City based Taxi Trips Data Analytics based on Hadoop and MapReduce for Online prediction of Trip time

MSc in Data Analytics
National College of Ireland
Dublin, Ireland

***Abstract*— In big cities like Chicago, taxis are in demand. In the revolutionized modern world, where 'grab a cab' is the upcoming market trend in most cities, taxis serve as the most preferable travel medium. Everyday large amount of data is being generated by passengers travelling through taxi. This data can be analyzed to understand the industry trends. We consider various factors such as the trip duration, trip miles, pick-up location, dropoff location, fare, tips, tolls, extras, total trip rate, payment type, company and so on. This vast amount of data available at Google Cloud platform is well suited for Big Data ecosystem for analysis. Hadoop Distributed framework is used for processing these parameters. The dataset is stored in Hadoop Distributed File System (HDFS) on a Linux system for parallel processing of data. MapReduce Techniques are used for and visualized using Excel.**

Keywords*— *Big Data, MapReduce, Hadoop Distributed File System

I. INTRODUCTION

Taxis offer a reliable and convenient way to get around the city, and are easy to hail at most locations in the loop and at airports. Analysing taxi data not only offer the best services to the passengers, but also tend to upend the harmful traffic congestion patterns, road accidents and unwanted consumption of fuels. Everyday a huge amount of data is generated and stored in big data environment. Big data environment leverages storing and analysis of huge amount of data. In this project, Chicago's taxi trips dataset is being used with the objective to store it on Hadoop and mine using MapReduce techniques and visualize the data for analysis.

In metropolitan cities, taxis serve as the time-consuming and safe mode to the customers. Large amount of data is being generated everyday when passengers make taxi trips. This data can be explored and analysed to understand industry trends and market-analysis. Chicago's taxi trips dataset is best suited for this project as it is large in size and contains many variables which can be used for analysis using Hadoop distributed environment. Chicago's taxi trips dataset is exported from Google's cloud platform which includes taxi trips from 2013 to 2016. For information purpose, the dataset is available on Google's cloud platform and can be used to explore and analyse. To explore the Chicago's taxi trips dataset, the dataset is stored in Hadoop Distributed File System and MapReduce technique is used for processing to answer the following research questions:

1. How does trip duration affect the total trip rates lasting less than 60 min?
2. Which drop-off areas charge the highest tip rate?
3. What are the maximum, minimum and average fares for rides lasting 20 min or more?

II. RELATED WORK

Chicago's taxi trips dataset is open publicly and is analysed by researchers to extract useful information from the data. [1] used spatiotemporal analysis and study of predictability for the analysis of taxi-GPS traces acquired in the city of Lisbon, Portugal to better understand urban mobility. In order to improve taxi profit, taxi traces were analyzed to identify the relationships between pick-up and drop-off locations. The scenario between taxi services i.e. what happens between the latest drop-off and next pick-up was analysed. These traces helped to explore the value of Points of Interest in analysis of taxi flow. [2] uses a dynamic equilibrium model of meeting frictions to quantify the impact of the imposed policies on medallion prices. Study reveals an analysis on the process that rules the meetings between passengers and taxicabs in New York City. In [3], study provides insights that reveal travel data patterns and city structures, which could potentially aid in developing urban transportation policies. On the basis of taxi-trip data, spatially embedded networks were built. These models were based on intra-city spatial interactions and the analysis used network science methods. To identify sub-regional structures, the community detection method is applied and to examine the properties of sub-regions, other network measures are used.

New opportunities arisen for data-driven analysis with increasing volumes of urban data can improve the lives of citizens via decision-making skills and other rules. In [4], study aims at focusing important urban data set: taxi trips. From economic activity to human behavior and mobility patterns, taxis are referred to as valuable sensors. Taxi trips data provide anomalous insights into many different aspects of city life. However, analysing this data presents many challenges. The data are complex, contain demographic and temporal factors in addition to multiple variables associated with each trip. Also, it is hard to explore data and perform

comparative analysis.(eg. Compare different regions over time). This problem was identified due to the size of the data there are on average 500,000 taxi trips in NYC. Besides analytics queries, a new model was proposed, which, supports origin-to-destination queries to enable the study of local mobility. A wide range of spatio-temporal queries are elaborated in [4]. It is flexible where not only queries can be composed but different aggregations and visualizations can be applied thereby allowing users to explore and compare results. Such scalable system offers interactive response times, detailed-level understanding and hidden details to generate visualizations for larger results. [5] models a taxi market in which entry and fares are regulated. Using data from meter inspections of New York City taxicabs, the model is estimated. Rigid relationship between vacant taxis and demand for taxi service are highlighted in this paper. The effects of the policy changes such as increase in the regulated fare and in the number of taxis medallions are evaluated by the estimates of the parameters of the model.

The literature reveals that these researchers did not use Hadoop and MapReduce for analysis of data. Such huge volume of datasets offers challenging needs to analyse them. This can be done by MapReduce processing on Hadoop framework by efficient and quick parallel execution of tasks taking place at different nodes. Further sections in this paper discusses about methodology which includes data processing, results, conclusion and future work.

III. METHODOLOGY

A. *Dataset Description*

The Chicago taxi trips dataset contains 23 attributes and 92,097,483 instances. The dataset is exported from Google's cloud platform. The dataset link is as follows: https://bigquery.cloud.google.com/table/bigquery-public-data:chicago_taxi_trips.taxi_trips?tab=preview&pli=1.
This dataset was chosen as a purpose to perform analytical processing and achieve the solution to the framed research question. Initially, the dataset is pre-processed using Pig scripting. The processed attributes are perfect to study the insights of the data. The dataset contains attributes like taxi id, trip start timestamp, trip end timestamp, trip duration, trip miles, pick area, drop-off area, fare, tips, tolls, extras, trip total, payment type and company. The dataset is analysed based on these characteristics. The objective is to process the output analytically using MapReduce task in Hadoop environment. To achieve the objective, operations are performed on the attributes trip duration, total trip rates, drop-off areas, tip rate and fares.

taxi_id	trip_start_timestamp	trip_end_timestamp	trip_seconds	trip_miles
85	13-01-2016 06:15	13-01-2016 06:15	180	0.4
2776	22-01-2016 09:30	22-01-2016 09:45	240	0.7
4237	23-01-2016 17:30	23-01-2016 17:30	480	1.1
5710	14-01-2016 05:45	14-01-2016 06:00	480	2.71
1987	08-01-2016 18:15	08-01-2016 18:45	1080	6.2
4986	14-01-2016 04:30	14-01-2016 05:00	1500	18.4
6400	26-01-2016 04:15	26-01-2016 04:15	60	0.2
7418	22-01-2016 11:30	22-01-2016 11:45	180	0
1078	25-01-2016 09:00	25-01-2016 09:00	480	1.3
6641	06-01-2016 23:15	06-01-2016 23:30	420	0
920	13-01-2016 18:30	13-01-2016 19:00	1380	5.1

Fig. 1. Schema of the Dataset

B. *Data Processing*

The following steps are followed for processing the data:

1. **Loading data using Pig Scripting**

   - Chicago's taxi trips dataset downloaded from Google's Cloud platform website is stored on the local desk. Pig scripting language is used for ETL process after storing the data on local disk. Data is processed in Hadoop environment using a complete package known as Apache Pig Latin scripting. Pig script is easy to use and understand which made it an ideal choice for ETL processing for dataset used in the project.

2. **Cleaning data using Pig Scripting**

   - For cleaning the data, null values were removed from the dataset using pig scripting and the results of the pig script were stored on the local disk.

3. **Moving data to HBase**

   - The results of the Pig script were stored in HBase after creating appropriate schema for the dataset. Apache HBase, a non-relational database is chosen as it works on top of Hadoop and HDFS. HBase is best suited for this project as it was easy to upload the huge Chicago's taxi trips dataset. HBase is an open source which makes it an ideal choice for this project to store the data.

4. **Reading the dataset in HBase**

   - The dataset was then read from HBase for MapReduce processing. Three MapReduce processing tasks are done on the dataset.

5. **MapReduce Processing**

   - Java is used for MapReduce processing in Eclipse environment.

6. MySQL Database for storage of output

- The output of the MapReduce task is stored in MySQL. MySQL is chosen because it is fast, easy to use and reliable for large databases.
- For visualizing the data, the output is stored back on the local disk.

7. Visualization using MS Excel

- Further, MS Excel is used for visualization purpose.

The steps followed in the project for processing the data are as shown in below Fig. 2:

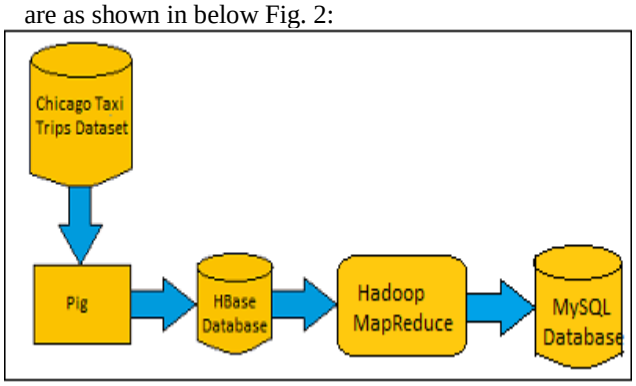

Fig. 2. Data Processing steps which shows HBase for storing the data, perform MapReduce job in Hadoop and further storing output in MySQL database.

For processing the data, three MapReduce tasks are performed.

1. **MapReduce 1**
   Mapper 1:
   Input – taxi trip duration, total trip rate

   Output:
   Key – trip duration
   Value – total trip rate

   Reducer 1 – trip duration, total trip rate
2. **MapReduce 2**
   Mapper 2:
   Input – drop-off areas, tips

   Output:
   Key – drop-off areas
   Value – tips
   Reducer 2 – drop-off areas, tips
3. **MapReduce 3**
   Mapper 3:
   Input – Fares, trip duration

   Output:
   Key – trip duration
   Value – fares
   Reducer 3 – trip duration, fares

Java programming is used in Eclipse environment to access the data stored in Hadoop. Java is used for MapReduce processing. This includes creating classes for Map. The Mapper's job is to process the input data. The input data in the form of file is passed to the mapper function. The mapper processes data and creates small chunks of data. The reduce stage is a combination of Shuffle and Reduce. The job of the Reducer is to process data that comes from the mapper. Pre build hadoop MapReduce libraries are used by the reducer to perform its job. Mapper processes data in the form Key, Value pair and gives output as a Key, Value pair to the reducer. A new set of output will be produced in after processing using MapReduce.

During MapReduce processing, Hadoop sends the Map and Reduce tasks to the appropriate servers in the cluster. The computation tasks takes place in parallel across different nodes for handling large dataset of Chicago taxi trips.
Eclipse environment is configured with Hadoop in-built libraries for MapReduce processing on the dataset stored in HDFS. For performing MapReduce task, three classes are created viz., Driver, Mapper and Reducer in Java. Hadoop's libraries viz., "org.apache.hadoop.mapreduce.Job", "apache.hadoop.mapreduce.Mapper", "apache.hadoop.mapreduce.Reducer" are imported to create new job for processing MapReduce task for the Driver class. Another library "apache.hadoop.mapreduce.Reducer" is imported for the Mapper class to process the data and perform mapping job. To perform reduce job, "apache.hadoop.mapreduce.Reducer" is imported for Reducer class.

In MapReduce 1, the input key is attribute trip duration and total trip rates as value. This key value pair is passed to the reducer. The reducer displays output for total trip rates lasting less than 60 min.
In MapReduce 2, the input key is attribute drop-off areas and tips as value. This key value pair is passed to reducer. The reducer shows the areas with highest tip rate from Chicago taxi trips dataset.
In MapReduce 3, the input key is attribute trip duration and fare as value. This key value pair is passed to reducer. The reducer shows the areas with highest tip rate from Chicago taxi trips dataset. The output for each of these MapReduce tasks is stored in MySQL. They are then moved to local file system for visualization and analysis.

IV. RESULTS

Tableau visualization software is used for visualizing the output of the MapReduce job.
Output of the MapReduce task 1 gives the trip duration by total trip rates from the year 2013 to 2016. The X-axis represents the total trip rates in US dollar and the Y-axis denotes the trip duration in seconds for Chicago's taxi trips.

Histogram shown in figure 3 illustrates how the trip duration affects the total trip rates lasting less than 60 min.

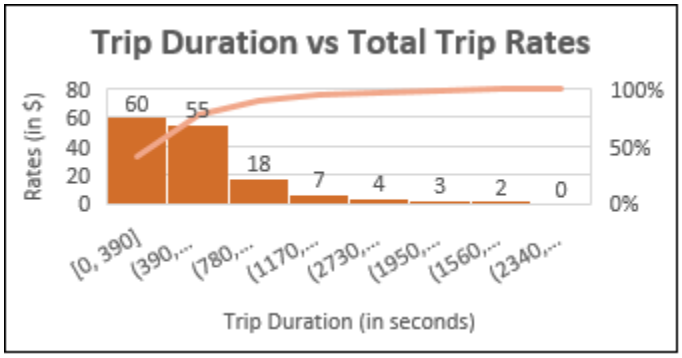

Fig. 3. Taxi Trip Duration vs Total Trip Rates

The above histogram shows the distribution for sample number of records from the Chicago taxi trip dataset. The trends highlight decreasing order of distribution. 60 records lie in the duration 0 to 390 seconds which is the maximum. The lowest was observed 1560 to 1950 with only 2 records.

Output of the MapReduce task 2 gives the drop-off areas with the highest tip rate. The X-axis represents the drop-off community areas and Y-axis indicates the tip charged by these areas. Scatter in figure 4 shows the distribution of highest tip rates in the drop-off community area.

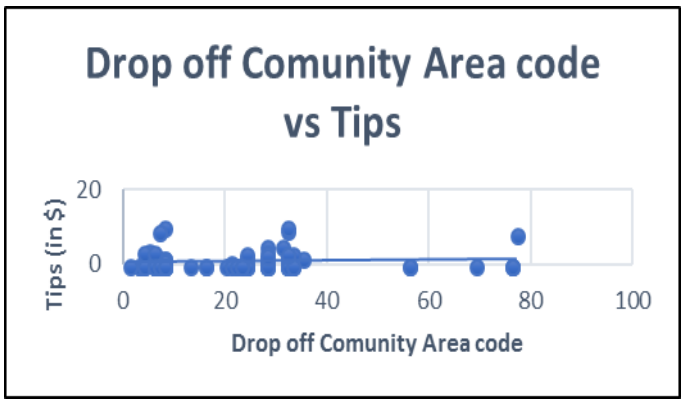

Fig. 4. Drop-off Community Area vs Trip

The maximum tips with the rates 8-10 dollars in the drop-off areas with the code range 10-30. A fair tip was offered by the drop-off areas with the codes 0-10 of about 1-4 dollars. 1-5 dollars were charged by the drop-off areas within the code range 25-35. Maximum drop-off areas did not charge any tip.

Below scatter plot shown in figure 4 indicates distribution of taxi fares in dollars against trip duration in seconds.

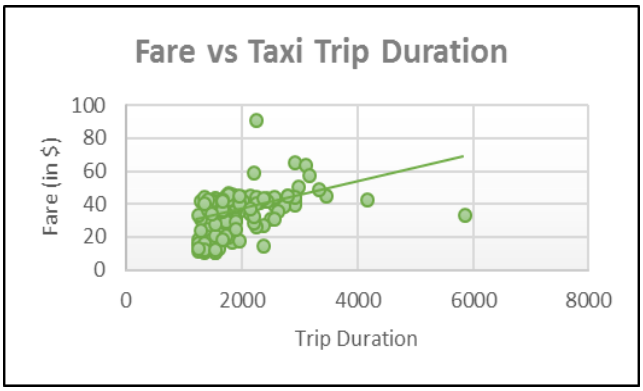

Fig. 4. Fare vs Taxi Trip Duration

Output of MapReduce task 2 gives taxi fares against trip duration lasting 20 minutes or more. The distribution of average fares is observed with trip duration lasting 20-30 minutes. This fare price lies in the range 10 to 50 dollars. Maximum rate is about 90 dollars lasting over 33 minutes. Rates charged in the range 60-70 dollars is recorded for trips lasting over 30 minutes. The fare price recorded is about 35 dollars for trip lasting about 60 minutes.

V. CONCLUSION

Chicago's taxi trips dataset is analysed using MapReduce job on Hadoop platform. Three analytical tasks were performed and the output is visualized using MS Excel. A decreasing trend was observed for the number of records in the dataset which affected total trip rates lasting less than 60 minutes. Further the project analysed fair distribution for tip rates charged by drop-off areas. Maximum, minimum and average fares is analysed for rides lasting 20 minutes or more. Many peculiar characteristics to understand different insights can be processed via effective Hadoop platform using MapReduce processing.

VI. FUTURE WORK

For faster processing of huge datasets in businesses and decision makings, Big Data Analytics are on the "Jagged Edge". Hadoop, which is regarded as the most cost effective framework can support any volume of data and scalable to any number of servers in a cluster. This combination of distributed filesystem, HDFS combined with a flexible processing engine, MapReduce form the basis to gain interest in the market. The amount of data is growing day by day and analysing it is all what that matters right now. For faster processing of Big Data Analytics, Hadoop using MapReduce is on the boom. A better alternative for MapReduce processing is Apache Spark.

REFERENCES

[1] Veloso, M., Phithakkitnukoon, S. and Bento, C., 2011, November. Sensing urban mobility with taxi flow. In Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Location-Based Social Networks (pp. 41-44). ACM.

[2] Lagos, R., 2003. An analysis of the market for taxicab rides in New York City. International Economic Review, 44(2), pp.423-434.

[3] Liu, X., Gong, L., Gong, Y. and Liu, Y., 2015. Revealing travel patterns and city structure with taxi trip data. Journal of Transport Geography, 43, pp.78-90.

[4] Ferreira, N., Poco, J., Vo, H.T., Freire, J. and Silva, C.T., 2013. Visual exploration of big spatio-temporal urban data: A study of new york city taxi trips. IEEE Transactions on Visualization and Computer Graphics, 19(12), pp.2149-2158.

[5] Flores-Guri, D., 2003. An economic analysis of regulated taxicab markets. Review of Industrial Organization, 23(3-4), pp.255-266.M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.