Using Decision Tree Classification Model to Predict Payment Type in NYC Yellow Taxi

Hadeer Ismaeil, Sherif Kholeif, Manal A.Abdel-Fattah
Information Systems Department, Faculty of Computers and Artificial Intelligence
Helwan University, Cairo, Egypt

***Abstract*—The taxi services are growing rapidly as reliable services. The demand and competition between service providers is so high. A billion trip records need to be analyzed to raise the spirit of competition, understand the service users, and improve the business. Although decision tree classification is a common algorithm which generates rules that are easy to understand, there is no implementation for classification on taxi dataset. This research applies the decision tree classification model on taxi dataset to classify instances correctly, build a decision tree, and calculate accuracy. This experiment collected decision tree algorithm with Spark framework to present the good performance and high accuracy when predicting payment type. Applied decision tree algorithm with different aspects on NYC taxi dataset results in high accuracy.**

Keywords*—Big data analytics; apache spark; decision tree classification; taxi trips; machine learning

I. Introduction

Big data describes the large volume of data, but there's no rule strictly defines the size of data. What really determines that the data is big, is the need for multiple physical or virtual devices to process this data as fast as possible. In [1], this data is generated by everything around us like systems, digital devices, and remote sensors. Big data is used for collecting and analyzing large and complex data sets to produce knowledge. In the past, storing, processing, and analyzing this volume of data were a problem; but new technologies solved this problem [2].

One of the tools that are used to analyze data is Apache Spark, which is open-source data analytics tools. Spark is based on MapReduce, but it's faster as it stores the data in the memory into RDD (Resilient Distributed Databases) [3]. Fig. 1 represents the difference between Spark and Hadoop performance.

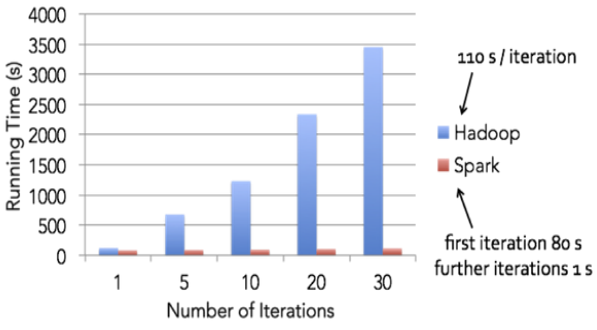

Fig. 1. Spark and Hadoop Performance. [3].

When talking about machine learning, Spark has two types of machine learning libraries: Spark MLlib, and Spark ML. Spark MLlib is the original API and it is based on RDD API, so it has more features than the new API (Spark ML). Spark ML provides high-level API, and it is based on data frames and dataset; it supports pipeline and easier to construct. Spark MLlib focuses on the basics of the algorithm leaving data preparation and pipelines to the user, but Spark ML works on all those aspects from data preparation to model training. Spark MLlib is the best choice when dealing with a stream of data. The new features will be added to Spark ML and this why this research will use Spark ML. [4].

Spark ML is based on pipeline concept which uses different stages to perform separate tasks from data cleaning to feature selection and applying machine learning algorithm. Pipeline stages consist of two basics transformers and estimators. Transformers use transform method which takes a data set as input and returns an enhanced dataset as a result. Estimators, when fit, returns a transform, it uses the fit method on data set to produce a model. A decision tree is an estimator that trains dataset to produce a model [4].

Classification is an important technique for assigning a data object to predefined class or category. One of the most commonly used algorithms, used to apply a classification technique, is the Decision tree algorithm. Decision tree one of the important algorithms in Classification technique. It builds a tree model. Decision tree algorithm is popular because it's easy to implement and understand. In addition, it is simple and fast to construct compared to other classification algorithms.

Decision tree algorithms extract knowledge from data, present it graphically and produce a clear and understandable rule. It is one of the most powerful and popular algorithms that can handle continuous and categorical variables using less computation. The most common used decision tree algorithms are ID3, C4.5, and CART. C4.5 is an improved version of ID3. [5][6].

Transportation plays a vital role in human life especially Taxi and Car services. As a business, it is a huge industry. Every minute there is a hundred of trips from just a single zone, which create a huge amount of data [7] [8]. NYC Taxi and Limousine commission provides an open Dataset that includes detailed trip record from NYC yellow taxi through 2017. Predicting payment method is considered important to business. Customers add their credit card only in trusted services. The greater the number of registered or used credit cards, the higher the service reliability is. It is also important to

the driver to know the payment method as some drivers prefer cash and others prefer credit for guaranteed payment. This dataset can be used to predict a lot of things like passenger count, Hotspot to reduce peak factor, Rush hour for extra charges and high demand at a specific time.

II. BACKGROUND

A. Big Data Analytics

The concept of big data has been around for years, and most firms now realize that if they capture all the data that flows into their operations, they can use analytics to extract tremendous value. Big data analytics assists businesses in harnessing their data and identifying new opportunities. As a result, smarter business decisions, more effective operations, higher profits, and happier consumers are the result. Big data analytics is the often-difficult process of analyzing large amounts of data to identify information such as hidden patterns, correlations, market trends, and customer preferences that can assist businesses in making better decisions. Data analytics tools and approaches provide organizations with a way to evaluate data sets and obtain new information on a large scale. Basic questions regarding business operations and performance are answered by business intelligence (BI) queries. Big data analytics is a type of advanced analytics that entails complicated applications that use analytics systems to power aspects like predictive models, statistical algorithms, and what-if analyses [9].

Without the right tools, methods, and techniques, big data analytics can be time-consuming, difficult, and computationally demanding. When the amount of data is too large to analyze and analyze on a single machine, Apache Spark and Apache Hadoop can help by using parallel and distributed processing. It is critical to first comprehend the concept of "big data" to comprehend the significance of parallel and distributed processing. The fast rate at which big data is generated necessitates that it be processed quickly, and the variety of big data implies that it contains several data kinds, including structured, semi-structured, and unstructured data. Because of the amount, velocity, and variety of big data, new, novel methodologies, and frameworks for collecting, storing, and analyzing the data were developed, which is why Apache Hadoop and Apache Spark were formed.[10].

B. Big Data Analytics Tools

Understanding parallel and distributed processing helps in understanding big data analytics tools and how they are used. Because both parallel processing and distributed processing entail breaking down computation into smaller sections, the two can be confused. The memory architecture distinguishes parallel computing from distributed computing. Parallel computing is the use of several processors to solve a problem at the same time. Distributed computing is the use of multiple computers to solve a problem at the same time. Because distributed computing is disk-based rather than memory-based, parallel computing processes have access to the same memory space as distributed computing workloads. Some distributed computing operations are performed on a single computer, while others are performed on multiple computers. Apache Hadoop and Apache Spark are both open-source systems for big data processing, although they differ in important ways. Hadoop processes data using MapReduce, whereas Spark employs resilient distributed datasets (RDDs). Hadoop uses a distributed file system (HDFS), which allows data files to be stored on several machines. Because servers and machines may be added to accommodate increasing data quantities, the file system is scalable. Because Spark lacks a distributed file storage system, it is mostly utilized for computation on top of Hadoop. Spark does not require Hadoop to run, although it can be used with Hadoop because it can construct distributed datasets from HDFS files. Spark does not provide a distributed file storage system; it is mostly utilized for computation on top of Hadoop. Spark does not require Hadoop to run, although it can be used with Hadoop because it can construct distributed datasets from HDFS files. The performance gap between Hadoop and Spark is significant. UC Berkeley researchers noticed that Hadoop is wonderful for batch processing but inefficient for iterative processing, so they invented Spark to address this. Spark program iteratively run 100 times quicker in memory than Hadoop and 10 times faster on disc. Spark's quickness is attributed to its in-memory processing. Instead, Hadoop MapReduce sends data to disc, which is read on the next iteration. It is substantially slower than Spark because data is reloaded from disc after each iteration [11].

C. Apache Spark MLlib

MLlib, an Apache Spark machine learning library, covers the major machine learning methods such as classification, clustering, regression, dimensionality reduction, transformers, and collaborative filtering. Some machine learning techniques can be applied to streaming data, which is useful for Spark Streaming applications [12].

D. Classification

Classification is a method of locating models that describe multiple data classes or concepts. By performing analysis, The class labels that are known for a set of training data or data objects during this model can be obtained. The primary goal of this model is to properly anticipate an unknown object's class label. The classification problem should include some input that is regarded as training data with class labels and is utilized to determine the class label for unlabeled test data or instances.

The primary challenge for categorization is data preparation. This procedure includes the following steps: selecting, pre-processing, data cleaning, data integration and transforming.

Decision Trees are the most common classification algorithms. A decision tree is described as a flow chart with a tree structure that includes a root node, non-leaf nodes, and leaf nodes. Each non-leaf node describes a test attribute, each branch describes a test outcome, and each leaf node retains the class label. The basic idea behind the decision tree is to divide the data recursively into subsets with the final goal of having each subset contain nearly homogeneous states of the target variable. The attribute selection measure must choose a splitting criterion that "best" separates the given dataset. Some well-known decision tree algorithms include ID3, C4.5, and CART, which use the Information Gain, Gain Ratio, and Gini Index as attribute selection measures, respectively. C5 is also a decision tree-based algorithm that is an improved version of

C4.5. When the decision tree is built, it is used to categorize another new instance by traversing from the root node to the leaf node and applying the test criteria at each non-leaf node. Each instance's class is the class of the leaf node [6].

III. Related Work

In [6], B. Charbuty, et. al. studied and compared the most popular decision tree algorithms, ID3, C4.5, CART, CHAID, and QUEST as the most common data classifier. Literature review related to Decision Tree was presented. The literature review compared different research papers work in terms of year of study, dataset, techniques, or algorithms used on the predefined dataset, and the resulted accuracy. The literature reviews the most recent research papers applied in different areas/datasets like medical, text classification, user smartphones classification, etc. this study proves the efficiency of decision tree in creating efficient and understandable rules and achieving high accuracy.

In [13] B. Roy, et. Al. applied a set of machine learning algorithm to predict trip duration between two locations. One of the algorithms used in this research was Decision tree Regression Model with varying max depth parameter and default values for other hyperparameters. Decision tree regression model with the biggest max depth achieved the best accuracy compared to the lower max depth. But other regression achieved better accuracy than decision tree regression model with different max depth.

In [14], S. Singh, et. al. studied and compared the most popular decision tree algorithms, ID3, C4.5, and CART. The survey described the advantage and disadvantage of each algorithm. The survey also contains a comparison between characteristics of each algorithm like the type of data suitable for each algorithm, speed, dealing with missing values and splitting formula. It also shows some application of decision tree such as business, industry, medicine and so on. Common datasets, tools, and problems of decision tree were introduced. The final observation of the survey was the dependency of splitting formula and dataset feature with the performance of the algorithm.

In [15], H. Sun, et. al. described how taxi data was collected through sensors and GPS. They analyzed taxi data to extract and filter data to get a valuable information and results. The results from analysis helped them propose a new application that adds new benefits to the taxi passengers and drivers. They described data attributes and how they used big-data tools to extract useful information, so they can notice relations between attributes. The analysis steps were described briefly to produce analysis results and patterns that was helpful to their mobile service.

In [16], X. Meng, et. al. discussed MLlib (machine learning library) and the need of this library to benefit from the great wealth of data nowadays. They described how spark is efficient with machine learning algorithms as it is iterative in nature. They also discussed other advantages of integrating MLlib with Spark. They presented the history of spark and MLlib development. They discussed briefly core features, performance, Scalability, and continuous improvements of MLlib library.

In [17], S. Salloum, et. al. presented the importance of big data analytics, Spark framework, and how Spark was initiated. They also presented the core features of Spark, Spark Components, how Apache Spark is perfect when dealing with iterative analysis and algorithms, and Apache Spark case studies in industry. They described Spark API's and Libraries and compared different libraries, compared machine learning packages (spark.ml and spark. MLlib), and the use of each of them. Other features and packages were introduced like Graph analysis, stream processing, batch streaming, and interactive analytics.

IV. Dataset

The dataset describes yellow taxi trips throughout January 2017. NYC Taxi and Limousine Commission (TLC) shares a billion of data through their website [18].

This data was collected by technology providers authorized under the Taxicab & Livery Passenger Enhancement Programs (TPEP/LPEP) and provided to NYC Taxi and Limousine Commission. This data was not generated but collected from real life. This dataset contains 1048575 trip records. Each trip record contains a timestamp for pickup and drop-off, the id of the vendor that provided the record, pickup and drop-off location, passenger count, trip distance, payment type and detailed information about the amount paid by the passengers. Data dictionary was illustrated in Table I.

TABLE I. Data Dictionary

Attribute Name	**Description**
Vendor ID	A code indicating the TPEP provider that provided the record. 1= Creative Mobile Technologies, LLC. 2= VeriFone Inc.
pickup Date Time	The date and time when the meter were engaged, it was categorized based on hour interval **0:3 = a; 4:7=b; 8:11=c; 12:15=d; 16:19=e; 20:23=f.**
Dropoff Date Time	The date and time when the meter were disengaged, it was categorized based on hour interval **0:3 = a; 4:7=b; 8:11=c; 12:15=d; 16:19=e; 20:23=f.**
Passenger Count	The number of passengers in the vehicle. The driver enters this value.
Trip Distance	The elapsed trip distance in miles reported by the taximeter.
PULocation ID	TLC Taxi Zone in which the taximeter was engaged
DOLocation ID	TLC Taxi Zone in which the taximeter was disengaged
Rate Code ID	The final rate code in effect at the end of the trip. 1= Standard rate 2=JFK 3=Newark 4=Nassau or Westchester 5=Negotiated fare 6=Group ride
Store and Fwd. Flag	This flag indicates whether the trip record was held in vehicle memory before it is sent to the vendor, aka "store and forward," because the vehicle did not have a connection to the server. Y= store and forward trip

	N= not a store and forward trip
Payment Type	A numeric code signifying how the passenger paid for the trip. 1= Credit card 2= Cash 3= No charge 4= Dispute 5= Unknown 6= Voided trip
Fare Amount	The time-and-distance fare calculated by the meter.
Extra	Miscellaneous extras and surcharges. Currently, this only includes the $0.50 and $1 rush hour and overnight charges.
MTA Tax	$0.50 MTA tax that is automatically triggered based on the metered rate in use
Improvement Surcharge	$0.30 improvement surcharge assessed trips at the flag drop. The improvement surcharge began being levied in 2015
Tip Amount	Tip amount – This field is automatically populated for credit card tips. Cash tips are not included, this column was categorized: 0.0 = no tips; 1.0 = tips
Tolls Amount	Total amount of all tolls paid in trip.
Total Amount	The total amount charged to passengers (Does not include cash tips).

In Fig. 2, a 2D correlation matrix was presented to declare the relations between the data variables.

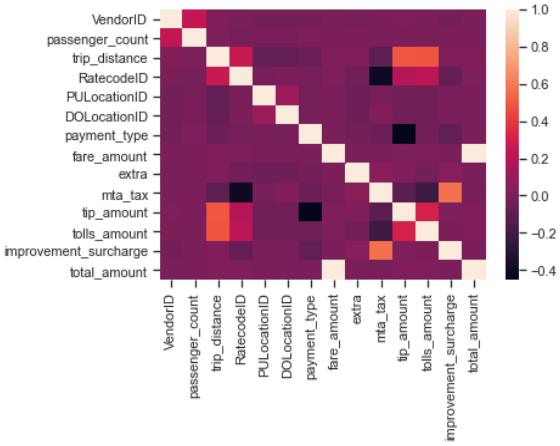

Fig. 2. Covariance Heatmap.

V. Data Preparation

In the dataset, first, the total amount column was removed because this column has a high correlation with other columns. Second, The Improvement surcharge column were removed because it has a constant value. Third, pickup-Date-Time and drop-off-Date-Time is timestamp column with thousands of unique instances, so dividing it into categories was the best solution for this research and each category has a range of hours. Fig. 3 introduces the framework of applying decision tree algorithm on taxi dataset.

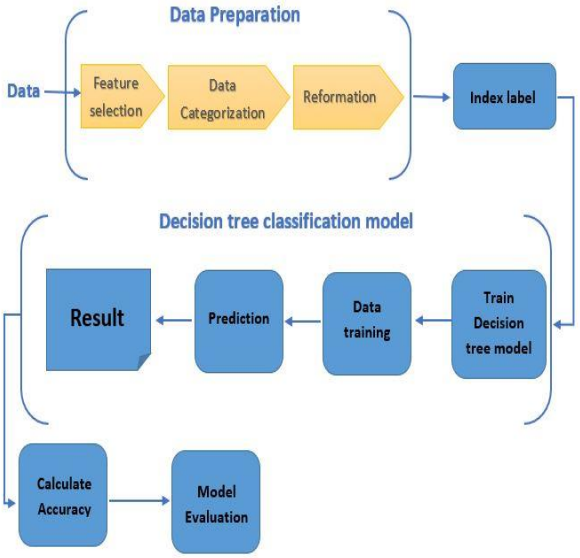

Fig. 3. The Framework of Applying Decision Tree Algorithm on Taxi Data.

VI. Classification on Spark

In [19] K. Zhao, et. al. they analyzed Uber and yellow taxi samples in NYC using three predictors: Neural Network, Markov predictor, and Lempel-Ziv-Welch predictors. They calculated accuracy based on different features; Neural Network was the only machine learning based method used in this comparison. Until now, there are not enough research papers applied in this dataset. Although Classification achieved good accuracy when applied in different applications like healthcare and business [20] [21] [22], and also decision tree was used in different applications and achieved high accuracy [5] [23], decision tree classification was not applied on taxi dataset yet. Based on those previous researches, decision tree classification was used with Apache Spark tool to calculate the accuracy of prediction. This research used Java language to apply classification algorithm on Apache Spark. This research used Spark version 2.2.1. When applying classification algorithm on taxi dataset, some issues appeared; some are related to the data as it needs some preparation before executing the algorithm and the others are related to the algorithm. After data preparation, the following steps were taken to apply the classification algorithm to NYC taxi dataset:

1) Load the data stored in JSON format.
2) Use "String Indexer Model" to index labels and add metadata to the label column.
3) Identify and index categorical feature.
4) Split the data into training and test sets of data.
5) Set Decision tree model.
6) Convert Indexed Labels to original labels.
7) Use Pipeline to chain indexer and tree.
8) Start training model and make prediction.
9) Compute test errors and accuracy.

VII. RESULTS

This model was executed in 5 minutes and 7.823 seconds. The algorithm was applied on 8GB Ram and 2 GHz processor core i7. The resulted accuracy of this model is 96.5%.

This research will use another evaluation metrics to evaluate this algorithm. There are four main categories that are used to calculate these metrics. In a supervised classification problem, there exists a true output and a predicted or model generated output. Therefore, the result of each data point will be assigned to one of the following categories:

- True Positive (TP): label is positive, and prediction is positive.
- True Negative (TN): label is negative, and prediction is negative.
- False Positive (FP): label is negative, but prediction is positive.
- False negative (FN): label is positive, but prediction is negative.

These four categories are the basics for most classification evaluation metrics. Metrics like precision and recall consider the type of error, while F-measure captures the balance between precision and recall and combine them to calculate F-measure. Precision (Positive Predictive Value) measures how many selected items from the dataset are relevant, while recall (True Positive Rate) measure how many relevant items are selected. For Example, if dataset contains 100 taxi trips containing 75 trips that were paid by credit cards and 25 trips were paid in cash. And let's suppose an algorithm for detecting credit card payment type identifies 60 trips paid with credit card, 55 were actually paid with a credit card (true positive) while the rest was paid in cash (false positive). Using the following formula, precision could be calculated as (PPV = 55/60 = 0.92) and recall (TPR = 60/75 = 0.8). High precision means that the algorithm returned relevant results more than irrelevant ones (from 60 trips the returned 55 that are really paid with credit card). High recall means that the algorithm returns most of the relevant results (there were 75 trips paid with a credit card in the dataset, but the algorithm identified only 55 trips).

$$\mathrm{PPV} = \frac{\mathrm{TP}}{\mathrm{TP+FP}} \tag{1}$$

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{P}} = \frac{\mathrm{TP}}{\mathrm{TP+FN}} \tag{2}$$

$$\mathrm{F} = 2 \cdot \frac{\mathrm{precision} \cdot \mathrm{recall}}{\mathrm{precision+recall}} \tag{3}$$

In binary classification, there are only two possible class labels. However, in Multiclass classification there are many class labels as in this research dataset, payment type are classified from 1 to 6 digits having 6 possible classes. Precision and recall were calculated for each class label then weighted precision, recall, and F-measure were calculated. The following table compares accuracy, test error, weighted precision, weighted recall, and weighted F-measure when changing the splitting criteria of the data into training and testing set. In the first case, 10% of data was held for testing. In the Second case, 20% of data was held for testing and so on.

The decision tree is a gradual algorithm which performs a recursive partitioning for the feature. Each partition is selected gradually by selecting the best split from several possible splits to maximize the information gain of a tree node. The node impurity is a measure of dividing the training data of the labels at the node into relatively homogenous subsets. There are two impurity measures for classification: Gini impurity and Entropy. Next Graphs compares evaluation metrics in terms of different node impurity. The node impurity measures the labels homogeneity at the node with two main impurity measures for classification. Table III illustrates the Gini impurity and Entropy formulas.

This research will use splitting criteria (0.8, 0.2) 80% for the training set and 20% held for the testing set as it gets the best results compared to other splitting criteria in Table II. Table IV compare time and number of evaluation metrics when using different impurity measures. Fig. 4, Visualize the difference between accuracy of Gini impurity and accuracy of Entropy, Entropy achieves the best Accuracy. Fig. 5, Visualize the difference between precision of Gini impurity and precision of Entropy, Gini impurity achieves highest precision with a slight difference of 0.03. Fig. 6, Visualize the difference between recall of Gini impurity and recall of Entropy, Entropy achieves the highest recall. Fig. 7, Visualize the difference between f-measure of Gini impurity and f-measure of Entropy, Entropy achieves the highest f-measure.

TABLE II. EVALUATION METRICS FOR DIFFERENT SPLITTING CRITERIA

	Splitting criteria [0.9,0.1]	**Splitting criteria [0.8,0.2]**	**Splitting criteria [0.7,0.3]**
Accuracy	96.5%	96.6%	96.6%
Test error	3.5%	3.4%	3.4%
Weighted precision	96.47%	96.57%	96.52%
Weighted recall	96.53%	96.6%	96.56%
Weighted F-measure	96.3%	96.4%	96.3%

TABLE III. NODE IMPURITY FORMULAS

Impurity	**Formula**	**Description**
Gini impurity	$\sum_{i=1}^{C} f_i(1-f_i)$	$\boldsymbol{f_i}$ is the frequency of label $\boldsymbol{i}$ at a node and $\boldsymbol{C}$ is the number of unique labels.
Entropy	$\sum_{i=1}^{C} -f_i\, log(f_i)$	

TABLE IV. EVALUATION METRICS FOR NODE IMPURITY

	Gini impurity	**Entropy**
Time	5:45s	5:58s
Accuracy	96.53%	96.6%
Test error	3.5%	3.4%
Weighted precision	96.55%	96.52%
Weighted recall	96.52%	96.57%
Weighted F-measure	96.29%	96.34%

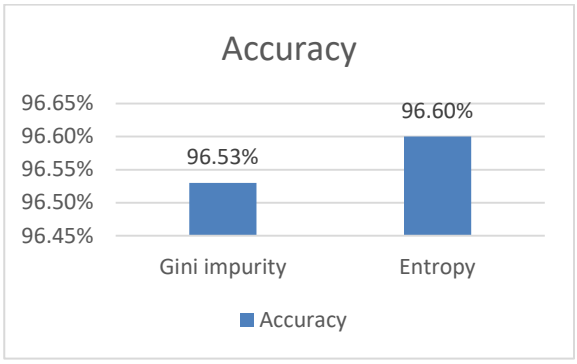

Fig. 4. Accuracy for Node Impurity.

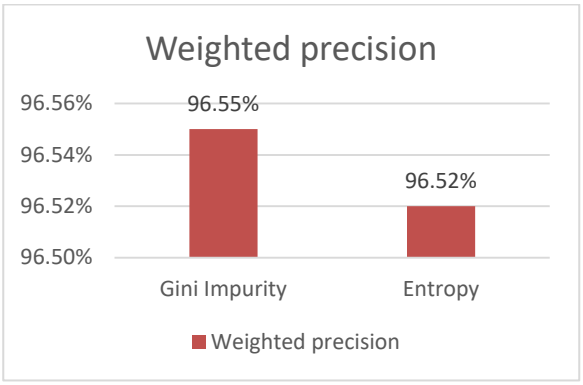

Fig. 5. Precision for Node Impurity.

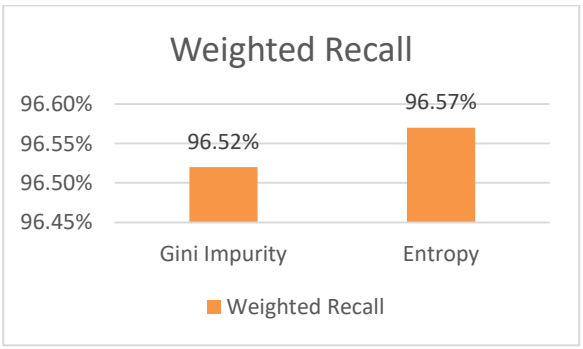

Fig. 6. Recall for Node Impurity.

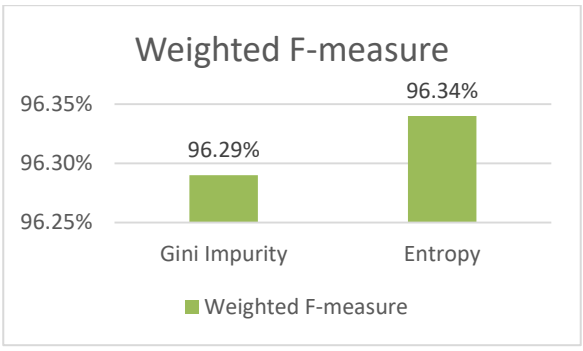

Fig. 7. F-measure for Node Impurity.

VIII. Conclusion

This research applied a decision tree classification model on NYC Taxi and Limousine Commission dataset to predict payment type with varying hyperparameters. The dataset contains detailed taxi trips, and each trip record describes the vendor of the data record, passenger count, pickup and drop-off location and timestamp and detailed receipt. Accuracy, precision, recall and f-measure were calculated with different splitting criteria and different node impurity. The experiment shows a promising result as the accuracy and other evaluation metrics is higher than 96%. For future work, the Decision tree classification model can be applied to the same dataset to predict passenger count, rush hour to predict extra charges, and high demand in a specific time zone. Also, this model can be applied to car services in Egypt like Uber and Careem. An application of Grid search cross validation can be applied too to get the best hyperparameters for this dataset.

References

[1] An introduction to big data", *Opensource.com*, 2018. [Online]. Available: https://opensource.com/resources/big-data. [Accessed: 13- Sep- 2018].

[2] Che, D., Safran, M. and Peng, Z. (2013). From Big Data to Big Data Mining: Challenges, Issues, and Opportunities. Database Systems for Advanced Applications. 7827, 1-15.

[3] L. Joseji, "6 Sparkling Features of Apache Spark! - DZone Big Data", dzone.com, 2014. [Online]. Available: https://dzone.com/articles/6-sparkling-features-apache. [Accessed: 24- Sep- 2018].

[4] H. Karau and R. Warren, High performance Spark. O'Reilly Media, 2017, pp. 219 - 251.

[5] H. Sharma and S. Kumar, "A Survey on Decision Tree Algorithms of Classification in Data Mining", International Journal of Science and Research (IJSR), vol. 5, no. 4, pp. 2094-2097, 2016.

[6] B. Charbuty and A. Abdulazeez, "Classification Based on Decision Tree Algorithm for Machine Learning", *Journal of Applied Science and Technology Trends*, vol. 2, no. 01, pp. 20-28, 2021.

[7] F. Wang, "Analysis of NYC Yellow Taxi data", NYC Data Science Academy Blog, 2016. [Online]. Available: https://nycdatascience.com/blog/student-works/analysis-of-nyc-yellow-taxi-data/. [Accessed: 11- Nov- 2018].

[8] M. Yazici, C. Kamga and A. Singhal, "A Big Data Driven Model for Taxi Drivers' Airport Pick-up Decisions in New York City", *2013 IEEE International Conference on Big Data*, pp. 37-44, 2013.

[9] A. Kushwaha, A. Kar and Y. Dwivedi, "Applications of big data in emerging management disciplines: A literature review using text mining", *International Journal of Information Management Data Insights*, vol. 1, no. 2, p. 100017, 2021.

[10] J. Wang, C. Xu, J. Zhang and R. Zhong, "Big data analytics for intelligent manufacturing systems: A review", *Journal of Manufacturing Systems*, 2021.

[11] S. Kumar and M. Singh, "Big data analytics for healthcare industry: impact, applications, and tools", *Big Data Mining and Analytics*, vol. 2, no. 1, pp. 48-57, 2019.

[12] M. Juez-Gil, Á. Arnaiz-González, J. Rodríguez, C. López-Nozal and C. García-Osorio, "Approx-SMOTE: Fast SMOTE for Big Data on Apache Spark", *Neurocomputing*, vol. 464, pp. 432-437, 2021.

[13] B. Roy and D. Rout, "Predicting Taxi Travel Time Using Machine Learning Techniques Considering Weekend and Holidays", *Proceedings of the 13th International Conference on Soft Computing and Pattern Recognition (SoCPaR 2021)*, pp. 258-267, 2022.

[14] S. Singh and M. Giri, "Comparative Study Id3, Cart and C4.5 Decision Tree Algorithm: A Survey", *International Journal of Advanced Information Science and Technology (IJAIST)*, vol. 3, no. 7, pp. 47-52, 2014.

[15] H. Sun and S. McIntosh, "Big Data Mobile Services for New York City Taxi Riders and Drivers", *IEEE International Conference on Mobile Services*, pp. 57-64, 2016.

[16] X. Meng, J. Bradley, B. Yavuz, E. Sparks, S. Venkataraman, D. Liu, J. Freeman, D. Tsai, M. Amde, S. Owen, D. Xin, R. Xin, M. J. Franklin, R.

Zadeh, M. Zaharia and A. Talwalkar, "MLlib: Machine Learning in Apache Spark", *Journal of Machine Learning Research*, vol. 17, pp. 1235-1241, 2016.

[17] S. Salloum, R. Dautov, X. Chen, P. Peng and J. Huang, "Big data analytics on Apache Spark", *International Journal of Data Science and Analytics*, vol. 1, no. 3-4, pp. 145-164, 2016.

[18] [Dataset] "NYC Taxi & Limousine Commission - Trip Record Data", *Nyc.gov*, 2018. [Online]. Available: http://www.nyc.gov/html/tlc/html/about/trip_record_data.shtml. [Accessed: 22- Sep- 2018].

[19] K. Zhao, D. Khryashchev, J. Freire, C. Silva and H. Vo, "Predicting Taxi Demand at High Spatial Resolution: Approaching the Limit of Predictability", *IEEE International Conference on Big Data (Big Data)*, pp. 833-842, 2016.

[20] V. Chaurasia and S. Pal, "A Novel Approach for Breast Cancer Detection using Data Mining Techniques", *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 2, no. 1, pp. 2456 - 2465, 2014.

[21] A. Linden and P. Yarnold, "Using data mining techniques to characterize participation in observational studies", *Journal of Evaluation in Clinical Practice*, vol. 22, no. 6, pp. 839-847, 2016.

[22] T. Bahari and M. Elayidom, "An Efficient CRM-Data Mining Framework for the Prediction of Customer Behaviour", *International Conference on Information and Communication Technologies*, vol. 46, pp. 725-731, 2015.

[23] Y. SONG and Y. LU, "Decision tree methods: applications for classification and prediction", *Shanghai Arch Psychiatry.*, pp. 130-135, 2015.