Predicting Potential Banking Customer Churn using Apache Spark ML and MLlib Packages: A Comparative Study

Hend Sayed, Manal A. Abdel-Fattah, Sherif Kholief
Information Systems Department
Faculty of Computers and Information
Helwan University
Cairo, Egypt

***Abstract*—This study was conducted based on an assumption that Spark ML package has much better performance and accuracy than Spark MLlib package in dealing with big data. The used dataset in the comparison is for bank customers transactions. The Decision tree algorithm was used with both packages to generate a model for predicting the churn probability for bank customers depending on their transactions data. Detailed comparison results were recorded and conducted that the ML package and its new DataFrame-based APIs have better-evaluating performance and predicting accuracy.**

***Keywords*—Churn prediction; Big data; Machine learning; Apache Spark; ML package; MLlib package; Decision tree**

I. Introduction

Recently, big data [1] technologies became more popular and are being used in many fields. It is a critical issue for most business owners to find the optimal solutions to automate their work and process their huge amount of data. A deluge of data is flooded all the time from many resources and there is a real competition in how to deal with it efficiently and with high performance. One of the most common needs is to predict customers churn depending on their data and activities. This need increases for businesses which are dealing with numerous clients such as telecommunications and banking[2], in this paper, the comparative study was conducted on transactions data of bank customers.

Churn prediction[3][4] is the process of predicting the intention of customers to leave. It is one of the most debated researches in last years. This study is conducted on a dataset of transactions of bank customers to predict their probability to leave.

Apache Spark has added solutions for MapReduce limitations and now it is widely used due to its high performance and efficiency in processing a huge amount of data that is 100 x times faster than Apache Hadoop [5]. Spark's machine learning APIs were based on Resilient Distributed datasets (RDD) in MLlib package, and now the primary API is the ML package which is a higher level API that is based on DataFrames that facilitate practical ML pipelines, especially feature transformations. AS mentioned in Apache Spark ML guide, DataFrames provide a more user-friendly API than RDDs. The many benefits of DataFrames include Spark Datasources, SQL/DataFrame queries, Tungsten and Catalyst optimizations, and uniform APIs across languages. MLlib package is in the maintenance mode with Spark 2.0. In this paper, a comparative study between the two packages is conducting in terms of accuracy, model training and model evaluation.

This research aims to highlight the practical differences between the two packages and list the pros and cons of each package depending on the real results of processing the same dataset using the same algorithm.

This paper is organized as follows, the 2nd section presents an overview of the related work. Then in the 3rd section, the used dataset and the steps of processing the data are discussed. Also, it is discussed in details the used algorithm and why it is chosen for this study. In 4th section evaluation and results are outlined. The conclusion of this research is summarizing the results of the comparative study.

II. Literature Review

A. Churn prediction

Customer churn [6] is the term used in the banking sector tries to denote the movement of customers from one Bank to another.

The Importance of Predicting Customer Churn [7]

- Avoiding losing revenue that results from a customer abandoning the bank.
- The cost of acquiring a new customer is 5x higher (Lee Resources 2010).
- Intensifies the competition among commercial banks.

According to these reasons, it is urgent for commercial banks to improve the capabilities to predict customer churn, thereby using good solutions for churn predicting to retain customers.

B. Churn prediction with big data

A large amount of data is being generated daily from different sources, which is much more expensive and much slower to be processed and analyzed[8]. Suitable and efficient solutions for storing, processing and analyzing a massive volume of data are critically needed to be able for churn prediction efficiently and accurately.

C. ML in Apache Spark

Applying machine learning on a big dataset needs a very large amount of physical resources for processing data, the need of having a platform that can efficiently perform very complex processing and analysis operations on enormous datasets increases daily[9]. Apache Spark is an efficient one of these platforms, it offers a set of modules for different machine learning tasks.

Gauri D et al., (2015)[10] proposed a churn prediction model for the telecommunications field that predicts churn probability for customers using their records data, which can help the company know which customer has an intention to leave or move to another service provider in the near future. Due to the increasing volume of customers' data, they are using Hadoop framework for data storing and processing. They applied the prediction model using the decision Tree C4.5 algorithm. The model generates the rules from training data and applies these rules on testing data to determine which customer may leave. The process of rules generation involves calculation of entropy for every attribute of each record along with the information gain. After applying rules on the test data, a list of the predicted churners and non-churners customers is returned as a result and added to a text file in HDFS.

Avishkar D. et al., (2016) [11] proposed an application for predicting customers attrition in the telecommunication field and enable business owners to take all needed procedures to retain the existing customers rather than increasing the number of customers. They take customer records as an input and give a prediction of customer churn as output. They applied prediction through analyzing customer behavior based on several attributes such as calls per day and using provided services. Because of the overall growth rate for over 35 percent over the past decade in terms of subscribers, they applied prediction using the Hadoop framework that made it much easier. They applied the decision tree approach using C4.5 data mining algorithm, which provides great accuracy in predicting churners. The result was obtained after applying the algorithm is a list of churners and non-churners.

Wei Z et al., (2016) [12] conducted a comparison of decision tree based churn prediction model between SPSS and Spark ML package and used a customers' data of insurance company as an example. The comparison was conducted on the execution flow of each, run-time, model evaluation, and model precision. Their results show that Spark ML is easier and more efficient than SPSS in applying a churn prediction model, especially for insurance companies.

D. Decision tree

The decision tree is a supervised learning algorithm that is used for regression or classification. Decision tree model has a tree-like structure that consists of nodes. Each node in this tree refers to a test on a single attribute, each branch represents an outcome of this test, and each leaf node holds a class label. The splitting of nodes is decided by /algorithms like information gain, chi square, gini index[13]. There are different algorithms used in the decision trees: ID3, C4.5, CART, C5.0, CHAID, QUEST, and CRUISE.

III. Methodology

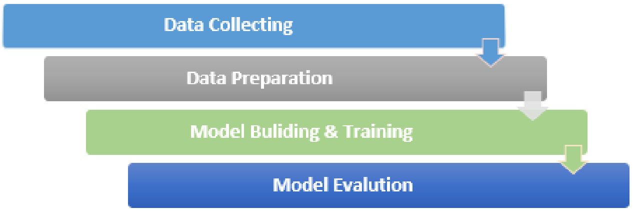

Fig. 1. Building ML classification model workflow

The workflow for building a machine learning classification model shown in Figure 1. After collecting data, it is prepared, transformed and processed to be ready for the training phase. A classification algorithm is applied to the prepared data for generating the classification model, then the model is being tested and evaluated using testing data. Methodology is divided for subsections, section 3.1 discusses the used dataset, subsection 3.2 presents the pre-processing steps required to prepare the dataset for the training phase. Then in section 3.3 Spark MLlib is illustrated with its components and work-flow, and the same discussion for spark ML package in section 3.4. In section 3.5 the results are outlined.

TABLE I. Dataset Attributes

Attribute	Description
Row Number	
Customer ID	
Surname	
Credit Score	
Geography	The location of the custormers of three countries where the bank is operating
Gender	
Age	
Tenure	The period of having the account in monthes
Balance	
NumOfProducts	
HasCrCard	If the customer has a credit card
IsActiveMember	
Estimated Salary	According to the different factors such as credit score, transactions company has used the data to calculate the estimated salary of the customer.
Churn	Indicates the customer has leaved or not

A. Dataset

A dataset of bank customers transactions is used in this study for predicting bank customers churn. The dataset is freely available online on Kaggle[1]. It contains 10k row and 14 columns, where each row represents a customer data and each column represents a single attribute. Table I illustrates the attributes of the used dataset and a description for non-descriptive attributes names.

B. Correlations Data Preparation

First, the irrelevant attributes (Row Number, CustomerID and Surname) were dropped. Then the strings or categorical attributes were mapped to numeric values. The mapped values are for Geography and Gender attributes, for Geography, there

[1] Kaggle: a platform for predictive modeling and analytics competitions, www.kaggle.com.

are three categories or values (France: 0.0, Spain: 1.0, Germany: 2.0) and there are two categories for Gender (Female: 0.0, Male: 1.0).

Then, a statistical analysis was performed on a part of the dataset to examine correlations between the numeric columns and generate scatter plots of them. As shown in Figure 2, it resulted that no high correlated pairs found, no more attributes were dropped.

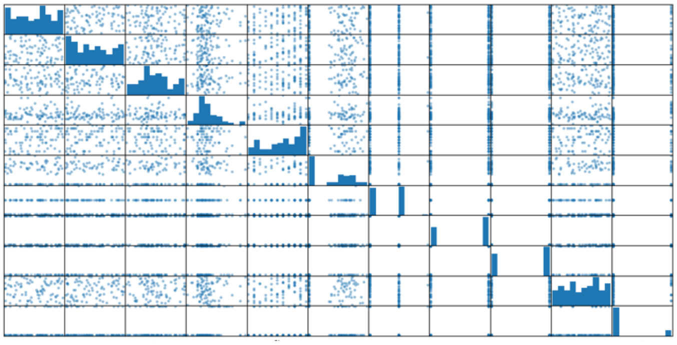

Fig. 2. Correlations scatter plots

After applying the training algorithms, despite the high accuracy, the sensitivity to the non-churners was much more than the sensitivity to the churner customers. By checking the count of each class, the count of churners was 2037, and for non-churners, the count was 7963. so the non-churners was down-sampled to a fraction of 2037/7963, that resulted in 2048 non-churners.

C. MLlib package

Spark MLLib package (spark.mllib) is the older package for machine learning on spark, it contains the original API that is built on top of Resilient Distributed Datasets (RDDs) which are immutable and partitioned collections of elements that can be operated on in parallel. Other advantages of RDDs are Persistence, fault-tolerance, lazy evaluation and typing. RDD-based API requires transforming the data into rows of type LabeledPoint before processing, each LabeledPoint consists of a label and a vector of features which represent the attributes.

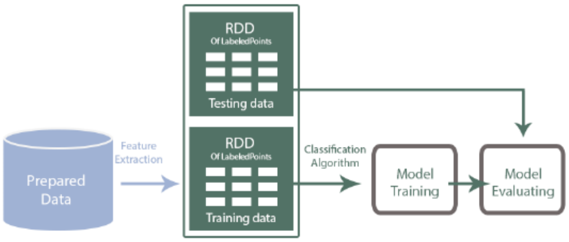

Fig. 3. MLlib Workflow

As shown in Figure 3, after transforming the data to RDD of LabeledPoints, each LabeledPoint consists of the label which represents the Churn column, and features column which is a vector of all features values. The RDD was divided into two sets (Training data & Testing data). Training data are used in applying the decision tree classification algorithm to generate the model using trainClassifier() function in MLlib DecisionTree module. The decision tree algorithm was applied with Gini impurity. After generating a model, the testing data was used for evaluating the model.

D. ML package

Spark ML (spark.ml) is a newer package that was introduced in Spark 1.2, it contains a newer machine learning APIs that are built on top of DataFrames, and it is currently the primary APIs for machine learning on Spark. As shown in Figure 4, The ML package workflow is represented by **ML Pipeline** that consists of some chained PipelineStages. Each **PipelineStage** can be either a **Transformer** or an **Estimator**.

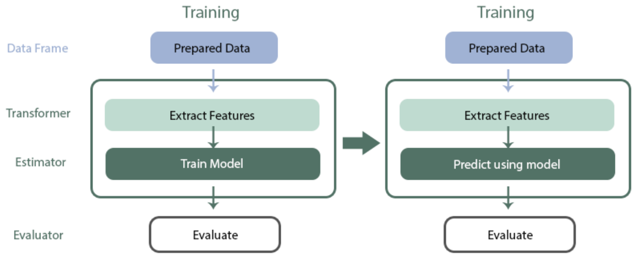

Fig. 4. General ML Workflow

The transformer is an algorithm that is transforming a dataFrame into another dataFrame while the estimator is an algorithm which can be fitted on a dataFrame to produce a model, which is a transformer. The pipeline itself is an estimator that is generating the classifier model. As shown in Figure 5, StringIndexer estimator is used for indexing the label column and VectorIndexer estimator to automatically check categorical features depending on a provided **maxCategories** value and generate a new column that contains a vector if features indices. The Spark ML decision tree classifier DecisionTreeClassifier() is an estimator that is added to the pipeline chain and used to generate the decision tree model. The researchers followed Spark python API docs [2] in implementation.

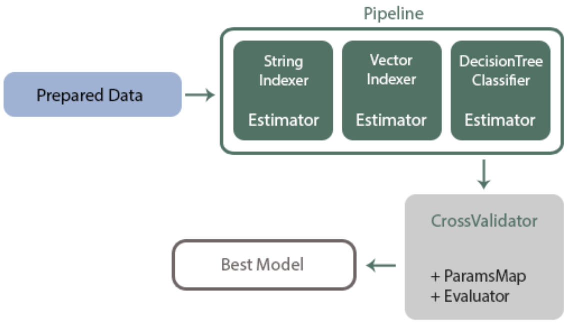

Fig. 5. Implemented ML Workflow

CrossValidator is used for best model selection using the generated pipeline as an estimator, and provided parameters

[2] Spark python API docs, https://spark.apache.org/docs/latest/api/python.

map that is generated using ParamGridBuilder() and, identifying the decision tree maxDepth values to search through. An evaluator MulticlassClassificationEvaluator() is used for testing the generated model, it works on a dataset with two attributes (Label and Prediction) and is used to get the predicting accuracy. CrossValidator takes a number to split the dataset into a set of folds that are used as separate training and testing datasets.

E. Results

Result 1, the resulted accuracy values for the two models were slightly close, however, the accuracy of the ML package model was better. As shown in Figure 6, the accuracy value for ML package model was 0.79 and the MLlib package model's accuracy was 0.73.

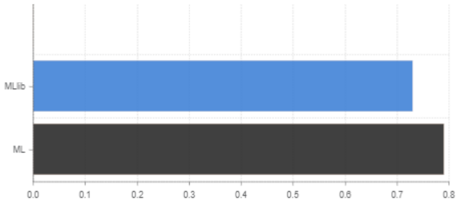

Fig. 6. Compared accuracy

Result 2, MLlib package needed less time for data transformations, applying the classification algorithm and training the model. The results are shown in Figure 7, it took only 6 seconds, and the model that was generated using ML package took 25 seconds.

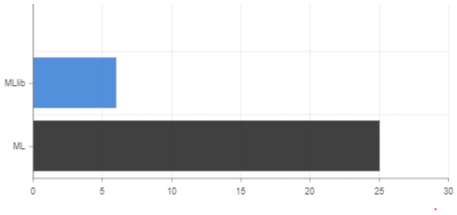

Fig. 7. Compared training time

Result 3, In contrast to the previous result, and as shown in Figure 8, the time needed for evaluating the model using the same testing data was much fewer in ML package model than the MLlib package model. It took only 5 seconds, and the model of MLlib needed 14 seconds.

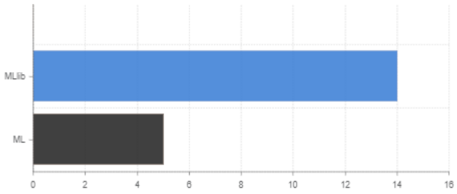

Fig. 8. Compared evaluation time

IV. Conclusion

In this paper, a comparative study between Apache Spark ML and MLlib packages was conducted, in terms of accuracy, model training and model evaluation. The researchers performed the comparison on bank customers transactions dataset and using decision tree algorithm for predicting customers churn probability. MLlib package with its RDD-based API has a better result for training time, which can be caused by the internal transformations in both packages. ML with its DataFrames-based API has better results for testing time and accuracy. So, the results indicate that ML churn prediction models can perform better and help in getting more accurate results faster. These results are useful for banks and any businesses that are dealing with numerous clients and records in predicting their churn probability. In the future, more detailed comparative studies will be done for more packages and platforms with different types of data and using different algorithms to check the best and most accurate models in different situations.

References

[1] N. K. Gyamfi, "Big Data Analytics : Survey Paper Scanned by CamScanner," no. February, 2017.

[2] N. Elgendy and A. Elragal, "Advances in Data Mining. Applications and Theoretical Aspects," vol. 7987, no. August, 2013. [Online]. Available: http://link.springer.com/10.1007/978-3-642-39736-3

[3] T. Y. Fei, L. H. Shuan, L. J. Yan, G. Xiaoning, and S. W. King, "Prediction on customer churn in the telecommunications sector using discretization and Naïve Bayes classifier," *International Journal of Advances in Soft Computing and its Applications*, vol. 9, no. 3, pp. 23–35, 2017.

[4] A. Keramati, H. Ghaneei, and S. M. Mirmohammadi, "Developing a prediction model for customer churn from electronic banking services using data mining," *Financial Innovation*, vol. 2, no. 1, p. 10, 2016. [Online]. Available: http://jfin-swufe.springeropen.com/articles/10.1186/s40854-016-0029-6

[5] T. Lin, Y. Li, and R. Mai, "Modeling and Control Method of Inductive Power Transfer System Based on LCL-S Topology," *Diangong Jishu Xuebao/Transactions of China Electrotechnical Society*, vol. 33, no. 1, pp. 104–111, 2018.

[6] K. Chitra and B. Subashini, "Customer Retention in Banking Sector using Predictive Data Mining Technique," p. 4, 2011.

[7] B. He, Y. Shi, Q. Wan, and X. Zhao, "Prediction of customer attrition of commercial banks based on SVM model," pp. 423–430, 2014.

[8] A. Bhattacharya and S. Bhatnagar, "Big Data and Apache Spark : A Review," no. 5, pp. 206–210, 2016.

[9] M. Assefi, E. Behravesh, G. Liu, and A. P. Tafti, "Big data machine learning using apache spark MLlib," *Proceedings - 2017 IEEE International Conference on Big Data, Big Data 2017*, vol. 2018-Janua, no. November, pp. 3492–3498, 2018.

[10] G. D. Limaye, J. P. Chaudhary, and N. Mumbai, "Churn Prediction Using MapReduce and HBase," *International Journal on Recent and Innovation Trends in Computing and Communication*, vol. 3, no. 3, pp. 1699–1703, 2015.

[11] A. Dalvi, B. Bhor, H. Chauhan, and P. Sawant, "Churn Prediction Using Hadoop," Tech. Rep., 2016. [Online]. Available: www.irjet.net

[12] W. Zhang, T. Mo, W. Li, H. Huang, and X. Tian, "The Comparison of Decision Tree Based Insurance Churn Prediction between Spark ML and SPSS," *Proceedings of International Conference on Service Science, ICSS*, pp. 134–139, 2017.

[13] B. Huang, M. T. Kechadi, and B. Buckley, "Customer churn prediction in telecommunications," *Expert Systems with Applications*, vol. 39, no. 1, pp. 1414–1425, 2012. [Online]. Available: http://dx.doi.org/10.1016/j.eswa.2011.08.024