Monash University

FIT5202 - Data processing for Big Data 2024 S2

Assignment 2A: Building Models for eCommerce Fraud Detection

Due: **(23:55 20/Sep/2024 End of Week 9)**
Worth: 15% of the final marks

Background

In the dynamic world of eCommerce, **M**onash **F**ashion **C**orporation (**MFC**, an imaginary company) has established itself as a leading online retailer of fashion products, catering to a diverse and global customer base. With a wide range of products, from trendy apparel to stylish accessories, MFC has built a reputation for quality and customer satisfaction. Our platform leverages advanced technologies to provide a seamless shopping experience, ensuring customers can easily browse, select, and purchase their desired items. However, as our business has grown, so too have the challenges associated with maintaining the integrity and security of our transactions. Unfortunately, the rise of digital commerce has been accompanied by increased fraudulent activities, which pose significant risks to our financial stability and customers' trust. To address these challenges, we are committed to implementing cutting-edge solutions to detect and prevent fraudulent transactions in real-time.

Problem Statement

Recently, we have observed a troubling surge in fraudulent transactions on our eCommerce platform. These fraudulent activities not only result in direct financial losses but also undermine the trust and confidence that our customers place in us. Traditional fraud detection methods, which often rely on manual reviews and rule-based systems, have proven to be insufficient in keeping pace with the sophisticated tactics fraudsters employ. The latency in detecting fraudulent transactions means that by the time a fraudulent activity is identified, the damage has often already been done. This situation necessitates the development of a more robust and real-time fraud detection system that can analyse vast amounts of transaction data, identify suspicious patterns, and take immediate action to mitigate potential threats. The challenge lies in creating a scalable and efficient solution capable of processing high volumes of data without compromising accuracy or speed.

At this stage, we are primarily combating one type of fraud called Card-Not-Present (CNP) Fraud. Here is a simple version of the CNP fraud procedure:

1) a fraudster places and pays for an order with a stolen credit card.
2) The payment may have gone through successfully at the time, so the merchant couldn’t detect the problem and ship out the ordered item.
3) A few days later, the rightful credit card holder noticed this unrecognised transaction and called their bank.
4) The bank reverses the transaction and charges the money back from the seller. However, the products have been shipped out, causing a loss for the company.

Objective of the Project

To tackle the pressing issue of fraudulent transactions, we seek talented students from Monash University to develop an advanced fraud detection application with Apache Spark MLLib and Spark Streaming processing. The primary objective of this project is to create a real-time fraud detection system that leverages machine learning to predict and prevent fraudulent transactions as they occur. The application can handle large-scale transaction data by utilising Apache Spark's powerful data processing capabilities, performing real-time analysis to identify and flag suspicious activities. The student will be tasked with designing and implementing machine learning models that can accurately classify transactions as legitimate or fraudulent based on historical data and identified patterns. Additionally, the application will need to provide visualisation capabilities so that the company operation team can have a real-time view of the business operation. Through this project, we aim to safeguard our financial interests and reinforce our valued customers' trust and confidence.

In the first part of this project (Assignment 2-Part A), you are tasked to explore the historical dataset and build your ML model using Spark MLLib. Primarily using customer information and their browsing behaviour to detect potential fraudulent transactions.
In the second part (Assignment 2-Part B), you will use your ML model and streaming data to predict/detect fraudulent transactions and visualise them.

The Datasets:

- category.csv: Contains product category information.
- customer.csv : Contains customer information.
- product.csv : Contains product information.
- transaction.csv : Contains sales transaction records, each sale is one record/row.
- browsing_behaviour.csv: Contains customers' browsing behaviour.
- customer_session.csv: Contains the relationship between a browsing session and customer information.
- fraud_transaction.csv: A list of transactions that are identified as fraud.
- A Metadata file is shared on Google Docs (metadata.docx), which contains the information about the dataset.

What you need to achieve

Use case 1	Based on customer information and browsing behaviours, predict whether a sales transaction is likely fraudulent.	Classification
Use case 2	Perform unsupervised learning using K-Mean clustering on user behaviour, aiming to discover common characteristics of a fraudster	K-Mean

Architecture

The following figure represents the overall architecture of the assignment setup. **Part A** of the assignment consists of preparing the data, performing data exploration and extracting features, and building and persisting the machine learning models.

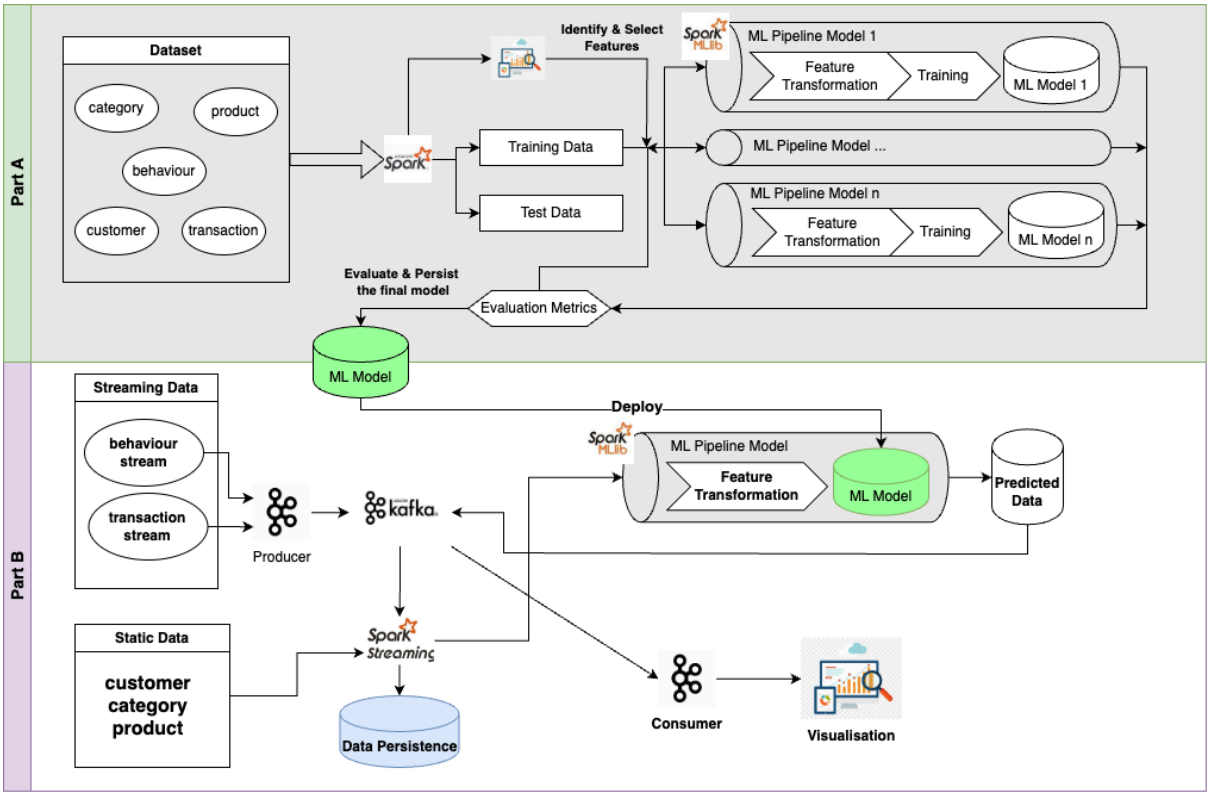

Fig 1: Overall Architecture for Assignment 2 (This assignment is Part A)

In both parts, you must implement the solutions using PySpark DataFrame/MLlib for the data pre-processing and machine learning pipelines. Excessive use of Pandas for data processing is discouraged. Please follow the steps to document the processes and write the codes in your Jupyter Notebook.

Getting Started

- Download the datasets from Moodle/Google Drive (dataset.zip).
- Download a template file for submission purposes:
    - ***A2A_template.ipynb*** file in Jupyter Notebook to write your solution. Rename it into the format (for example, ***A2A_xxxx0000.ipynb*****. xxxx0000** is your authcate ID.
- You will use Python 3+ and PySpark 3.5.0+ for this assignment (This environment is the same as we used in labs.)

IMPORTANT:

Please answer each question in your jupyter notebook file using code/markdown cell. Acknowledge any ideas or codes you referenced from others in the markdown cell or reference list.

If you use generative AI tools, all prompts you use should also be included in the reference section.

Part 1: Data Loading, Transformation and Exploration (30%)

In this section, you must load the given datasets into PySpark DataFrames and use *DataFrame functions* to process the data. Spark SQL usage is discouraged, and you can only use pandas to format results. For plotting, various visualisation packages can be used, but please ensure that you have included instructions to install the additional packages and that the installation will be successful in the provided docker container (in case your marker needs to clear the notebook and rerun it).

1.1 Data Loading (7%)

1. Write the code to create a SparkSession. For creating the SparkSession, you need to use a SparkConf object to configure the Spark app with a proper application name, to ensure the maximum partition size does not exceed 16MB, and to run locally with all CPU cores on your machine[1] (note: if you have insufficient RAM, reducing the number of cores is acceptable.) (2%)
2. Write code to define the schemas for the category, customer, product, browsing behaviour and transaction datasets, following the data types suggested in the metadata file[2]. (3%)
3. Using predefined schemas, write code to load the CSV files into separate data frames. Print the schemas of all data frames. (2%)

1.2 Data Transformation to Create Features (12%)

In the browsing behaviour dataset, there are 10 types of events:
VC(Viewing Category), VI(Viewing Item), VP(Viewing Promotion), AP(Add Promotion), CL(Click on a product/category) , ATC(Add a product to Shopping Cart), CO(CheckOut), HP(View HomePage), SCR(Mouse Scrolling), SER(Search for a product/category)

We categorise them into three different levels:

- L1(actions that are highly likely lead to a purchase): AP, ATC, CO
- L2(actions may lead to purchase): VC, VP, VI, SER
- L3(not very important - just browsing): SCR, HP, CL

[1] More information about Spark configuration can be found in https://spark.apache.org/docs/latest/configuration.html

[2] In this assignment, the “date/datetime” should be directly read as date/datetime format, instead of reading as Strings. Sample usage of schema for reading CSV file can be found in https://docs.databricks.com/data/data-sources/read-csv.html

Perform the following tasks based on the loaded data frames and create a new data frame. We will refer to this as feature_df, but feel free to use your own naming. **(2% each)**

1. For each transaction (linked to a browsing session), count the number of actions in each level and create 3 columns(L1_count, L2_count, L3_count).
2. Create two columns with a percentage ratio of L1 and L2 actions. (i.e. L1 ratio = L1/(L1+L2+L3) * 100%)
3. For each unique browsing session, based on event_time, extract the time of day as 4 groups: **morning**(6am-11:59am), **afternoon**(12pm-5:59pm), **evening**(6pm-11:59pm), **night**(12am-5:59am), add a column. (note: use medium time if a browsing session spans across different groups. For example, if a session starts at 10 am and ends at 1 pm, use 11:30 => (10+13)/2).
4. Join data frames to find customer information and add columns to feature_df: gender, age, geolocation, first join year. (note: For some columns, you need to perform transformations. For age, keep the integer only by rounding.)
5. Join data frames to find out the number of purchases the customer has made, add a column.
6. Attach the transaction labels for fraud/non-fraud.

1.3 Exploring the data (11%)

1. With the feature_df, write code to show the basic statistics: a) For each numeric column, show count, mean, stddev, min, max, 25 percentile, 50 percentile, 75 percentile; b) For each non-numeric column, display the top-5 values and the corresponding counts; c) For each boolean column, display the value and count. **(3%)**
2. Explore the dataframe and write code to present **two plots**[3] worthy of presentation to the company, describe your plots and discuss the findings from the plots. **(8%)**
    - One of the plots needs to be based on feature_df in regard to fraudulent behaviour; you're free to choose the other one.
    - Hint 1: You can use basic plots (e.g., histograms, line charts, scatter plots) to show the relationship between a column and the label or more advanced plots like correlation plots.
    - Hint 2: If your data is too large for plotting, consider using sampling before plotting.
    - 150 words max for each plot's description and discussion
    - Feel free to use any plotting libraries: matplotlib, seabon, plotly, etc.

Part 2. Feature extraction and ML training (50%)

In this section, you must use PySpark DataFrame functions and ML packages for data preparation, model building, and evaluation. Other ML packages, such as scikit-learn, would receive **zero** marks.

---

[3] This is an open question in which you would need to decide what plots to show.

- You can combine multiple features into one plot, but the plot should be clear and not contain an overwhelming amount of information.
- If you use subplots, each subplot is considered one plot, and the two-plot limit allows only two subplots for each activity data set.

2.1 Discuss the feature selection and prepare the feature columns (10%)

1. Based on the data exploration from 1.2 and considering the use case, discuss the importance of those features (For example, which features may be useless and should be removed, which feature has a significant impact on the label column, which should be transformed), which features you are planning to use? Discuss the reasons for selecting them and how you create/transform them[4]
    - 300 words max for the discussion
    - Please only use the provided data for model building
    - You can create/add additional features based on the dataset
    - Hint - Use the insights from the data exploration/domain knowledge/statistical models to consider whether to create more feature columns, whether to remove some columns

2. Write code to create/transform the columns based on your discussion above
    - **Hint: You can use one data frame for both use cases (classification and k-mean later in part 3) since you can select your desired columns as the input and output for each use case.**

2.2 Preparing Spark ML Transformers/Estimators for features, labels, and models (10%)

1. Write code to create Transformers/Estimators for transforming/assembling the columns you selected above in 2.1 and create ML model Estimators for Random Forest (RF) and Gradient-boosted tree (GBT) model.
    - **Please DO NOT fit/transform the data yet.**

2. Write code to include the above Transformers/Estimators into two pipelines.
    - **Please DO NOT fit/transform the data yet.**

2.3 Preparing the training data and testing data (5%)

1. Write code to split the data for training and testing purposes.
   Note: Due to the large dataset size, you can use random sampling (say 20% of the dataset) and do a train/test split or use one year of data for training and another year for testing.

2.4 Training and evaluating models (25%)

1. Write code to use the corresponding ML Pipelines to train the models on the training data from 2.3. And then use the trained models to predict the testing data from 2.3[5]

2. For both models (RF and GBT) and testing data, write code to display the count of TP/TN/FP/FN. Compute the AUC, accuracy, recall, and precision for the

[4] This is an open question in which you would need to decide what columns to use as features and what transformation(s) would be required for each feature. Include references when you use arguments from third parties or generative AI tools.

[5] Each model training might take from minutes to hours, depending on the complexity of the pipeline model, the amount of training data, the computing power of your laptop and the code efficiencies

above-threshold/below-threshold label from each model testing result using PySpark MLlib/ML APIs.
    - Draw a ROC plot.
    - Discuss which one is the better model (no word limit; please keep it concise)
3. Save the better model (you need it for Part B of Assignment 2).

(Note: You may need to go through a few training loops or use more data to create a better-performing model.)

Part 3. Customer Clustering and Knowledge sharing with K-Mean (10%)

In addition to building the previous models, the company would like to learn more about fraudulent behaviours. Instead of manually segregating, the company wants to explore a data-driven approach using k-mean clustering (i.e. separate genuine customers from fraudsters).

The company suspects that a fraudster may behave differently than a genuine customer (anomaly/outlier detection with k-means) in many ways:

1) A fraudster isn't genuinely interested in the product and will not spend much time browsing or searching for it.
2) A fraudster may try multiple stolen credit cards (several failed payments) before making a successful transaction, therefore they could have many failed payments.
3) A fraudster may favour high-value items/categories or low-value items/categories that are easy to resell.
4) Fraudsters may not have information about a credit card's limit; they may try to add/remove items from their shopping cart many times to find an amount that will work or place several small orders in a short period.

In this part, you need to perform the following tasks:

Task 1. Utilise the K-Mean clustering/hyperparameter tuning you learned in this unit to find the optimal K value and train the model. (note: behaviour-based grouping is not black and white, i.e. fraud/non-fraud; some behaviour may lead to a higher probability of fraud; therefore, K can be > 2.)

Task 2. Based on your model, identify the most common behaviour of fraudsters. Write a paragraph with 300 words maximum.

Note 1: This is an open question with no right or wrong answers. The evaluation will be based on the quality of your work. Feel free to include plots and/or other metrics to support your analysis/claim.

Note 2: You may ignore the domain knowledge and let the ML Algorithm do its work.

Part 4: Data Ethics, Privacy, and Security (10%)

In the era of big data, the convergence of vast quantities of information from various sources raises critical questions related to data ethics, privacy, and security. For example, in case of privacy, many companies are collecting much more data than they need from customers. In our case, we used a real-world data set with real customer information. How do you utilise those datasets with ethics, privacy and security in mind?

In this part of the assignment, you are tasked to explore these topics within the context of big data processing, drawing on contemporary research, real-world examples, and ethical considerations.
(word limit: 500 words, please include references)

(mandatory): Define the concepts of data ethics, privacy, and security within the big data domain.
(Choose one or more topics, you can also create your own topic) Explain the significance of these issues in today's data-driven world.

Data Ethics:

- Analyse how data ethics can influence big data processing;
- Examine real-world examples of how data ethics has been handled, both positively and negatively.
- Analyse the balance between technological advancements and ethical responsibilities

Data Privacy:

- Discuss the challenges and importance of maintaining privacy in big data.
- Investigate regulations and laws that govern data privacy, such as GDPR.
- Evaluate tools and techniques used to ensure privacy, and suggest improvements or new methodologies.

Data Security:

- Explore the potential security risks associated with big data processing.
- Assess the measures currently in place to secure big data, including encryption, authentication, and authorisation.

(**mandatory**)Summarise the key findings of your analysis.

Submission

You should submit your final version of the assignment solution online via Moodle.
You must submit the files created:

- Your jupyter notebook file A2A_authcate.ipynb
- **A pdf file** saved from jupyter notebook with all output following the file naming format as follows: **A1_authcate.pdf**

Note that both submitted (ipynb and pdf) files will be scanned using plagiarism detection software. The highest similarity score among students may be interviewed to prove the originality of the task.

Assignment Marking Rubric

Detailed mark allocation is available in each task. For complex tasks and explanation questions, you will receive marks based on the quality of your work.

In your submission, the jupyter notebook file should contain the **code and its output**. It should follow *programming standards, readability of the code, and organisation of code*. Please find the PEP 8 -- Style Guide for Python Code for your reference. Here is the link:

https://peps.python.org/pep-0008/ Penalty applies if your code is hard to understand with insufficient comments.

Other Information

Where to get help

You can ask questions about the assignment in the Assignments section in the Ed Forum, which is accessible on the unit's Moodle Forum page. This is the preferred venue for assignment clarification-type questions. You should check this forum regularly, as the responses of the teaching staff are "official" and can constitute amendments or additions to the assignment specification. Also, you can attend scheduled consultation sessions if the problem and the confusion are still unresolved.

Searching and learning on commercial websites/forums (e.g. Quora, Stack Overflow) is allowed. However, you should not post/ask assignment questions on those forums.

Plagiarism and collusion

Plagiarism and collusion are serious academic offences at Monash University. Students must not share their work with any other students. Students should consult the policy linked below for more information.

https://www.monash.edu/students/academic/policies/academic-integrity

See also the video linked on the Moodle page under the Assignment block.
Students involved in collusion or plagiarism will be subject to disciplinary penalties, which can include:

- The work not being assessed
- A zero grade for the unit
- Suspension from the University
- Exclusion from the University

Late submissions and Special Consideration

ALL Special Consideration, including within the semester, is now handled centrally. This means that students MUST submit an online Special Consideration form via Monash Connect. For more details, please refer to the **Unit Information** section in Moodle.

There is a **5% penalty per day including weekends** for a late submission. Also, the cut-off date is 7 days after the due date. No submission will be accepted (i.e. zero mark) after the cut-off date unless you have a special consideration.

Mark Release and Review

- Mark will be released within 10 business days after the submission deadline.
- Reviews and disputes regarding the mark will be accepted a maximum of 7 days after the release date (including weekends).

Generative AI Statement

As per the University's policy on the guidelines and practices pertaining to the usage

of Generative AI:

AI & Generative AI tools may be used SELECTIVELY within this assessment.

Where used, AI must be used responsibly, clearly documented and appropriately acknowledged (see Learn HQ).

Any work submitted for a mark must:

1. Represent a sincere demonstration of your human efforts, skills and subject knowledge that you will be accountable for.
2. Adhere to the guidelines for AI use set for the assessment task.
3. Reflect the University's commitment to academic integrity and ethical behaviour.

Inappropriate AI use and/or AI use without acknowledgement will be considered a breach of academic integrity.

The teaching team encourage students to apply their own critical thinking and reasoning skills when working on the assessments with assistant from GenAI. Generative AI tools may produce inaccurate content and this could have a negative impact on students' comprehension of big data topics.

Data source acknowledgement:

The dataset is a combination based on several real-world and synthetic datasets. Transaction records are from real-world data, user name, age, dob, salary etc. are randomly generated synthetic datasets.

We thank the authors/owners for sharing the original datasets.

1. eCommerce data from multi-category store | Kaggle
2. REES46
3. E-Commerce Data | Kaggle
4. Brazilian E-Commerce Public Dataset by Olist | Kaggle
5. Geoscape Geocoded National Address File (G-NAF) - Dataset - data.gov.au
6. Popular Baby Names - Dataset - data.sa.gov.au
7. Fashion Campus | Kaggle