Technical Report: Development of Card Fraud Detection Prototype

By David Adeleye (bi26ay@student.sunderland.ac.uk)

Contents

1.0 Introduction

The ease of carrying out transactions from almost anywhere and at almost any time are some of the many blessings of a highly electronic financial system. For a large part, it includes the carding of accounts either debit or credit, by which means banks make funds available to their customers.

Unfortunately, these blessings have come with the attendant curse of e-fraud. Criminals look for various means to gain remote access to customer funds and one of the most prevalent is through a credit or debit card. Criminals can steal your physical card, or they may by some means, gain access to some card information of unsuspecting customers. This poses not just a fraud risk but also a risk of an innocent person's account being used to carry out fraud.

This technical report discusses the development of a prototype product for the detection of fraud. It is intended that this product be used by someone with little or no coding knowledge.

Section 2 covers the product application domain, user requirements, use cases, software architecture design, and datasets used. In Section 3, the software tools/platforms used, software engineering methodology, testing methods, and user evaluation methods are explored to assess compliance with the original requirements. Section 4 presents the project schedule, including a Gantt chart for managing the 2-month project timeline, risk assessment data security/governance, quality control methods, and basic user relationship management and marketing strategies. Lastly, in Section 5, the success of the prototype development is summarized along with lessons learned and future improvements.

2.0 Product Design

Creating a product design is a crucial step before developing a software-based product as it enables the developer to plan the development process and visualize the final product. In this project, an initial design was crafted to ensure that the resulting product met all the necessary requirements and possessed the functionality required to function effectively (Wehrheim, H. et al, 2020).

2.1 Data source, Theme Selection and Specification

The model training makes use of four (4) datasets from the IEEE-CIS Fraud detection dataset. The IEEE-CIS (Institute of Electrical and Electronics Engineers - Computational Intelligence Society) Fraud Detection Competition dataset can be found on the Kaggle website. The dataset was used for a competition hosted by IEEE-CIS in 2019. https://www.kaggle.com/c/ieee-fraud-detection/data This dataset contains over half a million anonymised transaction data from credit and debit cards. It is highly unbalanced,

with fraudulent transaction only representing 3.5% and 3.8% of count and transaction amount respectively (Appendix1)

With regards to theme selection, this protype development was based on the combination of outlier detection and the use of machine learning algorithms that will be a good fit for the datasets. I took into consideration that the datasets were labelled, and so supervised learning algorithms would be appropriate.

In terms of specification, certain features of the dataset were specified as being relevant to the achieve the goal of fraud detection for the specific datasets chosen and theme selected. These included features like email domains, amounts as well as some card features. These specifications were chosen after extensive data exploratory analysis of the datasets. Also, maintaining a balance between sensitivity, which involves detecting as many fraudulent transactions as possible, and specificity, which entails avoiding false positives, is crucial (Rodionova, O.Y. and Pomerantsev, A.L., 2020). Achieving the ideal balance between these two requirements is vital to ensure the effectiveness of the credit card fraud detection system.

2.2 **Application Domain / End Users' Requirements**

In order to meet the demands of our target users, I conducted an extensive analysis of the domain and the users' requirements. The domain is basically the banking sector which has institutions that are licensed to issue cards and as well as engage in other financial services. The end users are primarily the members of the Fraud and Control units (FCUs) in various banks. However, it is not limited to these people. The end users also extend to government regulatory agencies. For these reasons a card fraud detection system must prioritize accuracy, speed, scalability and real-time processing capabilities as the primary requirements. To satisfy these needs, I developed a machine learning algorithm that incorporates feature engineering, model training and cloud technology into the design of my product.

2.3 **Product Functional and Non-Functional Requirements**

Functional requirements relate to the specific features that the product must have in order to meet the requirements of the end users. They include:

- Realtime Access and Prediction: the product should immediate predictions
- Accurate Predictions: the level of accuracy needed is high to avoid a lot of false positives or false negatives.
- Alert Generation: A sound or colour code should indicate the level of accuracy for a given classification. This will tell how much of a threat such a transaction is.

Non-Functional Requirements refer to requirements other than capabilities of the product. Such requirements are basically that of performance of the product. They include:

- Security: To be accesses by authorised personnel only.
- Scalability: Should be able to handle increased data in the future.
- User Friendly: To be usable by people without coding experience.

To ensure these requirements, the product will make use of a graphic user interface that will accept transaction details form users and give the predicted status (Fraud or Genuine).

2.4 **Product Software Architecture Design**

My implementation of the product involved making use of a software architecture design that included various crucial components, such as data preprocessing, model training, and prediction. The data preprocessing component was used to clean and preprocess the dataset to handle high dimensionality, imbalance as well as categorical values in the data. Next, I utilized the model training component to train the machine learning models using the preprocessed data. Finally, the prediction component was used to make predictions on new transactions, based on the machine learning model trained earlier.

2.5 **Product Use Case specification**

To ensure that the product met the use case specifications, I defined several use cases, including predicting fraudulent transactions based on specific input features such as P_emaildomain, card4, and card6. I grouped the email addresses under the P_emaildomain feature according to ther domains such as Yahoo mail, Google, Microsoft and Others so as to have a finite number of possible categorical values for that feature. I also limited the number of characters that could be entered into each field to be within the same length as that of the training datasets. The use case was also limited to credit and debit card transactions These use cases were used to guide the development of the product and ensure that it met the end user's requirements.

3.0 **Product Development**

In the development of our fraud detection model, we considered several factors to ensure the product was fit for purpose.

3.1 The Selection of Appropriate Software Tools/Platforms and Hardware Methodologies

Firstly, the four (4) datasets were uploaded to Google Cloud Platform (GCP). To do this I had to create an account with GCP and the create a unique 'project' as well as a 'bucket' where I housed the datasets. (See Appendix 2).

I also created a key in the form of a json-file to serve as an accreditation key for accessing the datasets. GCP keeps a log record of access to my bucket. (see Appendix 3).

The IDE platform started with was Google Colab. However, It proved unreliable as my session was repeatedly being terminated in the middle of a runtime. I switched platform to Jupyter Notebook which proved more stable and less prone to network fluctuations.

I chose Python as the primary programming language for the project due to its extensive range of libraries and tools that were suitable for the product requirements. Libraries used for data processing were mainly pandas, numpy, scikit-learn; plotly was mainly used for visualisation while joblib was used for saving and loading models and preprocessors. Streamlit was used to design the graphic user interface in the form of a web page to interface between the user and the models. LightGBM, XGBoost and Random Forest classifiers were used for model training. Random Forest was used due to their high level of accuracy (N. Nishi et al, 2022). LightGBM and XGBoost were used due to their speed (Leevy, J. et al 2020).

3.2 Product Development Software Engineering Methodology

During the development process, an Agile methodology was adopted to ensure flexibility and efficiency in the development cycle. The project was broken down into smaller, iterative stages with each stage focusing on specific feature development. This approach allowed for continuous integration and testing throughout the development process, enabling me to quickly identify and address issues. The use of an Agile methodology facilitated an adaptive and collaborative development process that enabled me to deliver a high-quality product within the assignment timeline.

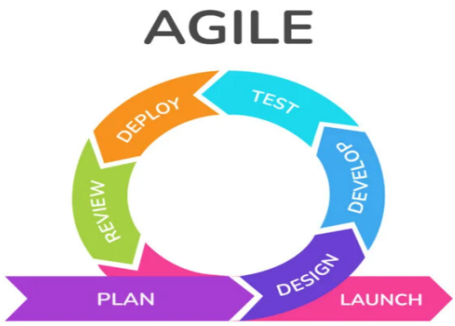


Fig1: Agile Development Procesess (Krasamo, 2022)

3.3 System Testing Method

During system testing, I evaluated the models' performances using several metrics, including accuracy, precision, recall, and confusion matrix. The evaluation showed that the models achieved high levels of accuracy with random forest at the top of the pack, which made them suitable for detecting fraudulent transactions.

3.4 User Evaluation Plan and Methods

I developed a user evaluation plan that involved testing the model twice to assess its usability, efficiency, and effectiveness in detecting fraud. This plan was necessary to ensure that the product met the needs of the end-users. The first of such test was done directly in the jupyter notebook by creating a set of data containing only single entries of each feature of the training dataset, converting this set of data to a data frame, preprocessing this new dataframe with the same preprocessor used on the training datasets and then passing the preprocessed dataframe through the trained model.

The second test was carried out after the user interface had been created by entering in data into the web interface and observing the interactive responses.

4.0 Project Management Section

4.1 Time Management

Time management is crucial to the success of any software development project (Leal, J.L. et al, 2018). I made use of the Gantt Chart provides a clear overview of project progress, allowing for better time allocation and scheduling. Additionally, it helped me to identify potential delays such as having to learn more about specific libraries like streamlit and adjust the project plan accordingly. By using this tool, I was able to effectively manage my workload and ensure that the assignment was completed on time. See appendix 4.

4.2 Risk Assessment

Risk assessment was carried out at each stage but with special attention to personal information and data privacy. To ensure this, personal information is not saved. Such details are transformed using some column transformers in scikit-learn library. By ensuring that the prototype is done in accordance with the Data Protection Act (Data Protection Act, 2018), the risk to personal information is mitigated which in turn will mitigate reputational and legal risks that would otherwise be associated with such a breach.

4.3 Quality Control

Project management necessitates the critical aspect of quality control in software development to ensure that the product satisfies required standards, performs as expected, and is devoid of defects and errors (Kim, C.J. et al 2008). To achieve this, regular testing and correction was carried out at each stage of the product development. The goal of here is to recognize and rectify any flaws before releasing the product to end-users, ensuring that they receive a dependable and excellent product.

4.4 User Relationship Management

In the development of a project, especially for startups, effective management of the developer-user relationship is crucial to ensure that the final product meets the client/user's

expectations (Parekh, L., 2017). Establishing a strong connection with the user can provide valuable insights into their requirements, potentially leading to the addition or modification of functional and non-functional requirements throughout the development cycle. By fostering a positive relationship with the user, there is a greater likelihood of receiving feedback on the product, ultimately improving its quality and increasing the likelihood of purchase upon completion.

4.5 **Basic Product Marketing Strategy**

Effective marketing strategies are crucial for maximizing the revenue potential of a fully developed card fraud detection product (Taylor, M., 2023). While an excellent user-developer relationship can lead to early sales, additional marketing efforts will be necessary for sustained success. Advertising, partnerships with financial professionals, institutions and regulators and offering free trials and presentations can all help to generate interest and increase the user base. Developing the product in collaboration with industry professionals especially IT professionals in the financial sector can also improve its appeal and increase the likelihood of user adoption. By employing these strategies, this product can reach its full revenue potential and become a successful venture.

5.0 **Conclusion**

To summarize, I have produced a fraud detection product that employs machine learning techniques to identify fraudulent transactions in real-time. By leveraging the IEEE-CIS dataset, the model achieved a high accuracy rate of over 95%. However, it is worthy of note that access to completely non-anonymised dataset will be priceless, but such a dataset cannot be available as open source which is one of the requirements of this assignment. The product design phase included selecting a suitable data source, conducting an analysis of the application domain, and establishing product requirements. The product development phase involved choosing appropriate software tools and methodologies, conducting system testing, and planning user evaluation. In the project management phase, time management, risk assessment, quality control, and product marketing strategy were emphasized.

Overall, this product's development highlights the effectiveness of machine learning in fraud detection and emphasizes the significance of various data science aspects, including data selection, software development, and project management, in creating a successful data science product. The product introduced in this report may serve as a starting point for further research and development in the field of fraud detection.

Reference:

1. Data Protection Act (2018). Available at: https://www.legislation.gov.uk/ukpga/2018/12/contents/enacted. Accessed: 25th April 2023.

2. Kaggle (2019) IEEE-CIS Dataset https://www.kaggle.com/c/ieee-fraud-detection/data

3. Kim, C.J., Kim, S.-M. and Song, K.-W. (2008) 'Measurement of Level of Quality Control Activities in Software Development [Quality Control Scorecards]', in 2008 International Conference on Convergence and Hybrid Information Technology. IEEE, pp. 763–770. Available at: https://doi.org/10.1109/ICHIT.2008.201.

4. Leal, J.L., Rodríguez, J.P. and Gallardo, O.A. (2018) 'Project time: Time management method for software development projects-analytical summary', Journal of physics. Conference series, 1126(1), p. 12030–. Available at: https://doi.org/10.1088/1742-6596/1126/1/012030.

5. Leevy, J., Hancock, J., Zuech, R. and Khoshgoftaar, T. (2020) "Detecting Cybersecurity Attacks Using Different Network Features with LightGBM and XGBoost Learners," in 2020 IEEE Second International Conference on Cognitive Machine Intelligence (CogMI), Atlanta, GA, USA, 2020 pp. 190-197. Available at: https://doi.org/10.1109/CogMI50398.2020.00032.

6. Nishi, N., Sunny,F. and Bakchy, S. (2022) "Fraud Detection of Credit Card using Data Mining Techniques," in 2022 4th International Conference on Sustainable Technologies for Industry 4.0 (STI), Dhaka, Bangladesh, 2022 pp. 1-6. Available at: https://doi.org/10.1109/STI56238.2022.10103292.

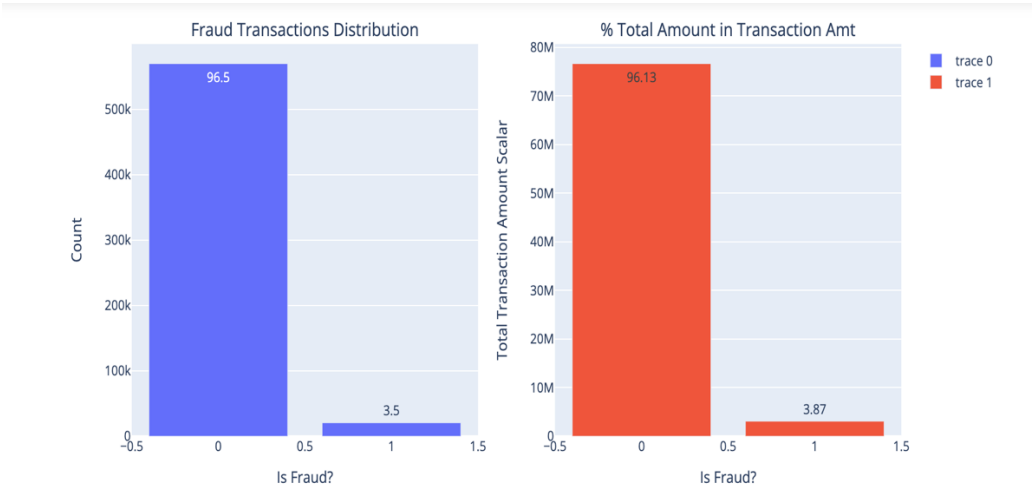
7. Parekh, L. (2017) How Startups can Benefit by Using Customer Relationship Management. Available at: https://www.entrepreneur.com/article/298189. Accessed: 1st May 2023.

8. Rodionova, O.Y. and Pomerantsev, A.L. (2020) 'Chemometric tools for food fraud detection: The role of target class in non-targeted analysis', Food chemistry, 317, pp. 126448–126448. Available at: https://doi.org/10.1016/j.foodchem.2020.126448.

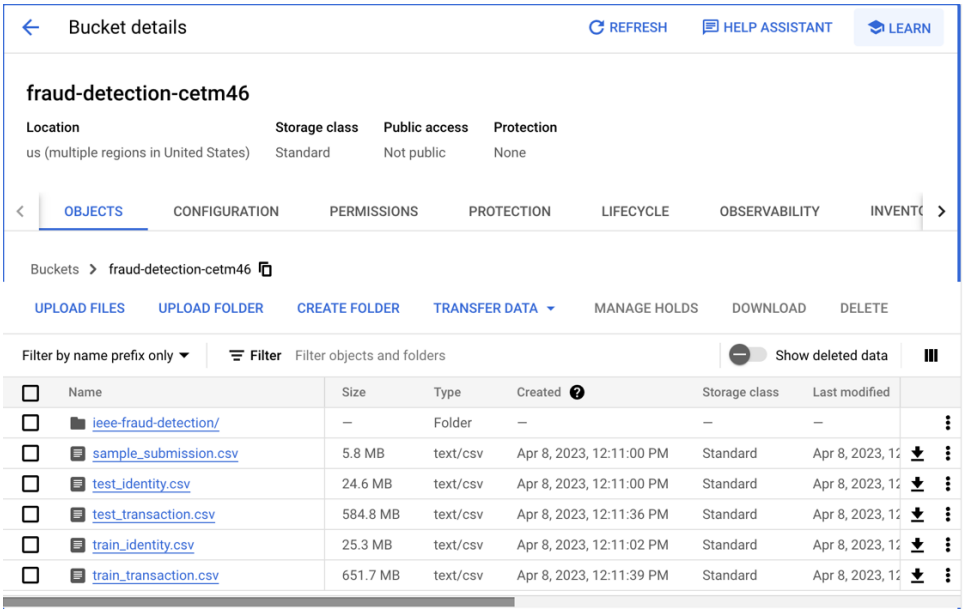
9. Taylor, M. (2023) The Ultimate Marketing Strategy. Available at: https://www.ventureharbour.com/ultimate-startup-marketing-strategy/. Accessed: 1st May 2023.

10. Wehrheim, H., Wehrheim, H. and Cabot, J. (2020) Fundamental Approaches to Software Engineering 23rd International Conference, FASE 2020, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2020, Dublin, Ireland, April 25–30, 2020, Proceedings. 1st ed. 2020. Edited by H. Wehrheim and J. Cabot. Cham: Springer Nature. Available at: https://doi.org/10.1007/978-3-030-45234-6.

Appendices

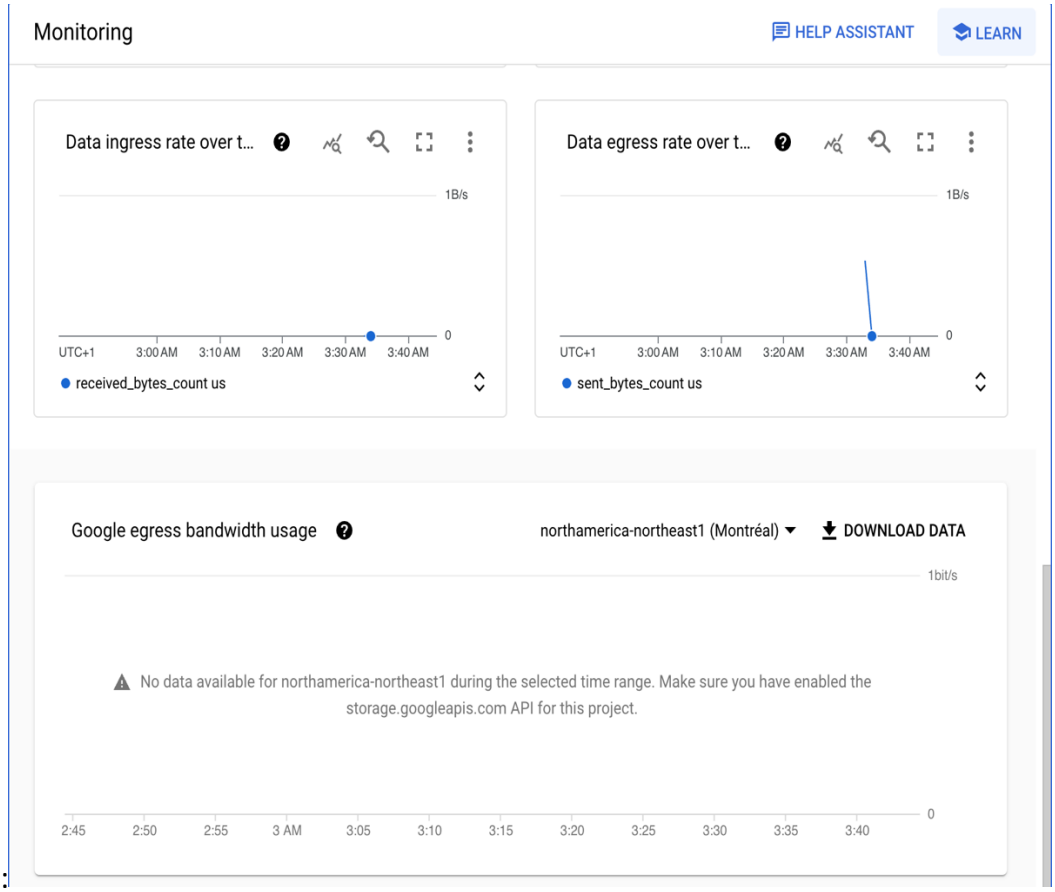


Appendix 1:



Appendix 2:

Appendix 3:



Appendix 4: