Liying Lum

lumliying@gmail.com

E-commerce Case Study

Company ABC

Contents

[1. Introduction 2](#_Toc165579937)

[2. Database design 3](#_Toc165579938)

[i. Normalisation 3](#_Toc165579939)

[3. Data processing 4](#_Toc165579940)

[i. Data Loading: 4](#_Toc165579941)

[ii. Data Cleaning and Preparation: 4](#_Toc165579942)

[iii. Data Type Conversion: 4](#_Toc165579943)

[iv. Text Processing: 4](#_Toc165579944)

[v. Feature Engineering with TF-IDF: 5](#_Toc165579945)

[4. Dashboard 6](#_Toc165579946)

[i. Book Review 6](#_Toc165579947)

[ii. Sentiment Analysis Prediction 7](#_Toc165579948)

[iii. Sales summary 8](#_Toc165579949)

[Regional Analysis: 9](#_Toc165579950)

[Detailed Sales Performance Analysis 10](#_Toc165579951)

[5. Personal Reflection on the Assessment 11](#_Toc165579952)

# Introduction

Company ABC, a dynamic e-commerce platform, aims to leverage its vast data spread across multiple source systems. The objective is to establish a robust data warehouse and analytical infrastructure to drive data-informed decisions.

The raw data includes files from various sources such as electronics, liquor, and books.

# Database design

(Postgres)

A diagram of a computer

Description automatically generated

## Normalisation

In the e-commerce platform, we want to ensure that data is organized efficiently to support various operations such as product inventory monitoring, customer reviews, and sales transactions. Normalization helps achieve this by structuring data into separate tables:

|  |  |
| --- | --- |
| Table Name | Description |
| *books* | Stores information about books, including titles, authors, descriptions, and categories. |
| *book\_reviews* | Captures reviews written by customers for different books, along with ratings and helpfulness votes. |
| *liquor\_sales* | Records sales transactions for liquor products, including details like invoice numbers, quantities sold, and sales amounts. |
| *liquor\_types* | Categorizes liquor products into different types, helping in analyzing sales trends and customer preferences. |
| *sales* | Logs transactions for electronic products, such as order IDs, product details, and purchase addresses. |
| *electronics\_category* | Categorizes electronic products into various types and categories for better organization. |

# Data processing

In modern business environments, the ability to extract actionable insights from vast amounts of data is crucial for informed decision-making and competitive advantage. This report illustrates the rationale behind the design of a comprehensive data processing and analysis pipeline tailored for sales data. By leveraging Apache Spark and PostgreSQL, this pipeline aims to streamline data cleaning, preprocessing, analysis, and storage, empowering businesses to derive meaningful insights from their sales datasets.

## Data Loading:

The script loads the data files from CSV files into Spark DataFrames, ensuring that any special characters or escape sequences are correctly handled. A validity check is implemented to filter out irrelevant file names based on predefined patterns. This ensures that only valid files are processed, minimizing errors and improving data quality.

Columns containing textual information such as book titles, descriptions, and authors are cleaned to remove leading/trailing whitespace.

## Data Cleaning and Preparation:

Various data cleaning operations are performed to enhance data quality and usability. These operations include handling null values, parsing timestamps, and extracting address components.

Handled missing values by imputing the review summary column with appropriate values for downstream text processing.

Additionally, state abbreviations are converted to full names to improve data consistency and readability, facilitating meaningful analysis and interpretation.

## Data Type Conversion:

Data type conversion is used to ensure consistency and compatibility across the dataset. By standardizing data types, the pipeline facilitates seamless integration and analysis, minimizing potential errors and inconsistencies.

## Text Processing:

**Tokenization** was used to break down textual data, such as book descriptions or user reviews into individual words or tokens for further analysis, as it enables the extraction of meaningful information from the text. This splits the text into a sequence of words, making it easier to analyze and process.

Utilised **BERT** pre-trained model to generate dense vector representations that capture the semantic similarity between electronic products. The pipeline can then quantify the similarity between the product names and classify them into the same cluster.

**Stopwords** was used for the book review columns to exclude common words like "the", "is", and "and", that often do not carry significant semantic meaning and can be removed to focus on relevant content helping to refine the analysis and improve the accuracy of results.

Chose to try using **logistic regression** to determine the sentiment polarity of textual data for the book review columns. With that it can identify if the reviews are positive or negative. Since there is also a column with the review score. I did a comparison with the sentiment analysis results to cross check the accuracy against the input review score.

## Feature Engineering with TF-IDF:

Once the textual data is tokenized and stopwords are removed, feature engineering techniques like Term Frequency-Inverse Document Frequency (TF-IDF) are applied to transform the text into numerical features.

With the text data tokenized and stopwords removed, it is being processed further with Term Frequency-Inverse Document Frequency (TF-IDF) to transform it into numerical features.

It calculates the importance of words in each document (e.g., review) relative to the entire corpus. It assigns higher weights to words that are frequent in a particular document but rare across the entire corpus, indicating their significance in conveying the document's meaning.

With both HashingTF and IDF to compute TF-IDF vectors for the tokenized text, these vectors represent the importance of words in each document, enabling further analysis and machine learning tasks.

# Dashboard

## Book Review

A close up of words

Description automatically generated

Through the integration of tokenization and the removal of stopwords, the books dataset can then be prepared for visualization. As word clouds are a dynamic graphical representation that can depict the frequency distribution of words within each text collection. From this it can be easily showcase the most frequent words used when users write a review. The size of each word indicates its frequency of occurrence.

At a glance into these word clouds, one can swiftly empowered to observe overarching themes, prevalent topics, and underlying sentiments associated with the books encapsulated within our dataset.

## Sentiment Analysis Prediction

A screenshot of a computer

Description automatically generated

The sentiment analysis dashboard provides insight into the sentiment of users’ reviews based on their review summaries. It utilizes machine learning techniques, specifically Logistic Regression, along with text processing techniques to predict the sentiment polarity of each review. The dashboard presents three main components:

1. Review Prediction: Displays the predicted sentiment polarity (positive or negative) for each review summary using Logistic Regression.
2. Actual Review Score: Shows the actual review score provided by the user, ranging from 1 to 5.
3. Raw Sentiment Value: Presents the raw sentiment value based on the defined threshold (e.g., >=3 for positive sentiment), allowing users to see the sentiment classification directly

From this, users can compare the predicted sentiment polarity with the actual review scores provided by users. It can provide insights into the accuracy of the sentiment prediction model and potential discrepancies between predicted sentiment and user ratings.

## Sales summary

The analysis provided by the Sales Summary Dashboard offers insights that can inform strategic decision-making and drive business growth.

A screenshot of a computer

Description automatically generated

By understanding regional variations, customer preferences, and product performance trends, businesses can optimize their sales strategies or inventories, improve operational efficiency, and capitalize on growth opportunities in the market.

The ability to drill down when hovering over attributes like *product category* and *region* allows users to get into specific data points enables businesses to gain granular insights and make data-driven decisions that positively impact their bottom line.

A close-up of a graph

Description automatically generated

### Regional Analysis:

A screenshot of a computer

Description automatically generated

**Identifying High-Performing Regions:**

In the bar chart, each bar represents a region corresponding to the total sales figure for that region. User can quickly identify which regions are generating the highest sales revenue.

This insight helps businesses allocate resources more effectively by prioritizing high-performing regions for marketing campaigns, sales initiatives, and resource allocation.

An example of this is if Region A consistently has higher sales compared to other regions, businesses may choose to invest more inventories in that region to further capitalize on its potential.

**Market Penetration:**

By comparing sales figures across regions, businesses can identify regions where their products are selling well and regions where sales are relatively low. This analysis helps businesses identify opportunities for expansion into new markets or intensifying efforts in underperforming regions to increase market share.

For example, if sales in Region B are significantly lower compared to other regions, it may indicate untapped market potential or the need for targeted marketing strategies to increase brand awareness and drive sales.

**Regional Trends:**

Monitoring sales trends over time in each region provides insights into regional variations in consumer behaviour. Businesses can identify seasonal trends, peak sales periods, and fluctuations in demand within each region.

Understanding regional trends enables businesses to adjust their strategies accordingly, such as stocking up on inventory during peak seasons or offering region-specific promotions to capitalize on consumer preferences.

For instance, if sales in Region C show a consistent increase during the summer months, businesses can tailor their marketing efforts and product offerings to align with seasonal demand, thereby maximizing sales potential.

### Detailed Sales Performance Analysis

A screenshot of a computer

Description automatically generated

Integrating a heat map visualization and a drill-down stacked bar chart to provide deeper insights into sales performance over time and by product category. The dashboard leverages interactive features to allow users to explore sales data at various levels of granularity, from monthly trends to weekly and daily breakdowns for individual products.

1. Heat Map:

Displays sales data across months and product categories, with color-coded cells representing total sales values. Colour intensity indicates the magnitude of sales values, with darker colours representing higher sales volumes.

Users can hover over each cell to view detailed sales information.

1. Drill-down Stacked Bar Chart:

The drill-down stacked bar chart provides a detailed breakdown of sales for a selected product and month.

Each bar in the stacked chart represents sales for a specific week of the selected month, enabling users to analyse weekly sales trends.

Different colours are used to differentiate sales figures for each day of the week within the selected week, enhancing clarity and visual appeal.

# Personal Reflection on the Assessment

In this assessment, I ventured into utilizing PySpark for processing datasets, apart from my usual methods. While PySpark presents its own set of challenges, particularly in debugging code segments, I have come to appreciate its efficiency in handling large volumes of data. However, reflecting on my approach, there are areas where enhancements could have been made.

During the course of this assignment, I recognized the potential advantages of incorporating lemmatization and standardizing capitalization into the text preprocessing pipeline for book reviews. These techniques could have significantly improved the quality of text analysis by reducing variations in word forms and ensuring consistency across the dataset. Furthermore, cleaning up all symbols in the text could have contributed to a more refined analysis, enhancing the accuracy of insights extracted from the reviews. Due to time constraints, I was unable to implement these enhancements successfully, but I acknowledge its potential value in future iterations of similar projects.

Additionally, another area for improvement lies in the organization of the codebase. Modularizing the code into smaller, reusable modules could have enhanced readability, maintainability, and reusability. By breaking down the code into logical units, it becomes easier to understand, modify, and extend, facilitating collaboration and reducing the likelihood of errors.

Overall, while this assessment provided valuable insights into the capabilities of PySpark, there is room for improvement in both the data processing techniques employed and the structure of the codebase. Moving forward, I will keep these learnings in mind into future projects to enhance efficiency and effectiveness in data engineering tasks.