

Churn Prediction in Telecommunication Industry: A Comparative Analysis of Boosting Algorithms

Opara John Ogbonna¹, Gilbert I.O. Aimufua¹,
Muhammad Umar Abdullahi² and Suleiman Abubakar¹

¹ Department of Computer Science,
Nasarawa State University,
Keffi,
Nigeria.

² Department of Computer Science,
Federal University of Technology,
Owerri,
Nigeria.

Email: johnopara100@gmail.com

Abstract

The issue of customer churn, which is a major problem in the telecommunications industry, poses several challenges, such as financial implications, client attrition, and increased marketing costs. The advancements achieved in the domain of machine learning and artificial intelligence have significantly expanded the possibilities for forecasting customer churn, presenting a promising resolution for effectively handling customer attrition and enhancing customer retention. This study presents a customer churn prediction model that uses machine learning approaches to assist telecom firms in enhancing customer retention and mitigating churn rates. The study employs machine learning techniques, such as Adaboost, Gradient Boosting, and Extreme Gradient Boosting (XGBoost), in order to evaluate extensive datasets and provide predictions on customer churn via a comparative evaluation. The methodology involves extracting data from the Kaggle data pool, doing further data preparation, and identifying relevant features. The Synthetic Minority Oversampling Technique (SMOTE) is used as a strategy to mitigate the challenges posed by imbalanced data. The dataset is partitioned into training and testing sets at a ratio of 75% to 25%. The XGBoost model demonstrated superior accuracy and recall, positioning itself as the top-performing model among the studied models. The attained accuracy rate was 89.51%. The XGBoost method has a recall rate of 92.48%, which is the highest of the three algorithms evaluated. Gradient boosting follows with a recall rate of 87.69%, while Adaboost achieves a recall rate of 85.13%. These findings underscore the potential of machine learning techniques for addressing the challenges posed by customer churn in the telecommunications industry.

Keywords: Churn Prediction, Telecom, Boosting Algorithms, Comparative Analysis, Customer Retention

INTRODUCTION

Cloud computing has gained significant significance in the telecommunications sector within the area of information technology (IT). The anticipation of client attrition has become crucial for formulating efficient customer retention tactics in response to intensified market rivalry. Customer churn, which refers to the phenomenon of consumers terminating their membership with a telecommunications service provider, has the potential to result in a

*Author for Correspondence

decrease in both revenue and market share (Wagh *et al.*, 2023). The costs related to gaining new customers in order to compensate for attrition are significantly elevated, underscoring the need of proactive retention initiatives.

The study conducted by Techsee in 2019 highlights the significance of customer effort in relation to turnover. There is a positive correlation between high levels of customer effort, characterized by a significant number of phone calls or extended periods of time required to resolve issues, and elevated rates of customer turnover. The impact of dissatisfied consumers on the discontinuation of services necessitates a comprehensive comprehension of customer behavior, as emphasized by Dalli (2022).

Machine learning algorithms are of great significance in effectively controlling client turnover. The algorithms often utilized in many domains encompass Logistic Regression, Random Forest, Support Vector Machines, Boosting, and Decision Trees (Dalli, 2022). According to Xiu (2017), the boosting strategy demonstrates enhanced accuracy and performance while dealing with imbalanced datasets, surpassing earlier ensemble approaches.

The objective of our research is to evaluate the efficacy of boosting algorithms, namely Gradient Boost, Extreme Gradient Boost (XGBoost), and Adaptive Boost (Adaboost), in forecasting customer attrition within the telecoms industry. This is consistent with the focus placed by Saleh and Saha (2023) on comprehending crucial factors such as service quality, customer happiness, subscription plan upgrades, and network coverage.

The study conducted by Bogaert and Delaere highlights the superior performance of ensemble approaches compared to individual classifiers. This finding aligns with the main objective of our research, which is to investigate dataset-related challenges in predictive analysis. To do this, we apply correlation tests and dataset resampling techniques. Taskin's (2023) research highlights the utilization of machine learning algorithms and dataset pretreatment techniques to improve predicted accuracy.

Fujo, Subramanian, and Khder (2022) propose the utilization of deep learning, more especially a Deep Back-propagation Artificial Neural Network (Deep-BP-ANN), as a recommended approach for customer churn prediction. This study expands upon previous research by examining the efficacy of boosting algorithms in the telecoms sector.

In their study, Azeem & Usman (2018) employ a fuzzy algorithm for the purpose of customer attrition prediction, effectively mitigating the impact of data noise. In their study, Vafeiadis *et al.* (2015) conducted a comparative analysis of several data mining techniques, with a particular emphasis on the algorithmic efficiency required to effectively process large datasets. Nevertheless, the scope of our study is limited to examining boosting algorithms and evaluating their appropriateness in forecasting user attrition.

The Hierarchy Multiple Kernel Support Vector Machine (H-MK-SVM) was proposed by Chen *et al.* (2018) as a solution for unbalanced datasets. Our research enhances the existing literature by conducting a focused evaluation of Gradient Boost, XGBoost, and Adaboost algorithms, especially in relation to issues associated with datasets. In their study, Shaaban *et al.* (2015) utilize an extensive data mining methodology that encompasses classification and clustering techniques for the purpose of customer churn prediction. The focus of our study is narrowed down to the examination of boosting algorithms and their relative efficacy.

The work conducted by Rahman *et al.* (2021) places significant emphasis on the efficacy of boosting classifiers, namely Adaboost and XGboost, in attaining a notable level of

accuracy. Our analysis is in line with this, especially evaluating the performance of Gradient Boost, XGBoost, and Adaboost algorithms in the telecommunications industry. In view of the foregoing, this research makes a valuable contribution by assessing the efficacy of boosting algorithms in the prediction of customer attrition in the telecoms industry. Prior research has investigated several machine learning approaches and the difficulties associated with datasets. However, this study offers a concentrated examination of boosting algorithms and their effectiveness in tackling churn-related problems. The objective of the study is to provide telecom companies with insights about the effectiveness of these algorithms in proactively implementing client retention measures.

METHODOLOGY

The research methodology serves as the core design or framework comprising the methods and processes utilized to obtain, collect, and evaluate data, all of which are precisely connected with the study subject at hand. The research technique of the proposed system is presented in Figure 1 and comprises of numerous steps. These steps comprise data gathering or collection, feature selection, categorization, and assessment.

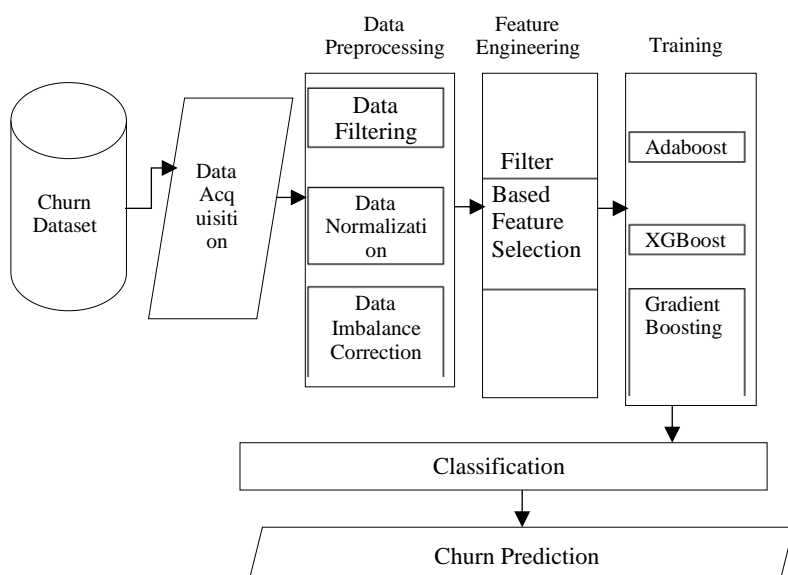


Figure 1: Research Design

Data Preprocessing: A variety of essential data preprocessing activities were conducted in anticipation of training a machine learning model in the telecommunications industry. The steps undertaken encompassed the management of missing values, the elimination of a non-informative 'phone number' column, the encoding of the 'churn' target label from Boolean to numerical values, the conversion of categorical data in the 'state' column using label encoding, the mapping of 'international plan' and 'voice mail plan' to numerical values, the standardization of feature scaling to ensure uniformity across attributes, and the application of the Synthetic Minority Oversampling Technique (SMOTE) to tackle imbalances within the dataset. The use of thorough data preparation techniques played a crucial role in enhancing the dataset for precise customer churn prediction using machine learning models inside the telecoms sector.

Feature Selection: The study applied a feature selection approach known as the filter selection technique, specifically the Chi-Square Selector, in a comparative manner. The chi-square test is a statistical procedure used to assess the level of dependence between two

variables. The concept in question exhibits resemblances to the coefficient of determination, denoted as R^2 . Nevertheless, it should be noted that the chi-square test is specifically designed for the analysis of categorical or nominal data. The chi-square statistic was computed to assess the association between each feature variable and the target variable, and the presence of a link between the variables and the target was noted. If the target variable exhibits independence from the feature variable, it may be inferred that the feature variable has been excluded from consideration. If the variables exhibit dependence, the feature variable was chosen.

Using the formula of Chi Square test:

$$\chi^2 = \frac{1}{d} \sum_{k=1}^n \frac{(O_k - E_k)^2}{E_k} \quad (1)$$

We have

$$\chi^2 = \frac{(A - E_A)^2}{E_A} + \frac{(B - E_B)^2}{E_B} + \frac{(C - E_C)^2}{E_C} + \frac{(D - E_D)^2}{E_D} \quad (2)$$

After Simple calculation we have

$$\chi^2 = \frac{N(AD - BC)^2}{(A+C)(B+D)(A+B)(C+D)} \quad (3)$$

Given

$$B = M - A$$

$$C = P - A$$

$$D = N - M - (P - A)$$

We have:

$$\chi^2 = \frac{N(AN - MP)^2}{PM(N - P)(N - M)} \quad (4)$$

Given a dataset, we can easily obtain:

A: the total number of positive instances that contain feature X,

M: the total number of instances that contain feature X

P: the total number of positive instances,

N: the total number of instances

Apparently, N, P, are constant for all the features,

Features Classifications: The classification procedure was executed using a comparative methodology, employing three boosting classifiers: Adaboost, Gradient Boosting, and XGBoost machine learning algorithms. The reduced features were later employed in the boosting process, using a data split of 75% for training and 25% for testing. The dataset was divided into two subsets. The first group, including 75% of the data, was used to train the boosting algorithms and construct a knowledge retention pattern using previous telecommunication datasets. The second subset, consisting of the remaining 25% of the data, was kept for evaluating the accuracy of each technique's predictions. The evaluation of the algorithms' performance was conducted based on widely accepted machine learning metrics. This article outlines a model that incorporates three case studies utilizing the boosting approach. The current case studies provide a comprehensive and comparative approach to assess the effectiveness of several boosting algorithms, including Adaboost, Gradient Boosting, and XGBoost Algorithm. The following section delineates the data sources that were acquired to create the study findings.

The utilization of the Chi-Square Selector (Reduced Dataset) in conjunction with Gradient Boosting is being implemented. The outcomes were acquired for each provided case model and later compared in order to determine the most effective model for identifying churn in a telecommunications dataset.

Adaboost Algorithm

Adaboost, also known as Adaptive Boosting, is a widely used machine learning methodology utilized for the purpose of performing classification problems. The methodology under consideration is within the category of boosting algorithms, which is a specific subset of ensemble learning approaches. The primary objective of these algorithms is to enhance the predictive abilities of less proficient learners, often shown by basic decision trees, by their amalgamation into a more robust learner.

Algorithm 1: Algorithm for Adaboost

1. Train a weak classifier using the current distribution of weights D_t .
2. Get a weak hypothesis $h_t: X \rightarrow \{-1, 1\}$ with error ϵ_t .
3. Choose $\alpha_t = \log((1 - \epsilon_t) / \epsilon_t)$.
4. Update the weights of the training samples based on the performance of the weak classifier.
5. For each sample i :
 - If the weak classifier correctly classifies the sample, multiply its weight by $e^{-(\alpha_t * y_i * h_t(x_i))}$.
 - If the weak classifier misclassifies the sample, multiply its weight by $e^{(\alpha_t * y_i * h_t(x_i))}$.
6. Normalize the weights so that they sum to 1.
7. Combine the weak classifiers into a strong classifier by weighting them according to their performance. The final hypothesis is given by $H(x) = \text{sign}(\sum(\alpha_t * h_t(x)))$, where $h_t(x)$ is the weak classifier trained in iteration t .
8. Return the final hypothesis.

XGBoost Algorithm

XGBoost, an acronym for eXtreme Gradient Boosting, is a highly optimized and efficient method for gradient boosting. It has garnered significant attention and widespread use in the field of machine learning competitions and several application domains. The proposed method incorporates the fundamental concepts of gradient boosting while introducing several modifications aimed at enhancing both computational efficiency and predictive accuracy.

Algorithm 2: Algorithm for XGBoost

Input:

- Training dataset $\{(x_i, y_i)\}$ where x_i represents the input features, and y_i is the corresponding target value.
 - Number of trees (iterations) T .
 - Maximum depth of each tree (max_depth).
 - Regularization parameters (λ for L1 and γ for L2).
 - Learning rate (η) for controlling the step size during gradient descent.
 - Objective function (e.g., MSE for regression or log-loss for classification)
1. Initialize $F_0(x)$ with a constant value (e.g., mean for regression, log-odds for classification).

2. For t in range(T):
 - a. Calculate the negative gradient (residuals) for each data point: $r_i = -\nabla L(y_i, F_{t-1}(x_i))$.
 - b. Train a decision tree $h_t(x)$ to fit the negative gradient r_i with a maximum depth of `max_depth`. Apply regularization terms (L1 and L2) if needed.
 - c. Compute the optimal weight γ_t for the newly created tree by solving a simple optimization problem: $\gamma_t = \operatorname{argmin}_{\gamma_t} \sum_i L(y_i, F_{t-1}(x_i) + \gamma_t * h_t(x_i)) + \lambda * \Omega(h_t)$, where $\Omega(h_t)$ represents the regularization term.
Update the ensemble model: $F_t(x) = F_{t-1}(x) + \gamma_t * h_t(x)$.
3. Output the final ensemble model $F_T(x)$.

Gradient Boosting Algorithm

Gradient Boosting is an effective data mining technique that showcases its versatility by being applicable to a diverse array of applications, encompassing both regression and classification tasks. The ensemble approach involves the construction of a predictive model by the creation of an ensemble of many decision tree structures. The core concept underlying Gradient Boosting is the integration of predictions derived from many "weak" models, often decision trees, to construct a coherent and resilient predictive model.

Algorithm 3: Algorithm for Gradient Boosting

Input:

- Training dataset $\{(x_i, y_i)\}$ where x_i represents the input features, and y_i is the corresponding target value.
 - Number of iterations T .
 - Learning rate (η), a hyperparameter controlling the step size during updates.
1. Initialize $F_0(x)$ as the initial prediction, often the mean (for regression) or the most frequent class (for classification) of y .
 2. For t in range(T):
 - a. Calculate the negative gradient (residuals) for each data point: $r_i = -\nabla L(y_i, F_{t-1}(x_i))$, where L is the loss function measuring the prediction error.
 - b. Train a weak learner (e.g., decision tree) $h_t(x)$ to fit the negative gradient r_i .
 - c. Update the ensemble model: $F_t(x) = F_{t-1}(x) + \eta * h_t(x)$.
 3. Output the final ensemble model $F_T(x)$.

Flowchart of the Model

The flowchart in Figure 2 illustrates the sequential activities and procedures employed in the evaluation of customer churn prediction using three prominent boosting algorithms. The assessment approach started with obtaining data from Kaggle, an online data repository. Subsequently, the data acquired via the process of data mining is submitted to a series of operations, including filtering and standardization. Additionally, a comprehensive examination was performed to detect any occurrences of missing values in the dataset, aiming to exclude any data points that had substantial deviations from the anticipated pattern or shown inconsistencies. Following that, the dataset with an imbalance was subjected to resampling using the Synthetic Minority Over Sampling approach in order to address the issue of limited representation of minority data. The third phase was feature selection, where the Chi-Square statistical method was utilized as a filtering mechanism to identify the most effective attributes with strong predictive ability. The allocation of reduced features was distributed in a percentage of 75% for training the Adaboost, Gradient boosting, and Extreme Gradient boosting classifiers, with the objective of enhancing information retention. Subsequently, the remaining 25% of the data was employed for the purpose of evaluating

and examining the classifiers' performance via testing. The evaluation was conducted using many criteria, such as accuracy, error rate, specificity, sensitivity, f1 score, and process time. The objective of the research was to ascertain the classifier that exhibited the highest level of effectiveness in predicting churn.

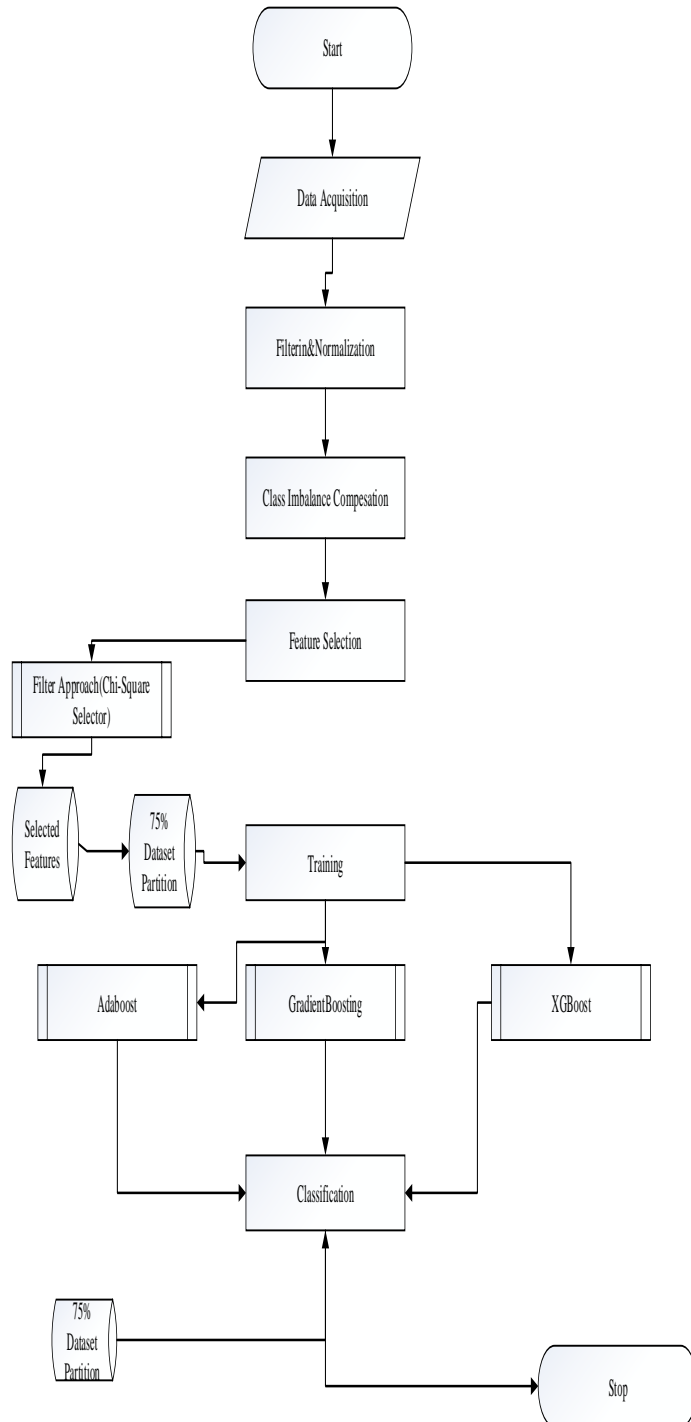


Figure 2: Flowchart of the Model

Performance Evaluation Matrices

Various metrics are frequently utilized to evaluate the efficacy of a classification model. These metrics encompass Accuracy, Confusion Matrix, Precision, Recall, F1-score, Error Rate, and Training time.

Classification Accuracy: is defined as the quotient obtained by dividing the number of accurate predictions made by the model by the total number of predictions made.

$$\text{Accuracy (\%)} = \frac{TP+TN}{TP+TN+FP+FN} \times 100 \quad (3.5)$$

Precision: Precision is calculated as the ratio of true positives to the sum of true positives and false positives.

The formula for precision is:

$$\text{Precision} = \frac{TP}{TP+FP} \quad (3.6)$$

Sensitivity; Sensitivity is a metric that aims to calculate the proportion of actual positive that was identified incorrectly in the model. It is also called recall or true positive rate. Mathematically, it is defined as the ratio of the true positive (TP) instances to the sum of true positive and false negative (FN) instances

$$\text{Sensitivity} = \frac{TP}{TP+FN} \quad (3.7)$$

Specificity

Specificity is another term for the real negative rate. It is theoretically defined as the ratio of true negative (TN) to total true negative and false positive (FP) cases.

Mathematically, it is expressed as:

$$\text{Specificity} = \frac{TN}{TN+FP} \quad (3.8)$$

F-Score

The F-score is a statistical measure used to assess the performance of a binary classification model by considering its ability to accurately forecast positive class instances. Precision and recall are utilized in the calculation of it.

It is mathematically calculated as:

$$\text{F1-score} = 2 * \frac{\text{precision} * \text{recall}}{\text{precision} + \text{recall}} \quad (3.9)$$

Error Rate (EER)

The error rate can be calculated by dividing the total count of incorrect predictions made on the test set by the total count of predictions made on the test set.

Mathematically, it is expressed as:

$$\text{Error Rate} = \frac{\text{Incorrect Predictions}}{\text{Total Predictions}} \quad (3.10)$$

RESULT AND DISCUSSION

Numerical Experimental Performance of the Proposed Models

This section assesses the numerical experimental performance of the proposed models and manipulating the sample and feature counts. The results are presented in a thorough manner through the utilization of tables and graphs.

Loading of Dataset and Dataset Filtering

The process of filtering the data entails the conversion of text variables into numeric variables and the removal of inconsistent items. This results in a dataset that is properly prepared and can be seamlessly incorporated into the system. The filtered and normalized data is shown in the subsequent section. During this stage, an incongruous element was also deleted. The variables pertaining to state, account length, area code, and phone number were omitted from the dataset since they did not exhibit sufficient statistical significance in the prediction models. Furthermore, the categorical churn type data was transformed into a numerical data type. Table 1 depicts the sequential processes included in the importation and filtration of the dataset.

Table 1: Dataset Loading and Filtering

	international plan	voice mail plan	number vmail messages	total day minutes	total day calls	total day charge	total eve minutes	total eve calls	total eve charge	total night minutes	total night calls	total night charge	total intl minutes	total intl calls	total intl charge	customer service calls	churn
0	2	1	25	265.1	110	45.07	197.4	99	16.78	244.7	91	11.01	10.0	3	2.70	1	2
1	2	1	26	161.6	123	27.47	195.5	103	16.62	254.4	103	11.45	13.7	3	3.70	1	2
2	2	2	0	243.4	114	41.38	121.2	110	10.30	162.6	104	7.32	12.2	5	3.29	0	2
3	1	2	0	299.4	71	50.90	61.9	88	5.26	196.9	89	8.86	6.6	7	1.78	2	2
4	1	2	0	166.7	113	28.34	148.3	122	12.61	186.9	121	8.41	10.1	3	2.73	3	2

Checking for Missing Values

The presence of missing values was evaluated for each attribute in the dataset, and no instances of missing values were found. Consequently, progressing to the subsequent stage. The illustration can be also seen in Figure 3.

```

Check for missing Values

#Checking missing values in train dat
print(data.isnull().sum())

#no missing value present in the trai

international plan      0
voice mail plan        0
number vmail messages  0
total day minutes      0
total day calls        0
total day charge       0
total eve minutes      0
total eve calls        0
total eve charge       0
total night minutes    0
total night calls      0
total night charge     0
total intl minutes     0
total intl calls       0
total intl charge      0
customer service calls 0
churn                  0
dtype: int64
    
```

Figure 3: Missing Value

Data Distribution of Target Variable

The dataset exhibits data imbalance, with 14.49% (483 instances) reflecting true labels showing customer migration to a different telecom provider, and 85.51% (2850 instances) representing false labels indicating consumers who additionally did not transfer. Figure 4 provides a more comprehensive representation of the information.

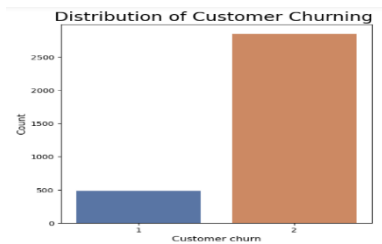


Figure 4: Data distribution in target variable

Feature Selection Technique

The technique of Filter Selection, especially the Chi-Square Selection, was utilized for feature selection. Before employing this methodology, a correlation analysis was performed to evaluate the association between the independent and dependent variables. A correlation study was performed to investigate the association between the independent variables and the dependent variable, churn. Figure 5 shows the resultant Chi-Square experiment for feature selection, which indicates acceptance of both the independent and dependent variables.

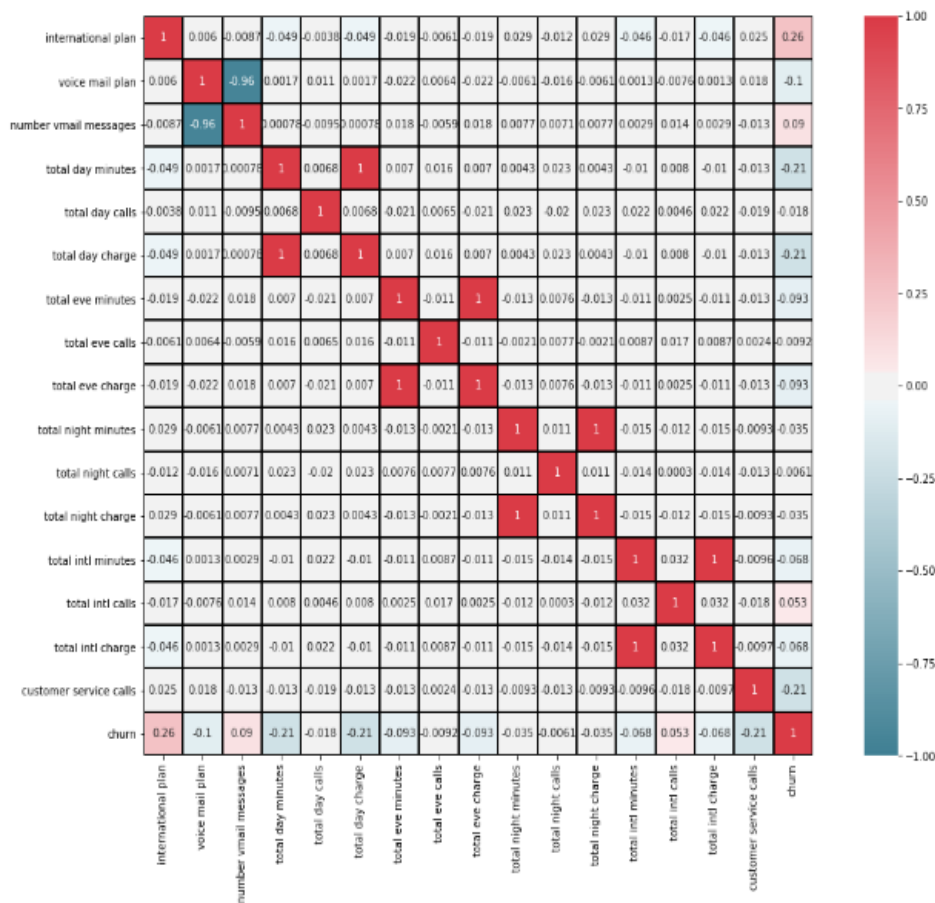


Figure 5: Correlation test of the features

Chi Square Selection output

The output provides a detailed breakdown of the selected features based on their associated probability values. The selection criteria employed here involved choosing features with p-values less than 0.05, indicating their statistical significance in relation to the dependent variable. Result of the Chi-Square feature ranking is shown in Table 2.

Table 2: Chi-Square Selected Features

No	Selected Factors	Probability Value
1	International plan	2.4931077033159556e-50
2	Voice mail plan	5.15063965903898e-09
3	Number vmail messages	0.020381511965517657
4	Total day minutes	2.0125191827544002e-09
5	Total day calls	0.02913941103845643
6	Total day charge	2.0125191827544002e-09
7	Total intl minutes	0.004895006642264651
8	Total intl calls	0.00012589314865942453
9	Total intl charge	0.004895006642264651
10	Customer service calls	2.243376519451627e-68

The statistical findings of the Chi-Square test for the feature variables that yielded a p-value are presented in Table 1. The statistical significance of the correlation study done to investigate the association between the independent characteristics and the dependent features is shown by the p-values, which are observed to be less than 0.05, as seen in Figure 4.6. If the magnitude of the target variable surpasses 0.05 times that of the feature variable, the feature variable will be eliminated. In contrast, when the value is below the designated threshold p , the feature variable will be included.

3.1.6 Data Imbalance

Figures 6 and 7 depict the distribution of particular attributes in correlation with the class designation. The aforementioned figures demonstrate a significant presence of data imbalance within the dataset. In order to address this issue in a comprehensive manner, it is crucial to implement appropriate strategies.

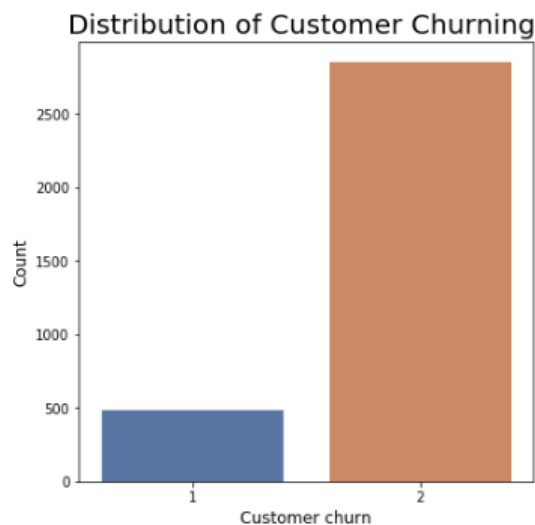


Figure 6: Churn and Non-Churners Distribution in dataset



Figure 7: Churn vs customer service call and voice mail distribution

Oversampling with Smote

For the purpose of trying to address the disparity in data distribution among the class groups within the dataset, the SMOTE approach was utilized to do oversampling. The dataset was augmented by the use of the Synthetic Minority Over-Sampling Technique (SMOTE), leading to a total of 4,530 instances in the final dataset.

Case 1: Chi-Square Selector (Reduced Dataset) + Adaboost Classification

Table 3 presents the results of the study conducted for each class using the Adaboost Algorithm. The table presents the achieved values of weighted accuracy, recall, f1-score, and support.

Classification Report

Table 3: Adaboost Classification Report

	precision	recall	F1-score	support
1	0.41	0.74	0.53	82
2	0.96	0.85	0.90	585
accuracy	0.00	0.00	0.84	667
Macro avg	0.69	0.80	0.72	667
Weighted avg	0.89	0.86	0.86	667

Evaluation Parameters for Adaboost Classification

Table 4 illustrates the evaluation criteria for the Adaboost Classifier, with particular emphasis on the selected reduced features determined by the Chi-Square Selector. The assessment encompasses many indicators, such as the f-score, specificity, sensitivity, accuracy, and mistake rate.

Table 4: Evaluation Parameters for Case 1 Classification Phase

Technique	F1-Score	Recall	Precision	Specificity	Sensitivity	Accuracy (%)	Error Rate
Adaboost	0.86	0.85	0.87	0.7439	0.8513	83.81	0.1619

Case 2: Chi-Square Selector (Reduced Dataset) + XGBoost Classification

The findings of the study, which employed the XGBoost Algorithm, are displayed in Table 5. The table displays the obtained values for weighted accuracy, recall, f1-score, and support.

Classification Report

Table 5: XGBoost Classification Report

	precision	recall	F1-score	support
1	0.05	0.32	0.08	82
2	0.00	0.00	0.00	585
accuracy	0.00	0.00	0.04	667
Macro avg	0.02	0.11	0.03	667
Weighted avg	0.01	0.04	0.01	667

Evaluation Parameters for XGBoost Classification

Table 6 presents the evaluation criteria of the XGBoost Classifier applied to the feature subset obtained using the Chi-Square Selector. The assessment criteria encompass the f-score, specificity, sensitivity, accuracy, and error rate.

Table 6: Evaluation Parameters for Case 2 Classification Phase

Technique	F-score	Recall	Precision	Specificity	Sensitivity	Accuracy (%)	Error Rate
XGBoost	0.86	92.48	0.92	68.29	92.48	89.51	0.1049

Case 3: Chi-Square Selector (Reduced Dataset) + Gradient Boosting Classification

The class-wise analysis obtained from the implementation of the Gradient Boosting Algorithm is presented in Table 7. The table displays the obtained values for weighted accuracy, recall, f1-score, and support.

Classification Report

Table 7: Gradient Boosting Classification Report

	precision	recall	F1-score	support
1	0.47	0.78	0.59	82
2	0.97	0.88	0.92	585
accuracy	0.00	0.00	0.87	667
Macro avg	0.72	0.83	0.75	667
Weighted avg	0.91	0.87	0.88	667

Evaluation Parameters for Gradient Boosting Classification

Table 8 presents the assessment parameters of the Gradient Boosting Classifier for the reduced features picked by the Chi-Square Selector. The assessment measures employed encompass the f-score, specificity, sensitivity, accuracy, and error rate.

Table 8: Evaluation Parameters for Case 3 Classification Phase

Technique	F1-Score	Recall	Precision	Specificity	Sensitivity	Accuracy (%)	Error Rate
Gradient Boosting	0.87	0.88	0.91	0.78	0.8769	86.51	0.1349

RESULTS

Analysis for Training Time

The term training time refers to the duration needed for the model to gain knowledge retention from the smaller dataset generated through the implementation of the Chi-Square Selector. The previously indicated methodology was applied to the Adaboost, XGBoost, and Gradient Boosting classifiers, in that order. The data shown in Figure 8 clearly demonstrates that the Adaboost Classifier outperforms its competitors in terms of optimum time.

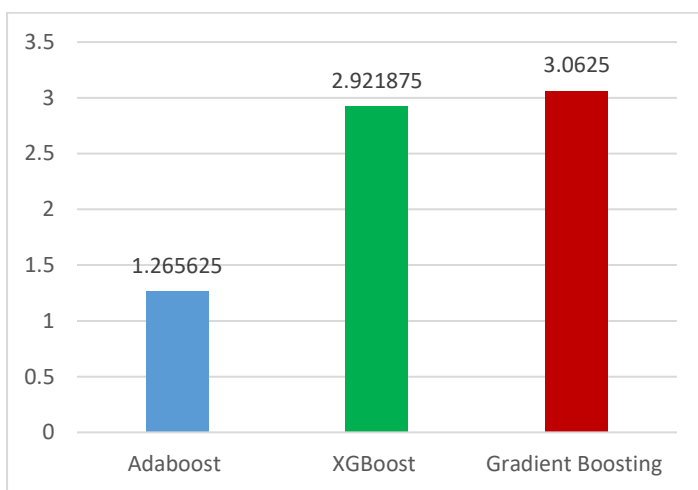


Figure 8: Training Time obtained for the three boosting techniques

Classification Accuracy

The findings suggest that among the models considered, Case 2 of the XGBoost Model exhibited the best level of classification accuracy, with a rate of 89.51%. This performance surpassed that of the Gradient Boosting Model, which achieved an accuracy of 86.51%, as well as the Adaboost Model, which achieved an accuracy of 83.81%. The findings suggest that XGBoost exhibited more efficacy in comparison to the alternative boosting method, and also achieved a higher prediction rate when compared to other machine learning algorithms evaluated on the same imbalanced dataset. The visual representation can be seen in Figure 9.

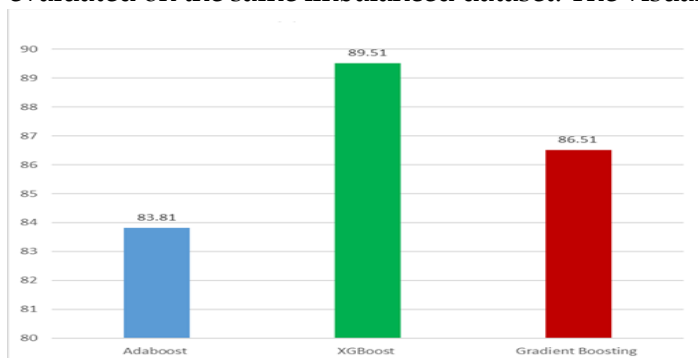


Figure 9: Classification Accuracy for the three boosting technique.

Sensitivity and Specificity

Sensitivity (SN) is calculated by dividing the number of accurate positive predictions by the total number of positive instances, while Specificity (SP) is determined by dividing the number of accurate negative predictions by the total number of negative instances. The maximum values for sensitivity and specificity are both 1. The data presented in Figure 10 demonstrate that the sensitivity and specificity rates are close to unity, suggesting a high level of prediction accuracy. The XGBoost Algorithm had superior predictive capabilities in terms of positive rate, achieving a score of 0.9248. However, it exhibited suboptimal performance in forecasting negative rates, with the gradient boosting approach outperforming it.

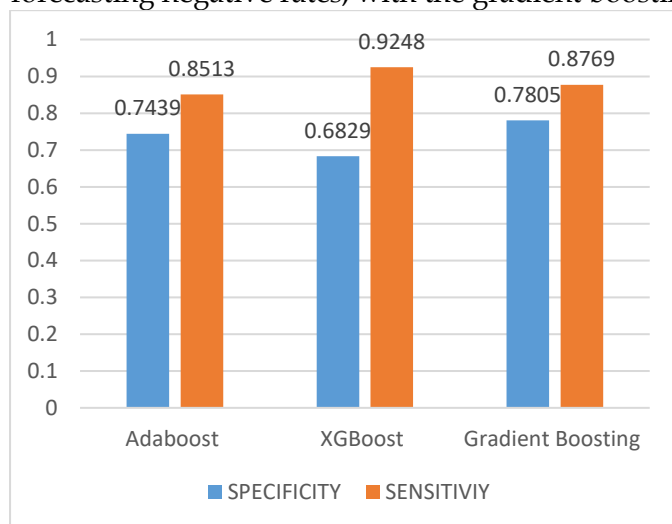


Figure 10: Sensitivity and Specificity Chart for the three boosting techniques

Comparative Analysis for Error Rate

The error rate is the minimum potential error rate of a classifier when making random classifications. The XGBoost classifier depicted in Figure 11 exhibits the lowest error rate of 0.1049, suggesting that it yields superior output results compared to other boosting algorithms when applied to the telecommunication dataset.

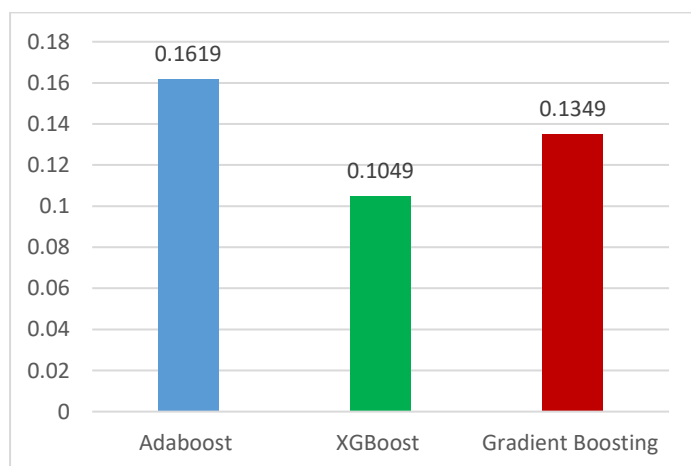


Figure 11: Error Rate for the three boosting techniques

Discussion

The table 9 shows the comparative evaluation parameters of the Adaboost, XGBoost and Gradient Boosting classifiers respectively.

Table 9: Comparative Evaluation Parameters for the Classification Phases

Technique	F-score	RECALL	PRECISION	SPECIFICITY	SENSITIVITY	ACCURACY (%)	ERROR RATE
Adaboost	0.86	85.13	0.87	0.7439	0.8513	83.81	0.1619
XGBoost	0.86	92.48	0.92	0.6829	0.9248	89.51	0.1049
Gradient Boosting	0.87	87.69	0.91	0.7805	0.8769	86.51	0.1349

When examining the performance metrics of Adaboost, XGBoost, and Gradient Boosting, distinct variations become apparent. The F1 scores exhibit a high degree of similarity, as both Adaboost and XGBoost algorithms attain a score of 0.86. However, Gradient Boosting algorithm demonstrates a little superior performance, reaching a score of 0.87. The F1 score holds significant importance as it effectively balances the metrics of accuracy and recall. When examining the metric of recall, XGBoost demonstrates a notable performance with a significantly high value of 92.48%. This number signifies the model's proficiency in correctly identifying and capturing a substantial part of positive instances inside the dataset. The Adaboost algorithm demonstrates a recall rate of 85.13%, whereas Gradient Boosting obtains a higher recall rate of 87.69%.

In relation to precision, XGBoost has a precision value of 0.92, indicating its proficiency in generating precise positive predictions. Adaboost demonstrates a precision of 0.87, whereas Gradient Boosting exhibits superior performance with a precision of 0.91. The importance of precision is paramount in situations where the occurrence of false positives might lead to major ramifications.

In terms of specificity, which refers to the capacity to accurately detect negative instances, Gradient Boosting has superior performance compared to other methods, achieving a specificity value of 0.7805. The Adaboost algorithm exhibits a specificity value of 0.7439, while XGBoost demonstrates the lowest specificity of 0.6829. The sensitivity, also known as recall, has the maximum value of 92.48% for XGBoost, followed by Gradient Boosting with 87.69% and Adaboost with 85.13%.

Accuracy is a comprehensive measure that encompasses the complete accuracy of a given entity or phenomenon. In this experiment, XGBoost demonstrates the highest level of accuracy, achieving a score of 89.51%. Following closely behind is Gradient Boosting with an accuracy of 86.51%, and Adaboost with an accuracy of 83.81%. In conclusion, the XGBoost algorithm exhibits the lowest error rate (0.1049), whereas Gradient Boosting demonstrates a slightly higher error rate (0.1349), and Adaboost exhibits the highest error rate (0.1619). In conclusion, it can be observed that all three methodologies exhibit commendable performance. However, XGBoost exhibits superior capabilities in terms of recall, precision, accuracy, and total error reduction. Consequently, based on the unique demands and priorities of the given work, XGBoost can be considered as a more suitable option. These findings are further supported by Gunnarsson et al. (2021) and Bogaert & Delaere (2023), who demonstrated that advanced deep learning models do not surpass XGBoost in terms of credit scoring performance. Similarly, the work of Kimura (2022) showed that XGBoost outperformed traditional classification algorithms and Random Forest in churn prediction. Furthermore, the study conducted by Oladipo et al. (2023) utilized an ensemble approach incorporating XGBoost and other boosting classifiers to develop a model, achieving a performance of 92.2%, outperforming CatBoost and Random Forest.

CONCLUSION

This paper presented and categorized customer turnover in a telecoms dataset using a comparative technique. The implementation objectives were met by using data mining to find

the best version of the three core models. Case 1, Case 2, and Case 3 models were used in the study. The Chi-Square Selector (Reduced Dataset) was originally used with Adaboost Classification. XGBoost Classification and Chi-Square Selector (Reduced Dataset) were used in the second case. Case 3 used Chi-Square Selector (Reduced Dataset) with Gradient Boosting Classification.

Machine learning-generated statistical criteria were used to evaluate the models. The dataset was mined many times using data filtering, imbalance correction, and engineering feature selection. Using these strategies, a more efficient and refined dataset was generated. The dataset was then fed to each boosting approach to summarize results. Model assessment revealed their predictive power and efficacy. A comparison of the data revealed the model with the greatest accuracy, precision, recall, F1 score, and other criteria. This selection approach produced the best telecommunications customer turnover forecasting and categorization model.

The study emphasized data preprocessing and feature engineering to improve model performance. By correcting data imbalances, identifying relevant traits, and improving the dataset, the models used the full power of boosting techniques to improve prediction accuracy. This study's detailed research and comparison of several models advances telecoms sector churn prediction. Using data mining to enhance churn prediction is stressed. This study can help telecom businesses design effective churn prediction systems and strategically allocate resources to customer retention. Thus, this reduces client turnover and boosts operational efficiency. New boosting algorithms, complicated feature engineering, and other factors affecting telecom customer turnover may be studied in the future.

We recommend XGBoost for forecasting churn in telecommunication datasets based on our empirical study. Customer Relationship Management (CRM) systems may provide a lot of data regarding inactive customers. This data may be used to create client retention and promotion programs. Scholars have found that XGBoost outperforms other boosting methods and a wide range of machine learning techniques. Future comparative study on algorithm efficacy can use this work as a foundation.

The results of this study offer a strong foundation for researchers and the scholarly community to further explore comparative evaluations utilizing various boosting algorithms, such as CatBoost and LightGBM.

Conflict of Interest: On behalf of all authors, the corresponding author states that there is no conflict of interest.

REFERENCES

- Adekunle, A. (2013). Nigeria; "Mobile Number Portability Now Live". Retrieved from <http://www.allafrica.com/view/group/main/id/00032116.html> 20 October 2013.
- Anjum, A., Raza, B., & Anwar, Z. (2017). Optimizing Coverage of Churn Prediction in Telecommunication Industry. *International Journal of Advanced Computer Science and Applications*, 8(5), 179-188.
- Azeem, M., & Usman, M. (2018). A fuzzy based churn prediction and retention model for prepaid customers in telecom industry. *International Journal of Computational Intelligence Systems*, 11(1), 66-78.
- Bogaert, M., & Delaere, L. (2023). Ensemble Methods in Customer Churn Prediction: A Comparative Analysis of the State-of-the-Art. *Mathematics*, 11(5), 1137.

- Brandusoiu, I., & Todorean, G. (2013). Churn Prediction in the Telecommunications Sector Using Support Vector Machines. *Annals of the Oradea University Fascicle of Management and Technological Engineering*, 22(1), 19-22.
- Chen, Z. Y., Fan, Z. P., & Sun, M. (2018). A Hierarchical Multiple Kernel Support Vector Machine for Customer Churn Prediction Using Longitudinal Behavioral Data. *European Journal of Operational Research*, 223(2), 461-472.
- Dalli, A. (2022). Impact of hyperparameters on Deep Learning model for customer churn prediction in telecommunication sector. *Mathematical Problems in Engineering*, 2022, 1-11.
- Den Poel, V. D., & Lariviere, B. (2014). Customer attrition analysis for financial services using proportional hazard models. *European Journal of Operational Research*, 157(1), 196-217.
- Fujo, S. W., Subramanian, S., & Khder, M. A. (2022). Customer churn prediction in telecommunication industry using deep learning. *Information Sciences Letters*, 11(1), 24.
- Gotovac, S. (2015). Modeling data mining applications for prediction of prepaid churn in telecommunication services. *Telecommunications Policy*, 39(3-4), 275-283.
- Gunnarsson, B. R., Vanden Broucke, S., Baesens, B., Óskarsdóttir, M., & Lemahieu, W. (2021). Deep learning for credit scoring: Do or don't?. *European Journal of Operational Research*, 295(1), 292-305.
- Hung, S., Wang, H., & Yen, D. C. (2016). Applying data mining to telecom churn management. *Expert Systems with Applications*, 31, 515-524.
- Idris, M., & Khan, A. (2012). Customer churn prediction for telecommunication: Employing various features selection techniques and tree-based ensemble classifiers. In 15th International Multitopic Conference (INMIC) (pp. 23-27). IEEE.
- Ismail, M. R., Awang, M. K., Rahman, M. N. A., & Makhtar, M. (2015). A Multi-Layer Perceptron Approach for Customer Churn Prediction. *International Journal of Multimedia and Ubiquitous Engineering*, 10(7), 213-222.
- Jhon, H. (2013). A customer profiling methodology for churn prediction (Doctoral dissertation).
- Keramati, A., Marandi, R. J., Aliannejadi, M., Ahmadian, I., Mozzafari, M., & Abbasi, U. (2014). Improved Churn Prediction in Telecommunications Industry Using Data mining Techniques. *Applied Soft Computing*, 24, 994-1012.
- Kim, K., Jun, C. H., & Lee, J. (2014). Improved Churn Prediction in Telecommunications Industry by Analyzing a Large Network. *Expert Systems with Applications*, 41(15), 6575-6584.
- Kimura, T. (2022). CUSTOMER CHURN PREDICTION WITH HYBRID RESAMPLING AND ENSEMBLE LEARNING. *Journal of Management Information & Decision Sciences*, 25(1).
- Kirui, C., Hong, L., Cheruiyot, W., & Kirui, H. (2013). Predicting Customer Churn in Mobile Telephony Industry Using Probabilistic Classifiers in Data Mining. *International Journal of Computer Science Issues*, 10(2), 165-172.
- Kisioglu, P., & Topcu, I. Y. (2016). Applying Bayesian belief network approach to customer churn analysis: a case study on the telecom industry of Turkey. *Expert Systems with Applications*, 38, 7151-7157.
- Kumar, A. S., & Chandrakala, D. (2016). A survey on customer churn prediction using machine learning techniques. *International Journal of Computer Applications*, 154(10), 18-25.
- Menardi, G., & Torelli, N. (2014). Training and assessing classification rules with imbalanced data. *Data Mining and Knowledge Discovery*, 28(1), 92-122.
- Mumin Yıldız, Y. A. (2015). Customer Churn Prediction in Telecommunication with Rotation Forest Method. *Stimulation Modelling Practice and Theory*, 55, 1-9.

- Oladipo, I. D., Awotunde, J. B., AbdulRaheem, M., Taofeek-Ibrahim, F. A., Obaje, O., & Ndunagu, J. N. (2023). Customer Churn Prediction in Telecommunications Using Ensemble Technique. *University of Ibadan Journal of Science and Logics in ICT Research*, 9(1).
- Ominike, A. (2016). Mobile Number Portability (MNP) in Nigeria, 41.
- Paul, (2015). Keep those customers [retaining and expanding existing customer base]. *Engineering Management*, 14(6), 22-23.
- Phua, C., Cao, H., Gomes, J. B., & Nguyen, M. N. (2015). Predicting Near-Future Churners and Win-Backs in the Telecommunications Industry. arXiv preprint arXiv:1210.6891.
- Rahman, S., Irfan, M., Raza, M., Moyeezullah Ghor, K., Yaqoob, S., & Awais, M. (2020). Performance Analysis of Boosting Classifiers in Recognizing Activities of Daily Living. *International journal of environmental research and public health*, 17(3), 1082. <https://doi.org/10.3390/ijerph17031082>
- Sabbath, S. F. (2018). Machine Learning technique for customer retention. *International Journal of Advanced Computer Science and Applications*, 9(2).
- Saleh, S., & Saha, S. (2023). Customer retention and churn prediction in the telecommunication industry: a case study on a Danish university. *SN Applied Sciences*, 5(7), 173.
- Seon, D., Ranganathan, C., & Babad, Y. (2014). Two-level model of customer retention in the US mobile telecommunications service market. *Telecommunications Policy*, 32(3), 182-196.
- Shaaban, E., Helmy, Y., Khedr, A., & Nasr, M. (2015). A Proposed Churn Prediction Model. *International Journal of Engineering Research and Applications*, 2(4), 693-697.
- Tiwari, A., Hadden, J., & Turner, C. (2015, March). A new neural network based customer profiling methodology for churn prediction. In *International Conference on Computational Science and Its Applications* (pp. 358-369). Springer, Berlin.
- Tomar, D., & Agarwal, S. (2014). A Survey in Pre-Processing and Post-Processing
- Umayaparvathi, V., & Iyakutti, K. (2015). Applications of Data Mining Techniques in Telecom Churn Prediction. *International Journal of Computer Applications*, 42(20), 1-5.
- Ullah, I., Raza, B., Malik, A. K., Imran, M., Islam, S. U., & Kim, S. W. (2019). A churn prediction model using random forest: analysis of machine learning techniques for churn prediction and factor identification in telecom sector. *IEEE Access*, 7, 60134-60149.
- Umayaparvathi, V., & Iyakutti, K. (2016). A survey on customer churn prediction in telecom industry: datasets, methods, and metric. *Int Res J Eng Technol*, 3(4), 1065-1070.
- Vafeiadis, T., Diamantaras, K. I., Sarigiannidis, G., & Chatzisavvas, K. C. (2015). A comparison of machine learning techniques for customer churn prediction. *Simulation Modelling Practice and Theory*, 55, 1-9.
- Verbeke, W., Dejaeger, K., Martens, D., Hur, J., & Baesens, B. (2016). New Insights into Churn Prediction in the Telecommunications Sector: A Profit Driven Data Mining Approach. *European Journal of Operational Research*, 218, 211-229.
- Vorraboot, P., Rasmequan, S., Chinnasarn, K., & Lursinsap, C. (2015). Improving classification rate constrained to imbalanced data between overlapped and non-overlapped regions by hybrid algorithms. *Neurocomputing*, 156, 1-13.
- Wagh, S. K., Andhale, A. A., Wagh, K. S., Pansare, J. R., Ambadekar, S. P., & Gawande, S. H. (2023). Customer churn prediction in telecom sector using machine learning techniques. *Results in Control and Optimization*, 100342.