

## WQD 7007 BIG DATA MANAGEMENT (GROUP PROJECT)

*Abdirahman Abdiwahab (S2116432), Fathia Farhana Binti Agusalim (17147394), Muhamad Hazwan Faiz Bin Zaid (S2134438), Muhammad Asyraff Bin Ponan (S2128251), Ruzana Binti Mohamed Aris (S2182858), Safwan Bin Shamsir (S2915293), Siyu Jiang (22060253)*

Faculty of Computer Science and Information Technology, Universiti Malaya

### PRODUCT SALES OPTIMISATION IN E-COMMERCE

#### 1. INTRODUCTION

##### 1.1 Overview

The rise of e-commerce has transformed the business landscape, providing companies with unprecedented opportunities to reach a global audience and maximize sales. Amazon.com, a company founded by Jeff Bezos is one of the recognisable multinational companies focusing on e-commerce, cloud computing, online advertising, digital streaming, and artificial intelligence (AI) to personalize the shopping experience for its customers. It is a well-known company that uses a big data approach to offer great services to its customers by building customer relationships and maximizing customer service (K.Vidyakala & N. Devi, 2015). Since Amazon has an unrivalled bank of data on online consumer purchasing behaviour that can mine from its 300 million customer accounts, Amazon has been using that data for many years to build recommender systems to recommend things to consumers who visit Amazon.com.

However, as businesses continue to navigate and optimize their online presence, they face numerous challenges in tapping into the full potential of e-commerce. Since 2003, Amazon has applied item-item similarity algorithms from collaborative filtering as the recommender engine and also has leveraged consumer click-stream data and past purchase data from its clients to show tailored results on customized web pages to each user (M. Rijmenam, 2013).

As big data can be described in terms of data management challenges due to the characteristics of big data such as volume, variety, veracity, velocity, variability, visualisation, and value, which commonly known as “7Vs”. Amazon is a perfect example of a company that fits into the characteristics of big data.

##### 1.2 Problem Statement

The financial health of an organization, which reflected in its profits and losses is served as a general metrics for assessing the effectiveness of its business strategies. The longevity and success of any business hinge on its ability to consistently generate profits. This profitability not only will influence its eligibility for bank financing and its attractiveness to investors, but also its potential for growth. Conversely, sustained losses can lead to operational downsizing and, in extreme cases, bankruptcy. Therefore, it is vital for businesses, including those in the e-commerce sector, to understand which products yield high profits and which products has a result in low profits. By identifying and analysing these profit trends, a business can optimize its performance and enhance its value.

##### 1.2 Objectives

The aim of this project is to enhance the sales revenue in E-Commerce through understanding customer behaviour and product trends for product optimisation and improvement to customer shopping experience. To fulfil the aim of this project, three (3) main objectives have been outlined:

- i. To determine most profitable and least profitable e-commerce product.
- ii. To evaluate product purchase trending and fluctuation
- iii. To identify any new discoveries from customer purchase trends

#### 2. METHODOLOGY

This section describes detail methodology as well as outline big data pipeline and architecture executed for this project.

##### a) Metadata

Metadata refers to data that provides information about other data. In this project, an open-source dataset from Kaggle website E-Commerce Data is being used to further develop an

E-Commerce framework lifecycle using Apache Hadoop. This raw dataset which consists of one-year transactional dataset from 2020 – 2021 as described in Table 1.

**Table 1: E-Commerce Metadata.**

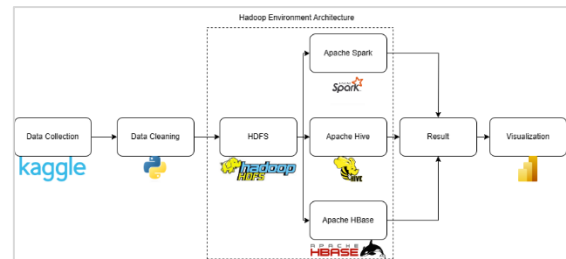
<b>Data Type</b>	Structured
<b>Dataset name</b>	E-commerce
<b>Format</b>	CSV File
<b>Number of records</b>	541,909 rows
<b>Number of variables</b>	8 columns
<b>Contains null value</b>	Yes
<b>Duplicate rows</b>	5,268 rows
<b>Contains outlier</b>	Yes

**Table 2: E-Commerce Data Description**

Features	Description
InvoiceNo	<ul style="list-style-type: none"> <li>• Invoice number.</li> <li>• Nominal, a 6-digit integral number uniquely assigned to each transaction.</li> </ul>
StockCode	<ul style="list-style-type: none"> <li>• Product code.</li> <li>• Nominal, a 5-digit integral number uniquely assigned to each transaction.</li> </ul>
Description	<ul style="list-style-type: none"> <li>• Product name.</li> <li>• Nominal</li> </ul>
Quantity	<ul style="list-style-type: none"> <li>• Quantities of each product per transaction. Negative value means returned product.</li> <li>• Numeric</li> </ul>
InvoiceDate	<ul style="list-style-type: none"> <li>• Invoice Date and time.</li> <li>• Numeric</li> </ul>
UnitPrice	<ul style="list-style-type: none"> <li>• UnitPrice in Sterling</li> <li>• Numeric, product price per unit.</li> </ul>
CustomerID	<ul style="list-style-type: none"> <li>• Customer code.</li> <li>• Nominal, a 5-digit integral number uniquely assigned to each customer.</li> </ul>
Country	<ul style="list-style-type: none"> <li>• Country name.</li> <li>• Nominal</li> </ul>

#### b) Big Data Pipeline and Architecture

The big data pipeline architecture is illustrated in in Fig. 1



**Fig. 1: Proposed Big Data Pipeline used for the project**

The methodology employed in this project encompasses several key steps. Initially, a raw e-commerce dataset is obtained from Kaggle. The dataset then undergoes a detailed data cleaning, data preprocessing, and feature engineering using Python on Google Colab, where missing values, duplicates, and format inconsistencies are addressed, and two new features have been inferred from the existing features. Upon completion of this phase, the cleaned dataset is saved as a CSV file.

Subsequently, the CSV file is loaded into the Hadoop Distributed File System (HDFS), a platform renowned for its distributed storage capabilities. Apache Hive, Apache Spark, and Apache HBase are then utilized to query and analyze the dataset stored in HDFS. It allows for the definition of tables and schemas and facilitates the execution of SQL-like queries on the dataset.

**Table 3: Big Data Framework Lifecycle and Tools**

Phase	Tools	Description
Data Cleaning	Google Colab	Handle raw dataset and clean the data for data quality and reliability using Python
Data Ingestion and Storage	HDFS	Load data from local machine and store raw dataset into HDFS
Data Query & Preprocessing	HBase, Hive, and Spark	Read and process the data using NoSQL (HBase) and SQ-like (HiveQL & PySpark) language
Data Visualisation	Microsoft Power BI	Visualise the processed and analysed data with appropriate graph

#### i. Google Colaboratory

Google Colab, also known as “Colaboratory”, is a powerful data analysis tool that enables users to build and run Python programs in a web browser. It has an easy-to-use interface and a diverse set of features for data processing, analysis, and collaboration (EdXD, 2022).

Google Colab allows users to connect to a variety of data sources, including those stored on Google Drive, GitHub, Kaggle, and other web platforms. By establishing these connections, Google Colab allows users to input massive amounts of data and generate interactive visual representations.

#### ii. Apache Hadoop (HDFS)

HDFS, or Hadoop Distributed File System, is a distributed file system that is part of the Hadoop ecosystem. It is designed to handle large data sets running on commodity hardware, and it can scale a single Apache Hadoop cluster to hundreds, or even thousands, of nodes (*What Is HDFS?* | IBM, n.d.).

HDFS operates by rapidly transferring data between nodes, making it suitable for applications that deal with large data sets. One of the key features of HDFS is its fault tolerance. It is designed to detect faults and automatically recover quickly. This is particularly useful in environments where failure of at least one server is inevitable due to the large number of commodity hardware components.

Leveraging HDFS for e-commerce datasets allows efficient management of large datasets, where the distribution of the e-commerce data across the HDFS cluster not only ensures higher availability but also allows the Hadoop cluster to break up work into smaller chunks and run those jobs on all the servers in the cluster for better scalability.

#### iii. Apache Hadoop (Spark - PySpark)

PySpark is a Python API for Apache Spark that allows user to use Python commands with the power of Apache Spark for data processing and analysis. This is suitable for data scientists who are familiar with Python language instead of Scala to use Spark engine in their data processing and analysis.

PySpark make use of Spark dataframe which is a table that is distributed across a cluster. One of the key differences between Spark dataframe and Pandas in Python is its lazy execution where the transformation of data is recorded and not applied immediately. Once the result is called, the transformation will be applied to the dataset as a single pipeline.

Utilising PySpark for e-commerce dataset allows efficient data preprocessing to be done for further analysis.

#### iv. Apache Hadoop (HBase)

Apache HBase is a Hadoop database that offers random real-time read or write access to the big data resources. Besides its scalability feature, Apache HBase is also open source, distributed and support NoSQL (Not Only SQL) which mainly consists of non-relational database (*Apache HBase – Apache HBase™ Home*, n.d.). We can imply that HBase is a column-oriented non-relational database that can store enormous volume of data (*What Is HBase?* | IBM, n.d.) where data is stored in individual columns and each row is defined by a unique row key. HBase also offers strictly consistent read and write access for big data processing (*Apache HBase – Apache HBase™ Home*, n.d.).

One interesting feature of HBase is database sharding which defined as massive size of databases is split into smaller and manageable part and each shard is being stored in different servers. This database sharding highlights the scalability and fault-tolerant features of HBase (Awati & Denman, 2022). According to IBM article (*What Is HBase?* | IBM, n.d.), since HBase is a NoSQL database management system, it does not support structured query language (SQL) at all unlike other relational databases.

#### v. Apache Hadoop (Hive)

Apache Hive is a distributed, fault-tolerant data warehouse system that enables analytics at a massive scale with easy data extraction, transformation, and loading (ETL). It allows users to read, write, and manage petabytes of data using SQL-like queries. Hive Metastore (HMS) is a critical component of many data lake architectures as it provides a repository of metadata for analysis and informed decision-making.

Hive is built on top of Apache Hadoop and supports storage on S3, and others through HDFS. The query requested by users or applications will be accepted by Hive Server 2 before creating an execution plan and automatically generates a YARN job to process the SQL queries. The YARN jobs will be generated as Apache Spark, Map Reduce, and Apache Tez jobs. Once the SQL query has been processed, the resulting queries will either return to the end-user or the application or will transmitted back to the HDFS.

#### vi. Microsoft Power BI

Microsoft Power BI is an effective business analytics instrument that enables individuals to depict and examine data from different origins. It offers an intuitive interface and a diverse array of functionalities for data representation, investigation, and collaboration. The ability of this tools to

integrate with various data sources either located on-premises or in cloud platforms as well as implementation of minimal or no code that empowers users to import substantial amounts of data and generate interactive visual representations has made the tools be more favourable to organisations without mastering specific programming language (Mihart, 2023).

Adoption of Power BI to visualise the e-commerce will be able to assist the organisation to obtain business insights and support decision making as well as developing impactful strategies to increase the sales.

### 3. RESULTS AND DISCUSSION

#### a) Data Cleaning with Google Collab

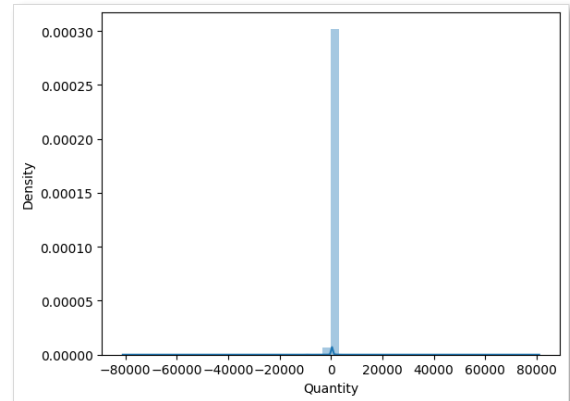
Firstly, the raw dataset was imported directly from Kaggle to Google Colab by using Kaggle API credentials which comprises of 541,909 total records. Then, exploratory data analysis (EDA) was performed to the dataset to inspect the cleanliness of the data. As a results from EDA process, there were 135,080 rows in CustomerID column and 1,454 rows in Descriptions column that contains null values. The null values in CustomerID column were dropped, which comprises of 25% of the total rows, due to each product transactions need to be linked to CustomerID for transactions verifications. There were also duplicate records, which probably due to double scanning on the Point-of-Sales (POS) system, also have been dropped.

During the EDA process, it is observed that there are several outliers in the Quantity column, which has been handled by using InterQuartile Range (IQR) method as shown in Fig. a.1 and Fig. a.2. In addition, there was also invalid values in StockCode column such as 'PADS', 'DOT', 'CRUK', and many more which has been excluded from the dataset.

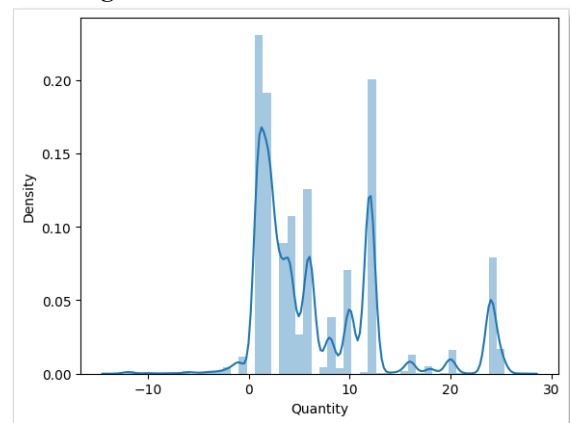
To achieve one of the project objectives which is to determine the most profitable and the least profitable products in every continent, new features named Continent has been produced by inferring to the value in Country column. The unknown value in the Country column has been dropped as well. Besides that, a new features named Status has been created by inferring the Quantity column, where if the value in Quantity column is negative value, it will return value 'Cancelled Order' in Status column, otherwise, it will return value 'Sold' in Status column.

In summary, a total of 169,056 records were eliminated from the dataset, resulting in a cleaned dataset comprising 372,853 records. This signifies

that approximately 31.2% of the raw data was removed during the cleaning process. Finally, the cleaned data has been imported into Hadoop Distributed File Systems (HDFS) for further analysis using Apache Hive, Apache HBase, Apache Spark and Microsoft Power BI. The detail of this data cleaning process is documented in this [link](#)



**Fig. a.1: Variations of data distribution before removing the outliers.**



**Fig. a.2: Variations of data distribution after removing the outliers.**

#### b) Data Ingestion and Storage with Apache Hadoop (HDFS)

As part of implementing data ingestion and storage with HDFS, a cluster was started using "start-all.sh" command in Hadoop. Once the cluster is started, a directory was created in HDFS for the cleaned dataset using "-mkdir" command. The cleaned dataset was then uploaded to HDFS using "-put" command.

```
hazwan987@MSI:~$ hadoop dfs -put cleaned_dataset.csv /user/wqd7007/
DEPRECATED: Use of this script to execute hdfs command is deprecated.
Instead use the hdfs command for it.
```

**Fig. b.1: Loading data to HDFS**

### c) Data Pre-Processing with Apache Spark – PySpark

The cleaned dataset was firstly loaded into Pyspark from HDFS using the spark.read.csv command. The first row of the csv file was used as the header for each column. The data type of each column also was determined as per Fig. c.1.

```
>>> ecom.printSchema()
root
|-- InvoiceNo: string (nullable = true)
|-- StockCode: string (nullable = true)
|-- Description: string (nullable = true)
|-- Quantity: integer (nullable = true)
|-- Status: string (nullable = true)
|-- InvoiceDate: date (nullable = true)
|-- UnitPrice: double (nullable = true)
|-- CustomerID: string (nullable = true)
|-- Country: string (nullable = true)
|-- Continent: string (nullable = true)
```

Fig. c.1: Data type of each column

New column was added to the dataset called “Total Amount” which is a product of each row’s unit price and quantity. This column will help to analyse the product revenue contribution.

The dataset has been separated based on their continent which are Europe, Asia, North America, South America, Oceania, and Africa for ease of analysis of each continent’s top and bottom product. The data also was filtered to only take invoices with “Sold” status as “Cancelled Order” status does not represent any revenue for the analysis. Both of this condition were called using “filter” function.

For each continent, the data was grouped by its stock code using the “groupBy” function to obtain the aggregated value of quantity sold and the total revenue received by each item throughout the dataset time frame. These columns are represented by “Total Quantity” and “Total Amount” and are sum of “Quantity” and “Total Revenue” column respectively. The grouped data was sort according to its revenue by descending and ascending in order to find the highest and lowest revenue product for each continent.

From the grouped data, we can see the highest revenue products are “Regency Cakestand 3 Tier” for Europe, Asia and South America, “Set 6 School Milk Bottles in Crate” for North America, “Red Retrospot Cake Stand” for Oceania, and “Classic Metal Birdcage Plant Holder” for Africa. Most of the continent have relatively large Total Quantity as compared to other products which shows high demand for these top products and would be a fast-moving product for their region with high revenue. However, for South America and Africa there are other products other than the top products

which were sold more, indicated by its Total Quantity but with lower Total Amount. This imply that the top products do have higher revenue due to its high unit price but is not a high demand product generally which might be slow to be sold consistently.

The least profitable products by region are “Pens Assorted Funny Face” for Europe, “World War 2 Gliders Asstd Designs” for Asia, “Set Of 4Pantry Jelly Moulds” for North America, “Emergency First Aid Tin” for South America, “Wrap Vintage Leaf Design” for Oceania and “Assorted Bottle Top Magnets” for Africa. For these lowest revenue products, the quantity sold for each item in their respective continent also is low in comparison to other products. This means that these products are not in high demand and provide the least revenue which are logical to reduce the stock at their continent to save the storage space to maximize the high demand products which can give better revenue to the sale.

```
>>> windowProd = Window.partitionBy("Continent").orderBy(col("TotalAmount").desc())
>>> df_product.withColumn("row",row_number().over(windowProd)).filter(col("row") <= 5).show(30)
```

StockCode	Description	Continent	TotalAmount	row
21380	CLASSIC METAL BIR...	Africa	38.25	1
22685	WOODEN CROQUET GA...	Africa	29.9	2
23298	SPOTTY BUNTING	Africa	29.7	3
22960	JAM MAKING SET WI...	Africa	25.5	4
22423	REGENCY CAKESTAND...	Africa	25.5	5
22423	REGENCY CAKESTAND...	Asia	707.4	1
23889	I LOVE LONDON BAB...	Asia	358.8	2
21250	VICTORIAN SEWING ...	Asia	356.4	3
22192	BLUE OTHER WALL C...	Asia	326.4	4
21217	RED RETROSPOT ROU...	Asia	322.2	5
22423	REGENCY CAKESTAND...	Europe	86785.9	1
47564	PARTY BUNTING	Europe	34534.98	2
85123A	WHITE HANGING HEA...	Europe	31899.01	3
85099B	JUMBO BAG RED RET...	Europe	24799.22	4
23298	SPOTTY BUNTING	Europe	23201.2	5
23358	SET 6 SCHOOL MILK...	North America	362.72	1
22423	REGENCY CAKESTAND...	North America	127.5	2
22649	STRAWBERRY FAIRY ...	North America	79.2	3
23173	REGENCY TEAPOT RO...	North America	59.7	4
23159	SET OF 5 PANCAKE ...	North America	49.92	5
21843	RED RETROSPOT CAK...	Oceania	727.35	1
22634	CHILD'S BREAKFAST ...	Oceania	488.9	2
22636	CHILD'S BREAKFAST ...	Oceania	367.2	3
20685	DOORMAT RED RETRO...	Oceania	349.5	4
22685	HERO BOARD COTTAG...	Oceania	345.6	5
22423	REGENCY CAKESTAND...	South America	175.2	1
22722	SET OF 6 SPICE TI...	South America	82.8	2
21438	SET/3 RED GINGHAM...	South America	81.36	3
22466	DOORMAT AIRMAIL	South America	67.5	4
22698	PINK REGENCY TEAC...	South America	61.2	5

Fig. c.2: Top 5 Products by Continent

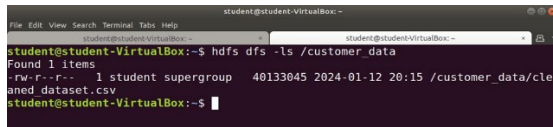
```
>>> windowProd = Window.partitionBy("Continent").orderBy(col("TotalAmount").asc())
>>> df_product.withColumn("row",row_number().over(windowProd)).filter(col("row") <= 5).show(30)
```

StockCode	Description	Continent	TotalAmount	row
22915	ASSORTED BOTTLE T...	Africa	5.84	1
21238	RED RETROSPOT CUP	Africa	6.8	2
23295	CHARLOTTE BAG VIN...	Africa	8.5	3
28677	PINK POLADOT BOWL	Africa	18.0	4
28676	RED RETROSPOT BOWL	Africa	18.0	5
84877	WORLD WAR 2 GLIDE...	Asia	0.29	1
22531	MAGIC DRAWING SLA...	Asia	0.42	2
22541	MINI JIGSAW LEAP ...	Asia	0.42	3
22539	MINI JIGSAW DOLLY...	Asia	0.42	4
22536	MAGIC DRAWING SLA...	Asia	0.42	5
22610	PENS ASSORTED FUN...	Europe	0.38	1
84227	HEN HOUSE W CHICK...	Europe	0.42	2
23366	SET 12 COLOURING	Europe	0.65	3
21268	VINTAGE BLUE TINS...	Europe	0.84	4
90184	PURPLE FRANGIPANI...	Europe	0.85	5
22993	SET OF 4 PANTRY P...	North America	1.25	1
10161G	WRAP BAD HAIR DAY	North America	2.5	2
22979	PANTRY WASHING UP...	North America	2.9	3
22965	3 TRADITIONAL BIS...	North America	4.2	4
37407	PIG PUG IN TWO CO...	North America	4.68	5
23232	WRAP VINTAGE LEAF...	Oceania	0.42	1
21583	TOYBOX WRAP	Oceania	0.42	2
21917	SET 12 KIDS WHIT...	Oceania	0.42	3
22465	MINI JIGSAW RUNNIES	Oceania	0.42	4
21918	SET 12 KIDS COLOU...	Oceania	0.42	5
22494	EMERGENCY FIRST A...	South America	15.0	1
23852	RECYCLED ACAPULCO...	South America	16.5	2
23853	RECYCLED ACAPULCO...	South America	16.5	3
23849	RECYCLED ACAPULCO...	South America	16.5	4
23850	RECYCLED ACAPULCO...	South America	16.5	5

Fig. c.3: Bottom 5 Products by Continent

#### d) Data Access with Apache HBase

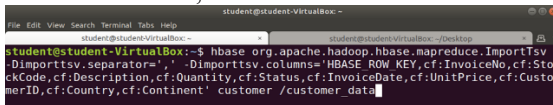
To start the HBase, firstly the dataset is uploaded to HDFS via directory `/customer_data`



```
student@student-VirtualBox:~$ hdfs dfs -ls /customer_data
Found 1 items
-rw-r--r-- 1 student supergroup 40133045 2024-01-12 20:15 /customer_data/cleaned_dataset.csv
student@student-VirtualBox:~$
```

Fig.d.1: Dataset upload to HDFS

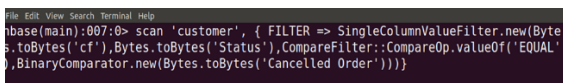
The dataset is imported to HBase using 'ImportTsv' command by defining the unique row key for the HBase table together with its column family which consists of the attributes in the original dataset. After that, 'customer' table is created.



```
student@student-VirtualBox:~$ hbase org.apache.hadoop.hbase.mapreduce.ImportTsv -Dimporttsv.separator=', ' -Dimporttsv.columns='HBASE_ROW_KEY,cf:InvoiceNo,cf:StockCode,cf:Description,cf:Quantity,cf>Status,cf:InvoiceDate,cf:UnitPrice,cf:CustomerID,cf:Country,cf:Continent' customer /customer_data
student@student-VirtualBox:~$
```

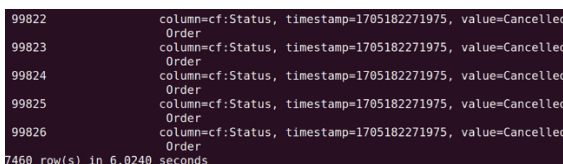
Fig.d.2: Dataset imported to HBase table 'customer'

After the dataset has been successfully imported to HBase, several commands were executed to scan and read the data. Figure d.3 and Figure d.4 demonstrate the query and output generated in identifying number of cancelled orders. Since HBase has a unique row identifier for each row, it can scan the whole dataset and find the specified output which is 'Cancelled Order' in the 'Status' column family.



```
hbase(main):007:0> scan 'customer', { FILTER => SingleColumnValueFilter.new(Bytes.toBytes('cf'), Bytes.toBytes('Status'), CompareFilter::CompareOp.valueOf('EQUAL'), BinaryComparator.new(Bytes.toBytes('Cancelled Order')) ) }
hbase(main):007:0>
```

Fig.d.3: Query for cancelled order

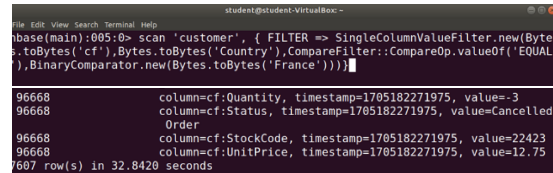


```
99822 column=cf:Status, timestamp=1705182271975, value=Cancelled Order
99823 column=cf:Status, timestamp=1705182271975, value=Cancelled Order
99824 column=cf:Status, timestamp=1705182271975, value=Cancelled Order
99825 column=cf:Status, timestamp=1705182271975, value=Cancelled Order
99826 column=cf:Status, timestamp=1705182271975, value=Cancelled Order
7460 row(s) in 6.0240 seconds
```

Fig.d.4: Result for cancelled order

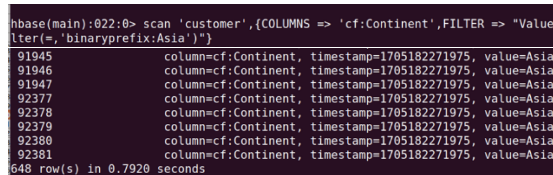
The result showed all the row ID with status of the order is cancelled. HBase also provided the total number of rows for those specific commands. Based on the e-commerce dataset there are 7,460 cancelled orders in total across all continents.

If business would like to identify total order in specific country and continent, HBase is capable to run the query and scan from the dataset. For example, HBase query is performed to scan the orders originating from France. Figure d.5 shows there is a total of 7,607 orders coming from France regardless of the status of the order. Same command can be executed to analyse the total orders coming from other location.



```
hbase(main):005:0> scan 'customer', { FILTER => SingleColumnValueFilter.new(Bytes.toBytes('cf'), Bytes.toBytes('Country'), CompareFilter::CompareOp.valueOf('EQUAL'), BinaryComparator.new(Bytes.toBytes('France')) ) }
96668 column=cf:Quantity, timestamp=1705182271975, value=-3
96668 column=cf:Status, timestamp=1705182271975, value=Cancelled Order
96668 column=cf:StockCode, timestamp=1705182271975, value=22423
96668 column=cf:UnitPrice, timestamp=1705182271975, value=12.75
7607 row(s) in 32.8420 seconds
```

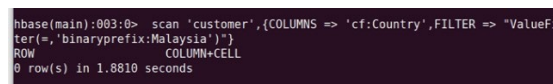
Fig.d.5: Example of query and results for number of orders in France



```
hbase(main):022:0> scan 'customer', { COLUMNS => 'cf:Continent', FILTER => 'ValueFilter(=, 'binaryprefix:Asia')' }
91945 column=cf:Continent, timestamp=1705182271975, value=Asia
91946 column=cf:Continent, timestamp=1705182271975, value=Asia
91947 column=cf:Continent, timestamp=1705182271975, value=Asia
92377 column=cf:Continent, timestamp=1705182271975, value=Asia
92378 column=cf:Continent, timestamp=1705182271975, value=Asia
92379 column=cf:Continent, timestamp=1705182271975, value=Asia
92380 column=cf:Continent, timestamp=1705182271975, value=Asia
92381 column=cf:Continent, timestamp=1705182271975, value=Asia
648 row(s) in 0.7920 seconds
```

Fig.d.6: Query and results for number of orders from Asia continent

As depicted in Fig.d.6, there are 648 orders coming from Asia. From the result, it is observed that the column family 'Continent' for all the unique row ID have the value of 'Asia'. If business would like to narrow down the search to specific country e.g. Malaysia, Fig.d.7 demonstrates sample of query used which identify 0 row indicating no order coming from the country.



```
hbase(main):003:0> scan 'customer', { COLUMNS => 'cf:Country', FILTER => 'ValueFilter(=, 'binaryprefix:Malaysia')' }
ROW COLUMN+CELL
0 row(s) in 1.8810 seconds
```

Fig.d.7: Query and results for number of orders from Malaysia

Based on all the queries performed by HBase, the queries executed does not able to meet one of the project objectives due to the limitations in HBase. HBase offers basic querying capabilities through row key scans, but it lacks a full-fledged query language like SQL as well ability to perform complex aggregation functions such as sum, average, and group by categories especially involving numerical data in the database itself (Taylor, 2023). HBase also happens to not be having query optimiser due to the fact that HBase is a non-relational database and normalisation and joining are difficult to be done using HBase (Kumar, 2023).

Thus, it is inferred that HBase capabilities is limited to random read, write and scan access and not including SQL-like queries to aggregate data and group by category and easy data summarization like Apache Hive. In a nutshell, HBase are good solutions to store massive volume of data but cannot perform effective data analysis.

#### e) Data Query with Apache Hive

Execution of data analysis with Hive is being first being performed by creating a table 'retail' where each column name is defined with its data types, and the file is stored in HDFS. Fig.e.1 and Fig.e.2 demonstrates command that being executed



to create the table as well as verifying the table that is loaded in Hive.

```

hive> create table cancelled_orders (
  invoiceid string,
  stockcode string,
  description string,
  quantity int,
  status string,
  invoicedate timestamp,
  unitprice double,
  customerid string,
  country string,
  continent string,
  totalamount double)
  partitioned by (year int, month int, day int)
  rowformat serde 'org.apache.hadoop.hive.serde2.lazy.LazyBinarySerDe'
  storageformat 'orc'
  tblproperties ('orc.compress=ZLIB')
)

```

Fig.e.1. Create a new table

```

hive> select *
  from cancelled_orders
 limit 5;

```

Fig.e.2. Sample of 5 rows of data

After loading the data and creating the table in Hive, a new table is created to add a new column named 'TotalAmount', which refers to the total sales amount for each transaction and it's calculated by Quantity \* UnitPrice. The first five (5) rows of new table are shown in Fig.e.3 where Fig.e.4 shows the column names and data types in the new table.

```

hive> select *
  from cancelled_orders
 limit 5;

```

Fig.e.3. Include a new column 'TotalAmount'

```

invoiceid      string
stockcode      string
description     string
quantity       int
status         string
invoicedate    timestamp
unitprice      double
customerid     string
country        string
continent      string
totalamount    double

```

Fig.e.4. List of column name and its datatype

To analyse the e-commerce data, Fig.e.5 to Fig.e.9 outline the result generated that can draw business insights.

#### 1. Hights total sales from continent

```

Europe 5023116.470001148
Asia 17742.7699999999982
Ocenia 17596.500000000001
North America 3913.02
South America 1143.6000000000001
Africa 1002.3099999999998

```

Fig.e.5. Continent with highest total sales

#### 2. Highest total sales based on country and continents

```

Africa South Africa 1002.3099999999998
Asia Singapore 5901.2300000000005
Europe United Kingdom 4317429.7300000339
North America Canada 2131.0399999999995
Ocenia Australia 17596.500000000001
South America Brazil 1143.6000000000001

```

Fig.e.6. Highest total sales based on country and continents

It can be observed that Europe stands out with a significantly higher total sales amount compared to other continents, and The United Kingdom stands out with a significantly higher total sales amount compared to other countries, indicating that United Kingdom as a a major market in Europe for the e-commerce platform. The United Kingdom's significant lead in sales could suggest a focus on customer retention and expansion in that market.

#### 3. Top five products with highest cancelled orders

```

REGENCY CAKESTAND 3 TIER 170
JAM MAKING SET WITH JARS 86
SET OF 3 CAKE TINS PANTRY DESIGN 70
ROSES REGENCY TEACUP AND SAUCER 50
STRAWBERRY CERAMIC TRINKET BOX 50

```

Fig.e.7. Top 5 product with highest cancelled order

According to the outcome shown in Fig.e.7, it can be observed that REGENCY CAKESTAND 3 TIER has the highest no. of cancellation order with 170 orders, followed by JAM MAKING SET WITH JARS with a total of 86 cancellations and SET OF 3 CAKE TINS PANTRY DESIGN with 70 cancelled orders. Product order on ROSES REGENCY TEACUP AND SAUCER and STRAWBERRY CERAMIC TRINKET BOX both have an equal number of cancellations, with 50 each.

This result reveals there might be potential issues during the e-commerce service such as quality issues or inaccurate descriptions, inventory or supply chain management problems causing delays in order fulfilment or stockouts which leads to cancellations by customer. Hence, business can conduct an in-depth analysis on quality management and product lifecycle delivery check to minimise the cancel order status.

#### 4. Number of cancelled orders from continent

```

Europe 7282
North America 83
Ocenia 65
Asia 30

```

Fig.e.8. Number of cancelled order from each continent

Looking into a broader scale on the no. of cancelled order for each continent, Fig.e.8, highlights that Europe has the most cancelled orders (7,282 orders), and this is mainly due to due to customer dissatisfaction, payment issues, logistics problems, etc. It would be crucial to investigate the reasons behind these cancellations to address any underlying problems.

The low number of cancellations in Asia and Oceania might indicate a smaller customer base,

higher customer satisfaction, or more efficient order fulfilment. It's also possible to include strict cancellation policies or clauses to reduce the cancellation impact.

5. Number of cancelled orders in each continent in monthly basis

2010-12	Europe	578
2010-12	Ocenia	3
2011-01	Asia	3
2011-01	Europe	582
2011-02	Europe	383
2011-02	Ocenia	1
2011-03	Asia	6
2011-03	Europe	552
2011-04	Asia	8
2011-04	Europe	462
2011-04	Ocenia	1
2011-05	Europe	508
2011-05	Ocenia	1
2011-06	Europe	575
2011-06	Ocenia	2
2011-07	Europe	509
2011-07	Ocenia	56
2011-08	Asia	7
2011-08	Europe	509
2011-09	Europe	656
2011-09	Ocenia	1
2011-10	Europe	805
2011-10	North America	77
2011-11	Europe	873
2011-12	Asia	6
2011-12	Europe	290
2011-12	North America	6

Time taken: 3.744 seconds, Fetched: 27 row(s)

**Fig.e.9. Number of cancelled orders by continents by months**

Figure 9 shows that Europe is consistently has the highest number of cancelled orders every month throughout the year. The numbers vary, with the highest being 873 cancelled orders in November 2011. North America shows a significantly lower number of cancellations compared to Europe. Oceania has very few cancellations, mostly in single digits, except for July 2011, when there were 15 cancellations. Asia also has relatively low cancellation numbers, with cancellations mostly in the range of tens, with a peak of 383 in January 2011.

This result shows that there could be seasonal trends, such as increased cancellations during certain months, which could be correlated with holidays or sales events. For example, Europe has the highest number of cancellations in October and November, probably due to the increased logistics burden and price volatility caused by holidays such as Halloween, Thanksgiving and Christmas.

Some mitigation that can be implemented is via introducing dynamic pricing to manage price fluctuations effectively, so customers feel they are getting fair prices even during high-demand periods, and keep customers informed about their order status, any potential delays, and expected delivery times. Transparency can reduce cancellations.

## f) Data Visualisation with Microsoft Power BI

As part of interpreting the data more effectively and in interactive way, data visualisation is being employed for business to better visualise the result. The cleaned dataset is imported into Power BI, and the data type of each column is being verified and transformed before uploaded to the dashboard to ensure the correct assignment for each variable. This step is important as using an incorrect data type can result in inaccurate analyses or calculations. In addition, certain visualisations are specific to particular data types.

```

= Table.TransformColumns([e-commerce], {{"Product", type text}, {"Quantity", type text}, {"Status", type text}, {"OrderDate", type text}, {"Total", type text}, {"Customer", type text}})

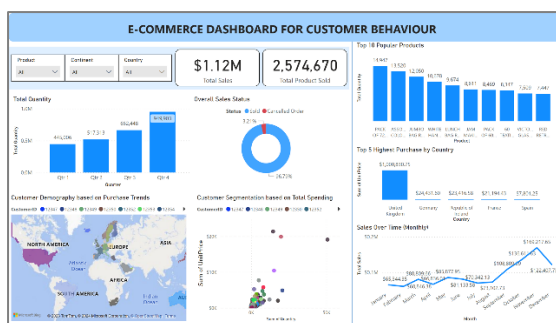
```

**Fig. f.1: Query to transform e-commerce datatype**

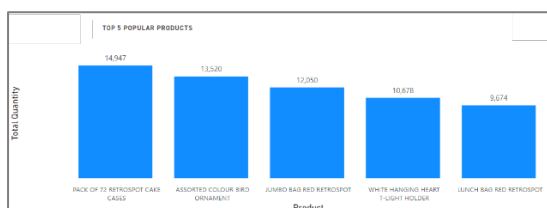
Once the data type has been transformed, visualisation dashboard is established as shown in Fig. f.2. The dashboard can be sliced down, and the graphs are also interactive to one another. Though from the data preprocessing conducted by pySpark, Hbase and Hive have addressed most of the business objectives, there are also additional discoveries from the visualisation that can be addressed. For example, Fig f.3 shows that, top popular products from e-commerce is Pack of 72 Restrospot Cake Cases as compared to top revenue addressed by Europe is Regency Cakestand 3 Tier" highlighted from pySpark. Throughout the 1-year period of customer purchase history, Fig. f.4 shows that there is a noticeable peak order in November, where total sales reach the highest point on the chart at \$169,217.65. This could be to several reasons such as a seasonal increase in demand for products due to online shopping beginning in November such as early Christmas shopping.

Additionally, customer behaviour patterns can be inferred from the graph in Figure f.5, which demonstrates that the majority of purchases are clustered below a total quantity of 10,000 and a sales figure of \$10,000. This could reflect the diverse nature of the customer base, which ranges from regular consumers and small businesses to bulk buyers, each exhibiting unique purchasing patterns. Moreover, the data suggests that members of loyalty programs are likely to buy more, which corresponds to higher sales for specific customers identified in the dataset.

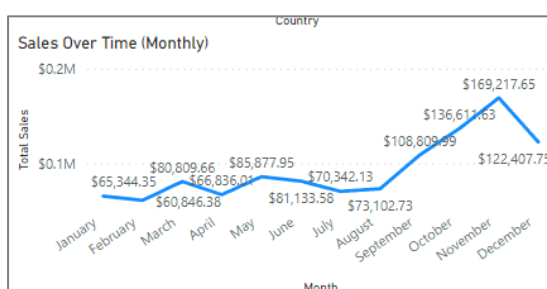




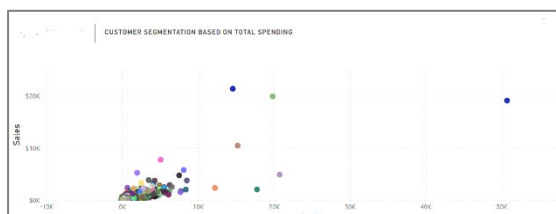
**Fig. f.2: Data Visualisation for E-Commerce Dashboard with Power BI**



**Fig. f.3: Overall Top 5 Popular Products**



**Fig. f.4: Monthly Product Trending Sales**



**Fig. f.5: Customer demography based on purchase quantity and sales**

#### 4. CONCLUSION

Based on the implementation of different big data tools applied for this project, the project has successfully established big data pipeline for e-commerce to meet the project objectives. Table 3 has demonstrated summary of comparison on employing data query and processing from PySpark, HBase and Hive to extract the business insights. It is suggested that comparison on computational work performance of these models is not evaluated as execution of the process for each tool were performed by different users' machines that have different hardware configurations (CPU, RAM,

storage, network capacity). Some user machines have high-end server that will perform differently as compared to a mid-range one. Thus, comparative assessment to assess which big data tools that perform better for data query and preprocessing is not performed. Furthermore, the focus of the project is to study customer purchase behaviour based on the online shopping from the e-commerce platform.

**Table 4: Comparison between different tools**

Features	pySpark	HBase	Hive
Data Handling	It is an analytical framework for performing large-scale analytics.	This tool is extensively used for transactional processing for fast retrieval of massive data volumes.	It is a distributed data warehouse platform to store and manage massive data volumes.
Query Language	SQL	NoSQL key/value store	Hive QL
Operation	Operates on the server side of any cluster.	Run in real-time on its database.	Operates on the server side of any cluster.
Scalability	Highly flexible and scalable.	Easily scalable	Easy to understand and scalable

The selection of tools for exploring big data resources depends on their specific use and the requirements needed to match each tool's functionality and capabilities. Hence adoption of higher-capability tools like Apache Pig that offers powerful transformation and processing capabilities can be further explored. Furthermore, advancement of the technologies with cloud platform platforms such as Google Cloud, Microsoft Azure, and Amazon Web Services (AWS) could be employed to handle big data for e-commerce as well uncovering more meaningful patterns, thereby enabling more informed decisions in sales optimization.

#### 5. REFERENCES

- [1] Van Rijmenam, M. (2023, April 26). *How Amazon is leveraging big data*. Dataflok. <https://dataflok.com/read/amazon-leveraging-big-data/>
- [2] Neal. (2022, October 3). *A Brief History Of AWS – And How Computing Has Changed*. Digital Cloud Training. <https://digitalcloud.training/a-brief-history-of-aws-and-how-computing-has-changed/>

- [3] *Data lakes and Analytics on AWS - Amazon Web Services*. (n.d.). Amazon Web Services, Inc. <https://aws.amazon.com/big-data/datalakes-and-analytics/>
- [4] A, M. (2023, April 11). *How Amazon uses data science and analytics to drive e-commerce success*. <https://www.linkedin.com/pulse/how-amazon-uses-data-science-analytics-drive-success-michael-ampof#:~:text=Amazon>
- [5] Mrkonjić, E. (2022, November 21). *How Amazon uses big Data | SeedScientific*. [SeedScientific](https://seedscientific.com/how-amazon-uses-big-data/). <https://seedscientific.com/how-amazon-uses-big-data/>
- [6] *What is Hadoop? - Apache Hadoop Explained - AWS*. (n.d.). Amazon Web Services, Inc. <https://aws.amazon.com/what-is/hadoop/#:~:text=Amazon>
- [7] Taylor, D. (2023, September 18). *Amazon data collection: what info does the company gather? — Reason Automation*. Reason Automation. <https://www.reasonautomation.com/content/amazon-data-collection-what-info-does-the-company-gather>
- [8] O'Flaherty, K. (2022, February 27). *The data game: what Amazon knows about you and how to stop it*. *The Guardian*. <https://www.theguardian.com/technology/2022/feb/27/the-data-game-what-amazon-knows-about-you-and-how-to-stop-it>
- [9] Invisibly. (2023, April 7). *How does Amazon use big Data? 2023 Amazon Big Data Insights*. Invisibly. <https://www.invisibly.com/learn-blog/how-amazon-uses-big-data/>
- [10] Digicode. (2023, July 14). *Why companies should opt for big data*. <https://www.linkedin.com/pulse/why-companies-should-opt-big-data-digicode>
- [11] Marr, B. (2021, August 12). *Amazon: Using Big Data to understand customers*. Bernard Marr. <https://bernardmarr.com/amazon-using-big-data-to-understand-customers/>
- [12] Mills, T. (2019, November 6). *Five Benefits Of Big Data Analytics And How Companies Can Get Started*. *Forbes*. <https://www.forbes.com/sites/forbestechcouncil/2019/11/06/five-benefits-of-big-data-analytics-and-how-companies-can-get-started/?sh=6814c04b17e4>
- [13] A. Verma, N. Sethi and N. Jai, "Beyond Hadoop for e-commerce Big Data Analysis through Amazon," 2018 International Conference on Advanced Computation and Telecommunication (ICACAT), Bhopal, India, 2018, pp. 1-4, doi: 10.1109/ICACAT.2018.8933660.
- [14] Awati, R., & Denman, J. (2022, January 12). *sharding*. SearchOracle. <https://www.techtarget.com/searchoracle/definition/sharding>
- [15] Kumar, A. (2023, June 30). *Coding Ninjas Studio*. Coding Ninjas. <https://www.codingninjas.com/studio/library/hbase-pros-cons>
- [16] Apache HBase – Apache HBase™ Home. (n.d.). <https://hbase.apache.org/>
- [17] Taylor, D. (2023, November 16). *HBase Advantages, Disadvantages & performance bottleneck*. Guru99. <https://www.guru99.com/hbase-limitations-advantage-problems.html>