**Sbs23074 - MSc in Data Analytics-Integrated CA2 Semester 2**

Author: Osuade Adereti

e-mail: [sbs23074@student.cct.ie](mailto:sbs23074@student.cct.ie)

Student ID: SBS 23074

Address of repository: <https://github.com/osuade>

**RESEARCH PAPER ON ADVANCED DATA ANALYTICS AND BIG DATA STORAGE & PROCESSING**

Contents

[Research Aims 3](#_Toc149576978)

[Section 1- Background 4](#_Toc149576979)

[Section 2 – Literature Review 5](#_Toc149576980)

[Section3–Critical Analysis/Deliverables Discussion 7](#_Toc149576981)

[Section 4 –Research Methodology And Conclusion 10](#_Toc149576982)

[Section 6 – Acknowledgements 11](#_Toc149576983)

[Section 7 – References 12](#_Toc149576984)

## Research Aims

The aims of this research project are as follows:

- To demonstrate the utilisation of a distributed data processing environment (Apache Spark) to solve the given problem as part of analysis to be carried out.

-To discuss the storage and processing of big data using advanced data analytics techniques.

- To perform a comparative analysis of the capabilities of two data bases (YSCB and MongoDB).

-To carry out sentiment extraction of the given dataset (ProjectTweets).

-To explore 2 methods of time-series forecasting and design dynamic and interactive dashboard

-To carry out a semantic analysis of a project tweets gleaned from API.

## 

## Section 1- Background

Apache Spark is a fast, distributed computing technology. It employs horizontal clustering for fast and efficient computation. Apache Spark provides its computational framework on top of Hadoop MapReduce (MR) model, and it employs MR model for an extended computational framework subsuming interactive database queries and online processing through streaming. The most striking attribute of Spark is in-memory computation, which reduces the read/write latency of intermediate data during processing.

The Spark programming paradigm is very powerful and express a uniform programming model supporting the application development in multiple programming languages.

Apache Spark supports programming in Scala, Java, Python and even though there is no functional parity across all the programming languages supported.

Apache Spark is a cluster computing framework for large-scale data processing. It does not use MapReduce as an execution engine and it uses its own distributed runtime for executing work on a cluster. However, Spark has many parallels with MapReduce, in terms of both API and runtime.

Spark can handle different workloads such as batch program, iterative codes, interactive database queries and streaming data. The Spark is faster than Hadoop distributed processing, and it is attributed to the reduced amount of read/write tasks to the hard disk. It stores the intermediate value of the variables in memory during the execution.

The cluster management in Apache Spark is performed in three different ways viz. Standalone, Hadoop Yarn, and Mesos. Spark can access the data from local file system or any distributed file system like Hadoop Distributed File System (HDFS) (Vadlamani and Sk, 2017, p. 3).

Cassandra is a free and open source, distributed, wide-column store, NoSQL database management system designed to handle large amounts of data across many commodity servers, providing high availability with no single point of failure. Cassandra offers support for clusters spanning multiple datacenters, with asynchronous masterless replication allowing low latency operations for all clients.

Cassandra introduced the Cassandra Query Language (CQL) which is a simple interface for accessing Cassandra, as an alternative to the traditional Structured Query Language (SQL). CQL adds an abstraction layer that hides implementation details of this structure and provides native syntaxes for collections and other common encodings. Language drivers are available for Java, Python, C++ and others.

The keyspace in Cassandra is a namespace that defines data replication across nodes.

The Limitations to the use of Cassandra are as follows:

-Up to Cassandra 1.0, it was not row-level consistent which means that inserts and updates into table that affect the same row that are processed at approximately the same time may affect the non-key columns in inconsistent ways

-It is wide column store, thus it is a hybrid between a key-value and a tabular database management system. Tables may be created, dropped, and altered at run-time without blocking updates and queries.

-Cassandra cannot do joins or subqueries but it emphasizes denormalization through features like collections.

## Section 2 – Literature Review

From experiential observation, Apache Spark as big data analytics tool is much faster and gives accurate results than Hadoop (Kamal *et al.*, 2020, p. 24).

In ANN model development, the ML libraries are required, which are imported from Keras. One of the well-known libraries of Keras and behind it TensorFlow, is supported. Keras framework is much easier to use instead of directly using Tensorflow (Rahman *et al.*, 2021, p. 12).

Apache Spark MLlib offers fast, flexible, and scalable implementations of a variety of machine learning components, ranging from ensemble learning and principal component analysis (PCA) to optimization and clustering analysis. Apache Spark MLlib also offer options for distributed processing by parallel processing and support of big data tools that utilize distributed architectures. These criteria will decrease the processing time required and, at the same time, increase the time available to interpret analytics results. This becomes very important when the machine learning task has many predictions to calculate(Asseﬁ *et al.*, 2017, p. 11).

Big data technologies can be described as the tools or technologies that are used to efficiently process data that has been classified as big data. Some of the big data technologies include, Apache Hadoop, Apache Spark, Apache Cassandra, and Apache HBase.

Apache Spark has emerged as a widely used open-source engine. Spark is a fault-tolerant and general-purpose cluster computing system providing APIs in Java, Scala, Python, and R, along with an optimized engine that supports general execution graphs. Moreover, Spark is efficient at iterative computations and is thus well-suited for the development of large-scale machine learning applications (Nguyen *et al.*, no date, p. 1).

Apache Spark has huge potential to contribute to the Big Data-related business in the industry. Some of the common benefits of Apache Spark are its speed, multilingualism, advanced analytics, and open-source community.

Data mining (DM) deals with preparation of data obtained from various information sources (e.g. databases, text files, streams) as well as data modeling using a variety of techniques, depending on the goal that one wants to achieve (e.g. classification, clustering, regression, association rule mining, etc.). DM uses machine learning (ML) techniques to discover new knowledge from the existing information. DM is, nowadays, mostly considered within the wider scope of data science, which also encompasses statistics, big data techniques and data visualization. Data preparation is a vital step in the process of data analysis, and it includes data preprocessing and data manipulation (sometimes also called wrangling). Preprocessing aims at cleaning, integrating, transforming and reducing the original raw data so that it can become usable for data analysis, while wrangling transforms the preprocessed dataset into a data format that can be easily manipulated by the data modeling algorithms (Stancin and Jovic, 2019, p. 1).

Cassandra is a Java-based system that can be managed and monitored via Java Management Extensions (JMX). The JMX-compliant nodetool utility can be used to manage a Cassandra cluster (adding nodes to a ring, draining nodes, decommissioning nodes, and so on). Nodetool also offers a number of commands to return Cassandra metrics pertaining to disk usage, latency, compaction, garbage collection, and more.

Its main features include the following:

-Distributed – Every node in the cluster has the same role.

-Supports replication and multi data center replication- Cassandra is designed as a distributed system, for deployment of large numbers of nodes across multiple data centers.

-Scalability- Designed to have read and write throughput both increase linearly as new machines are added, with the aim of no downtime or interruption to applications.

-Fault-tolerant- Data is automatically replicated to multiple nodes for fault-tolerance. Replication across multiple data centers is supported. Failed nodes can be replaced with no downtime.

-Tunable consistency- Cassandra is typically classified as an AP system, meaning that availability and partition tolerance are generally considered to be more important than consistency in Cassandra.

-MapReduce support- Cassandra has Hadoop integration, with MapReduce support. There is support also for Apache Pig and Apache Hive

-Query Language- Cassandra introduced the Cassandra Query Language (CQL). CQL is a simple interface for accessing Cassandra, as an alternative to the traditional Structured Query Language (SQL).

-Eventual consistency- Cassandra manages eventual consistency of reads, upserts and deletes through Tombstones.

Based on the above, Cassandra was selected for processing and storage of the ProjectTweets for proper analysis using pyspark with jupyter notebook for python programming. The insights gained from this analysis were invaluable to the stakeholders and peers as per the justifications and documentations done on it in this report.

## Section3–Critical Analysis/Deliverables Discussion

## 3.1 - Apache Cassandra data storage was used for processing carried out in this project for both preparation and processing of the data in project tweets in spark environment as attached in the screencast using Pyspark on Ubuntu on my virtual box machine after installing Cassandra and following these steps:

-Start the distributed file system using Ubuntu software

-Start -dfs.sh

-Run Cassandra

-hduser@muhammad-vm:/$ cd /usr/local/cassandra

- hduser@muhammad-vm:/$ cd /usr/local/cassandra $ bin/cassandra -p cassandra. Pid

-Create keyspace on Cassandra

-hduser@muhammad-vm:~$ cd/usr/local/cassandra/

-hduser@muhammad-vm:/usr/local/cassandra $ bin/cqlsh

-Create keyspace Tweets with replication = {‘class’: ‘simplestrategy’, ‘replication\_factor’:1}

-cqlsh > use tweets;

-cqlsh > create table tweets (daytimestamp, scoredecimal, primary key(day))

-run pyspark using jarfile for mysql and connector for Cassandra

3.2 – The rationale and justification for using Apache Cassandra data processing storage system is because of its unique characteristics which include the following:

-Apache Cassandra is an open source, distributed, decentralized, elastically scalable, highly available, fault-tolerant, tuneably consistent, column-oriented database that bases its distribution design on Amazon’s Dynamo and its data model on Google’s Big table.

-Cassandra’s interface allows it to be accessed from a variety of languages like Python, Scala, C++, and Ruby.

-It also performs blazingly fast writes, it can store hundreds of terabytes of data, and it is decentralized and symmetrical, so the failure point is eliminated.

-It provides schema-free data models and is highly available.

-In terms of NOSQL Cassandra handles multi-data awareness better than others.

-It differs greatly from a relational database management system in many ways.

Python is a programming language of choice because it is open source community which is flexible and can work easily with pyspark. Python is a popular and versatile programming language which has become a popular choice among data scientists for its ease of use, extensive libraries, and flexibility. Python provide and efficient and streamlined approach to handing complex data structure and extracts insights.

Data wrangling are carried out easily using python to do the missing value and processing of data because of its flexibility. Appropriate Exploratory Data Analysis (EDA) was carried out on the dataset by inspecting and visualizing the data to understand its characteristics better, uncover the patterns, relationships and insights, and identify potential issues or abnormalities before building reliable models and making sound decisions after cleaning the data and checking for any outliers and null values. Python, Seaborn and pandas are used to perform EDA on the dataset.

The method used to prepare the dataset for Machine Learning (ML) is to use Pandas to load the dataset. Pandas is a fast, powerful, flexible and easy to use open-source data analysis and manipulation tool thru data frame using the read\_csv. The head and tail of the dataset must be examined to know the data types and check for null values before describing the data frame to know the count, mean, standard deviation, minimum, percentiles and maximum values of the dataset. Other methods of preparing data for Machine Learning (ML) are feature engineering, log transformation and feature scaling or normalisation.

## 3.3 – Comparative Analysis of two databases (YCSB and MongoDB)

## YCSB is an open-source specification and software package for benchmarking SQL and NoSQL database management solutions’ relative performance with the data model that functions as a key-value store which has one entity (User) with ten variables by default.

YSCB also provided workloads to determine how the benchmark should be run for comparative basis which include the amount of fields per record (field count) or the proportion of read, write, and update operations to complete.

Most NoSQL databases make trade-offs like optimising for reads vs writes, latency vs durability, and synchronous vs asynchronous replication, among others.

The performance of databases can be compared under these three major headings:

-Data Models- Cassandra and Hbase used BigTable model while MongoDB and CouchDB used Document model

-Design Options-Read or write optimized, Synchronous or Asynchronous replication, Latency or Durability and Data partitioning (row-based or column-based storage)

-Workloads Evaluation – YCSB has five predefined workloads (A to E) which are:

A-Update heavy

B-Read heavy

C-Read only

D-Read latest

E-Short ranges

YCSB Statistics Report (which are repeatable and can be used to target against different systems for comparisons) include the following:

-Percentile latency e.g 95th percentile and 99th percentile

-Histogram buckets

-Time series

However, MongoDB is a relatively new breed of data of database that has no concept of tables, schemas, SQL, or rows and there is no need for time-consuming configuration or set-up. It is a powerful, flexible, and scalable general-purpose database and user-friendly.

MongoDB is a document-oriented database, not a relational one. It allows complicated hierarchical relationships to be represented with just a single record by allowing embedded documents and arrays.

Major characteristics of MongoDB include :

-It is a general-purpose database, which means that in addition to creating, reading, updating and deleting data, it has most the functionality you would from a database management system, as we as a few unique characteristics.

-Indexing: MongoDB supports both generic and unique secondary indexes, as well as compound, geographic, and full-text indexing

-Aggregation: MongoDB has an aggregation system that is built on data processing pipelines.

-Special Collection and Index Types-It has time-to-live (TTL) collections for data that should expire after a given amount of time, such as sessions, and fixed-size (capped) collections for data that should be kept for a long time, such as logs.

-File Storage-It supports an easy-to-use protocol for storing large files and file metadata.

MongoDB Vs Relational DBMS can be compared as the following:

* Collections vs Table
* Document vs Row
* Field vs Column
* Schema-less vs Schema-oriented
* Example: Mongo Document

Key Components of MongoDB Data storage are:

-\_id: This is a field required in every MongoDB document which represents a unique value in it.

-Collection: A collection exists within a single database.

-Database: This is a container for collections like in RDMS wherein it is a container for tables. Each database gets its own set of files on the file system. A MongoDB server can store multiple databases.

-Document: A record in a MongoDB collection is basically called a document which consists of field name and values.

-Field: A name-value pair in a document. A document has zero or more fields which are analogous to columns in relational databases.

## 3.4 – Analysis of any change of sentiment that occurs over time period selected

This is demonstrated on jupyter notebook attached to this report.

## 3.5 – Forecasting Time Series And Results Presentation

This is demonstrated on jupyter notebook attached to this report.

## Section 4 –Research Methodology And Conclusions

The methodology adopted for the analysis of these datasets (tweets) provided here include the following:

-The ProjectTweet.csv file was read after importing the necessary libraries.

-Check query values to see duplications and dropping them

-The dates were formatted in an acceptable manner

-Hastags were checked to flatten the list

-NLTK was imported and stop words

-The texts were cleaned

-Vader was used to carry out sentiment analysis of the tweets

-Tweet information were checked to see null values in time series

-Tweet 2-4 were selected and grouped into hourly average, daily average, minute average and second.

At the end, it was noticed that aggregate by day has the least percentage of null and thus, it was used to insert missing values. The aggregate by second has the highest of null.

## Section 6 – Acknowledgements

Everyone has been a major source of inspiration, motivation, and support for this course.

Warm appreciation to my family and friends who encouraged me to do this programme to enhance my knowledge by taking up this new challenge.

This research work has provided me with a good baseline understanding of Advanced Data Analytics and key technologies in Big Data Storage and Processing and I am ready to tackle the challenging subject in earnest by progressing further on the research work to build cognitive systems.

Finally, I would like to express my gratitude to Springboard and CCT Management for providing the learning platform to upskill my knowledge in this field of Data Analytics with my fellow colleagues.

## Section 7 – References

Asseﬁ, M. *et al.* (2017) ‘Big Data Machine Learning using Apache Spark MLlib’.

Gupta, A. *et al.* (2017) ‘A Big Data Analysis Framework Using Apache Spark and Deep Learning’. arXiv. Available at: http://arxiv.org/abs/1711.09279 (Accessed: 25 September 2023).

Kamal, S. *et al.* (2020) ‘Traditional machine learning and big data analytics in virtual screening: a comparative study’, *International Journal of Advanced Computer Research*, 10, pp. 72–88. Available at: https://doi.org/10.19101/IJACR.2019.940150.

Khan, S., Liu, X. and Alam, M. (no date) ‘A Spark ML – driven preprocessing approach for deep learning-based scholarly data applications’.

Nguyen, H.H. *et al.* (no date) ‘Apache Spark for Data Science’.

Rahman, Md.A. *et al.* (2021) ‘A smart method for spark using neural network for big data’, *International Journal of Electrical and Computer Engineering (IJECE)*, 11(3), p. 2525. Available at: https://doi.org/10.11591/ijece.v11i3.pp2525-2534.

Shyam R. *et al.* (2015) ‘Apache Spark a Big Data Analytics Platform for Smart Grid’, *Procedia Technology*, 21, pp. 171–178. Available at: https://doi.org/10.1016/j.protcy.2015.10.085.

Vadlamani, R. and Sk, K. (2017) *Big Data Analytics Enabled Smart Financial Services: Opportunities and Challenges*, p. 39. Available at: https://doi.org/10.1007/978-3-319-72413-3\_2.