Predicting Telecommunication Customers Churn Using Machine Learning.

Abdul-Kamil Fusheini
Computer Science Department,
College of Science, Engineering, and Technology,
Norfolk State University, Norfolk, 23504, Virginia, USA
a.fusheini@spartans.nsu.edu; aakamyl@gmail.com

Abstract **- The customer churn problem poses a significant challenge for telecommunication companies (Telcos), given the higher costs associated with acquiring new customers than retaining existing ones. In this paper, I review a customer churn prediction approach proposed by researchers in my reference article "Telecommunication Customers Churn Prediction using Machine Learning". This approach is based on usage patterns, employing various machine learning models, including linear regression, random forest, Support Vector Machine (SVM), K-Nearest-Neighbor (KNN), and decision tree. To make a significant contribution to the approach, I consider two new models namely artificial neural network and naïve Bayes in addition to the best model in the reference article namely random forest, and one other model which is decision trees. I evaluate the prediction performance of these models using the same metrics namely accuracy, recall, precision, F1-score, and area under the curve (AUC) to allow for comparisons. My findings show that the random forest model outperforms all the others, matching the remarkable accuracy of 95.5% achieved by my reference article on the dataset utilized. This study complements and corroborates the work in my reference article and therefore contributes to the understanding of customer churn prediction in the telecommunications industry, providing valuable insights for Telcos to devise effective retention strategies.**

Keywords *– churn prediction, machine learning, telecommunication service, telco, classification, decision tree, random forest, artificial neural network, naïve Bayes.*

I. INTRODUCTION

In the evolving telecommunications industry, marked by a surge in connected devices and subscribers, telecom companies (Telcos) are in a relentless battle to expand their customer base. However, amidst this fervent competition, they encounter a formidable challenge known as customer churn - the departure of existing customers and termination of their contracts [1]. This phenomenon, often driven by suboptimal service quality and heightened competition from rivals offering more enticing deals, poses a significant threat to Telcos' profitability and market share.

Recognizing the exorbitant costs of acquiring new customers [2], Telcos adopt proactive measures to retain their existing customer base. One such strategy gaining prominence is the anticipation of potential churners through predictive analytics and the subsequent implementation of personalized retention initiatives tailored to preserve customer loyalty.

In this context, machine learning becomes a pivotal tool for Telcos to forecast customer churn based on intricate usage data patterns. However, developing an effective churn prediction model presents formidable hurdles, including the need for extensive data volumes, intricate feature spaces, and the pervasive challenge of class imbalance [3]. To mitigate these obstacles, innovative techniques such as oversampling [4] and re-sampling are deployed to bolster the robustness of the predictive models.

The quest for optimal churn prediction models has spurred extensive research into various machine learning classifiers. Studies have explored many algorithms, ranging from logistic regression and decision trees to artificial neural networks (ANNs) and support vector machines (SVMs). Notably, the random forest has emerged as a favored choice, exhibiting superior performance [1], particularly after addressing class imbalance through innovative re-sampling methods. Nonetheless, the absence of a universally applicable churn prediction model underscores the inherent variability in algorithm performance, contingent upon the unique characteristics of the dataset under scrutiny.

Against this backdrop, the reference article, and my study endeavor to preprocess customer usage data meticulously and discern the most apt machine learning model for churn prediction. By evaluating predictive models based on a comprehensive array of metrics - including recall, accuracy, precision, F1-score, and area under the curve (AUC) - the aim is to furnish Telcos with actionable insights to fortify their customer retention strategies and navigate the dynamic telecommunications landscape with agility and foresight.

II. EXPLORATORY DATA ANALYSIS

A. Dataset

To allow for comparisons, I used the dataset described in my reference article which is new and different from those I have used in class assignments over the semester. This new dataset is sourced from the Kaggle database [5], originally released by the CrowdAnalytix community for their churn prediction competition [2]. The dataset does not mention the identity of the Telco and includes 20 categorical columns and the target column. There are 3333 unique customer records represented in the rows. The dataset has no null or missing values, and the names and definitions of the columns are the following:

- state: The geographical state of the customer.
- account_length: The duration of customer account tenure in days.
- area_code: The area code corresponding to the customer's location.
- phone_number: The unique phone number associated with the customer.
- international_plan: Indicates whether the customer is subscribed to an international plan (YES/NO).
- voicemail_plan: Indicates whether the customer is subscribed to a voicemail plan (YES/NO).
- number_vmail_messages: The number of voicemail messages sent by the customer.
- total_day_minutes: Total duration of calls made by the customer during the day.
- total_day_calls: Total number of calls made by the customer during the day.
- total_day_charges: Total charges incurred for calls made by the customer during the day.
- total_eve_minutes: Total duration of calls made by the customer in the evening.
- total_eve_calls: Total number of calls made by the customer in the evening.
- total_eve_charges: Total charges incurred for calls made by the customer in the evening.
- total_night_minutes: Total duration of calls made by the customer at night.
- total_night_calls: Total number of calls made by the customer at night.
- total_night_charges: Total charges incurred for calls made by the customer at night.
- total_intl_minutes: Total duration of international calls made by the customer.
- total_intl_calls: Total number of international calls made by the customer.
- total_intl_charges: Total charges incurred for international calls made by the customer.
- customer_service_calls: Total number of customer service calls made by the customer.
- churn: Indicates whether the customer churned (TRUE) or not (FALSE).

B. Data Analysis

The data analysis is conducted in Python3 using Jupyter Notebook version 6.5.4 and Anaconda Navigator. Various packages were employed, including 'Pandas' for data manipulation, 'Numpy' for scientific computing, 'Scikit-learn' for data mining and analysis, and 'Imblearn' for resampling techniques to address the class imbalance. 'Matplotlib', 'Seaborn', and 'Scipy' were used to generate graphs. I did exploratory data analysis as a first step leveraging data visualization techniques to discern patterns within the dataset.

A bar chart for churn distribution in Figure 1 revealed severe class imbalance, with as low as 14.49% for churn customers. To mitigate this imbalance, the Synthetic Minority Over Sampling Technique (SMOTE) is employed to generate synthetic data points based on the nearest neighbors, thereby enhancing prediction accuracy on the test data. This greatly improved AUC but reduced accuracy slightly as a trade.

Further exploration plotted a histogram for the total day charge feature in Figure 2 a heatmap visualization for pairwise correlations of all features in Figure 3, and a dendrogram for all features in Figure 4.

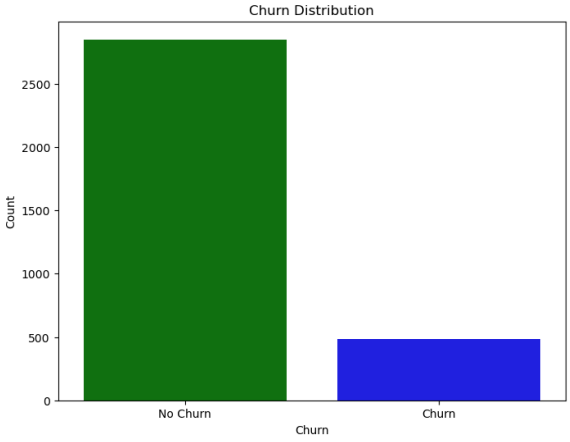

Fig. 1. Churn distribution

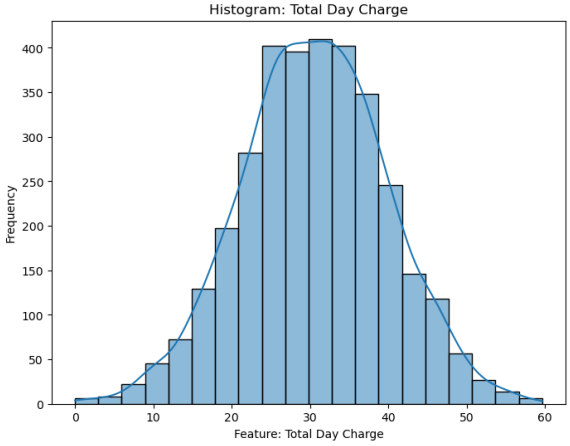

Fig. 2. Histogram for total day charge

The heatmap in Figure 3 reveals the correlation between pairs of columns. A high correlation exists between the number of voicemail messages and the voicemail plan feature as customers with an active voicemail plan are likely to send more voicemail messages. Similarly, a strong correlation exists between total day minutes and total day charges, which is expected since the telecom company typically charges customers based on call duration. This correlation pattern is also evident in international, evening, and night calls.

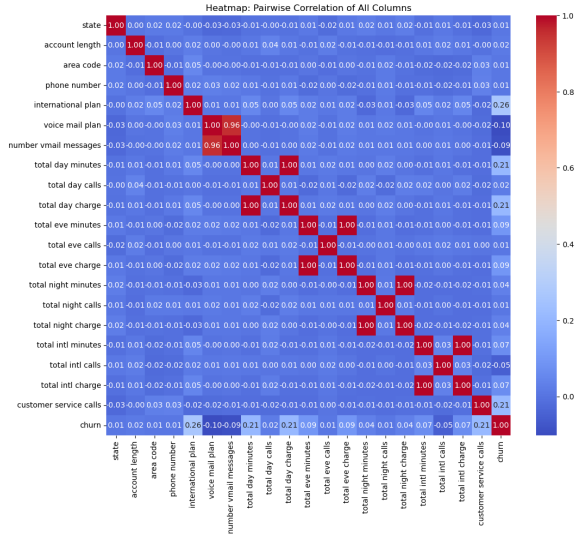

Fig. 3. Heatmap of pairwise correlation of the columns

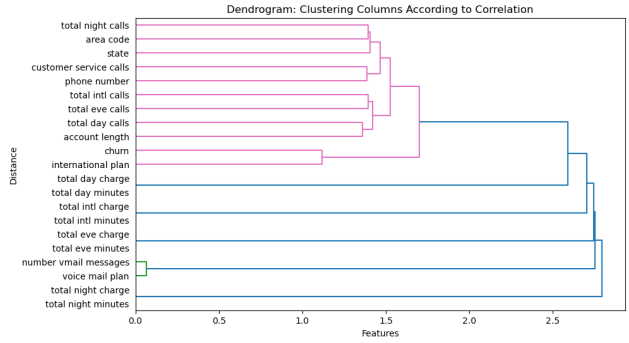

Fig. 4. Dendrogram of clustering the features according to correlation

In the dendrogram, four sets of features are closely clustered together, indicating a strong correlation. These feature pairs consist of total night charges and total night minutes, total evening charges and total evening minutes, total day charges and total day minutes, and total international charges and total international minutes.

III. PREDICTION PERFORMANCE

A confusion matrix is a table that is often used to describe the performance of a classification model on a set of test data for which the true values are known. It allows visualization of the performance of an algorithm by showing the counts (in this case the proportion) of true positive (TP), true negative (TN), false positive (FP), and false negative (FN)

predictions. These are used to calculate the performance metric.

- True Positives (TP) are the correctly classified positive samples.
- True Negatives (TN) are the correctly classified negative samples.
- False Positives (FP) are the incorrectly classified positive samples.
- False Negatives (FN) are the incorrectly classified negative samples.

Table I displays the confusion matrix.

TABLE I. CONFUSION MATRIX

	Predicted Class	
Actual class	Negative	Positive
Negative	True Negative (TN)	False Positive (FP)
Positive	False Negative (FN)	True Positive (TP)

- Precision: it measures the proportion of correctly classified positive samples among all samples classified as positive. It is calculated using (1).

$$Precision = TP / (TP + FP) \quad (1)$$

- Recall (Sensitivity): recall measures the proportion of correctly classified positive samples among all actual positive samples. It is calculated using (2).

$$Recall = TP / (TP + FN) \quad (2)$$

- F1-Score: F1-Score is the harmonic mean of precision and recall. It provides a balance between precision and recall, especially when there is an imbalance between the number of positive and negative samples. It is calculated using (3).

$$F1 = 2((Precision \times Recall)/(Precision + Recall)) \quad (3)$$

- Accuracy: it measures the proportion of correctly classified samples among the total number of samples. Accuracy is calculated using (4).

$$Accuracy = (TP + TN)/(TP + FP + TN + FN) \quad (4)$$

- Area Under the Curve (AUC): AUC measures the ability of the model to distinguish between positive and negative classes across different threshold values. It represents the area under the Receiver Operating Characteristic (ROC) curve. In this measure, two parameters were designed which are True Positive Rate (TPR) which is equal to Recall, and False Positive Rate (FPR), as in (5).

$$FPR = FP/(TN + FP) \quad (5)$$

The Receiver Operating Characteristic (ROC) curve is plotted using these two parameters. The curve shows the trade-off between the two parameters.

For the prediction, the dataset is split into training data and test data using the 'train_test_split' function in 'scikit-learn' module according to Table II below.

TABLE II TRAIN TEST SPLIT

Model	Random Forest	Decision Tree	Neural Network	Naïve Bayes
Test Split	20%	25%	20%	25%
Random State	5%	5%	0%	25%

A. *Random Forest*

Random forest employs a strategy of random subset selection to grow decision trees, utilizing a majority vote to arrive at predictions. While exhibiting bias toward predicting no-churn customers, the model demonstrates robust performance, with minimal misclassifications. The RF model correctly predicted 0.988 of actual no-churn customers and 0.75 of actual churn customers resulting in a bias of 0.238 towards predicting no-churn customers. The confusion matrix is shown below in Table III.

TABLE III CONFUSION MATRIX

	Predicted Class	
Actual class	Negative	Positive
Negative	0.988	0.012
Positive	0.25	0.75

B. *Decision Tree*

Decision tree creates a binary tree for classification, selecting attribute splits to maximize information gain. Despite a propensity towards predicting no-churn customers, the model

demonstrates slightly better performance in identifying true churn customers than the RF model and a slightly worse performance in predicting no-churn customers. The DT model correctly predicted 0.969 of actual no-churn customers and 0.77 of actual churn customers resulting in a bias of 0.199 towards predicting no-churn customers. The confusion matrix is shown below in Table IV.

TABLE IV CONFUSION MATRIX FOR DECISION TREE

	Predicted Class	
Actual class	Negative	Positive
Negative	0.969	0.031
Positive	0.23	0.77

C. *Artificial Neural Network (ANN)*

Artificial neural networks are inspired by the structure and functioning of the human brain. It consists of interconnected nodes organized into layers - an input layer, one or more hidden layers, and an output layer. Each node, or neuron, receives input signals, performs a computation, and passes the result to the next layer. ANN classifiers can learn complex patterns and relationships in data, making them applicable to solving classification problems. I considered exploring this technique to make a new contribution to customer churn modeling for the chosen dataset and similar ones. The ANN model showed a severe propensity towards predicting no-churn customers with a bias of 0.977 as it correctly predicted 1.0 of actual no-churn customers and 0.023 of actual churn customers. This shows that it is more sensitive to the class imbalance problem despite resampling using SMOTE. The confusion matrix is shown below in Table V.

TABLE V CONFUSION MATRIX FOR ARTIFICIAL NEURAL NETWORK

	Predicted Class	
Actual class	Negative	Positive
Negative	1	0
Positive	0.977	0.023

D. *Naive Bayes Classifier (BN)*

The Naive Bayes technique is the other new machine learning technique I considered adding to the scope of my reference paper. This classifier is a probabilistic machine learning technique based on Bayes' theorem with the "naive" assumption of independence between features. Despite this simplification, it's remarkably effective in many real-world applications, especially in text classification and spam filtering. The classifier calculates the probability of each class given a set of input features and then selects the class with the highest probability as its prediction. It's computationally efficient, easy to implement, and works well with high-dimensional datasets. However, it may perform poorly if the independence assumption doesn't hold for the data. In my work, BN correctly predicted 0.928 actual no-churn customers and 0.509 actual churn customers with a bias of 0.419 toward predicting no-churn customers. This model therefore performed better than the NN model but less than both the DT and RF models. The confusion matrix is shown below in Table IV.

TABLE VI. CONFUSION MATRIX FOR NAIVE BAYES

	Predicted Class	
Actual class	Negative	Positive
Negative	0.928	0.072
Positive	0.491	0.509

E. *Comparison of Models*

The predictive models are evaluated based on accuracy, precision, recall, F1-score, and AUC. These metrics were retrieved using the accuracy_score, classification_report, and roc_auc_score functions in the "sklearn" module. Table VII shows these metrics for all the models and Figure 5 is a radar chart showing the performance comparison.

Random forest emerges as the top performer, achieving the highest accuracy and F1-score and a minute margin, 0.001, less than the highest precision, recall, and AUC. The accuracy of 95.5% matches the best accuracy achieved in my reference paper for random forest.

Decision tree was the second-best model for this dataset. Neural networks produced the highest precision but did not perform well on other metrics.

Naïve Bayes was not the best performer on any metric but overall performed better than neural networks.

TABLE VII. EVALUATION METRIC COMPARISON

Predictive Models	Accuracy	Precision	Recall	F1-Score	AUC
Random Forest	0.955	0.934	0.869	0.898	0.869
Decision Tree	0.942	0.881	0.87	0.875	0.87
Neural Network	0.871	0.935	0.511	0.488	0.511
Naïve Bayes	0.871	0.725	0.715	0.722	0.718

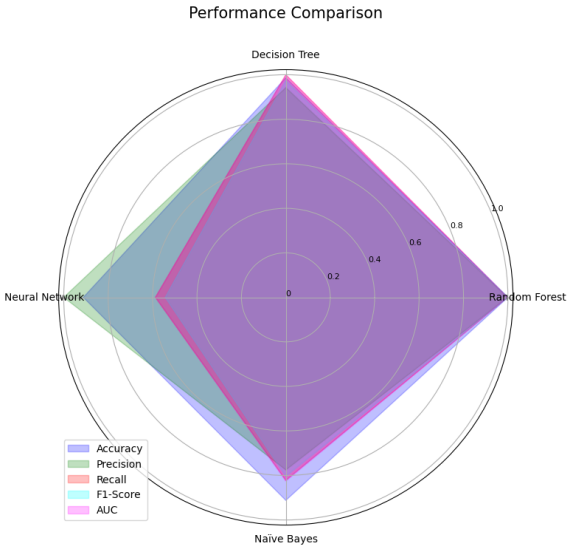

Fig. 5. Evaluation metric comparison

IV. RELATED REGRESSION PROBLEM

As a final semester project, there is a guideline from the course instructor to identify and solve a regression problem related to the classification problem addressed above following my reference article and their approach.

The conclusion drawn from the above classification study in my reference paper [1] is that reducing customer service calls help to mitigate customer churn. This is logical as potential churn customers who are dissatisfied with service(s) are likely to make more calls to customer service for support than those who are satisfied with network service.

When customers make calls to support centers, personnel, and resources must be available and ready to receive the calls, log requests and complaints, and finally respond to or resolve the request or complaint. This typically requires efficient resource planning and an effective forecast of the expected number of customer service calls is fundamental to the planning process.

I explored two machine learning regression techniques namely random forest and neural network to predict the number of customer service calls for each customer for this same dataset. The random forest model was chosen because it performed best in the classification problem for this dataset and because it incorporates multiple decision tree-based learners. Neural network was chosen because it gave the highest precision in the classification problem for this dataset.

The number of customer service calls predicted for each customer can be aggregated to form the total expected for all customers and based on this effective resource planning can be made to cater for all customer service calls readily and efficiently. This would minimize customer service calls in the end as customers would not have to call many times for the same issue. In the long run, the regressor model would help mitigate customer churn which is the original problem being addressed.

The regression models had the same test split as their classifier counterparts in this study.

V. REGRESSION MODELS PERFORMANCE

The metrics considered for regression models were mean absolute score (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and accuracy. The accuracy was computed using (6) and (7).

$$Error\ (\%) = (MAE/average\ of\ target) \times 100 \quad (6)$$

$$Accuracy\ (\%) = 100 - Error(\%) \quad (7)$$

The RF regressor model yielded a Mean Absolute Error of 1.01, Mean Squared Error of 1.76 Root Mean Squared Error of 1.33 for the same dataset.

The NN regressor model yielded a Mean Absolute Error of 1.05, Mean Squared Error of 1.76 Root Mean Squared Error of 1.33 for the same dataset.

The two models performed nearly on par in the considered metrics yielding an accuracy of 85%.

VI. CONCLUSION

This study explores the efficacy of various machine learning algorithms in classifying churners and non-churners among telco customers. Leveraging a dataset comprising 3333 customer records, the study employs SMOTE to address class imbalance and evaluates predictive models based on multiple performance metrics. Random forest emerges as the optimal model with an accuracy of 0.955, precision of 0.934, recall of 0.869, F1-score of 0.898, and AUC of 0.869 confirming the study in my reference paper. Decision tree also performed well in all the metrics. Neural networks produced the highest precision but did not perform well on other metrics. Naïve Bayes was not the best performer on any metric but overall, performed better than neural networks.

The primary objective of predictive analysis is to anticipate customers who may churn and take measures to prevent or minimize churn rates. It is recommended that telcos utilize the random forest predictive model to identify potential churners. Notably, key features identified by the random forest for churn prediction include customer service calls, total day minutes, and total day charges. It is noteworthy that total day charge and total day minutes exhibit a correlation, where higher total day minutes correspond to increased total day charges. To mitigate the risk of customer churn, the company should focus on reducing customer service calls and lowering the total charge per minute.

It also considered the related regression problem of reducing customer service calls and explored the random forest and neural network regression techniques to support telcos overcome customer churn.

ACKNOWLEDGEMENT

The author would like to thank Dr. Isaac Osunmakinde and his coursemates in CSC-611 Machine Learning (Spring 2024) at Norfolk State University for supporting this work.

REFERENCES

[1] N. I. A. R. a. M. H. Wahid, "Telecommunication Customers Churn Prediction using Machine Learning," in IEEE 15th Malaysia International Conference on Communication (MICC), Malaysia, 2021.

[2] CrowdAnalytix, "Churn Prediction in Telecom," [Online]. Available: https://www.crowdanalytix.com/contests/why-customer-churn. [Accessed 20 April 2024].

[3] A. S. R. A. M. Q. A. K. a. A. R. S. A. Qureshi, "Telecommunication subscribers' churn prediction model using machine learning," in *Eighth International Conference on Digital Information Management (ICDIM 2013), Islamabad*, Islamabad, 2013.

[4] R. R. Z. S. A. a. M. M. J. Andrews, "Churn Prediction in Telecom Sector Using Machine Learning.," *International Journal of Information 8, no. 2,* 2019.

[5] Kaggle.com, "Churn in Telecom's dataset," [Online]. Available: https://www.kaggle.com/datasets/becksddf/churn-in-telecoms-dataset. [Accessed 18 April 2024].