**CCT College Dublin**

**Assessment Cover Page**

*To be provided separately as a word doc for students to include with every submission*

|  |  |
| --- | --- |
| **Module Title:** | *Advanced Data Analytics*  *Big Data Storage and Processing* |
| **Assessment Title:** | *MSC\_DA\_BD\_ADAv5* |
| **Lecturer Name:** | *David McQuaid*  *Muhammad Iqbal* |
| **Student Full Name:** | Jay Gilbert |
| **Student Number:** |  |
| **Assessment Due Date:** | *10/11/2023* |
| **Date of Submission:** |  |

**Declaration**

|  |
| --- |
| By submitting this assessment, I confirm that I have read the CCT policy on Academic Misconduct and understand the implications of submitting work that is not my own or does not appropriately reference material taken from a third party or other source. I declare it to be my own work and that all material from third parties has been appropriately referenced. I further confirm that this work has not previously been submitted for assessment by myself or someone else in CCT College Dublin or any other higher education institution. |

**Project Report: Analysis of Twitter Sentiment**

1. **Introduction**

**Twitter comprises of over 1,600,000 tweets retrieved through the Twitter API. This dataset contains valuable real time data, including tweet IDs, timestamps, user details, and the text of the tweets. This assignment unlocks insights from this dataset by exploring the sentiment trends in contains.**

**1.1 Objectives**

* **To load, process and store the ProjectTweets.csv dataset into a suitable database for analysis.**
* **To employ Apache Spark, a distributed data processing environment, to process the dataset efficiently.**
* **To conduct sentiment analysis using Text\blob library and to explore sentiment trends over time.**
* **To present the findings through an interactive and dynamic dashboard.**
  1. **Description of Dataset**

**The dataset used contains a vast collection of Twitter posts, know as “tweets”. Each tweet provides insights like the emotions, opinions, and sentiments of the users. The dataset comprises of:**

* **ids: The unique identifiers for each tweet**
* **date: Timestamps reflecting the date and time when each tweet was posted.**
* **Flag: Describes whether the tweet was associated with a specific query or labelled as “NO\_QUERY”.**
* **User: The usernames of the users who authored each tweet, offering insights into the source of content.**
* **Text: The text of the tweets.**

1. **Data Preparation Storage**

**2.1 The dataset used in this assignment is titled “ProjectTweets.csv”. The dataset was collected from Twitter and contains a substantial number of tweets related to a specific topic or keyword, making it suitable for analysis.**

**2.2 Data Preprocessing**

**Before conducting ny analysis, it was essential to preprocess the provided Twitter dataset. Data preprocessing included:**

* **Date Parsing: The original dataset provided dates in a non-standard format (e.g. “Sat May 16 23:58:44 UTC 2009”). To make the data useable to time series analysis, the dates were parsed into a datetime format. By doing this we could then analyse the data overtime more easily.**
* **Text cleaning: The text data in the tweets contained various inconsistencies, including letter casing and special characters. Cleaning was achieved by converting all text to lower case. This standardization helps with later analysis, as it would not be as case sensitive.**
  1. **Data Storage**

**To store and manage the dataset, SQLite database was employed, this is due to the fact it is lightweight and its self contained nature, making it suitable for this assignment. The dataset was saved as a table in the SQLite database, here we could interact with it easily and perform SQL queries for different analysis.**

* 1. **Distributed Data Processing with Apache Spark**

**Apache Spark offers several advantages in our data preprocessing framework:**

* **Parallel Processing: Apache Spark enabled parallel processing of the dataset, making it more suitable for operations like sentiment analysis and time series forecasting.**
* **Scalability: The project considered scalability by utilizing Spark’s ability to scale horizontally, accommodating growing data needs without significant architectural changes.**
* **Data Transformations: Spark provided a robust framework for efficient data transformations, including filtering, mapping, and aggregating data, particularly valuable for preparing sentiment analysis.**

**In summary, data storage and processing activities involved initial data preparation, SQLite database storage for effective data management, and strategic use of Apache Spark for efficient data transformations and advanced analysis.**

1. **Rationale and Justification**

**3.1 Choice of Programming Language**

**The choice of Python as the programming language was driven by its extensive libraries and packages, making it well-suited for data analysis and machine learning. Python supports more robust data manipulation and visualisation, aligning with the projects requirement ( Jones et al. 2001).**

**3.2 Data Wrangling**

**Data preprocessing choices, including date parsing and text cleaning, were made to ensure data quality and consistency, which is crucial for accurate sentiment analysis and forecasting.**

**3.3. Machine Learning Models and Algorithms**

**Selecting TextBlob library for sentiment analysis based on its simplicity and effectiveness for the specific task at hand (Loria, 2018). While more complex machine learning models could have been considered, TextBlob provided satisfactory results for this project.**

1. **Comparative analysis**

**A comparative analysis of at least two databases, Mongo DB and Cassandra, was conducted using the yahoo cloud serving benchmark(YCSB). The following metrics were recorded and analysed for comparison (Cooper et al., 2010).**

* **Read and write Throughput: Measuring the rate at which data can be read from and written to each database.**
* **Latency: Analysing the time it takes for a database to respond to red and write requests.**

1. **Analysis of sentiment change**

**An analysis of sentiment change over a selected time period was conducted. This analysis involved:**

* **Sentiment Trend Analysis: Tracking Sentiment trends over time through visual representation and statistical measures.**
* **Identification of Significant Shifts: Identifying any significant shifts I sentiment and providing explanations of these changes.**

1. **Sentiment Forecast**

**The project includes a sentiment forecast for the next 1 week, 1 month, and 3 months. This forecast is based on historical sentiment data and utilizes time series forecasting methods. Two forecasting methods, ARIMA and Prophet, were employed to provide robust predictions (Hyndman & Athanasopoulos, 2018; Taylor & Letham, 2018).**

1. **Presenting of Results**

**7.1 Design and Layout of the Dynamic Dashboard**

**The dynamic dashboard was designed following Tuffs principles for effective data generation it includes:**

* **Sentiment Trend Charts: Visual representations of sentiment trends over time, allowing users to track changes and patterns.**
* **Comparative Database Performance Metrics: Tables and charts displaying the results of the comparative analysis between MongoDB and Casandra.**
* **Sentiment Forecast: Predicted sentiment values for the next week, month, and three months, presented through interactive charts.**

**7.2 Components of the Dashboard**

**The dashboard allows users to interact with the data, customize their view, and gain insights easily. Users can explore sentiment trends, compare database performance, and visualise sentiment forecasts.**

* 1. **Functionality of the Dashboard**

**The dashboard allows users to interact with the data, customize their view, and gain insights easily. Users can explore sentiment trends, compare database performance, and visualize sentiment forecast.**

* 1. **Visual Aids and Captioning**

**Visual aids, including charts, graphs, and tables are used to visually represent the data and analysis results. Each visual elements accompanied by captions and explanations provide context and insights into the findings, ensuring clarity and understanding.**

**References:**

**Jones, E., Oliphant, T., Peterson, P., et al. (2001). SciPy: Open Source Scientific Tools for Python.**

**Loria, S. (2018). TextBlob: Simplified Text Processing. https://textblob.readthedocs.io/en/dev/**

**Cooper, B. F., Silberstein, A., Tam, E., Ramakrishnan, R., & Sears, R. (2010). Benchmarking Cloud Serving Systems with YCSB. In Proceedings of the 1st ACM Symposium on Cloud Computing (SoCC'10).**

**Hyndman, R. J., & Athanasopoulos, G. (2018). Forecasting: principles and practice. OTexts.**

**Taylor, S. J., & Letham, B. (2018). Forecasting at scale. The American Statistician, 72(1), 37-45.**