Big Data Prediction in Hydrogen Gas Power Plant using Apache Spark

• Manvi Chandra (Rapp, Santa Monica)
• Jongwook Woo (Dept of Information Systems, California State University Los Angeles)

Ⅰ. Introduction

The Hydrogen research and fueling facility of Califronia State University Los Angeles (CalStateLA) was established in the year 2014. The station is capable of producing hydrogen using renewable resources and the process involved is called as electrolysis. It is the first station to sell hydrogen fuel by kilogram into the public. In this paper we have collected and taken the data set containing the fueling details of hydrogen fuel generated in the facility. The various factors that affect the fueling process are analyzed. We build the prediction model using Decision Forest Regression algorithm and achieve 94% of accuracy.

Hydrogen gas fueling facility at California State University Los Angeles [11] has provided us with the data set which was recorded during each fill of Hydrogen gas. We had an opportunity of exploring the various parameters which can improve the fueling performance.

Analytics in this domain can help the facility in various ways like understanding the factors that affect the fueling process which in turn will affect the cost and the efficiency of fuel. The data is increasing drastically with every day and analysis should be done in appropriate manner for the facility to function efficiently. We chose to used decision forest regression in our analysis and prediction of vehicle pressure which is one of the factor affecting fill.

Ⅱ. Related Work

Observing the research done in myriad papers related to analysis and prediction, analysis is usually performed taking into consideration very few fields of the raw dataset. In this paper the data analysis is performed by making use of the raw dataset to the fullest by creating machine learning model which portray analysis Hydrogen Gas Power Plant Dataset. This research leverages relationship between many components which are responsible for gaining knowledge about strategies and information which can be of practical significance. Also most significantly, this analysis can be of theoretical importance, proving the use of Big Data Analytics [7] in improving the process of prediction.

Also, observing various papers, a usual sight is the use of either Hadoop MapReduce codes or HiveQL codes being utilized for analyzing datasets. More commonly, only HiveQL is usually utilized for analyzing datasets as it is complicated to write code using Hadoop MapReduce. This paper is distinct compared to other papers. In this paper, the dataset analysis is done using Spark. Spark is a competitor to MapReduce at some extent. As speed is of great concern these days, Apache Spark is gaining popularity. Spark can be up to 100 times faster than Map-Reduce due to its capability to do in memory processing. It is open source, and one of the most active projects in Big Data. Spark basically has the ability to bring the top end data analysis, the exact epitome of performance and high end sophistication which were initially obtained by utilizing expensive systems to commodity Hadoop Clusters and it runs in the same clusters which leverages users to experiment more with the available data.

III. Big Data Analysis and Prediction

Big data is considered as an umbrella. It accommodates data that are gathered from years together. The name big data comes from the humungous amounts of zeroes, ones etc. and other data which are gathered in a year, month, day, hour and also every second. Such type of data is portrayed on spreadsheets etc. Big data does not have a definite structure and therefore is gathered in different shapes and sizes. Therfore, Big Data can be defined as non-expensive frameworks that can store a huge variety of data set and process it in distributed parallel systems [5, 6].

The analytical solutions include important amounts of structured and multi structured data. Currently emerging analytical processing technologies make things happen. In order to analyze and manage big data, non-relational systems like Hadoop distributed processing system is gaining popularity.

1. Hadoop

Apache Hadoop is an open source framework for distributed storage and distributed processing of large data sets. The core of Apache Hadoop is HDFS (Hadoop Distributed File System) and Map Reduce. Hadoop breaks the file into large blocks and then distribute the across different nodes. Hadoop supports parallel processing [9].

Hadoop Distributed File System is scalable, distributed and portable file system which is written in java. A Hadoop cluster has got one name node and multiple data nodes. The metadata of the file is saved in the name node. The data that makes up the file is stored in blocks in the data node.

Map Reduce engine comprises of job tracker and task tracker. The job tracker is one the cluster and is responsible to schedule the map task and the reduce task on appropriate task tracker. The task tracker is responsible for accomplishment of parallelism for map and reduce tasks.

2. Apache Spark

Apache Spark was originally developed by University of California, Berkeley’s AmpLab. It is an open source cluster computing framework.

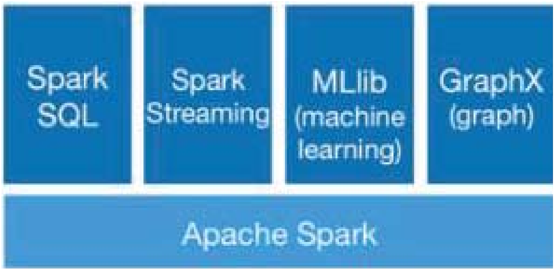


Fig. 1. Spark Cores

The programmers using spark are exposed to application programming interface centered around RDD (resilient distributed dataset). The spark core is responsible for the usual Input and output functionalities, scheduling and the tasks are dispatched in a distributed manner. Spark SQL is built on top of the spark core and is responsible for working with the structured data.

It uses Data frames which supports structured and semi structured data. Spark streaming is responsible for fast scheduling of streaming applications. It consumes the data in small batches and apply RDD transformations to perform streaming analytics.

MLlib is spark's scalable library in which machine learning algorithms are shipped simplifying the pipelines including classification, regression, clustering etc. GraphX is an application programming interface for graph and parallel graph computation. The paper adopts Spark MLlib to build a predictive model for the prediction analysis.

Ⅳ. Predictive Analysis

Machine Learning is to focuses on pattern recognition and artificial intelligence. It explores the patterns to predict the future or to detect abnormal trend. It basically learns from the existing data set and automates creation of models using various algorithms. Machine learning allows to find out hidden insights without actually being programmed explicitly.

Machine Learning is to study and construct systems that can learn from the existing data. Machine Learning algorithms are broadly classified in two categories namely:

- Supervised Learning
- Unsupervised Learning

In case of Supervised learning, the data to be given to machine learning algorithm is labeled and is known as the training data set. The machine is responsible of classifying the new data based on the labels by creating an inference function which can be used for scoring new instances.

For example, if you want to determine chairs using your machine learning algorithm in JPG'S, then a large dataset could be provided to your algorithm that has chairs labeled and then the algorithm can be used to recognize the chairs in new images.

In case of Unsupervised learning, the machine is not provided with any training data set and the machine to discover information about the new data. It refers to the problem of exploring a structure in unlabeled data. In this case the expected result is unknown.

For instance, we will have no idea how a chair looks like. In case of unsupervised learning the algorithm has to groups the similar patterns in order to find patterns similar to that of chair in order to recognize chairs.

1. System Requirements

The experimental systems in the paper is executed in cloud computing, which is Microsoft Azure HDInsight that is Hadoop cluster with Spark engine with the following specifications:

- Number of worker nodes: 2
- Number of cores used: 8
- Total RAM: 14 GB

2. Flow diagram

The figure 2 demonstrates the flow diagram used to create the model and the process of predicton. In the first step we got the data from California State University's Hydrogen Plant facility in the form of excel files. We collected the entire data into one csv file.We created a Blank experiment in Microsoft Azure Machine learning Studio. During the next step we loaded our data set to the Microsoft Azure Machine learning Studio platform.

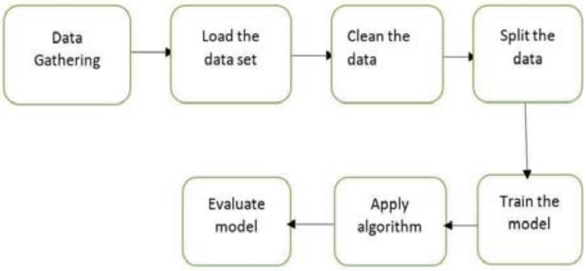


Fig. 2. Flow diagram

Once the data is loaded, uneccesary data has to be removed, if there are duplicate rows then one of them have to be removed ,the data cleaning process involves finding if there is any missing value, any special charaters, semantic errors etc and resolving the erros in order to get a clean dataset. Once the data is cleaned it is split into training and test sets. Then the model is trained by applying the algorithm in order to predict the desired value .In our case we have used Decision Forest Regression Algorithm .Once the model is trained the new data is evaluated based on the inference function created by the labeled values in order to determine the accuracy of are model.

3. Microsoft Azure Machine Learning Studio

Microsoft Machine Learning Studio is a collaborative tool which uses drag and drop technique to do various kinds of predictive analytics on our data set. In this project we created a model to predict vehicle pressure using decision forest regression algorithm.

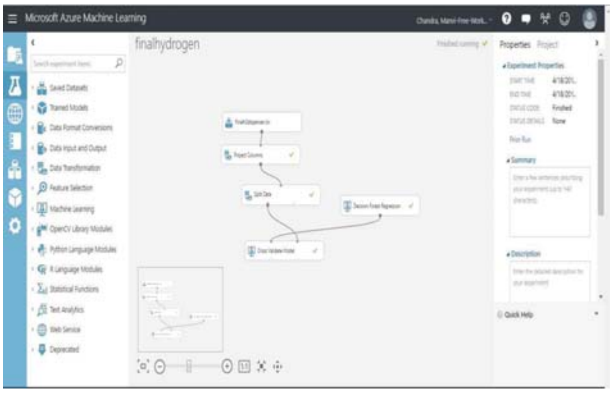


Fig. 3. Machine Learning Studio Model

The figure 3 shows the prediction model for Hydrogen Gas Power Plant dataset. In this model first the data set is loaded and the data is cleaned. Then the columns necessary for prediction are loaded. The data set is then split into Training and test data and then the model is trained and evaluated using Decision Forest regression.

4. Decision Forest Regression Algorithm

It is an ensemble of Decision forest that is an ensemble learning method for classification, regression and other tasks, that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees.

The algorithm uses the idea of bagging and the selection of features in random order to build a collection of decision trees with controlled variance.

The model that is built in the paper can be useful in the prediction of the State of charge that depends on the vehicle pressure prediction.

STATE OF CHARGE (SOC):

Ratio of hydrogen density within the vehicle storage system to the full-fill density. SOC is expressed as a percentage and is computed based on the gas density as per formula below:

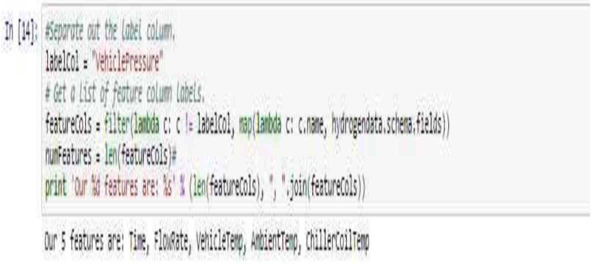
$$SOC\ (\%) = \frac{\rho\,(P,\ T)}{\rho\,(NWP,\ 15°C)}\ x\ 100$$

In the above equation SOC stands for the State of charge which depends on P which is the vehicle pressure and T which is the vehicle temperature.

4.1 Spark MLlib Model

The steps involved in creating the spark model along with the code snippets are as stated below:

- Separating the labeled column.

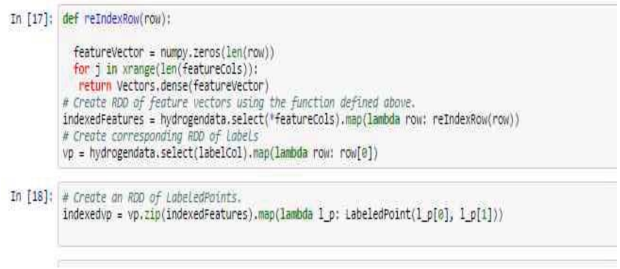


```
In [14]: #Separate out the label column.
         labelCol = "VehiclePressure"
         # Get a list of feature column labels.
         featureCols = filter(lambda c: c != labelCol, map(lambda c: c.name, hydrogendata.schema.fields))
         numFeatures = len(featureCols)#
         print 'Our %d features are: %s' % (len(featureCols), ", ".join(featureCols))

         Our 5 features are: Time, FlowRate, VehicleTemp, AmbientTemp, ChillerCoilTemp
```

Fig. 4. Separating Labelled Column

- Creation of RDD



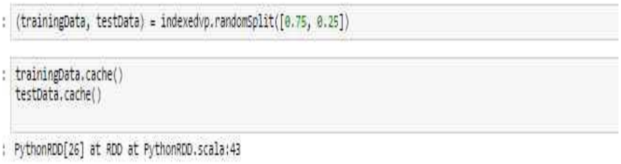
```
In [17]: def reIndexRow(row):

           featureVector = numpy.zeros(len(row))
           for j in xrange(len(featureCols)):
            return Vectors.dense(featureVector)
         # Create RDD of feature vectors using the function defined above.
         indexedFeatures = hydrogendata.select(*featureCols).map(lambda row: reIndexRow(row))
         # Create corresponding RDD of labels
         vp = hydrogendata.select(labelCol).map(lambda row: row[0])

In [18]: # Create an RDD of LabeledPoints.
         indexedvp = vp.zip(indexedFeatures).map(lambda l_p: LabeledPoint(l_p[0], l_p[1]))
```

Fig. 5. Creation of RDD

- Splitting the data into training and test sets.



```
: (trainingData, testData) = indexedvp.randomSplit([0.75, 0.25])

: trainingData.cache()
  testData.cache()

: PythonRDD[26] at RDD at PythonRDD.scala:43
```

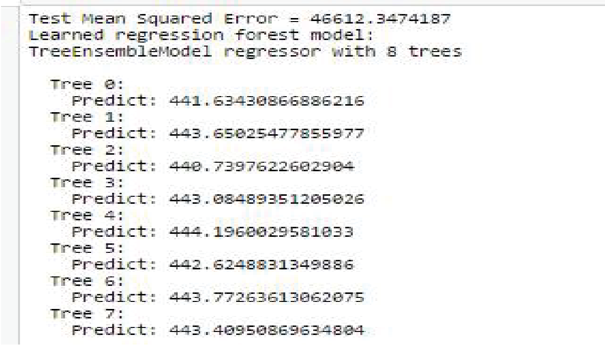
Fig 6: Splitting of data

- Training the dataset using Decision forest regression algorithm.
- Evaluation of the result.

4.2 Results and Observations

The model predicts the vehicle pressure (Pressure of hydrogen gas within the vehicle Hydrogen Storage System) based on various parameters like flow rate, vehicle temperature, ambient temperature, time and chiller coil temperature using the decision forest regression with a lot of accuracy.

The test-mean-square-error is predicted relatively accurate:



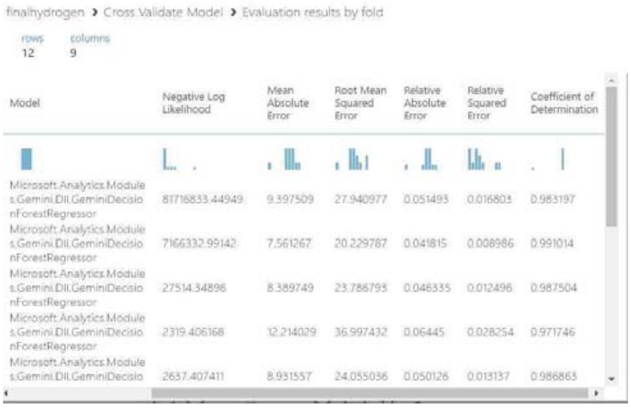
```
Test Mean Squared Error = 46612.3474187
Learned regression forest model:
TreeEnsembleModel regressor with 8 trees

  Tree 0:
    Predict: 441.63430866886216
  Tree 1:
    Predict: 443.65025477855977
  Tree 2:
    Predict: 440.7397622602904
  Tree 3:
    Predict: 443.08489351205026
  Tree 4:
    Predict: 444.1960029581033
  Tree 5:
    Predict: 442.6248831349886
  Tree 6:
    Predict: 443.77263613062075
  Tree 7:
    Predict: 443.40950869634804
```

Fig. 7. TMSE error prediction using spark model

4.3 Microsoft Machine Learning Studio Model

The model is built in Machine Learning studio in the figure 8. The coefficient of determination is approximately 97% for this model as well in the figure 8.



finalhydrogen › Cross Validate Model › Evaluation results by fold

rows 12 columns 9

Model	Negative Log Likelihood	Mean Absolute Error	Root Mean Squared Error	Relative Absolute Error	Relative Squared Error	Coefficient of Determination
Microsoft.Analytics.Modules.Gemini.Dll.GeminiDecisionForestRegressor	81716833.44949	9.397509	27.940977	0.051493	0.016803	0.983197
Microsoft.Analytics.Modules.Gemini.Dll.GeminiDecisionForestRegressor	7166332.99142	7.561267	20.229787	0.041815	0.008986	0.991014
Microsoft.Analytics.Modules.Gemini.Dll.GeminiDecisionForestRegressor	27514.34896	8.389749	23.786793	0.046335	0.012496	0.987504
Microsoft.Analytics.Modules.Gemini.Dll.GeminiDecisionForestRegressor	2319.406168	12.214029	36.997432	0.06445	0.028254	0.971746
Microsoft.Analytics.Modules.Gemini.Dll.GeminiDecisio	2637.407411	8.931557	24.055036	0.050126	0.013137	0.986863

Fig. 8. Model result

V. Conclusion

According to our analysis of the Hydrogen Gas Power Plant Data Set, we were able to explore and to create a machine learning model using Apache Spark, followed by creation of Microsoft Machine Learning Studio supporting the result in the training and test data set.

We were able to predict Vehicle pressure with almost 97% accuracy for SOC, which can help Hydrogen Gas Power Plant in order to detect abnormal temperature and pressure to maintain their fueling performance.

참고문헌

[1] Matei Zaharia, Mosharaf Chowdhury, Tathagata Das, Ankur Dave, Justin Ma, Murphy McCauley, Michael J. Franklin, Scott Shenker, Ion Stoica," Resilient Distributed Datasets: A Fault-Tolerant Abstraction for In-Memory Cluster Computing",UC Berkley,2012

[2] W. Xue, J. Shi, and B. Yang. X-RIME:” Cloud-based large scale social network analysis” In SCC, pages 506 - 513, 2010.

[3] J. Dean and S. Ghemawat. MapReduce: Simplified data processing on large clusters. In OSDI, 2004.

[4] Matei Zaharia, Mosharaf Chowdhury, Michael J. Franklin, Scott Shenker, Ion Stoica,” Cluster Computing with Working Sets”, UC Berkley,2010

[5] Woo Jongwook and Xu Yuhang, 2011. “Market Basket Analysis Algorithm with Map/Reduce of Cloud Computing”, The 2011 international Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA 2011), Las Vegas (July 18-21, 2011)

[6] Woo Jongwook, Oct 28 2013. “Market Basket Analysis Algorithms with MapReduce”, DMKD-00150, Wiley Interdisciplinary Reviews Data Mining and Knowledge Discovery, Volume 3, Issue 6, pp445-452, ISSN 1942-4795

[7] CalStateLA Hydrogen Station, http://www.calstatela.edu/ecst/h2station

저자소개

Manvi Chandra
2016: MS, Information Systems, California State University Los Angeles
2015: Honors Award of Special Recognition for scholastic achievement
2011: Maha Rishi Dayanand University, Rhotak, India.
현 재: 데이터분석가 at Rapp, Santa Monica, USA 관심분야: Big Data 분석 예측

Jongwook Woo
1991: MS, Electronic Eng
Yonsei University .
1998: MS, Computer Sci
USC.
2001: PhD, Computer Eng
USC.
현 재: Full Professor
경영정보학과
California State University Los Angeles
관심분야: Big Data 분석 예측