Indonesian Journal of Electrical Engineering and Informatics (IJEEI)
Vol. 5, No. 2, June 2017, pp. 174~184
ISSN: 2089-3272, DOI: 10.11591/ijeei.v5i2.297

Analysing Transportation Data with Open Source Big Data Analytic Tools

Y. Beeharry[1*], T. P. Fowdur[1], V. Hurbungs[2], V. Bassoo[1], V. Ramnarain-Seetohul[3]
[1,2,4]Department of Electrical and Electronic Engineering, University of Mauritius, Réduit, Mauritius
[3]Department of Software and Information Systems, University of Mauritius, Réduit, Mauritius
[5]Department of Information & Communication Technology, University of Mauritius, Réduit, Mauritius
e-mail: y.beeharry@uom.ac.mu

Abstract

Big data analytics allows a vast amount of structured and unstructured data to be effectively processed so that correlations, hidden patterns, and other useful information can be mined from the data. Several open source big data analytic tools that can perform tasks such as dimensionality reduction, feature extraction, transformation, optimization, are now available. One interesting area where such tools can provide effective solutions is transportation. Big data analytics can be used to efficiently manage transport infrastructure assets such as roads, airports, bus stations or ports. In this paper an overview of two open source big data analytic tools is first provided followed by a simple demonstration of application of these tools on transport dataset.

Keywords*: Big Data, analytic tools, transportation*

1. Introduction

Over the years, the traditional way of processing and analyzing data was to take data from operational systems such as Enterprise Resource Planning (ERP), Customer relationship management (CRM), or Supply Chain Management (SCM) systems and centralize the data in a Data Warehouse. This data was structured in nature and business intelligence tools enabled businesses to define key metrics and get answers to already known issues [1]. Businesses are now being overloaded with modern sources of information such as social networks, online media, sms, email, blogs and mobile activities.

With the urge to remain competitive, businesses can no longer rely on traditional methods of decision making and must therefore be able to process and analyse all possible sources of information to ensure business continuity [2]. This large set of data is referred to as Big Data which combines both structured and unstructured data as compared to traditional data frameworks. Big Data is characterized by 4 V's:

a. Volume: Very large data sets with both structured and unstructured data which can be in terms of Terabytes, Petabytes, etc…
b. Velocity: Speed at which data is coming / generated. For example: high speed data flow from IoT sensors, Twitter feeds, Facebook likes, among others.
c. Variety: Data comes from different sources. For example: smartphones, wearable devices, IoT devices and sensors, and other mobile devices.
d. Veracity: All data which are being captured are not of good quality. Part of this data may carry some level of uncertainty.

Some emerging big data applications are:

a. Healthcare: W. Raghupathi and V. Raghupathi [3] outlined an architectural framework and methodology that describes the potential and promise of big data analytics in healthcare. This framework would enable healthcare providers to obtain insight from their clinical and other data repositories, and make informed decisions. One such example would be to diagnose and treat patients in cost-effective ways by analyzing patient records, disease patterns, and faster development of vaccines. Additionally, D. Madhavi and B. V. Ramana [4] proposed a de-identified personal

Received April 28, 2016; Revised April 23, 2017; Accepted May 10, 2017

health care system which uses Map Reduce Pig querries which are required to be executed on the datasets for health care.

b. Manufacturing: J. Lee et al [5] highlight the trends of industrial big data environment and smart manufacturing. Key impact areas which they have identified are: machine health prediction, transparency and organization across production lines, reduced labour costs, and optimized machine maintenance.
c. Traffic management: Y. Lv et al [6] proposed a deep architecture model using autoencoders as building blocks to represent traffic flow features for prediction. Their experiments demonstrate that the proposed method for traffic flow prediction has better performance. The main objectives of traffic management are: better travel decisions, reduced traffic congestion and carbon emissions, and improved traffic flow.

Many organizations have the expertise and equipment for handling large quantities of structured data. However, the faster flows and increasing volumes of data, leaves them with the inability to "mine" the data and derive actionable intelligence in a timely way. Not only is the volume of this data growing too fast for traditional analytics, but the speed with which it arrives and the variety of data types necessitates new types of data processing and analytic solutions [7]. Big data doesn't always fit into neat tables of rows and columns. There are also many new data types, both structured and unstructured, that can be processed to yield insight into a business or condition [8]. Popular machine learning toolkits such as R [9] or Weka [10] were not built for these kinds of workloads. Although Weka has distributed implementations of some algorithms available, it is not on the same level as tools that were initially designed and built for terabyte-scale. Some of the open-source Big Data Analytics Tools are: Mahout [11, 12], MLlib [13, 14], H2O [15], SAMOA [16], and SparkR [17].

The remainder of this paper is organised as follows. Section II gives an overview of the open source analytical tools. Section III gives a detailed overview of related works. Section IV presents the application and testing of the two proposed tools. Finally Section V concludes the paper.

2. Big Data in Transportation

Researchers have developed a complete transportation decision-making system called the TransDec for the city of Los Angeles. The system acquires data from different sources in real time and the amount of data that arrives per minute is around 46 megabytes [18]. The system gathers traffic occupancy and volume data from more than 8,900 traffic loop detectors. Data from buses and train are collected; the data is detailed and contains GPS location updated every two minutes and delay information calculated by taking into account pre-defined schedules. Information from ramp meters that are located at the entrance of highways is also used. The system also accepts text format information about traffic incidents. All the data is then cleaned, reducing the input rate from 46 megabytes per minute to 25 megabytes per minute. Analytical techniques are applied to produce precise traffic patterns. TransDec can also predict clearance times and resulting traffic jams after accidents [19].

London is a growing city with a population of around 8.6 million and is expected to hit 10 million by 2030. Transport for London is using big data to keep London moving and plan for challenges arising from a growing city. The system gathers data from the 'Oyster' cards, GPS location of around 9,200 buses, 6,000 traffic lights and 1,400 cameras [19]. Transport for London uses big data tools to develop accurate travel patterns of customers across rail and bus networks. The information is then used by authorities to plan closures and diversions. The system also enables authorities to send targeted emails to customers providing them alternative routes thus minimizing the impact of scheduled and unscheduled changes [20].

In Sweden, the Stockholm train operators are big data analytics to predict train delays up to two hours before they arise. The traffic control centre can then decide to make additional train available to remedy the situation [21].

The Land Transport Authority (LTA) of Singapore is using big data to better serve their customers. LTA uses data generated from logins to their public WIFI system available in the MRT stations to produce a real time crowd heat map on each of their platforms. Additional trains are added to the network if needed [22].

3. Open Source Analytic Tools

The few open-source Big Data analytics tools mentioned (Mahout, MLlib, H2O, SAMOA, and SparkR), are all frameworks which are scalable and contain abstractions for machine learning algorithms on streaming data. The setup of these distributed frameworks is very time-consuming. In the above list, the easiest tools which can be used are H2O and SparkR with MLlib. H2O can be run on one local machine and has a Graphical User Interface (GUI) with all the necessary documentation on steps to follow for data processing and analysis. The Databricks community edition provides an online interface where users can create their own notebooks using the R language amongst others, using the Spark framework for real-time analytics. As such, the processes for transport data analysis with H2O and SparkR are presented in this work.

3.1. H2O

H2O is a fast scalable open source software for distributed in-memory predictive analytics, machine learning and deep learning. It is based on pure Java and Apache v2 Open Source. It provides simple deployment with a single jar and automatic cloud discovery. H2O allows data to be used without sampling and provides reliable predictions quicker. For this reason it is suitable for several organisations such as PayPal, Nielsen, Cisco, etc. [23].

H2O can support billions of data rows in-memory even if the cluster size is relatively small. This is possible by the use of sophisticated in-memory compression techniques. The H2O platform has its own built-in Flow web interface so as to make analytic workflows become user-friendly to users who do not have engineering background. It also includes interfaces for R, Python, Scala, Java, JSON and Coffeescript/JavaScript. The H2O platform was built alongside (and on top of) both Hadoop and Spark Clusters and is typically deployed within minutes [23-25].

Several common machine learning algorithms are supported by H2O. Examples include: Generalized Linear Modelling (GLM) such as linear regression, logistic regression, etc, Naive Bayes, principal components analysis, time series analysis, k-means clustering etc. Best-in-class algorithms such as Random Forest, Gradient Boosting and Deep Learning at scale are also implemented by H2O [23-25]. A typical architecture for H2O is shown in Figure 1 [24].

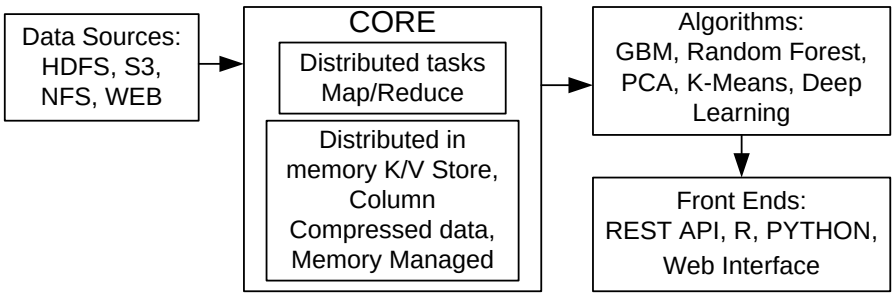

Figure 1. H2O Architecture

As observed in Figure 1, it supports: several data source, distributed memory and tasks, a range of algorithms and multiple front-ends [24].

3.2. SparkR

R is a common tool for building machine learning models. However, its effectiveness is constrained by the processing power of a single machine. It is now possible to handle complex machine learning problems with the power of clustered computers using a dedicated library called MLib which is provided by Apache Spark. Spark MLlib is an open source API that is part of the Apache Software Foundation. Spark DataFrames and MLlib provide tooling to make it easier to integrate existing workflows developed on tools such as R and Python, with Spark. For example, SparkR allows users to call MLlib algorithms using familiar R syntax [26]. Figure 2 illustrates the Spark ecosystem.

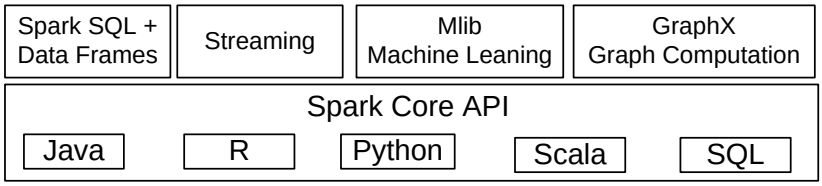

Figure 2. The Spark ecosystem

As observed in Figure 2, Spark supports programming platforms such as R, SQL, Python, Scala and Java. Additionally, it has several libraries which can provide functionalities such such as graph computations, stream data processing, and real-time interactive query processing in addition to machine learning [26].

MLlib provides distributes and fast implementations of common learning algorithms. Various linear models are available to address regression problems. To cater for classification problems, powerful algorithms such as Naïve Bayes, Random Forest, and Decision Tree are provided. For collaborative filtering, least squares with explicit and implicit feedback can be used. Unsupervised learning algorithms such as K-Means, and Principle Component Analysis (PCA) for dimensionality reduction are also part of MLlib. A number of low-level primitives and basic utilities for convex optimization, statistical analysis, feature extraction, and distributed linear algebra, [27] are also provided in the library.

4. Application and Testing

In this section, the open source tools: H2O, and SparkR on Databricks have been used to perform analytics on the transport related data obtained from [28]. A detailed explanation on the configurations, and coding are given in the following sub-sections. The machine learning algorithm used in H2O and SparkR on Databricks is: Generalised Linear Model (GLM).

4.1. H2O

The web-based user interface of H2O can be accessed by executing the jar file and accessing the specified url through the web-browser as explained in [25]. The dataset to be used can be imported using the ***importFiles*** option in the list on the homepage of H2O as shown in Figure 3.

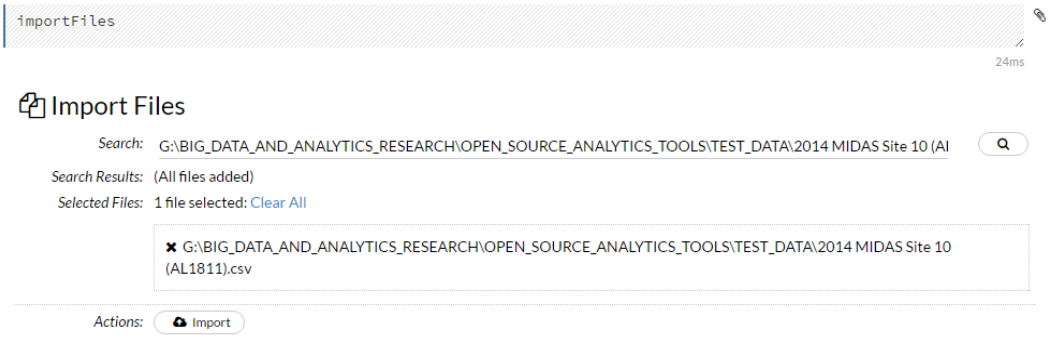

Figure 3. Import data set on the cluster to be processed in H2O

The data frame is then split into Training set and Test set. 75% of the data is used to train the model and 25% is reserved for testing purposes. Figure 4 shows the section where the frame is selected and the percentage splits specified.

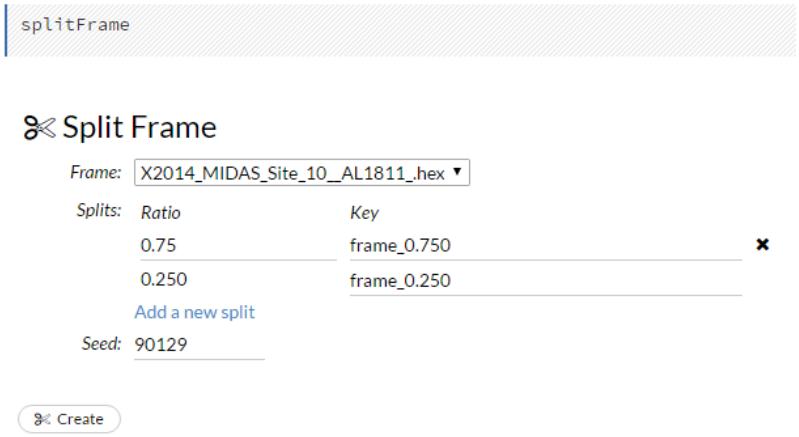

Figure 4. Split of Data Frame

4.1.1. GLM with H2O

The model is then built using the training data set as input. The Generalized Linear Model (GLM) is used with the "Speed Value" column selected as the response column. The "Local Date" and "Local Time" columns are ignored. These are shown in Figure 5. The equation being modelled is: Speed Value ~ Day Type ID + Total Carriageway Flow + Total Flow vehicles less than 5.2m + Total Flow vehicles 5.21m-6.6m + Total Flow vehicles 6.61m-11.6m + Total Flow vehicles above 11.6m.

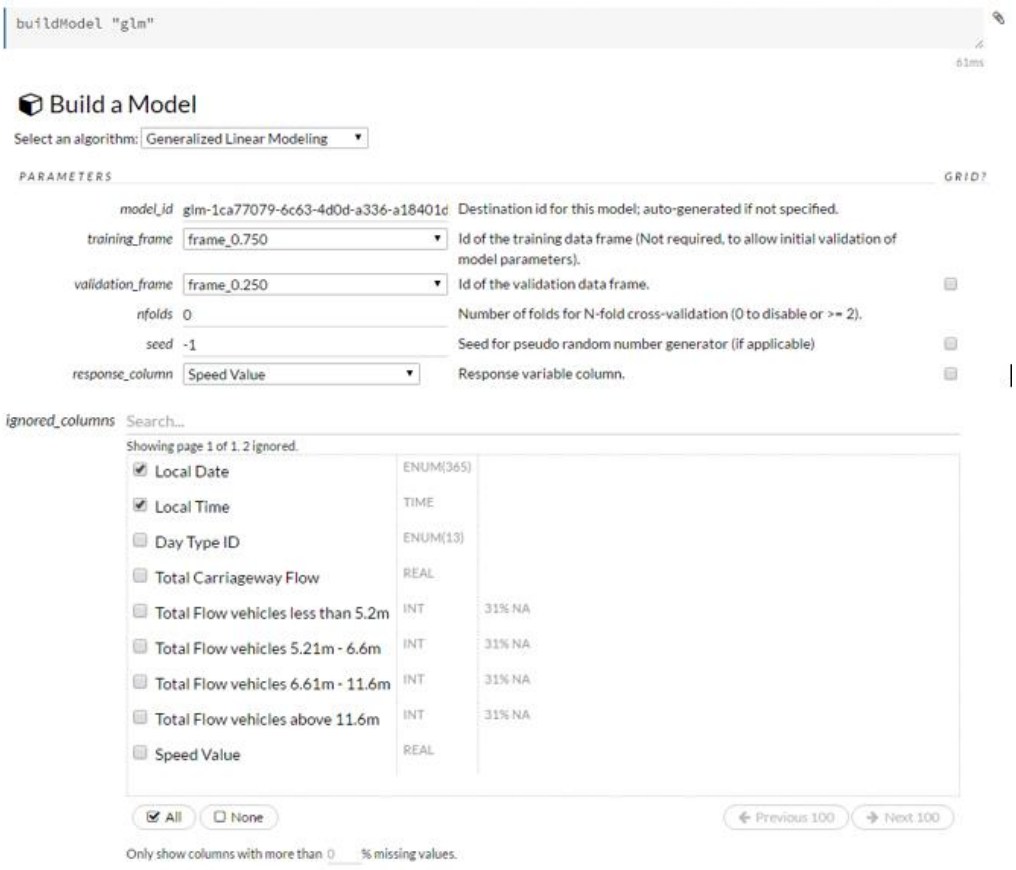

Figure 5. Building a GLM

The output metrics on the training model is shown in Figure 6.

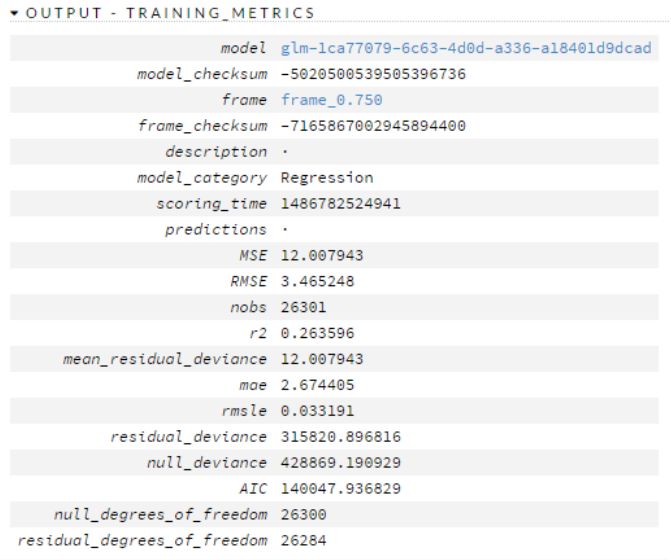

Figure 6. Output metrics for training model

With the training model obtained, prediction can be performed on the test set data. A comparison can then be performed on the predicted and already known “Speed Value”. Figure 7 shows the step where the model is used to predict the “Speed Value” for the test set.

Figure 7. Prediction using training model on test data set

The predicted “Speed Value” can be merged with the test data set and compared with that already known. Figure 8 shows part of the combine frame.

combined-prediction-68aa3ef9-c998-43b3-be69-89782b2532b3

DATA

Previous 20 Columns | Next 20 Columns

Row	predict	Local Date	Local Time	Day Type ID	Total Carriageway Flow	Total Flow vehicles less than 5.2m	Total Flow vehicles 5.21m – 6.6m	Total Flow vehicles 6.61m – 11.6m	Total Flow vehicles above 11.6m	Speed Value
1	106.5318	01/01/2014	34200000	14	63.5000	·	·	·	·	108.5600
2	106.0124	01/01/2014	31500000	14	32.5000	·	·	·	·	108.6200
3	105.9538	01/01/2014	30600000	14	29.0	·	·	·	·	110.6100
4	105.7779	01/01/2014	25200000	14	18.5000	·	·	·	·	106.9800
5	105.7444	01/01/2014	23400000	14	16.5000	·	·	·	·	107.1400
6	105.5224	01/01/2014	14400000	14	3.2500	·	·	·	·	108.1300
7	105.5601	01/01/2014	12600000	14	5.5000	·	·	·	·	112.0900
8	105.6271	01/01/2014	9000000	14	9.5000	·	·	·	·	108.9200
9	105.8197	01/01/2014	6300000	14	21.0	·	·	·	·	106.6200
10	106.5988	01/01/2014	67500000	14	67.5000	·	·	·	·	104.4600
11	106.0543	01/01/2014	77400000	14	35.0	·	·	·	·	104.7300
12	105.9705	01/01/2014	80100000	14	30.0	·	·	·	·	103.7100
13	105.7611	01/01/2014	84600000	14	17.5000	·	·	·	·	106.3300
14	105.7025	01/01/2014	85500000	14	14.0	·	·	·	·	108.8100

Figure 8. Part of combined data frame of test data set and predicted “Speed Value”

An analysis of the percentage difference between the predicted and already known “Speed Value” for the test data set can also be performed.

4.2. SparkR with Databricks

The dataset to be used can be imported using the ***Create Table*** option in the list on the homepage of Databricks as shown in Figure 9.

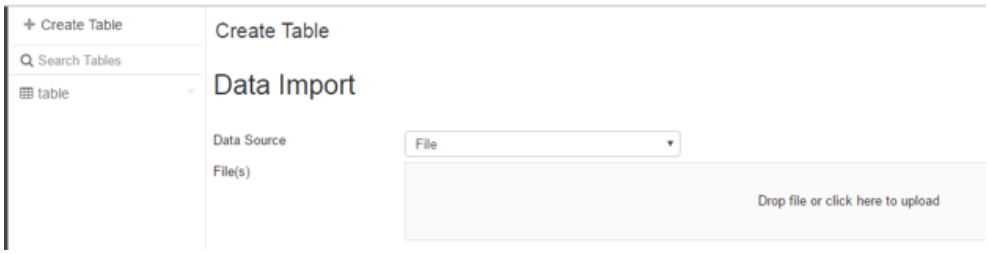

Figure 9. Import data set on the cluster to be processed in Databricks

The code to read the data into a data-frame for further processing is shown in Figure 10. The NA's are replaced by 0's in this case and the data-frame is then split into training (75%) and test (25%) sets as shown in Figure 11. The Generalized Linear Model (GLM) is built as shown in Figure 12. With the training model obtained, prediction can be performed on the test set data. A comparison can then be performed on the predicted and already known "Speed Value". Figure 13 shows the step where the model is used to predict the "Speed Value" for the test set. The predicted "Speed Value" can be merged with the test data set and compared with that already known. Figure 14 shows part of the combine frame.

```
> sparkDF <- read.df(sqlContext, source = "csv", path = "/FileStore/tables/d0d0l8ls1486873447321")
  #df = read.csv("/FileStore/tables/d0d0l8ls1486873447321/2014_MIDAS_Site_10__AL1811_-2878f.csv", header = TRUE)
  df = read.csv("/dbfs/FileStore/tables/d0d0l8ls1486873447321/2014_MIDAS_Site_10__AL1811_-2878f.csv", header = TRUE)
```

Figure 10. Read data into data-frame with SparkR

```
> df[is.na(df)] <- 0

Command took 0.02 seconds -- by yogesh536@hotmail.com at 2/12/2017, 2:28:36 PM on My Cluster

> smp_size <- floor(0.75 * nrow(df))
  df_train <- df[(1:smp_size), ]
  df_test <- df[-(1:smp_size), ]

Command took 0.01 seconds -- by yogesh536@hotmail.com at 2/12/2017, 2:28:38 PM on My Cluster

> df <- createDataFrame(sqlContext, df)
  df_train <- createDataFrame(sqlContext, df_train)
  df_test <- createDataFrame(sqlContext, df_test)
```

Figure 11. Replace NA's with 0's and split data set into training and test sets

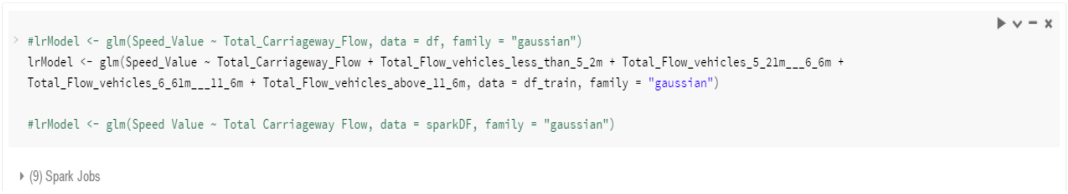

Figure 12. Generalized Linear Model with SparkR

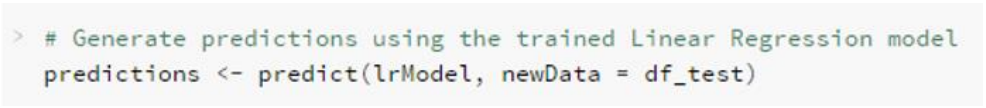

Figure 13. Prediction using training model on test data set with SparkR

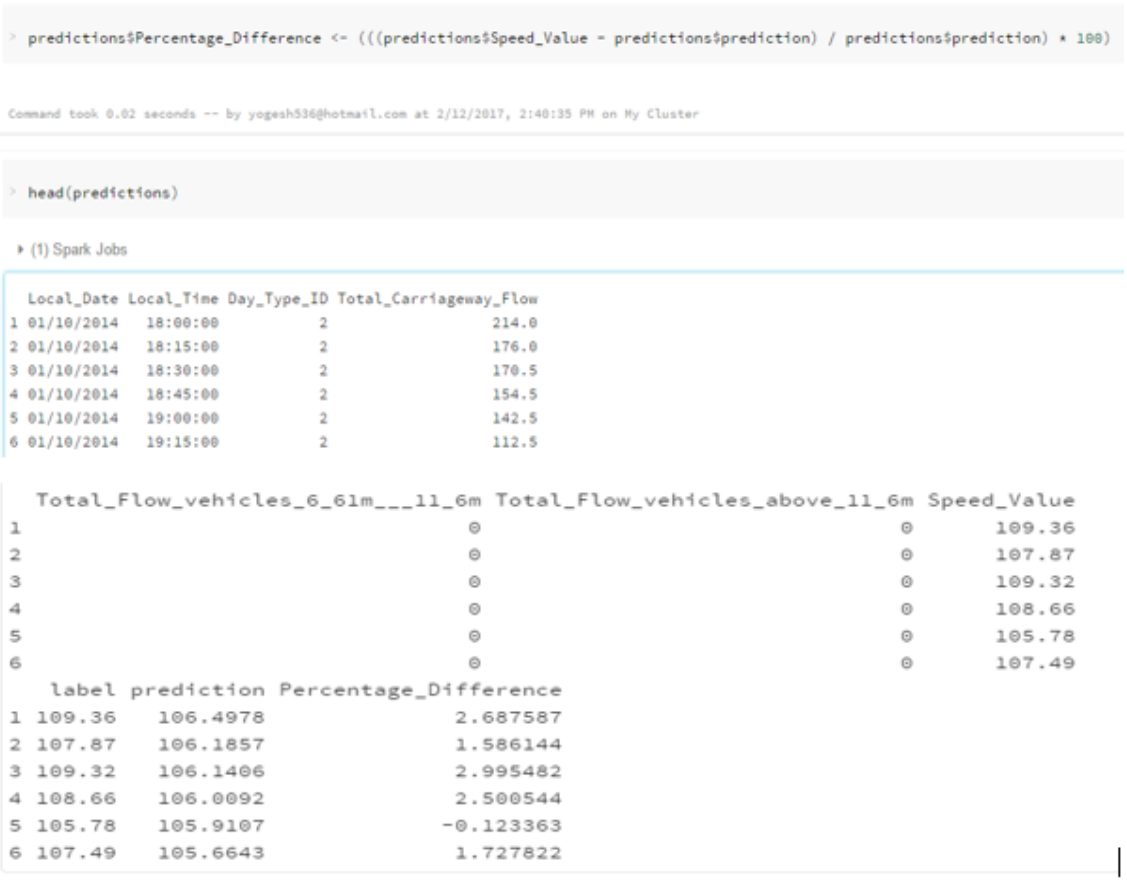

Figure 14. Part of combined data frame of test data set, predicted “Speed Value”, and the percentage difference between them

5. Conclusion

This paper exemplifies the use of open source big data analytical tools, to show how predictive modelling can be applied to data pertaining to transport. Only two big data analytical tools (H2O and SparkR) have been chosen for demonstration in this paper. H2O can be easily used by researchers in different fields with the basic knowledge of machine learning model training and testing using the GUI. SparkR on the other hand, eases the tasks of experts in statistics familiar with the R programming language and willing to venture in research works pertaining to Big Data Analytics. However, as future work, other analytical tools can be used on

different datasets, e.g., health data, manufacturing among others. Furthermore, these tools can be used with infrastructures such as Hadoop to handle massive datasets from several sources.

References

[1] Aptera. Big Data vs Traditional Approaches to Enterprise Reporting. 2015. [Online]. Available: Available: http://blog.apterainc.com/business-intelligence/big-data-vs-traditional-approaches-to-enterprise-reporting. [Accessed 24 February 2017].

[2] H Bagheri, AA Shaltooki. Big Data: challenges, opportunities and cloud based solutions. International Journal of Electrical and Computer Engineering. 2015; 5(2).

[3] W Raghupathi, V Raghupathi. Big Data Analytics in Health Care: Promise and Potential. Journal of Health Information Science and Systems. 2014; 2(3): 1 - 10.

[4] D Madhavi, BV Ramana. De-Identified Personal Health Care System Using Hadoop. International Journal of Electrical and Computer Engineering. 2015; 5(6).

[5] J Lee, HA Kao, S Yang. Service innovation and smart analytics for Industry 4.0 and big data environment. in Proceedings of the 6th CIRP Conference on Industrial Product-Service Systems, Cincinati, USA. 2014.

[6] Y Lv, Y Duan, W Kang, Z Li, FY Wang. Traffic Flow Prediction With Big Data: A Deep Learning Approach. IEEE Transactions on Intelligent Transportation Systems. 2015; 16(2): 865 - 873.

[7] S Landset, TM Khoshgoftaar, AN Ritcher, T Hasanin. A survey of open source tools for machine learning with big data in the Hadoop ecosystem. Journal of Big Data. 2015; 2(24): 1 - 36.

[8] P Russom. Executive Summary: Big Data Analytics. 2011. [Online]. Available: https://www.tableau.com/sites/default/files/whitepapers/tdwi_bpreport_q411_big_data_analytics_tableau.pdf. [Accessed 24 February 2017].

[9] The R Foundation. The R project for Statistical Computing. The R Foundation, 2017. [Online]. Available: https://www.r-project.org/. [Accessed 22 February 2017].

[10] IBM. IBM Big Data Platform. [Online]. Available: https://www-01.ibm.com/software/in/data/bigdata/enterprise.html. [Accessed 22 January 2017].

[11] Apache Mahout. What is Apache Mahout. 2014 - 2016. [Online]. Available: http://mahout.apache.org/. [Accessed 22 January 2017].

[12] CE Seminario, DC Wilson. Case study evaluation of Mahout as a recommender platform. in 6th ACM Conference on recommender engines (RecSys), Dublin. 2012.

[13] D Singh, CK Reddy. A survey on platforms for big data analytics. Journal of Big Data. 2014; 2(8).

[14] J Zheng, A Dagnino. An initial study of predictive machine learning analytics on large volumes of historical data for power system applications. in IEEE International Conference on Big Data, Washington. 2014.

[15] MM Najafabadi, F Villanustre, TM Khoshgoftaar, N Seliya, R Wald, E Muharemagic. Deep learning applications and challenges in big data analytics. Journal of Big Data. 2015; 2(1): 1 - 21.

[16] GDF Morales, A Bifet. SAMOA: Scalable Advanced Massive Online Analysis. Journal of Machine Learning Research. 2015; 16: 149 - 153.

[17] Databricks. Announcing SparkR: R on Spark. 2015. [Online]. Available: https://databricks.com/blog/2015/06/09/announcing-sparkr-r-on-spark.html. [Accessed 26 February 2017].

[18] H Jagadish, J Gehrke, A Labrinidis, Y Papakonstantinou, J Patel, R Ramakrishnan, C Shahabi. Big Data and its technical challenges. Communications of the ACM. 2014; 57(7): 86 - 94.

[19] SL Weinstein. Innovations in London's Transport: big Data for a better customer experience. November 2015. [Online]. Available: http://2015.data-forum.eu/sites/default/files/1600-1640%20Weinstein_SEC.pdf. [Accessed 26 February 2017].

[20] LS Weinstein. How TfL uses 'big data' to plan transport services. EuroTransport, 2016. [Online]. Available: http://www.eurotransportmagazine.com/19635/past-issues/issue-3-2016/tfl-big-data-transport-services/. [Accessed 26 February 2016].

[21] K Barrow. Big Data predicts train delays before they occur. Railjournal.com, 2017. [Online]. Available: http://www.railjournal.com/index.php/commuter-rail/big-data-predicts-train-delays-before-they-occur.html. [Accessed 26 February 2017].

[22] Infocom Media Development Authority. Smart Nation big on Big Data. 14 November 2016. [Online]. Available: https://www.imda.gov.sg/infocomm-and-media-news/buzz-central/2016/6/smart-nation-big-on-big-data. [Accessed 26 February 2017].

[23] A Candel, V Parmar, E LeDell and A Arora. Deep Learning with H2O. 2016. [Online]. Available: http://docs.h2o.ai/h2o/latest-stable/h2o-docs/booklets/DeepLearningBooklet.pdf. [Accessed 24 February 2017].

[24] T Nykodym, P Maj. Fast Analytics on Big Data with H2O. 2017. [Online]. Available: http://gotocon.com/dl/gotoberlin2014/slides/PetrMaj_and_TomasNykodym_FastAnalyticsOnBigData.pdf. [Accessed 20 February 2017].

[25] H2O.ai. Fast Scalable Machine Learning API. H2O, [Online]. Available: http://h2o-release.s3.amazonaws.com/h2o/rel-tverberg/4/index.html. [Accessed 17 February 2017].
[26] Databricks. Making Machine Learning Simple: Building Machine Learning Solutions with Databricks. Databricks. 2016. [Online]. Available: http://cdn2.hubspot.net/hubfs/438089/Landing_pages/ML/Machine-Learning-Solutions-Brief-160129.pdf. [Accessed 24 February 2017].
[27] X Meng, J Bradley, B Yavuz, E Sparks, S Venkataraman, D Liu, J Freeman, DB Tsai, M Amde, S Owen, D Xin, R Xin, MJ. Franklin, R Zadeh, M Zaharia and A Talwalkar. MLlib: Machine Learning in Apache Spark. Journal of Machine Learning Research. 2016; 17(1): 1235 - 1241.
[28] UK Data Gov. Opening up Government. UK Data Gov. 2017. [Online]. Available: http://tris.highwaysengland.co.uk/detail/trafficflowdata. [Accessed 17 February 2017].