Customer Churn Prediction in Telecommunications Industry

Big Data Parallel Programming (DT8034): Project Report Analysis of Program (1)

By Mohammad Hammoud

Contents

1. Introduction

In today's highly competitive telecommunications market, retaining customers and minimizing churn is critical for businesses to maintain profitability and grow their customer base. Customer churn refers to the loss of customers to competing service providers. This project aims to develop a machine learning model that accurately predicts customer churn in the telecommunications industry using Apache Spark's MLlib library. By identifying customers at risk of churning, companies can implement targeted retention strategies to enhance customer loyalty, mitigate revenue loss, and improve overall business performance.

We will utilize the Telco Customer Churn dataset, available on Kaggle (https://www.kaggle.com/blastchar/telco-customer-churn), containing approximately 7,000 data points with comprehensive information on customer attributes, churn status, and service usage. The dataset will be preprocessed to extract relevant features for training various machine learning models, including DecisionTreeClassifier, RandomForestClassifier, GBTClassifier, and LogisticRegression.

The primary goal is to predict the likelihood of a customer churning, which will be approached as a binary classification problem. The target variable is the customer churn status, represented as Yes or No. The performance of the selected machine learning algorithms will be compared based on their accuracy, training time, and other relevant metrics to identify the best-performing model for predicting customer churn.

By creating and deploying a robust machine learning model with high predictive accuracy, we aim to support telecommunications companies in their efforts to retain customers, enhance customer loyalty, and ultimately improve overall business performance.

2. Data Preprocessing

The initial phase of any machine learning project involves preprocessing the data to prepare it for training with the chosen machine learning algorithms. This section outlines the steps taken to preprocess the Telecom customer churn dataset.

2.2 Data Cleaning and Transformation

We began by loading the Telecom customer churn dataset into a PySpark DataFrame, displaying the first row, number of rows, and column names to understand the structure of the data. We then examined the schema of the DataFrame to identify the data types for each column. Numeric columns were selected and cast to appropriate data types, and we inspected the summary statistics DataFrame in a transposed format to identify any unusual values.

For instances where TotalCharges could not be cast to double, we created a new column with null values, filled null values with the previous value using a window function, and discarded the original and intermediate columns. Next, we applied the StringIndexer to string columns and dropped the original string columns to prepare the data for machine learning algorithms. We then displayed the resulting data frame to verify any data changes.

2.3 Exploratory Data Analysis

We further examined the summary statistics using the describe() function to understand the dataset and converted the PySpark DataFrame to a Pandas DataFrame for easier manipulation. We created separate DataFrames for churned and non-churned customers, plotting heatmaps to compare the mean values of features for both groups as shown in Figure 1.

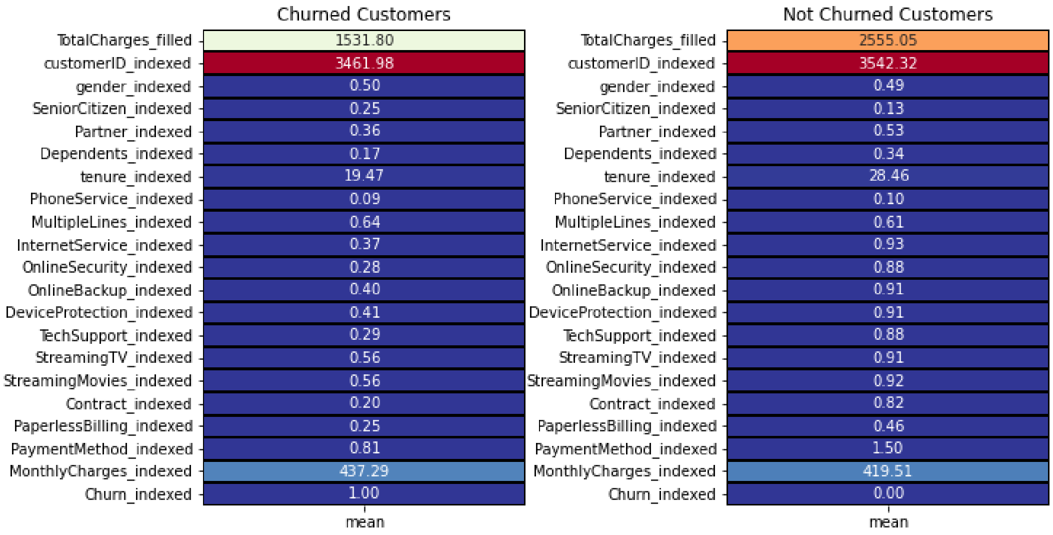

Figure 1: Compare mean values of features for Churned and Not Churned groups.

We reviewed the schema to determine the data types for each column, classifying them into numerical and categorical features. We calculated the percentages of churned and non-churned customers, creating pie charts and bar charts to visualize these figures. The categorical features were divided into customer information, Services Signed Up for, and Payment Information. We used bar charts to visualize the relationship between each feature within these groups and churn.

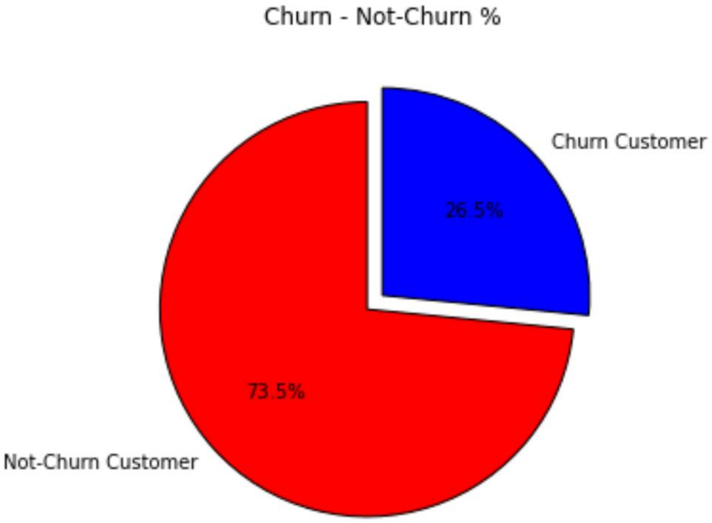

Figure 2: Percentage of Churn against Not Churn.

Further dataset exploration involved creating a list of columns to plot, designing subplots for each column, and plotting countplots using Seaborn. We converted the PySpark DataFrame to a Pandas DataFrame for additional visualization and plotted countplots for 'Contract_indexed', 'PaperlessBilling_indexed', and 'PaymentMethod_indexed' as shown in Figure 3. We also calculated the percentages for each category in 'gender', 'SeniorCitizen', 'Partner', and 'Dependents' as shown in Figure 4, plotting pie charts for each in Figure 5.

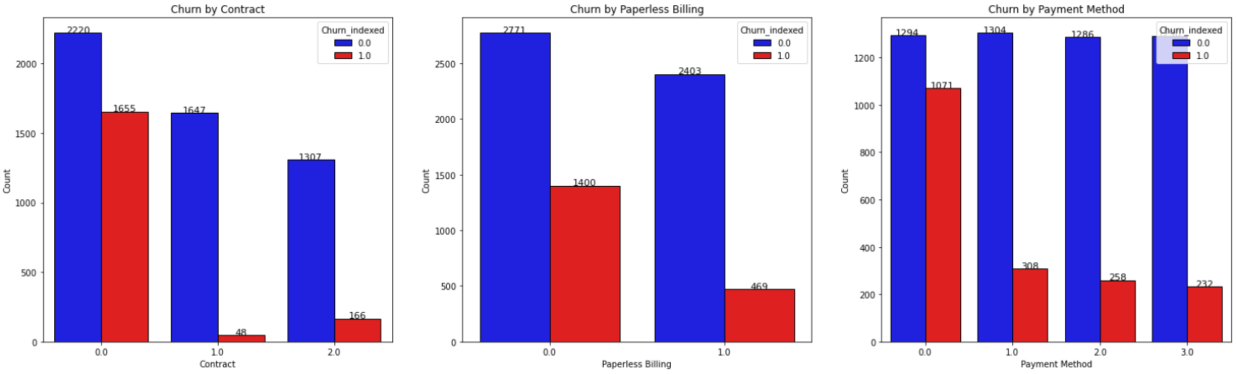

Figure 3: Countplots to visualize relationship between churn and 'Contract_indexed', 'PaperlessBilling_indexed', and 'PaymentMethod_indexed'.

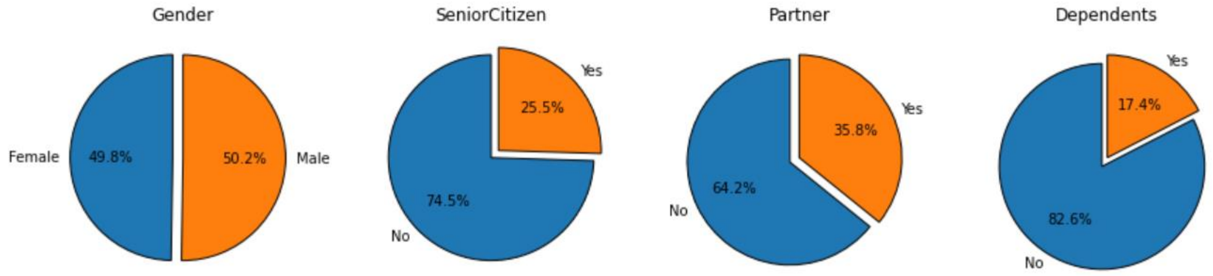

Figure 4: Percentages for each category.

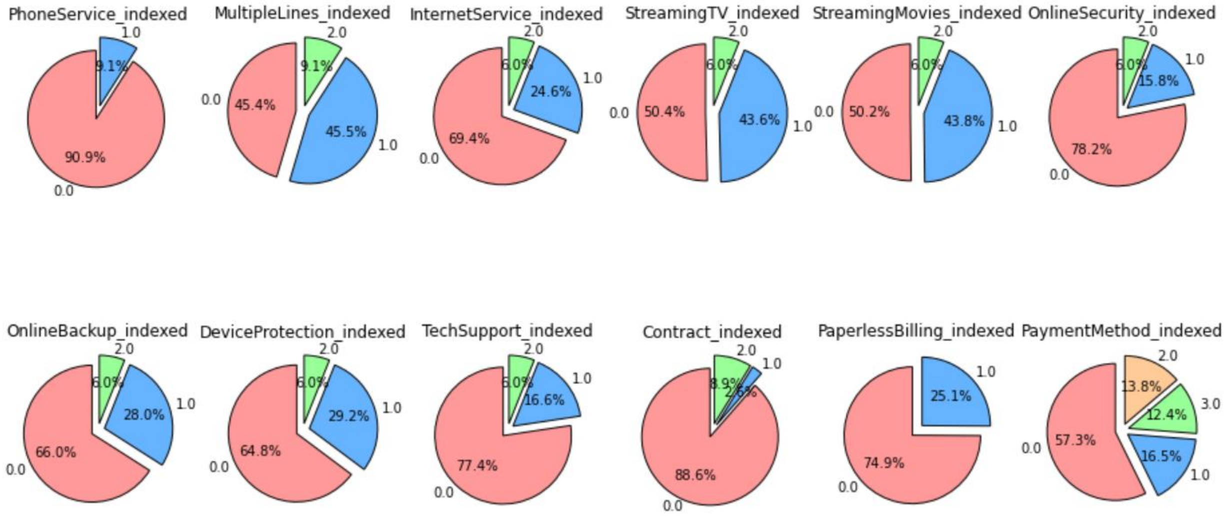

Figure 5: A set of pie charts to visualize the distribution of the churned customers for each service.

By filtering churned customers, we analyzed the distribution of churned customers for each service, plotting pie charts to represent the distribution of churned customers per service.

2.4 Feature Selection

To identify the most relevant features for machine learning algorithms, we created a heatmap to visualize correlations between features as shown in Figure 6 and 7. We dropped columns with low correlation to 'Churn_indexed' to form a new DataFrame (df1).

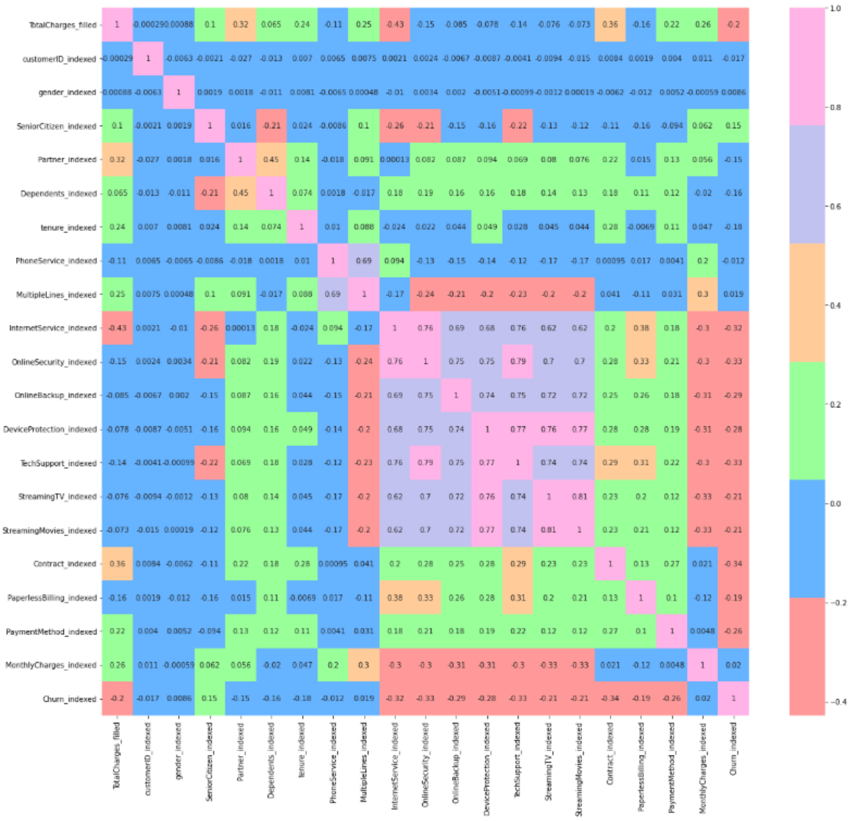

Figure 6: Heatmap to visualize correlations between features.

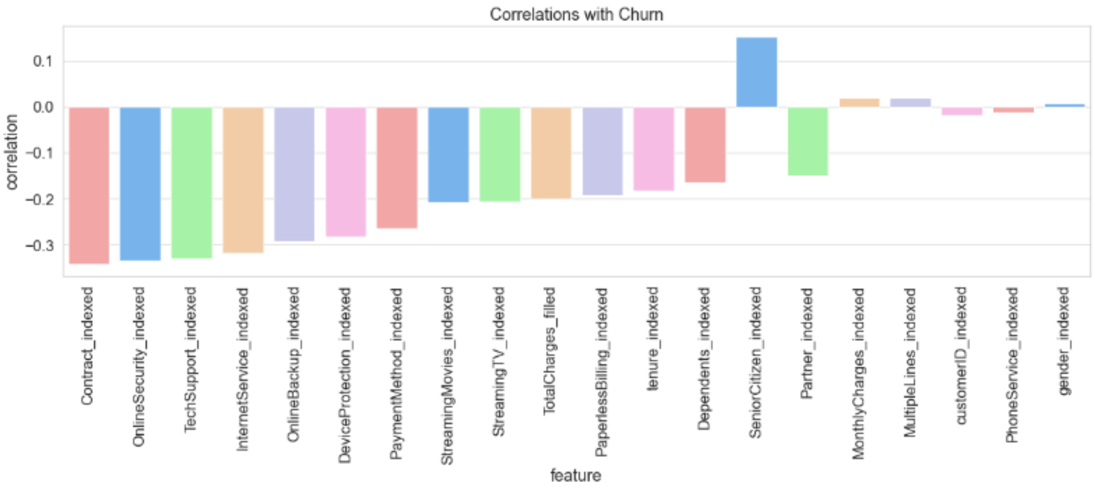

Figure 7: A bar plot of the correlations between the features and 'Churn_indexed'.

2.5 Data Balancing

To balance the dataset, we oversampled negative cases and combined them with positive cases to create a new DataFrame (balanced_df). We then reviewed the schema of the balanced DataFrame to ensure data types for each column remained consistent as shown in Table 1. This balanced dataset was then used for training and evaluating the chosen machine learning algorithms, ensuring that the models are not biased towards any particular class.

Table 1: shows the distribution of churned (1.0) and non-churned (0.0) customers in balanced_df.

Churn_indexed	Count
1.0	1869
0.0	1891

3. Spark Implementation

In this section, Various classification models utilizing machine learning algorithms can be employed to predict customer churn in the telecommunications industry. We focus on tree-based and logistic regression algorithms, including DecisionTreeClassifier, RandomForestClassifier, GBTClassifier, and LogisticRegression.

3.2 DecisionTreeClassifier

We started by implementing the DecisionTreeClassifier, a tree-based method that recursively splits the input space for predictions. We trained a decision tree model for customer churn prediction by specifying the features and target variables. We selected the model's parameters, such as the maximum depth, to optimize the model's performance. After training the model, we evaluated its performance using the test dataset and calculated relevant metrics, such as accuracy and the area under the ROC curve.

3.3 RandomForestClassifier

Next, we implemented the RandomForestClassifier, an ensemble technique aggregating multiple decision trees to enhance prediction accuracy and prevent overfitting. We trained a random forest model for customer churn prediction, specifying the features and target variable and the number of trees to be used in the ensemble. We adjusted the model's parameters to optimize its performance. After training the RandomForestClassifier, we evaluated its performance using the test dataset and calculated the accuracy and the area under the ROC curve.

3.4 GBTClassifier

We then implemented the GBTClassifier, another ensemble method that combines multiple weak learners (decision trees) in a stage-wise fashion to improve prediction accuracy. We trained a gradientboosted trees model for the customer churn prediction problem, specifying the features and target variable, and selected the model's parameters, such as the maximum number of iterations, to optimize its performance. After training the GBTClassifier, we evaluated its performance using the test dataset and calculated relevant metrics, such as accuracy and the area under the ROC curve.

3.5 LogisticRegression

Finally, we implemented the LogisticRegression algorithm, a linear model that predicts the probability of a binary response based on one or more input features. We trained a logistic regression model for the customer churn prediction problem, specifying the features and target variable, and adjusted the model's parameters, such as regularization strength, to optimize its performance. After training the LogisticRegression model, we evaluated its performance using the test dataset and calculated relevant metrics, such as accuracy and the area under the ROC curve.

By employing these algorithms, we developed various classification models for predicting customer churn in the telecommunications industry. We fine-tuned and evaluated the models to identify the best-performing approach, ultimately assisting telecommunications companies in retaining their customers and enhancing their overall business performance.

4. Results

In this section, we present the results obtained from applying the machine learning algorithms discussed in the previous section to the preprocessed Telco Customer Churn dataset. We will compare the performance of each model based on their accuracy, training time, and other relevant metrics.

4.2 DecisionTreeClassifier

The DecisionTreeClassifier model achieved an AUC-ROC of 75.06% on the test dataset. The model's training time was 18.42 seconds. Although the model provided reasonable predictive performance, its interpretability and simplicity may prove valuable for understanding the decision-making process behind customer churn prediction.

4.3 RandomForestClassifier

The RandomForestClassifier model yielded an AUC-ROC of 77.83% on the test dataset, outperforming the DecisionTreeClassifier. The model's training time was 77.28 seconds. This improvement can be attributed to the ensemble approach of RandomForest, which reduces overfitting and enhances the generalizability of the model.

4.4 GBTClassifier

The GBTClassifier model demonstrated an AUC-ROC of 69.03% on the test dataset, which is lower than the performance of the RandomForestClassifier. The model's training time was

417.48 seconds. The stage-wise boosting approach of the GBTClassifier contributes to its improved predictive performance compared to other tree-based methods, but in this case, it did not outperform the RandomForestClassifier.

4.5 LogisticRegression

The LogisticRegression model achieved an AUC-ROC of 83.35% on the test dataset. The model's training time was 11.83 seconds. While the performance was higher than that of tree-based models, the linear nature of the LogisticRegression model may limit its ability to capture complex relationships between features.

4.6 Model Comparison and Conclusion

Based on the results as shown in Table 2, the LogisticRegression exhibited the highest AUC-ROC in predicting customer churn, followed by RandomForestClassifier, DecisionTreeClassifier, and GBTClassifier. However, the training time and complexity of each model should also be considered when selecting the best approach for a specific use case. In this context, LogisticRegression proves to be an efficient choice due to its high accuracy and relatively short training time, making it a suitable model for the customer churn prediction problem in the telecommunications industry.

Table 2: Shows AUC, training time, and evaluation time for each model.

Algorithm	**AUC-ROC**	**Training Time (sec)**	**Evaluation Time (sec)**
DecisionTreeClassifier	0.7506	18.42	0.47
RandomForestClassifier	0.7783	77.28	0.99
GBTClassifier	0.6903	417.48	183.64
LogisticRegression	0.8335	11.83	0.45

5. Conclusion

In conclusion, this project aimed to develop a machine-learning model that accurately predicts customer churn in the telecommunications industry. By leveraging Apache Spark and its MLlib library, we could preprocess the Telco Customer Churn dataset and apply various machine learning algorithms, including DecisionTreeClassifier, RandomForestClassifier, GBTClassifier, and LogisticRegression.

Our analysis revealed that the LogisticRegression model achieved the highest AUC-ROC score, followed by RandomForestClassifier, DecisionTreeClassifier, and GBTClassifier. While each algorithm has its merits, the LogisticRegression model emerged as the most suitable for this problem due to its high AUC-ROC score and relatively short training time.

By implementing a robust machine learning model with high predictive accuracy for customer churn, telecommunications companies can better identify customers at risk of churning and deploy targeted retention strategies. This will enhance customer loyalty and mitigate revenue loss, ultimately improving the overall business performance in the highly competitive telecommunications industry.

Future work on this project may include exploring other machine learning algorithms, feature engineering techniques, and hyperparameter optimization methods to enhance the model's performance further. Additionally, the model's practical deployment in real-world settings should be examined to assess its effectiveness and adaptability in a constantly evolving telecommunications landscape.