Global Scientific JOURNALS

GSJ: Volume 12, Issue 1, January 2024, Online: ISSN 2320-9186
www.globalscientificjournal.com

Research Article

Customer churn prediction model enhacement for the telecommunication industry using data transformation methods and feature selection

Zaineb Boujelbene[*], Mohamed Yessine Labidi[*], Achref Lemjid [*], Lotfi Ncib [**], Mohamed khalil Zghal [*]

[*] Private Higher School of Engineering and Technology
[**] Ridcha Data

[*] zaineb.boujelbene@esprit.tn

Abstract

In the realm of the telecommunications industry, customer churn stands as a paramount challenge. To address this issue, researchers and analysts make extensive use of customer relationship management data and employ diverse machine learning models and data manipulation techniques to pinpoint potential churners.
In the telecommunications industry, addressing customer churn is pivotal. Our research builds upon prior work, introducing substantial enhancements with the primary goal of advancing the effectiveness of the telecom sector, we embarked on an exploration of multiple machine and deep learning models, coupled with data transformation methods. Our optimization efforts encompassed the application of novel feature selection techniques and the fine-tuning of hyperparameters. Subsequently, we conducted experiments on the same publicly available telecommunications datasets, previously utilized in related research, using widely accepted evaluation metrics like AUC, precision, recall, and F-measure.
Our suggested methods specially using the bayesian optimization enhanced the accuracy of predictions to reach a maximum of 96% and a minimum of 93% .
Key words: Churn prediction, Telecommunication, feature selection methods, DT methods, Hyperparameters.

Introduction

The development and rapid digitalization of the world have revolutionized business practices, leading to the emergence of new models such as subscription-based services. While these innovations offer possibilities, they also bring forth challenges that demand contemporary solutions. One significant challenge faced by the telecommunication industry is customer churn, where customers frequently switch between providers, impacting the revenue, profitability, and reputation of telecom companies [18]. In this fiercely competitive landscape, proactive customer retention measures have become essential to stay ahead [18].
Customer knowledge plays a vital role in understanding and addressing churn. Customer relationship management systems facilitate interactions between businesses and customers, enabling the collection, storage, and analysis of valuable customer

data [2]. Over the years, these systems have evolved and become more sophisticated, leveraging technology and data analysis tools to gain comprehensive insights into customer behaviors.

The vast amount of available data and information has grown exponentially, allowing for the storage and processing of large datasets. However, this abundance of data also necessitates the automatic extraction of meaningful information and knowledge. To prevent telecom churn, firms can utilize data mining techniques and machine learning algorithms to derive valuable insights from stored data. By doing so, they can identify potential churners and implement targeted retention strategies effectively .

In summary, the digital era has transformed the way businesses operate, and subscription based services have become a prominent result of this evolution.// Among the various challenges faced by the telecommunications industry, customer churn stands out as a critical concern. However, leveraging customer knowledge through sophisticated customer relationship management systems and advanced data analysis techniques empowers telecom companies to proactively combat churn and retain their valuable customer base.

1 Litterature Review

The issue of the customer churn prediction (CCP) has been addressed through various methods, including machine learning models, data mining techniques, and hybrid approaches [3–5]. Numerous machine learning (ML) and data mining strategies have been employed to analyze and understand the complexities associated with the CCP. Several customer churn prediction models have been developed that leverage various machine learning algorithms and data transformation (DT) methods. In particular, eight different classifiers combined with six different DT methods was used to develop a numberof models to handle the CCP (Customer churn prediction) problem. The classification algorithms used include K-Nearest Neighbor (KNN), Naïve Bayes (NB), Logistic Regression (LR), Random forest (RF), Decision tree (DTree), Gradient boosting (GB), Feed-Forward Neural Networks (FNN), and Recurrent Neural Networks (RNN). On the other hand, the DT methods that have been applied are: Log, Rank, Box-cox, Z-score, Discretization, and Weight-of-evidence (WOE). To optimize the machine learning classifiers, univariate technique has been performed to select the most effective features and grid search method has been used to find the best hyperparameters. Extensive experiments have been conducted on four different publicly available datasets and the models have been evaluated using various information retrieval metrics such as AUC, Precision, Recall, and F measure. The experimental results clearly demonstrate that the DT methods have a positive impact on CCP models. In the same context, we briefly review some of the most relevant papers below. The study by Amin et al [7] is more recent, they investigate the importance of data transformation methods in the context of the CCP in TCI problem. However, they have used only one dataset for model training and another dataset for independent testing. In the study made by Coussement et al [8] two DT methods, Discretization and Weight-of-evidence, have been implemented. However, the authors experimented with only one dataset.

Sana JK, Abedin MZ, Rahman MS and Rahman MS3 [6] studies are very relevant to our work, they used 4 different datasets with a consistant results focusing on data transformation.

On the other hand, the experiments of this study have been focusing on trying different feature selection and the hyperparameter tunning methods which explin the improvement of the results.

A concrete decision on the best combination of DT method and prediction model with the right feature selection techniques to construct a successful CCP model was provided.

Specifically, the Weight-of-evidence (WOE) for data transformation, followed by model training with Logistic Regression (LR) or Feed-Forward Neural Networks (FNN).

The following tables summarise the different techniques used in the literature.

Table 1. Feature selection methods used in prior studies

Study	**Feature selection Methods**
Amin et al.	No
Coussement et al.	No
Amin et al.	**Yes**
Makhtar et al.	No
Amin et al.	No
Burez et al.	No
Qureshi et al.	No
Etaiwi et al.	No
Melian et al.	No
Andreea et al.	**Yes**
Sana et al.	**Yes**

Table 2. Hyperparameter tuning methods used in prior studies

Study	**Hyperparameter tuning Methods**	
	Grid search CV	**Bayesian optimization**
Amin et al.	No	No
Coussement et al.	No	No
Amin et al.	No	No
Makhtar et al.	No	No
Amin et al.	No	No
Burez et al.	No	No
Qureshi et al.	No	No
Etaiwi et al.	No	No
Melian et al.	No	No
Andreea et al.	No	No
Sana et al.	**Yes**	No

Table 3. Data transformation methods used in prior studies

Study	**Data Transformation Methods**							
	Log	**Rank**	**Box-Cox**	**Z-Score**	**Discretization**	**WOE**	**OHE**	**Y-J power**
Amin et al.	**Yes**	**Yes**	No	No	No	No	No	No
Coussement et al.	No	No	No	No	**Yes**	**Yes**	No	No
Amin et al.	**Yes**	**Yes**	**Yes**	**Yes**	No	No	No	No
Makhtar et al.	No	No	No	No	No	No	No	No
Amin et al.	No	No	No	No	No	No	No	No
Burez et al.	No	No	No	No	No	No	No	No
Qureshi et al.	No	No	No	No	No	No	No	No
Etaiwi et al.	No	No	No	No	No	No	No	No
Melian et al.	No	No	No	No	No	No	No	No
Andreea et al.	No	No	No	No	No	No	No	No
Sana et al.	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**	No	No

2 Our Contributions

The proposed solution builds upon the existing one by introducing several enhancements to improve the customer churn prediction (CCP) models. First, a new data transformation (DT) method, Yeo-Johnson power transformation, has been added to the existing set of DT methods (Log, Rank, Box-cox, Z-score, Discretization, and Weight-of-evidence). Secondly, to select the most effective features, Pearson's correlation coefficient has been employed as a feature selection technique.
Moreover, the proposed solution utilizes various wrapper feature selection methods, including Forward selection with K-Nearest Neighbor (KNN), L1 Regularization (Lasso) with Logistic Regression and Feed-Forward Neural Networks (FNN), and Tree-based Feature Importance with Decision Tree and Random Forest. These methods help to identify important features for each classifier and potentially enhance model performance.
Furthermore, variance thresholding is applied with all classifiers to retain only the features with significant variability, thus reducing noise and potentially improving the model's generalization.
Finally, Bayesian Optimization is utilized for hyperparameter tuning, which efficiently searches for the best combination of hyperparameters for each classifier.

3 Proposed Methodology

3.1 Techniques Used

In this study, various modeling techniques have been employed to address classification challenges. Traditional machine learning models, such as decision trees(DT), Random forests, logistic regression, K Nearest Neighbor (KNN), Gaussian Naive Bayes, Gradient Boosting, AdaBoost, XGBoost and Support Vector Machines (SVM), offer a robust approach based on well-established algorithms. On the other hand, deep learning models, including Artificial Neural Networks (ANN), Feed-forward Neural Networks (FNN), and Recurrent Neural Networks (RNN), leverage complex architectures inspired by the human nervous system. These deep approaches excel in recognizing intricate patterns and predicting sequences, promising significant advancements in tasks such as forecasting temporal behavior in telecommunications data. Each approach, whether traditional or deep, brings distinct advantages depending on the nature and complexity of the processed data.

3.2 Data Transformation Methods

Data transformation is the process of converting raw data into a format or structure that would be more suitable for model building and also data discovery in general. It is an imperative step in feature engineering that facilitates discovering insights.
To improve the customer churn prediction (CCP) models, several data transformation (DT) methods were implemented : Label Encoding , Yeo-Johnson Power Transformation.

In this study we used one hot encoding as it is one of the most powerful techniques for handling categorical data in machine learning , yet easy to compute and simple to interpret, some of the advantages of this method are that it preserves all the information in the categorical variables ensuring no loss of data , avoids misinterpretations that can occur with label encoding and prevents biases in some algorithms.
Yeo-johnson power transformation was also used , it offers several benefits when applied to feature engineering , it stabilizes the variance reducing the impact of heteroscedasticity and the influence of extreme outliers by compressing or stretching the data , it is flexible by allowing a wide range of transformations by adjusting the parameter.

3.3 Feature Selection Methods

It is one of the main components of feature engineering which presents the process of selecting the most important features to input in machine learning algorithms, its techniques are employed to reduce the number of input variables by eliminating redundant or irrelevant features and narrowing down the set of features to those most relevant to the machine learning model.

3.3.1 Filter Methods

Filter methods in machine learning are employed for feature selection prior to the model training phase. These techniques focus on identifying the most pertinent features based on their statistical properties.
Variance Threshold, a straightforward method, eliminates features with insufficient variance, particularly those with zero variance across all samples.
Pearson's correlation coefficients serve as a measure of linear relationships between variables, guiding feature selection by emphasizing high correlation with the target variable while promoting low correlation among features. This approach ensures that selected variables contribute distinct information to the model.
Tree-based Feature Importance method assesses feature significance, commonly applied in decision trees and random forests. This technique calculates importance by evaluating how much each feature reduces the weighted impurity in a tree, with higher values indicating greater relevance within the model.
SelectPercentile method chooses a subset of features based on their individual statistical properties rather than taking into account the interactions between features. The main idea is to preserve the top percentage of features that demonstrate the highest scores.
Overall, these filter methods play a crucial role in enhancing the efficiency and relevance of machine learning models by selecting features based on their intrinsic statistical properties.

3.3.2 Wrapper Methods

Wrapper methods are employed in machine learning to assess multiple models through procedures that systematically add or remove predictors, aiming to identify the optimal combination that maximizes overall model performance. These procedures are often designed following the concept of a Greedy algorithm, characterized by making locally optimal choices at each stage of the process.
A specific wrapper method, **Forward Selection**, initiates the process with one predictor and incrementally adds more predictors in iterations. In each iteration, the best-performing predictor among the remaining original predictors is added based on predefined performance criteria. This iterative and criteria-driven approach allows the method to gradually build a predictive model, selecting predictors that contribute most effectively to overall model performance.

3.3.3 Embedded Methods

Embedded methods in machine learning involve feature selection techniques integrated into the model training process with the goal of identifying the most relevant features and excluding irrelevant ones. Particularly beneficial when dealing with a large number of features to prevent overfitting, embedded methods enhance model performance and generalization on new data by focusing on key features.
Regularization techniques, such as **L1** & **L2 regularizations**, play a crucial role in controlling model complexity.

L1 regularization adds a penalty term proportional to the absolute values of the model's coefficients, encouraging them to be close to zero. This simultaneous feature selection and regularization help prevent overfitting. The mathematical representation of L1 regularization includes a penalty term in the cost function.

$$Cost = \sum_{i=0}^{N}(Y_i - \sum_{j=0}^{M} X_{ij}W_j)^2 + \lambda \sum_{j=0}^{M} |W_j|$$

Additionally, **L2 regularization**, known as Ridge regression, differs from L1 by using the squared magnitude of coefficients as the penalty term.

$$Cost = Loss\ \ Function + \lambda \sum_{j=0}^{M} W_j^2$$

The highlighted part in the cost function represents the L2 regularization element. Choosing an appropriate lambda parameter is vital, as it balances the regularization effect. While a small lambda preserves Ordinary Least Squares, a large lambda prevents overfitting, striking a balance to optimize model performance.

3.4 Hyperparameters Tuning Methods

Hyperparameters are preset configurations for machine learning models, like learning rate and regularization strength, set before training so they control various aspects of its process and can significantly impact a model's performance.
Hyperparameter tuning is a crucial part of the machine learning workflow , it involves finding the best combination of these settings to maximize and fine-tune the model for an optimal performance and a generalization to unseen data.

3.4.1 GridSearchCV

GridSearch involves defining a grid of hyperparameter values to explore. It systematically searches through all possible combinations of hyperparameters within the predefined grid and evaluates the performance of the model for each combination using a cross-validation strategy. The goal is to find the combination of hyperparameters that results in the best model performance, as measured by a specified evaluation metric (e.g., accuracy, F1 score, mean squared error).

3.4.2 Bayesian Optimization

It is a probabilistic model-based approach that uses Bayesian inference to find the optimal hyperparameters for a machine learning model. It works by iteratively selecting hyperparameters to evaluate based on the results of previous evaluations. The goal is to find the set of hyperparameters that maximizes the performance on a validation set by choosing its parameter combinations in an informed way, it enables itself to focus on those areas of the parameter space that it believes will bring the most promising validation scores.

4 Implementation

4.1 Dataset Used

This study was performed on the same four publicly available benchmark datasets that were used on the existing solution mentioned in the Paper ”A novel customer churn prediction model for the telecommunication industry using data transformation methods and feature selection”.
This table briefly describes these datasets:

Table 4. Summary of the datasets used in this study

Description	Dataset 1	Dataset 2	Dataset 3	Dataset 4
N° of samples	10000	5000	3333	7043
N° of attributes	50	20	21	21
N° of class labels	2	2	2	2
% of churn samples	26.50	85.86	85.50	26.54
% of non-churn samples	73.50	14.14	14.50	73.46
Data source	URL1	URL2	URL3	URL4

- URL1: https://github.com/jlopez873/Customer%Churn_Analysis_KNN/blob/main/data
- URL2: https://data.world/earino/churn
- URL3: https://www.kaggle.com/becksddf/churn-in-telecoms-dataset/data
- URL4: https://www.kaggle.com/blastchar/telco-customer-churn

The datasets include information about customers that churned during the last month in a column named “Churn”, services that each customer has been using such as phone, multiple lines, internet, online security, online backup, device protection, tech support, and streaming TV and movies. Additionally, they include demographic information about the customers(gender, age range, and if they have partners and dependents) and finally customer account information (how long they've been a customer, contract, payment method, paperless billing, monthly charges, and total charges).

4.2 Exploratory Data Analysis

Among the four datasets that we worked with, we opted to focus on one specific dataset to provide a comprehensive overview of our visualization process. The selected dataset, named "Dataset 4" contains about 7043 customers, including 5174 (73.5%) non churners and 1869 (26.5%) churners, with 26 variables devided by the type, 15 categorical, 4 Numeric and 6 boolean ones.

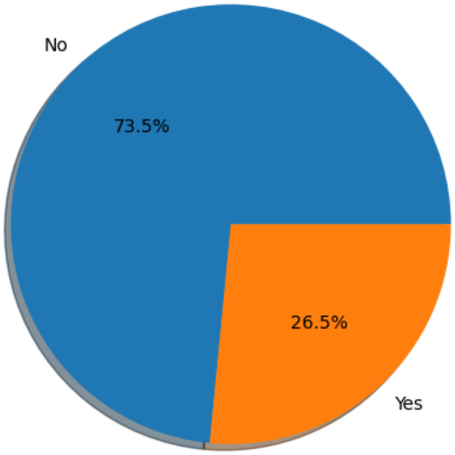


Fig 1. Dataset 4 Churn Label Distribution

4.3 Data Preprocessing

Data cleaning is a functional first step in understanding the data and where the quality issues are coming from. It involves detecting and correcting or deleting corrupt or inaccurate records from a set of records taken from a table or database, before moving on to analysis. Without appropriate data quality, the final analysis will have poor precision.
In our case, the fact that there are no missing values suggests that our dataset is complete for the specific columns we analyzed and we can build models without worrying about imputation or treatment of missing data, which simplifies the following steps.

- The sample IDs and/or descriptive texts which are used only for informational purposes are ignored.
- Redundant attributes have been removed.
- Missing numerical values were replaced with zero (0) and missing categorical values were treated as a separate category.
- Following previous literature, categorical values were encoded such as 'yes' or 'true' as 1, while 'no' or 'false' as 0. For other categorical variables, a **Label Encoder** from the scikit-learn Python library was employed to generate numeric representations. Additionally, numerical values underwent normalization using the **Yeo Johnson Power Transformation** method to ensure a more Gaussian-like distribution for numerical features.
- Correlation matrix was presented to help us better understand our data and identify the factors that influence customer churn.

4.4 Feature Selection

In this work, we strategically utilized various feature selection methods to enhance the performance of the models we developed. Here's a breakdown of our approach:

- **Pre-model Training phase:**
  We employed filter methods such as Variance Threshold, Pearson's correlation coefficients, and the SelectPercentile method to identify features based on a user-defined statistical test and significant properties.
- **Model Training Phase:**
  We used wrapper methods, starting with the Forward Selection wrapper method, to incrementally add predictors based on predefined performance criteria in order to construct an optimized predictive model.
  Additionally, we employed the Tree-based Feature Importance method to evaluate the relevance of features based on their contribution to reducing impurity in decision trees and random forests.
  Finally for the embedded methods ,Regularization techniques, including L1 and L2 regularizations, were implemented to control model complexity. This approach facilitated the identification of the most relevant features and exclusion of potentially irrelevant ones, preventing overfitting and improving the model's generalization on new data, thus enhancing model training efficiency.

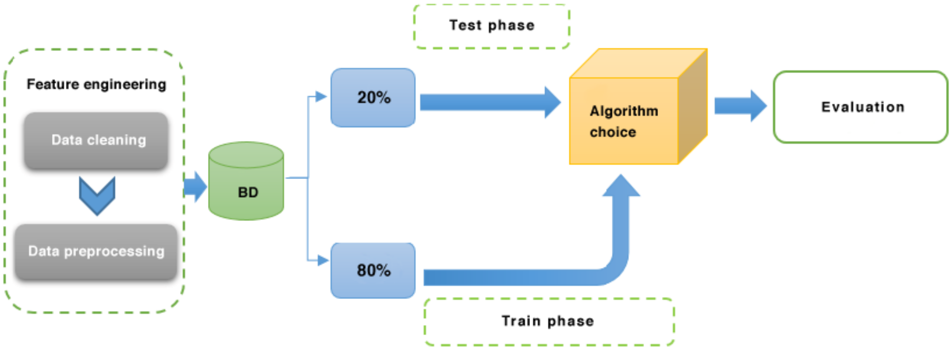


Fig 2. Architecture of the preposed research model

4.5 Validation method and steps

After data preprocessing, we applied several machine learning approaches to it for the purpose of making predictions. When implementing them, training and testing data need to be transformed in the same way. This is usually achieved by feeding the **80**% training dataset to building the data transformation or Feature Selection algorithm and then apply that algorithm to the **20**% test set.
To address the issue of imbalanced data, we incorporated stratified sampling during the train-test split. Stratification ensures that the distribution of classes is maintained in both the training and testing sets.

The simulations were conducted on a system equipped with an 11th Gen Intel(R) Core(TM) i7-10750H processor running at 2.60GHz, with 16.0 GB of RAM memory. The system operated on a 64-bit operating system, x64-based processor, and ran Windows 11 Home Single Language. Detailed specifications for the various classifiers used in the study, which were determined through iterative testing, are provided in the tables below.

Table 5. Dataset 4 - Model Specification

Technique Used	Specifications

K-Nearest Neighbors	N_neighbors = 3
	Weights = uniform
	Algorithm = auto
	Leaf_size=30
	Metric= minkowski

Random Forest	N_estimators = 100
	Criterion= entropy
	Random_state=None

Logistic Regression	Penalty = L2
	Intercept_scaling = 1
	Solver = lbfgs

Decision Tree	Criterion=gini
	Splitter=best
	Max_depth = None

Support Vector Machine	Gamma = auto
	kernel = rbf

Gradient Boosting	Learning Rate = 0.01 N_estimators = 100

Light Gradient Boosting Machine	N_estimators = hp.quniform('n_estimators', 50, 150, 1)
	Max_depth=hp.quniform('max_depth', 3, 15, 1)
	learning_rate=hp.loguniform('learning_rate', np.log(0.01), np.log(0.5))
	Max_evals = 100
	Algo = TPE (Three-structured Parzen Estimator)

XGBoost	N_estimators = 90
	Learning_rate = 1.0
	Algorithm=SAMME

AdaBoost	N_estimators = 90
	Learning_rate = 1.0
	Algorithm=SAMME

Gaussian Naive Bayes	Priors = None
	Var_smoothing = $1e^{-09}$

Artificial Neural Network	Dense_Layer=16, Trans_Function=ReLu
	Dense_Layer=8, Trans_Function=ReLu
	Dense_Layer=1, Trans_Function=Sigmoid
	Optimizers = RMSProp
	Loss = Binary Crossentropy

Feedforward Neural Network	Dense_Layer=16, Trans_Function=ReLu
	Dense_Layer=8, Trans_Function=ReLu
	Dense_Layer=1, Trans_Function=Sigmoid
	Optimizers = RMSProp
	Loss = Binary Crossentropy

Recurrent Neural Networks	Threshold = 0.01
	LSTM_Layer=32
	Dense_Layer=1, Trans_Function=Sigmoid
	Optimizers = RMSProp
	Loss = Binary Crossentropy

4.6 Performance measures

Different performance measures such as accuracy, recall, precision and F1 score value of the suggested predictive system for consumer behavior were assessed in this study.

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN}$$

$$Precision = \frac{TP}{TP+FP}$$

$$Recall/POD/Sensitivity = \frac{TP}{TP+FN}$$

$$POFA = \frac{FP}{TN+FP}$$

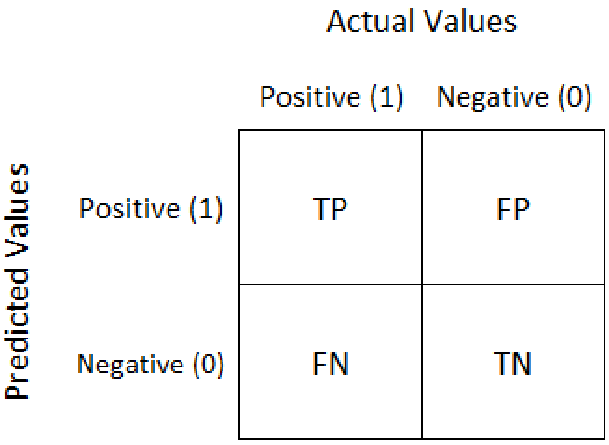
$$Specificity = \frac{TN}{TN+FP}$$

$$F1 - Score = \frac{2 * precision * recall}{precision + recall}$$

$$AUC = \frac{1 + POD - POFA}{2}$$

Where:

- **TP (True Positives):** is an outcome where the model correctly predicts the positive class.
- **TN (True Negatives):** is an outcome where the model correctly predicts the negative class.
- **FP (False Positive):** is an outcome where the model incorrectly predicts the positive class.
- **FN (False Negative):** is an outcome where the model incorrectly predicts the negative class.



		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Fig 3. Confusion Matrix

5 Results & Discussion

In our study, centered on churn prediction within the telecommunications sector, we prioritized the critical role of data transformation during algorithm implementation. Consistency between our training and testing datasets was paramount, necessitating the consistent application of data transformation methods to both sets. This involved initially applying the transformation algorithm to the training dataset and seamlessly extending its use to the test dataset.

Within this context , the Yeo-Johnson Power Transformer [18] consistently emerged as the superior choice across various classifiers, consistently delivering higher accuracy and F1-Score values. Notably, when paired with the Random Forest classifier, the Yeo-Johnson transformation achieved an impressive 94% accuracy and a 78% F1-Score. Similarly, with gradient boosting, we observed a 94% accuracy and a 79% F1-score. These results underscore the significant advantages of the Yeo-Johnson transformation in augmenting model performance and resilience within our specific dataset.

Moreover, our study conducted a thorough comparative analysis, evaluating diverse feature selection techniques in a classification context. We assessed these techniques using key metrics, including accuracy, precision, recall, and F1-score. Feature selection methods were categorized into two groups: during the training phase and before it. For the former, we employed the Forward Selection wrapper method, incrementally adding predictors based on predefined performance criteria to construct an optimized predictive model. Additionally, the Tree-based Feature Importance method was utilized to evaluate feature relevance based on their contribution to reducing impurity in decision trees and random forests.

These results were obtained across a range of classifiers, including 'KNN', 'DT', 'RF', 'LR', 'SVM', 'GB', 'XGB', 'AdaBoost', 'GNB', 'ANN', 'FNN', and 'RNN'. Notably, our findings revealed consistently in an average values a high accuracies ranging from 83% as a minimum to 94% as a maximum, with F1 scores proportionally elevated, reaching a maximum of 79%. For instance, the 'RF' (RandomForest) classifier achieved an accuracy of 93.09% and an F1-score of 78.19%, while the 'XGB' (XGBoost) classifier posted an accuracy of 94.20% and an F1-score of 78.61%. Importantly, these results demonstrate a substantial improvement compared to the original study, showcasing the efficacy of our refined approach in churn prediction within the telecommunications domain.

Table 6. Dataset 3 - Performance of the Classifier Models

Classifier	Accuracy	Precision	Recall	F1-Score	AUC
KNN	88.61%	71.07%	38.23%	50.45%	67.16%
DecisionTree	89.84%	66.54%	64.31%	64.90%	81.41%
RandomForest	93.09%	84.65%	66.32%	74.60%	85.25%
Logistic Regression	85.99%	54.16%	22.5%	32.84%	58.51%
SVM	91.36%	86.01%	48.36%	62.81%	66.90%
Gradient Boosting	94.32%	87.17%	72.80%	79.53%	85.22%
XGBoost	94.30%	87.34%	71.13%	78.41%	84.69%
AdaBoost	86.40%	55.98%	32.00%	40.61%	64.38%
Gaussian NB	85.96%	71.33%	27.00%	30.50%	71.00%
ANN	83.00%	43.00%	49.00%	46.00%	80.00%
FNN	90.47%	71.22%	53.60%	58.64%	84.30%
RNN	93.00%	84.00%	65.00%	74.00%	82.00%

Table 7. Dataset 4 - Performance of the Classifier Models

Classifier	Accuracy	Precision	Recall	F1-Score	AUC
KNN	89.60%	69.00%	68.80%	59.20%	62.21%
DecisionTree	90.11%	78.51%	73.44%	74.21%	82.33%
RandomForest	93.00%	84.24%	74.35%	74.78%	86.15%
Logistic Regression	89.63%	81.20%	80.71%	66.98%	71.16%
SVM	90.55%	87.20%	69.47%	68.33%	70.98%
Gradient Boosting	92.14%	82.33%	74.00%	69.22%	80.41%
XGBoost	91.52%	79.20%	74.33%	59.61%	71.67%
AdaBoost	89.79%	62.44%	58.22%	49.88%	69.71%
Gaussian NB	87.16%	72.11%	41.55%	47.96%	70.00%
ANN	89.83%	80.05%	79.82%	79.49%	81.37%
FNN	87.66%	80.19%	79.61%	79.04%	83.94%
RNN	90.12%	82.16%	78.00%	79.30%	80.02%

Figures 4 and 5 below show the accuracy on both the datasets 3 and 4, using a bar graph plot, and line graph plot of the Accuracy and F1-Score measures for some of the classifiers used in this work. In both of the graphs, the distinct difference in the values for both of the datasets can be visualized.

The accuracy line plot helps to visualize that SVM, Gradient Boosting, XGBoost and FNN achieved better efficacy on the third dataset than the fourth; Conversely, for the other classifiers, the opposite holds true: they exhibited better performance on the fourth dataset such that AdaBoost Classifier and ANN.

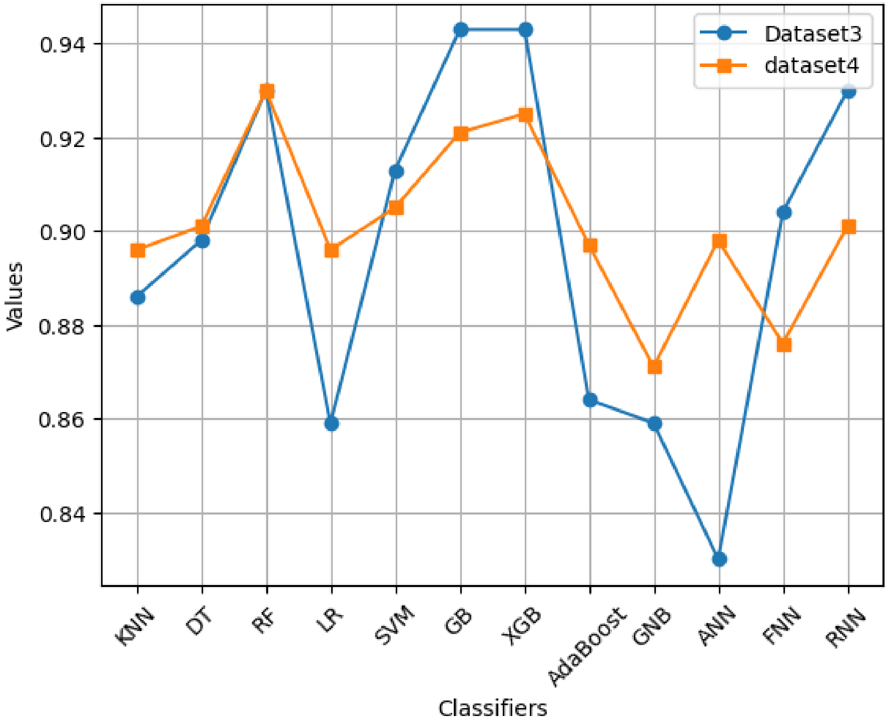


Fig 4. Accuracy Score Plot of different Classifiers on both datasets

The bar graph plot below shows that the accuracy of each classifier is displayed along with the matching F1-Score ratings. We can see that despite having a high accuracy value, some models showed quite a low F1-Score value such that GNB and Logistic Regression for all datasets.

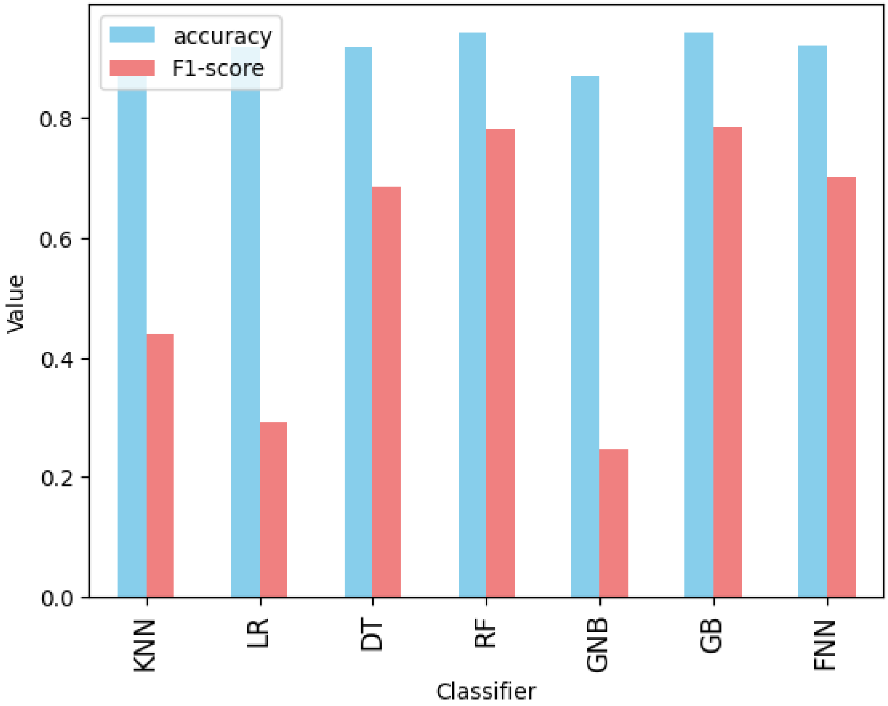


Fig 5. Accuracy and F1-Score Plot using Yeo-Johnson

The graph beneath shows how Bayesian Optimization performs on four different datasets using four different metrics which are Accuracy, Precision, Recall, and F1-Score. The datasets are labeled as Dataset1, Dataset2, Dataset3, and Dataset4. Each dataset has a different color and shape for its line.

- **Dataset1 (blue square)** has consistent performance across all metrics, with values around 0.9.
- **Dataset2 (red diamond)** has variable performance, with the highest value for Precision (0.95) and the lowest value for Recall (0.75).
- **Dataset3 (green triangle)** also has variable performance, but with the highest value for Recall (0.9) and the lowest value for Precision (0.8).
- **Dataset4 (orange circle)** has the highest value for Accuracy (0.93) and the lowest value for F1-Score (0.82).

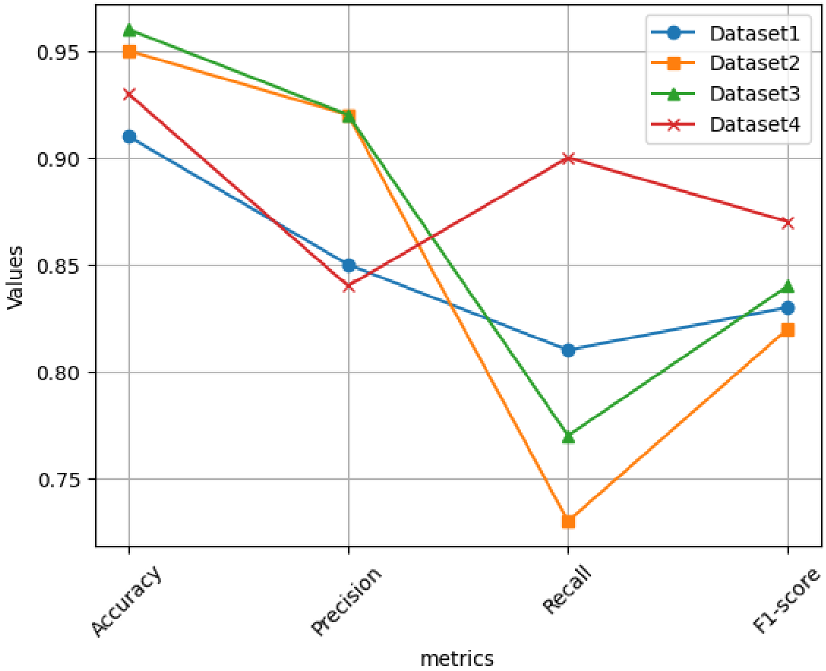


Fig 6. Performance Metrics Visualization for Bayesian Optimization across All Datasets

The next and last figure shows that the new methods have higher accuracy levels than the previous ones for most of the classifiers, except for Gaussian Naive Bayes. The best performing classifier is Recurrent Neural Network, followed by Feedforward Neural Network and Gradient Boosting. The worst performing classifier is Decision Tree, followed by Gaussian Naive Bayes and K-Nearest Neighbors. The figure suggests that the new methods are more effective for complex and nonlinear classifiers than for simple and linear ones.

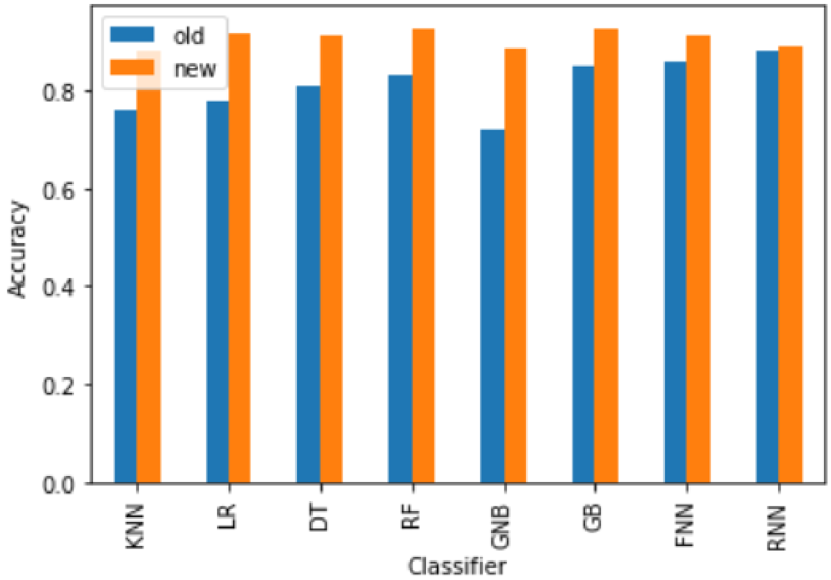


Fig 7. Classifier Performance between Previous and New Methods

Conclusion

One of the most threatening issues that encounter companies is loosing its clients. In the beginning, a company typically focuses on acquiring new clients, then grows by offering additional products to existing clients or trying to get them to use their products more. If all is going well, there comes a point when the company is large enough that it must also choose a slightly more defensive strategy and focus on retaining existing customers. Despite the best user experience, there will always be a group of clients who are not satisfied and decide to leave the company then faces the problem of how to prevent these (voluntary) departures as effectively as possible.

This report has shed light on the significant challenge of customer churn in the telecommunications industry, facilitating interactions between businesses and customers, enabling the collection, storage, and analysis of valuable customer data.

The improvements made to the existing framework open up new possibilities for enhancing customer retention strategies and ultimately strengthening the telecom companies revenue, profitability, and reputation.

One of the key improvements is the integration of various wrapper feature selection methods, such as Forward selection with K-Nearest Neighbor, L1 Regularization with Logistic Regression and Feed-Forward Neural Networks, and Tree-based Feature Importance with Decision Tree and Random Forest, brings an added layer of refinement. These methods aid in selecting the most critical features for each classifier, leading to more accurate predictions and increased model performance.

Also, the application of variance thresholding with all classifiers contributes to noise reduction and better generalization. By retaining only the features with significant variability, the models become more robust and capable of handling unseen data effectively.

In conclusion, the proposed solution offers a comprehensive and sophisticated approach to address the challenge of customer churn in the telecommunications industry. By leveraging advanced data transformation, feature selection, and hyperparameter tuning techniques, telecom companies can better understand customer behaviors and proactively combat churn.

References

1. Oskarsdo ttir M, Bravo C, Verbeke W, Sarraute C, Baesens B, Vanathien J. Social Network Analytics for Churn Prediction in Telco: Model Building, Evaluation and Network Architecture. Expert Systems with Applications. 2017; 85.

2. Amin A, Anwar S, Adnan A, Nawaz M, Aloufi K, Hussain A, et al. Customer Churn Prediction in Tele- communication Sector using Rough Set Approach. Neurocomputing. 2016.

3. Idris A, Khan A. Customer churn prediction for telecommunication: Employing various various features selection techniques and tree based ensemble classifiers. 2012. p. 23–27.

4. 8. Hung SY, Yen D, Wang HY. Applying data mining to telecom chum management. Expert Systems with Applications. 2006; 31:515–524. https://doi.org/10.1016/j.eswa.2005.09.080

5. He Y, He Z, Zhang D. A Study on Prediction of Customer Churn in Fixed Communication Network Based on Data Mining. 2009. p. 92–94.

6. Sana JK, Abedin MZ, Rahman MS, Rahman MS. A novel customer churn prediction model for the telecommunication industry using data transformation methods and feature selection. PLoS ONE 17(12): e0278095. https://doi.org/10.1371/journal.pone.0278095

7. Amin A, Shah B, Khattak AM, Lopes Moreira FJ, Ali G, Rocha A, et al. Cross-company customer churn prediction in telecommunication: A comparison of data transformation methods. International Journal of Information Management. 2019; 46:304–319. https://doi.org/10.1016/j.ijinfomgt.2018.08.015

8. Coussement K, Lessmann S, Verstraeten G. A comparative analysis of data preparation algorithms for customer churn prediction: A case study in the telecommunication industry. Decision Support Systems. 2017; 95:27–36. https://doi.org/10.1016/j.dss.2016.11.007

9. Conant GC, Wolfe KH. Turning a hobby into a job: how duplicated genes find new functions. Nat Rev Genet. 2008 Dec;9(12):938–950.

10. Ohno S. Evolution by gene duplication. London: George Alien & Unwin Ltd. Berlin, Heidelberg and New York: Springer-Verlag.; 1970.

11. Magwire MM, Bayer F, Webster CL, Cao C, Jiggins FM. Successive increases in the resistance of Drosophila to viral infection through a transposon insertion followed by a Duplication. PLoS Genet. 2011 Oct;7(10):e1002337.

12. Makhtar M,Nafis s, Mohamed MA, Awang MK, Rahman MNA, Mat Deris M. Churn classification model for local telecommunication company based on rough set theory. Journal of Fundamental and Applied Sciences. 2017; 9(6):854–68.

13. Burez J, Van den Poel D. Handling class imbalance in customer churn prediction. Expert Systems with Applications. 2009; 36(3 PART 1):4626–4636. https://doi.org/10.1016/j.eswa.2008.05.027

14. Etaiwi W, Biltawi M, Naymat G. Evaluation of classification algorithms for banking customer's behavior under Apache Spark Data Processing System. Procedia Computer Science. 2017; 113:559—564. https://doi.org/10.1016/j.procs.2017.08.280

15. Amin A, Shah B, Khattak AM, Baker T, u Rahman Durani H, Anwar S. Just-in-time Customer Churn Pre- diction: With and Without Data Transformation. In: 2018 IEEE Congress on Evolutionary Computation (CEC); 2018. p. 1–6.

    International Journal of Information Management. 2019; 46:304–319. https://doi.org/10.1016/j.ijinfomgt.2018.08.015

16. Melian DM, Dumitrache A, Stancu S, Nastu A. Customer Churn Prediction in Telecommunication Indus- try. A Data Analysis Techniques Approach. Postmodern Openings. 2022; 13(1Sup1):78–104. https://doi.org/10.18662/po/13.1Sup1/415

17. Andreea DUMITRACHE AAMN, STANCU S. Churn Prediction in Telecommunication Industry: Model Interpretability. Journal of Eastern Europe Research in Business and Economics. 2020; 2020 (2020).

18. I.-K. Yeo and R. A. Johnson, A new family of power transformations to improve normality or symmetry, Biometrika, vol. 87, no. 4, pp. 954-959, 2000.