total Amount Lost by State and Age Group")

    filtered_age_df = df[df['age_group'].isin(selected_age_groups)].copy()
    amount_lost_grouped = filtered_age_df.groupby(['state',
'age_group'])['amount_lost'].sum().reset_index()

    #checkbox_top3 = st.checkbox("Top 3 States", key="top3")
    #checkbox_next5 = st.checkbox("Bottom 5 States", key="next5")

    cb_col1, cb_col2 = st.columns(2)
    with cb_col1:
        checkbox_top3 = st.checkbox("Top 3 States", key="top3")
    with cb_col2:
        checkbox_next5 = st.checkbox("Bottom 5 States", key="next5")
```

```
    state_totals =
amount_lost_grouped.groupby('state')['amount_lost'].sum().sort_values(ascending=False)
    top_states = state_totals.head(3).index.tolist()
    next_states = state_totals.iloc[3:8].index.tolist()

    if checkbox_top3:
        amount_lost_grouped =
amount_lost_grouped[amount_lost_grouped['state'].isin(top_states)]
    elif checkbox_next5:
        amount_lost_grouped =
amount_lost_grouped[amount_lost_grouped['state'].isin(next_states)]

    fig4 = px.bar(
        amount_lost_grouped,
        x='state',
        y='amount_lost',
        color='age_group',
        barmode='group',
        title="",
        labels={'amount_lost': 'Total Amount Lost', 'state': 'State', 'age_group': 'Age Group'}
```

The graph below contains two checkboxes which acts as filter specific to the graph. The filters are to show the top 3 states with the highest amount lost and the remaining 5 states. The purpose of these filters is to provide a deeper analysis for all states but separately due to the lack of size and space.

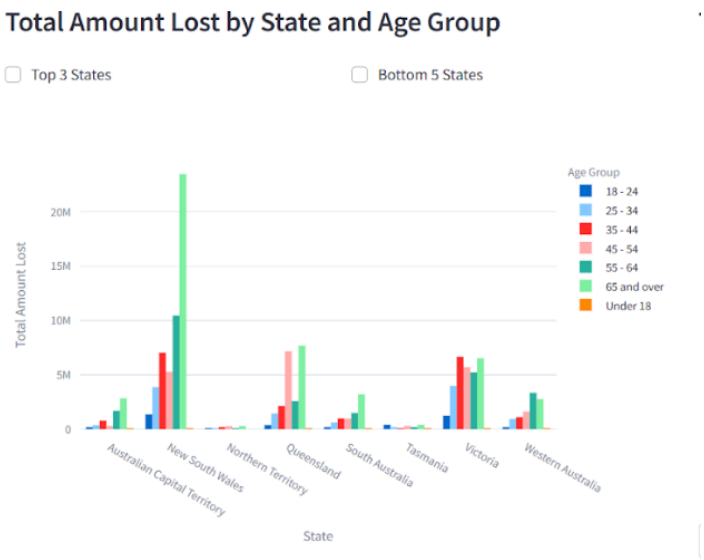


Figure 29: Group column chart for Amount lost by State and Age Group

```
with col5:
    st.subheader("Total Amount Lost by Scam Type")

    amount_lost_by_scam = df.groupby('scam_type')['amount_lost'].sum().reset_index()
    amount_lost_by_scam = amount_lost_by_scam.sort_values(by='amount_lost',
ascending=True)

    fig5 = px.bar(
        amount_lost_by_scam,
        x='amount_lost',
        y='scam_type',
        orientation='h',
        title="",
        labels={'amount_lost': 'Total Amount Lost ($)', 'scam_type': 'Scam Type'},
        color_discrete_sequence=['#EF553B']
    )
    fig5.update_layout(margin=dict(l=200, t=50, r=25, b=25),
yaxis=dict(tickmode='linear'))
```

```
st.plotly_chart(fig5, use_container_width=True)
```

The code above creates a bar chart that shows the total amount lost for each scam type. It contains two breakdowns of the chart for amounts less than 1 million and larger than 1 million. This breakdown is placed under the "See More" button for users to click if they intend to. Since the highest value is extremely big, it makes the rest of the graph look insignificant. This creates a bias in users interpretation.

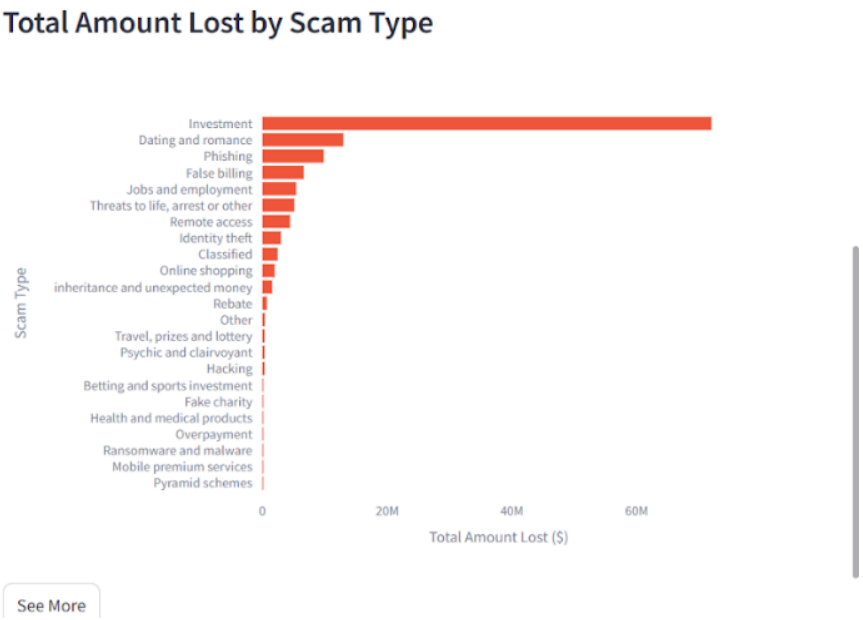


Figure 30: Bar chart showing Amount Lost for each scam type

```
# "See More" section for detailed breakdown
    if "show_drilldown" not in st.session_state:
        st.session_state.show_drilldown = False

    if st.button("See More", key="seemore_button"):
        st.session_state.show_drilldown = not st.session_state.show_drilldown

    if st.session_state.show_drilldown:
        st.subheader("Detail Breakdown of Scam Types")
        tab_high, tab_low = st.tabs(["Losses ≥ $1M", "Losses < $1M"])
```

```
    with tab_high:
        df_above_1m = amount_lost_by_scam[amount_lost_by_scam['amount_lost'] >=
1_000_000]
        fig_high = px.bar(
            df_above_1m,
            x='amount_lost',
            y='scam_type',
            orientation='h',
            title="Scam Types with Losses ≥ $1M",
            labels={'amount_lost': 'Total Amount Lost ($)', 'scam_type': 'Scam Type'},
            color_discrete_sequence=['#FFD700']
        )
        fig_high.update_layout(margin=dict(l=200, t=50, r=25, b=25))
        st.plotly_chart(fig_high, use_container_width=True)

    with tab_low:
        df_below_1m = amount_lost_by_scam[amount_lost_by_scam['amount_lost'] <
1_000_000]
        fig_low = px.bar(
            df_below_1m,
            x='amount_lost',
            y='scam_type',
            orientation='h',
            title="Scam Types with Losses < $1M",
            labels={'amount_lost': 'Total Amount Lost ($)', 'scam_type': 'Scam Type'},
            color_discrete_sequence=['#FFDAB9']
        )
        fig_low.update_layout(margin=dict(l=200, t=50, r=25, b=25))
        st.plotly_chart(fig_low, use_container_width=True)
```

The code above creates a button which shows two tabs of the previous graph. This separates between amounts lost less than 1 million and amounts lost higher than 1 million. Users can switch between the two tabs to compare all scam types and their respective amount lost.

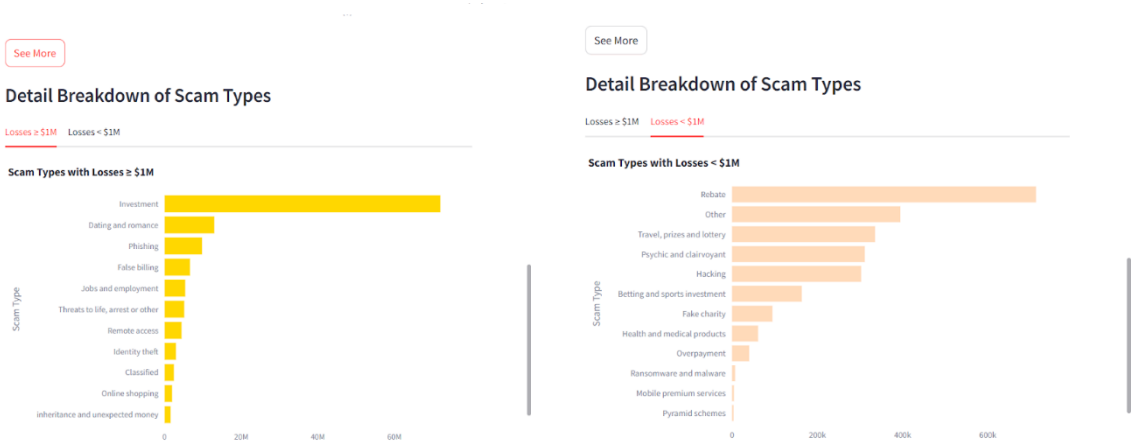


Figure 31: Two breakdown tabs for different amount lost

5.1.1 Predictive Dashboard Development

```
import streamlit as st
import pandas as pd
import plotly.express as px
```

The following Python code imports three essential libraries used for building an interactive dashboard. It imports Streamlit library, Pandas library for Pandas data manipulation and analysis and plotly for interactive visualization charts.

```
# ---------------------------
# SETUP
# ---------------------------
st.set_page_config(page_title="Prediction Dashboard", layout="wide")
st.title("Australia Scam Prediction Dashboard using Random Forest")

# Load data
df = pd.read_csv("Updated_RF.csv")
df.columns = df.columns.str.strip().str.lower().str.replace(' ', '_')
```

The code sets up the page layout by setting the page title as well as the tab title on the browser. It then loads the predicted dataset to create useful and informative graphs to show the predictions of the model. The image below shows the general layout of the dashboard.

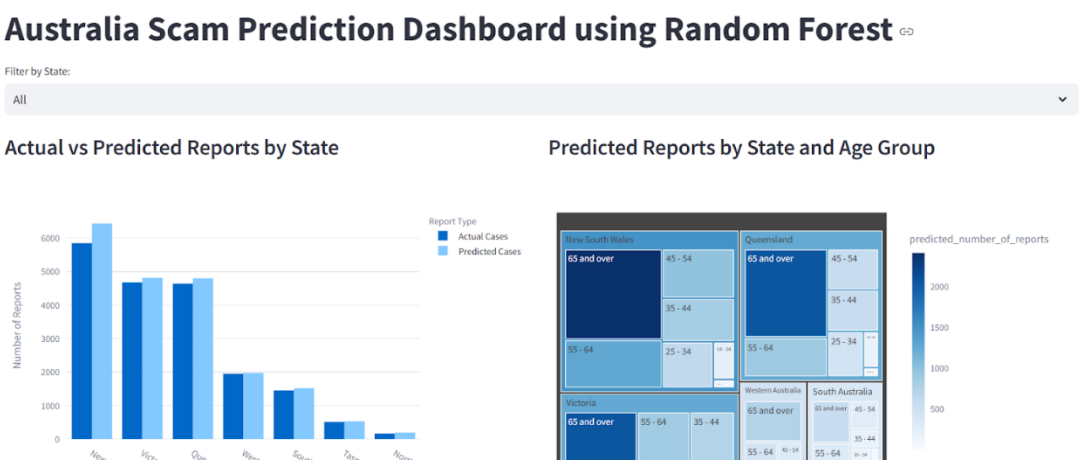


Figure 32: Predictive dashboard's page and layout view

```
# ---------------------------
# FILTER: Select State (Anchor)
# ---------------------------
states = sorted(df['state'].dropna().unique())
selected_state = st.selectbox(" 📍 Filter by State:", options=["All"] + states)

if selected_state != "All":
    filtered_df = df[df['state'] == selected_state]
else:
    filtered_df = df.copy()
```

The following filter provides options to users to select whichever state they want to view. This filter is applied to all graphs in the predictive dashboard. This is called an anchor filter. When selected, every graph will move according to what is chosen.

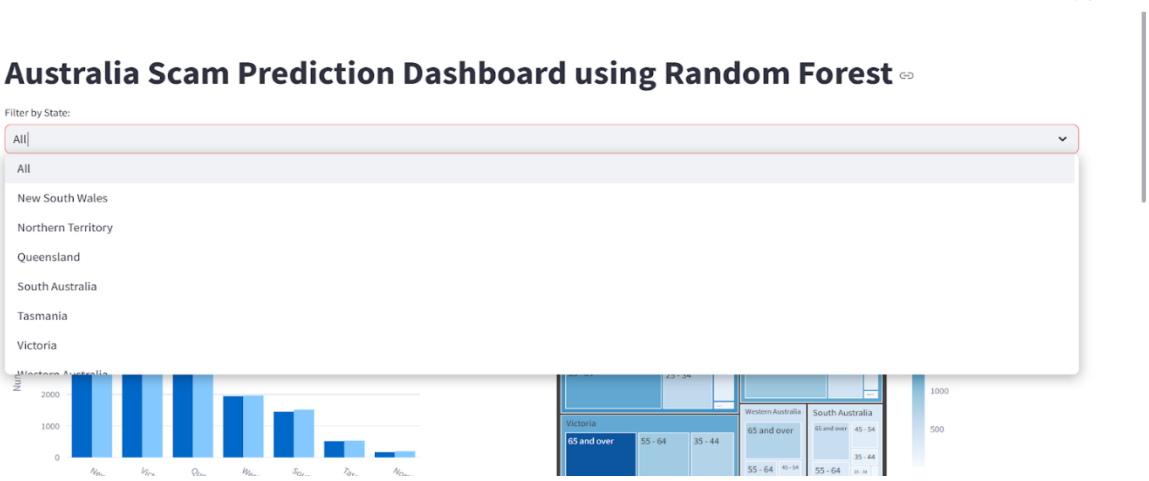


Figure 33: State filter dropdown menu

```
# Create two columns for each row
col1, col2 = st.columns(2)
```

The codes function is to position graph 1 and graph 2 side by side compared to top and bottom if it was not specified. The layout is consistent with the overall dashboard to achieve uniformed design.

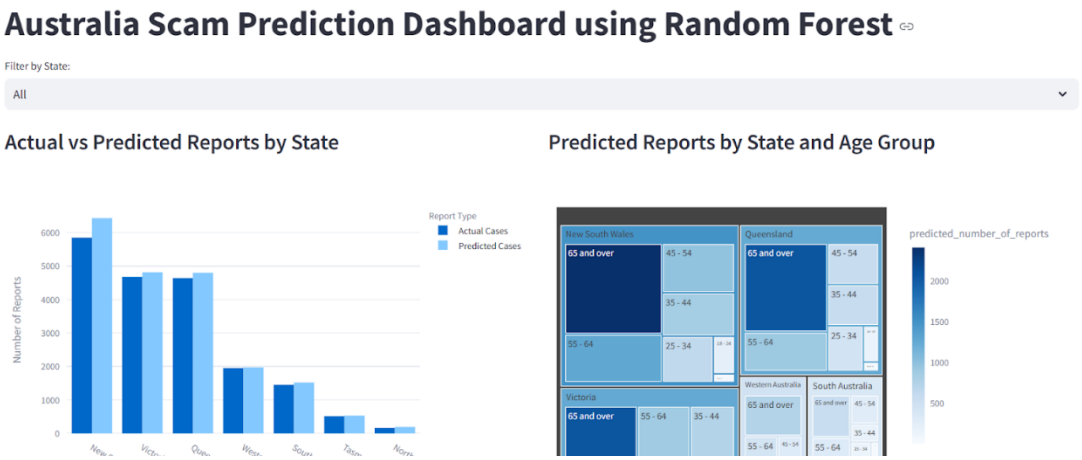


Figure 34: Graphs positioned side by side in a container

```
# ---------------------------
# VIZ 1 and VIZ 2
# ---------------------------
with col1:
    st.subheader("Actual vs Predicted Reports by State")

    viz1_data = df.groupby('state')[['actual_number_of_reports',
'predicted_number_of_reports']].sum().reset_index()
    viz1_data = viz1_data.sort_values(by='actual_number_of_reports', ascending=False)

    viz1_data = viz1_data.rename(columns={
        'actual_number_of_reports': 'Actual Cases',
        'predicted_number_of_reports': 'Predicted Cases'
    })

    fig1 = px.bar(
        viz1_data,
        x='state',
        y=['Actual Cases', 'Predicted Cases'],
        barmode='group',
        title="",
        labels={'value': 'Number of Reports', 'state': 'State', 'variable': 'Report Type'}
    )
    fig1.update_layout(margin=dict(t=50, l=25, r=25, b=25))
    st.plotly_chart(fig1, use_container_width=True)
```

Graphs above is the output of the written code that shows the comparison between the actual number of reports and predicted number of reports across all states. The difference between the two is represented by a shade of blue where actual values is of darker blue whereas predicted values is lighter blue.

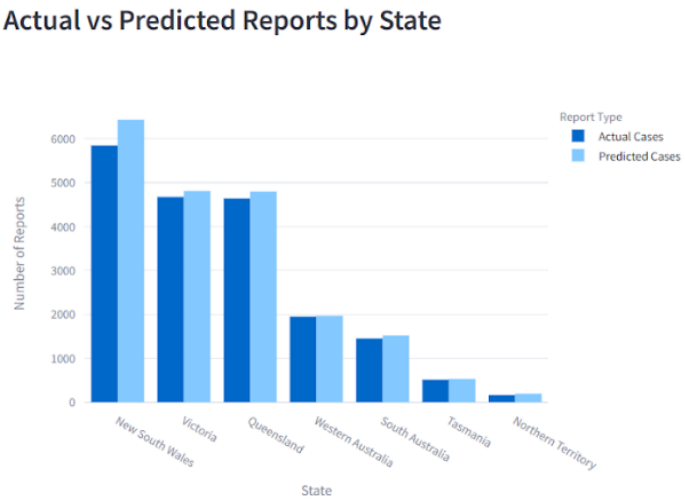


Figure 35: Comparison between actual vs predicted values in column chart

```
with col2:
    st.subheader("Predicted Reports by State and Age Group")

    viz2_data = filtered_df.groupby(['state', 'age_group'])['predicted_number_of_reports'].sum().reset_index()
    viz2_data = viz2_data.sort_values(by='predicted_number_of_reports', ascending=False)

    fig2 = px.treemap(
        viz2_data,
        path=['state', 'age_group'],
        values='predicted_number_of_reports',
        color='predicted_number_of_reports',
        color_continuous_scale='Blues',
        title=""
    )
    fig2.update_layout(margin=dict(t=50, l=10, r=10, b=10))
    st.plotly_chart(fig2, use_container_width=True)
```

The code above is to create a treemap chart for number of predicted reports grouped by age group and state. The darker the color indicates a high value of reported cases while lighter color means lesser number. States appear bigger contains the most saturated cases reported while smaller sized states reflects lower reported cases.

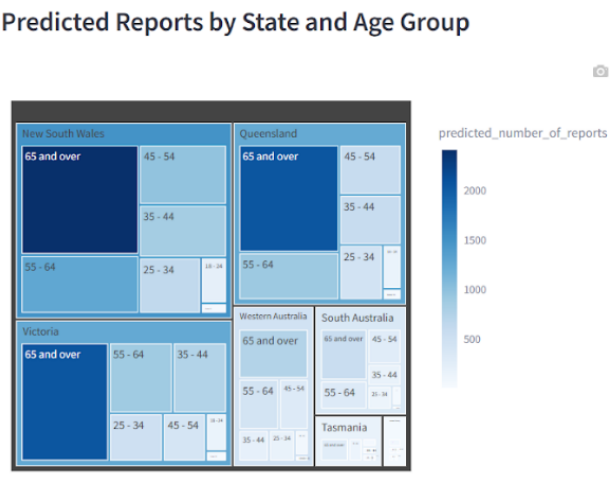


Figure 36: TreeMap diagram for number of reports by State and Age Group

```
# ---------------------------
# VIZ 3 and VIZ 4 in second row
# ---------------------------
col3, col4 = st.columns(2)
```

The code above is to position graph 3 and graph 4 side by side instead of top and bottom. This is consistent throughout the dashboard to provide an organized and professional look.

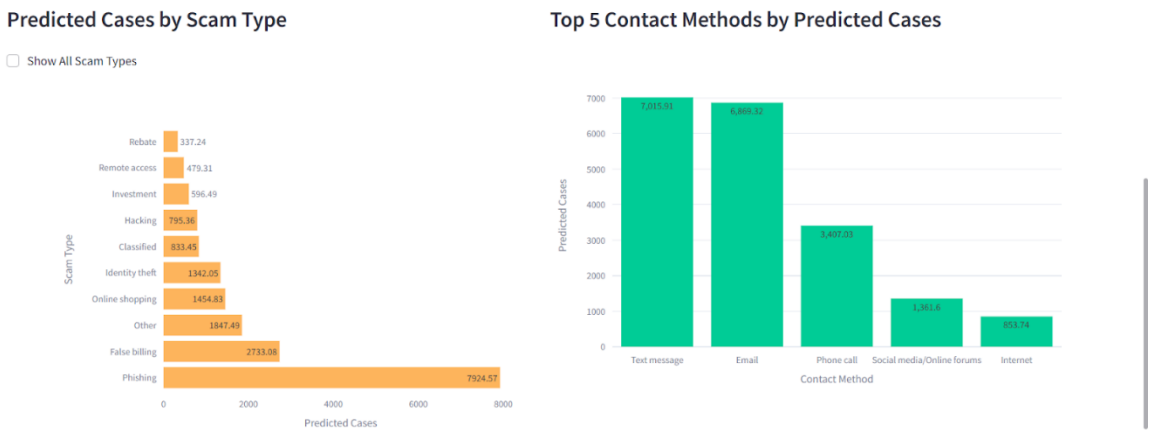


Figure 37: Graphs positioned next to each other

```
with col3:
    st.subheader("Predicted Cases by Scam Type")

    viz3_data                                                                    =
filtered_df.groupby('scam_type')['predicted_number_of_reports'].sum().reset_index()
    viz3_data = viz3_data.sort_values(by='predicted_number_of_reports', ascending=False)

    show_all = st.checkbox("Show All Scam Types", value=False, key="show_all_scam")

    if not show_all:
        viz3_data = viz3_data.head(10)
        show_labels = True
    else:
        show_labels = False  # Don't show labels when showing all

    fig3 = px.bar(
        viz3_data,
        x='predicted_number_of_reports',
        y='scam_type',
        orientation='h',
        text='predicted_number_of_reports' if show_labels else None,
        title="",
        labels={'predicted_number_of_reports': 'Predicted Cases', 'scam_type': 'Scam Type'},
        color_discrete_sequence=['#FDB45C']
    )
    fig3.update_layout(margin=dict(l=200, t=50, r=25, b=25), yaxis=dict(tickmode='linear'))
    st.plotly_chart(fig3, use_container_width=True)
```

The graph by default will show only top 10 scam types if users do not click the check box “Show All Scam Types”. This is to increase readability and focus only the top values. The x axis is the number of predicted cases while the y axis is the type of scams. The graph only shows label when 10 values are shown but disappear when the check box is ticked.

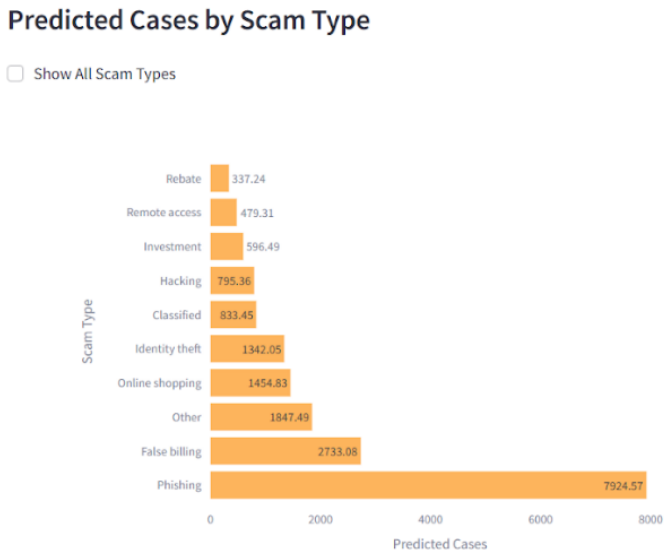


Figure 38: Bar chart showing top 10 number of predicted cases by scam types

```
with col4:
    st.subheader("Top 5 Contact Methods by Predicted Cases")

    viz4_data =
filtered_df.groupby('contact_method')['predicted_number_of_reports'].sum().reset_index()
    viz4_data = viz4_data.sort_values(by='predicted_number_of_reports',
ascending=False).head(5)

    fig4 = px.bar(
        viz4_data,
        x='contact_method',
        y='predicted_number_of_reports',
        text_auto=True,
        title="",
```

```
        labels={'predicted_number_of_reports': 'Predicted Cases', 'contact_method': 'Contact Method'},
        color_discrete_sequence=['#00CC96']
    )
    fig4.update_layout(margin=dict(t=50, l=25, r=25, b=25))
    st.plotly_chart(fig4, use_container_width=True)
```

Finally, the code for the last graph. We specify the graph title, what is the x and y value for axis, sort them in ascending order, put labels on all columns and specify the color and the layout.

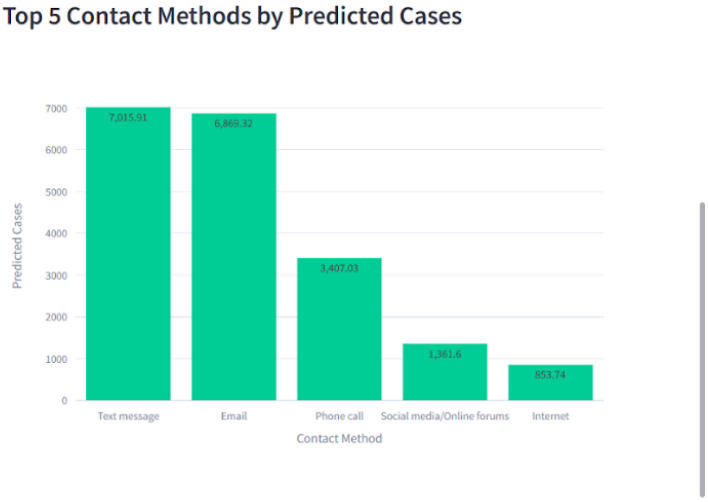


Figure 39: Column chart of top 5 highest predicted cases by contact method

5.2 User Testing

System	Australia Scam Cases Dashboard			
Purpose	To test filter functionalities and confirm it operates as intended			
Page: Main Page				
Feature	Description	Expected Result	Actual Result	Comment
1	Side panel buttons • Main Dashboard (Descriptive) • Predictive Dashboard • Model Comparison	Switch between different pages – Main Dashboard, Predictive Dashboard, Model Comparison	Switches between different pages – Main Dashboard, Predictive Dashboard, Model Comparison	
2	Side panel filter • Anchor filter • Scam category filter • Age group filter	To view different outcomes of the graph, make comparison	View different outcomes of the graph, makes comparison	
3	Total amount lost by state & age group graph checkbox • Top 3 • Bottom 5	To view and switch between top 3 states and bottom 5 states	Views and switches between top 3 states and bottom 5 states	
4	Total amount lost by scam type “See more” button • Graph breakdown • 2 tabs (More and Less than 1M)	View graph breakdown in two tabs. Provide in-depth details of graph	Views graph breakdown in two tabs. Provides in-depth details of graph	

Page: Predictive				
Feature	Description	Expected Result	Actual Result	Comment
5	Side panel filter • Contact method filter	Select or deselect values to make changes on Top 5 contact methods graph	Selects or deselects values that changes Top 5 contact methods graph	
6	Anchor filter • State Show all scam type Checkbox • Predicted cases by scam type graph	Make changes across all graphs Show all scam types in one view	Changes view across all graphs Shows all scam types in one view	
Page: Model Comparison				
Feature	Description	Expected Result	Actual Result	Comment
7	Select your own graph • Choose between descriptive & predictive • Select 2 graphs	Display 2 graphs selected by users	Display 2 graphs selected by users	
Test By				
Date				
Pass/Fail				
Comment/Feedback				
Acceptance of Testing				
Verified by		Accepted by		

Name	Name
Date	Date

CHAPTER 6

CONCLUSION

6.0 Overview

Chapter 8 summarizes the end of Project 2 phases, model development and building dashboard. It also provides thorough conclusions for each chapter from chapter 1 to chapter 7. It also discusses the challenges and limitations of the project.

6.1 Summary of all chapters

Chapter 1 introduces the overall idea of the project. It focuses on the background of scams as a serious global cybercrime issue. The problem statement highlights the ever-increasing advancements of scammers and the limitation of awareness being done currently. The project's objective are 123. The chapter also discusses how the project aligns with how it can be useful for decision making and spreading awareness.

Chapter 2 delivers a complete review of the cybercrime situation which includes histories, types, impacts and modern tactics of scams. It explores factors that influences scam vulnerability and make comparison between machine learning approaches used in several studies. Ultimately, Random Forest was decided as the ideal model to be used for the project.

Chapter 3 details data acquisition method and steps taken in preprocessing based on the CRISP-DM methodology. The dataset was taken from an Australian portal named ScamWatch that includes various data types from temporal, numerical and loads of categorical. By using Microsoft Excel, I was able to perform data cleaning and transformation techniques such as removal of duplicates, data formatting and text replacement. The chapter finish off with the descriptive dashboard that visualizes 5 key insights like number of reports by scam type and state that will be used later on for predictive analysis.

Chapter 4 focuses on the development of predictive model using Random Forest algorirthm. Step by step process are discussed from data preparation, data conversion, feature encoding, to training and testing the model. Then the data must be converted back into readable format to ease data interpretation. While chapter 5 talks about the process of developing descriptive and predictive dashboard using Streamlit. The chapter delivers the final outcome of project 2.

6.2 Project outcome

The outcome of this project are the combined results from both Project 1 and Project 2. Project 1 is more towards the introduction or starting point of the proposed topic. This is where problem statements are addressed and objectives are set. Essentially the project aims to shed some light to the growing cybercrime issue, where scams, in particular, are becoming more and more serious not just in Malaysia but anywhere else. The lack of attention and preventive measures poses a threat to our nation as scam victims are losing their hard-earned money to scammers. Aside from that, several related studies were studied to identify their approach and model used.

As for Project 2, we explored the use of Python using Jupyter Notebook for data manipulation and model development and discover Streamlit to be useful to create interactive dashboard that can integrate the model's output. Ultimately, in the end, I managed to accomplish all objectives outlined in Chapter 1, which are to explore relationship between variables, predict the number of scam cases in Australia, and to build interactive dashboards that visualizes scam patterns based on features. In addition to that, every project requirements were met from researching literature reviews to performing data transformation and preparation, and finally model and dashboard development.

6.3 Problems encountered

During Project 1, one specific challenge I faced was about data cleaning. Normally in class, I deal with small, simple and well formatted dataset. This time around, I had to manage a large dataset with thousands of rows with predefined format that is foreign to me. New techniques were discovered to deal with these kinds of data and that was a bit difficult for me. However, with a quick search on "How to" in Youtube, I quickly learn them and applied the method straight away.

Challenges in Project 2 was even more tricky as I have never built or learn how to build a machine learning model. Countless terms and functions, libraries and packages that was never heard forced me to adapt to the new environment that I am currently in. A couple of dead ends that I had found myself in, such as what tool to use, where do I begin, where do I learn these new skills. Fortunately, with the help and advice from my supervisor, Dr Zaihisma and other analytics lecturers, I was able to find a solution to these challenges.

6.4 Limitations

One limitation of this project is that the analysis is based on a dataset sourced from Australia. Therefore, the insights might be biased or a touch different to what we experience in Malaysia. For example, a scam type in the dataset may not exists in our country due to technological limitations or totally irrelevant. Additionally, the data is based on the year 2024 from January to July. The trends may shift upward or downward over time making the insights slightly outdated.

6.5 Future Improvements

Granted, this project is not completely perfect and is not up to industry standards. In the future, I hope to incorporate the ability to upload other datasets into the model that can make other predictions that it currently offers. This allows outside parties to use the model for their own personal and business use. Another improvement that I wish to add is to gain access to Malaysia's own scam dataset to get a clearer picture of its own scam landscape.

6.6 Chapter Summary

To conclude, this chapter provide a complete summary of all the chapters in this project that offers a clear understanding of its objectives, progress and achievements. It also outlines the key problems faced in the entirety of the project and its limitations.

REFERENCES

Ben McKimm. (2024, 20 Sept). Average Salary in Australia by Age, State, and Industry Revealed https://manofmany.com/lifestyle/advice/average-salary-australia

Zurairah Jais, Salbiah Nur Shahrul Azmi, Noor Fariza Mohd Hasini, Shahrul Niza Samsudin, Nor Izzuani Izhar, Nor Ainee Idris, Sarjan Azwan Amirulsyafiq Abu Hassan. (2024, 25 June). ONLINE SCAM: A SYSTEMATIC LITERATURE REVIEW https://unimel.edu.my/journal/index.php/JULWAN/article/viewFile/1689/1348

Mohamad Rizal Abd Rahman. (2020). Online Scammers and Their Mules in Malaysia https://journalarticle.ukm.my/16141/1/39507-135466-1-PB.pdf

Bernama. (2024, July 17) High Cases of Scam Victims, Low Resilience in Malaysia - National Scam Awareness Survey 2024. Retrieved from https://www.bernama.com/en/news.php/news.php?id=2318397#:~:text=PETALING%20JAYA%2C%20July%2016%20

Ipsos. (2023). IPSOS PRESS RELEASE : SCAM IN MALAYSIA https://www.ipsos.com/sites/default/files/ct/news/documents/2023-12/Ipsos%20Press%20Release%20-%20Scam%20in%20Malaysia_0.pdf

Luqman Amin. (2024, May 17). Nearly one-third of Malaysians prone to investment scams, study shows.Retrieved from https://theedgemalaysia.com/node/712072

Ipsos. (2023, December 18). [PRESS RELEASE] - Scam in Malaysia. Retrieved from https://www.ipsos.com/en-my/press-release-scam-malaysia

Assoc Prof Dr Chong Wei Ying (2024, March 13). SCAMS: A PREVENTABLE PHENOMENON STILL PREVALENT. Retrieved from https://www.bernama.com/en/thoughts/news.php?id=2276953

Maleen Balqish Salleh. (2022, August 5). PDRM: Over RM5.2 billion lost to scams in two years. Retrieved from

https://theedgemalaysia.com/article/pdrm-over-rm52-billion-lost-scams-two-years

Vietnam Investment Review. (2024, August 14). Malaysia loses 700 million USD due to online scams.Retrieved from https://vir.com.vn/malaysia-loses-700-million-usd-due-to-online-scams-113572.html#:~:text=Malaysia%20lost%20a%20total%20of,Digital%20Minister%20Gobind%20Singh%20Deo

Fraud.com.(n.d.). The History and Evolution of Fraud. Retrieved from https://www.fraud.com/post/the-history-and-evolution-of-fraud

Comply Advantage. (2024, April 3). Top 5 fraud trends in 2024 and how to mitigate them. Retrieved from https://complyadvantage.com/insights/top-fraud-trends/#:~:text=1.-,Synthetic%20identity%20fraud%20remains%20the%20most%20common%20form%20of%20identity,phishing%2C%20smishing%2C%20and%20vishing

HO YEW KEE. (n.d.) The Psychology of Scams https://sid.org.sg/common/Uploaded%20files/Bulletin/forNewsletter/May_ThePsychologyofScams.pdf

Cyber Talents. (n.d.). What is Cybercrime? Types, Examples, and Prevention Retrieved from https://cybertalents.com/blog/what-is-cyber-crime-types-examples-and-prevention

Remilty, (2024, August 8) 11 Common Money Transfer Scams and How to Avoid Them. Retrieved from https://blog.remitly.com/money-transfer/money-transfer-scams/

Jonathan Kwaku Afriyie, Kassim Tawiah, Wilhemina Adoma Pels, Sandra Addai-Henne, Harriet Achiaa Dwamena, Emmanuel Odame Owiredu, Samuel Amening Ayeh, John Eshun. (2023, March). A supervised machine learning algorithm for detecting and predicting fraud in credit card transactions. https://www.sciencedirect.com/science/article/pii/S2772662223000036#sec6

Avenga. (2022, November 28). Solving financial fraud detection with machine learning methods. Retrieved from

https://www.avenga.com/magazine/fraud-detection-machine-learning/

Ohad Shalev. (2024, November 21). The Best Machine Learning Algorithms for Fraud Detection. Retrieved from https://sqream.com/blog/fraud-detection-machine-learning/

Atikah Hanisah Mohd Hanif, Nurazean Maarop, Norshaliza, Kamaruddin, Ganthan Narayana Samy. (2024, January 12). Machine Learning Approach in Predicting Fraudulent Job Advertisement. https://hrmars.com/papers_submitted/20532/machine-learning-approach-in-predicting-fraudulent-job-advertisement.pdf

Asha RB, Suresh Kumar KR. (2021, January 23). Credit card fraud detection using artificial neural network. https://www.sciencedirect.com/science/article/pii/S2666285X21000066

Vijay Kanade. (2022, September 2). What Is a Support Vector Machine? Working, Types, and Examples. Retrieved from https://www.spiceworks.com/tech/big-data/articles/what-is-support-vector-machine/

RobiticsBiz. (2022, September 10) Pros and cons of Support Vector Machine (SVM) Retrieved from https://roboticsbiz.com/pros-and-cons-of-support-vector-machine-svm/

Hrvoje Smolic. (2024, May 13). Decision Tree. Retrieved from https://graphite-note.com/a-comprehensive-guide-to-decision-trees-everything-you-need-to-know/#elementor-toc__heading-anchor-8

Geeksforgeeks. (2024, 21 November). How to Calculate Entropy in Decision Tree? Retrieved from https://www.geeksforgeeks.org/how-to-calculate-entropy-in-decision-tree/

Vijay Kanade. (2023, April 3). What Is Linear Regression? Types, Equation, Examples, and Best Practices for 2022. Retrieved from https://www.geeksforgeeks.org/how-to-calculate-entropy-in-decision-tree/