International Journal of Scientific Research in Science and Technology
Available online at : www.ijsrst.com
OPEN ACCESS
Print ISSN: 2395-6011 | Online ISSN: 2395-602X
doi : https://doi.org/10.32628/IJSRST

Enhancing Customer Churn Analysis in the Banking Sector through Machine Learning Algorithms and Hyper Parameter Tuning

Dharani Pitchala[1], Dr. D. Vivekananda Reddy[2]

[1]M. Tech Student, Department of Computer Science and Engineering, S.V. University College of Engineering, Tirupati, Andhra Pradesh, India

[2]Associate Professor, Department of Computer Science and Engineering, S.V. University College of Engineering, Tirupati, Andhra Pradesh, India

ARTICLEINFO

Article History:

Accepted: 01 Nov 2023
Published: 30 Nov 2023

Publication Issue
Volume 10, Issue 6
November-December-2023
Page Number
172-182

ABSTRACT

In the dynamic landscape of the banking sector, understanding and mitigating customer churn is crucial for maintaining a stable and profitable customer base. This study explores the application of machine learning algorithms, namely Random Forest, Decision Tree, Gradient Boosting, and Artificial Neural Networks (ANN), for predicting and analyzing customer churn in the banking industry. The research focuses on evaluating the performance of these algorithms both before and after hyperparameter tuning, aiming to enhance predictive accuracy and model robustness. The initial phase involves preprocessing and feature engineering to optimize the input data, ensuring the algorithms receive relevant and meaningful information. Subsequently, four distinct machine learning algorithms are employed to build predictive models. The Random Forest algorithm excels in ensemble learning, Decision Tree offers interpretability, Gradient Boosting provides boosting techniques, and ANN leverages neural networks for complex pattern recognition. To improve the models' performance, hyperparameter tuning is conducted, involving an exhaustive search for optimal hyperparameter configurations. This process aims to fine-tune the algorithms and maximize their predictive power. Comparative analyses are then performed to measure the effectiveness of the algorithms before and after hyperparameter tuning, using metrics such as accuracy, precision, recall, and F1-score. The results demonstrate the impact of hyperparameter tuning on model performance, highlighting improvements in prediction accuracy and robustness. The findings contribute valuable insights to banking institutions seeking to implement effective customer churn prevention strategies. The study's methodology and results serve as a guide for leveraging machine learning algorithms and hyperparameter tuning in other industries facing similar customer retention challenges.

Keywords : Customer Churn, Banking Sector, Machine Learning, Random Forest, Decision Tree, Gradient Boosting, Artificial Neural Networks, Hyper parameter Tuning, Predictive Analytics

I. INTRODUCTION

The banking sector operates in a highly competitive environment where retaining customers is paramount for sustained success and growth. Customer churn, the phenomenon of customers disengaging from a financial institution, poses a significant challenge for banks. Identifying potential churners and implementing proactive strategies to retain them are critical for maintaining a stable and profitable customer base. Traditional methods of churn analysis often fall short in capturing the intricate patterns and dynamic factors that contribute to customer attrition. In recent years, machine learning algorithms have emerged as powerful tools for predicting and understanding customer behavior. This study focuses on the application of advanced machine learning techniques—specifically Random Forest, Decision Tree, Gradient Boosting, and Artificial Neural Networks (ANN)—in analyzing customer churn within the banking sector.

MACHINE LEARNING ALGORITHMS

The chosen algorithms bring diverse strengths to the analysis. Random Forest excels in ensemble learning, aggregating the predictions of multiple decision trees to enhance accuracy and reduce overfitting. Decision Trees provide transparency and interpretability, allowing for a clear understanding of the decision-making process. Gradient Boosting, through iterative training, refines the model's predictions, boosting overall performance. Finally, Artificial Neural Networks leverage the complexity of neural structures to identify intricate patterns in large datasets.

HYPER PARAMETER TUNING

While these algorithms exhibit promising capabilities, their performance can be further enhanced through hyper parameter tuning. Hyper parameter tuning involves optimizing the settings that govern the learning process of the algorithms. This study aims to investigate the impact of hyperparameter tuning on the predictive accuracy and generalization capability of the chosen machine learning models.

OBJECTIVES

1. Evaluate the effectiveness of Random Forest, Decision Tree, Gradient Boosting, and ANN in predicting customer churn in the banking sector.
2. Investigate the impact of hyperparameter tuning on the performance of these machine learning algorithms.
3. Provide insights and recommendations for banking institutions to implement effective customer retention strategies based on the analysis.

SIGNIFICANCE

Understanding customer churn and implementing preemptive measures is crucial for banks to sustain growth and competitiveness. The integration of machine learning algorithms, coupled with hyperparameter tuning, offers an advanced analytical approach to address this challenge. The findings of this study aim to contribute valuable insights for banking professionals, data scientists, and researchers seeking to optimize customer churn analysis using state-of-the-art machine learning techniques.

The organizational framework of this study divides the research work in the different sections. The literature review is presented in section 2. Further, in section 3, Existing System discussed. Moreover, in next section IV, briefly explain about Data and data set and data processing and the description of machine learning and different learning mechanisms are mentioned. In section V, Experimental work is shown with diverse learning models. Conclusion and future work are presented by last sections VI.

II. LITERATURE SURVEY

Clement et al. [2] have presented new features categorized as contract-related, call pattern description, and call pattern changes description features derived from traffic figures and customer profile data. The given features were evaluated using Naïve Bayes and Bayesian network and obtained results were compared to results obtained using decision tree. Results have shown that probabilistic classifiers have shown higher true positive rate than decision tree but decision tree performs better in overall accuracy.

Essam et al. in [11] have introduced a simple model based on data mining to track customers and their behavior against churn. A dataset of 500 instances with 23 attributes has been used to test and train the model using 3 different techniques i.e., Decision trees, SVM and Neural networks for classification and k-means for clustering. Results indicate that SVM has been stated as the best suited method for predicting churn in telecom.

Umman Tuğba Şimşek Gürsoy [12] have compared regression techniques with decision tree based techniques. Results have shown that in logistic regression analysis churn prediction accuracy is 66% while in case of decision trees the accuracy measured is 71.76%. Hence decision tree based techniques are better to predict customer churn in telecom.

V. Umayaparvathi and K. Lyakutti [13] have used Neural Networks and Decision trees to build the churn prediction model. According to the results, Decision trees have 98.88% of predictive accuracy and an error rate of 1.11167%. Similarly neural networks have shown the predictive accuracy of 98.43% with 1.5616% of error rate. As is indicated by the results, decision trees have outperformed neural networks for churn prediction. According to the authors, selection of right combination of attributes and fixing the proper threshold values may produce more accurate results.

Saad et al. [14] have applied different machine learning algorithms such as linear and logistic regression, ANN (Artificial Neural Networks), K-means clustering, Decision Trees to identify churners and active customers. The best results were obtained using exhaustive CHAID, a variant of standard decision trees.

Ning Lu [15] has proposed a model with an “Implementation zone” where customers with highest churn probability can be addressed for retentive actions. The author has also proposed a further improvement in performance by analyzing other classification techniques as well or using a hybrid approach for more accurate results.

Vladislav and Marius in [17] have presented quality measures of six churn prediction models including regression analysis, naïve Bayes, decision trees, neural networks etc. They have also pointed out the links between churn prediction and customer lifetime value. According to the authors, new prediction models need to be developed and combination of proposed techniques can also be used.

Khalida et al. in [20] have used a specific training sample set was used to conduct an experiment on customer churn factor using decision tree. According to the authors, rule information can be easily understood by decision tree. An attempt has been made to identify various factors responsible for customer churn such as area.

Amal et al.[21] have reviewed that generally Decision tree based techniques, neural network trees and Regression techniques are applied in churn prediction. Decision tree based techniques outperform all other in terms of accuracy. On the other hand, neural networks outdo other techniques due to size of data sets. From the presented literature work, it can be concluded that in most of the cases, Decision trees have outperformed other techniques for predicting churn in telecommunication industry.

III. EXISTING SYSTEM

Existing System Customer churn prediction has been performed using various techniques, including data mining, machine learning, and hybrid technologies. These techniques enable and support companies in identifying, predicting, and retaining churn customers. They also help industries in CRM and decision making. Most of them used decision trees in common as it is one of the recognized methods to find out the customer churn, but it is not appropriate for complex problems [1].

But the study shows that reducing the data improves the accuracy of the decision tree [2].In some cases, data mining algorithms are used for customer prediction and historical analysis. The techniques of regression trees were discussed with other commonly used data mining methods like decision trees, rule-based learning, and neural networks [3].

IV. PROPOSED METHOD

In this system, we use various algorithms like Random Forest, and Decision Tree to find accurate values and which helps us to predict the churn of the customer. Here we implement the model by having a dataset that is trained and tested, which makes us have maximum correct values. Fig.1 shows the proposed model for churn prediction and describes its steps. In the Initial step, data pre-processing is performed in which we do filtering data and convert data into a similar form, and then we make feature selection. In the further step prediction and classification is done using the algorithms like Random Forest, Decision Tree. Training and testing the model with the data set, we observe the behavior of the customer and analyze them. In the final step, we do analysis based on the results obtained and predict the customer churn.

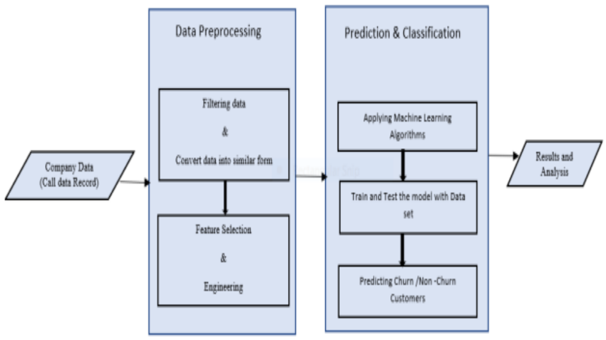

Figure 1: Proposed Model for Customer Churn Prediction

A. DATA SET

The dataset used in this research work is BigML churn in Telecom's Dataset from UCI Machine Learning Repository [3]. The dataset is extremely large and contains detailed information of all the parameters which are extremely important for predictive churn analysis. The rich set of attributes presented by the dataset helped in identifying customer churn more effectively. It consists of both churned and not-churned customer types. The tool used for dataset visualisation is Tableau Public Software. The data is input in CSV format and then visualised using various visual elements like charts, graphs, and maps, facilitating in understanding trends, outliers, and patterns in data

B. DATA PREPROCESSING

Step 1: Importing the required libraries the entire research is performed on Python 3.7.3 programming language. Python has earned the title of one of the most popular languages for machine learning tasks due to its vast collection of libraries. The machine learning libraries used in this research are:

- NumPy
- Matplotlib
- Pandas
- SkLearn

Step 2: Importing the Dataset The dataset is present in CSV format consisting of tabular data stored in plain text. The read_csv() method of pandas library is used to create a data frame of the given dataset.

Step 3: Handling Missing Values The dataset chosen for this research had been prepared extremely carefully and did not contain any missing values.

Step 4: Encoding Categorical Data Machine learning algorithms only deal with numerical values and not label values. Hence, the following attribute values are label encoded and converted to Boolean values:

- International Plan
- Voicemail Plan
- Churn

Step 5: Splitting the dataset into train and test set The telecom dataset is partitioned into two subsets, one for training the ML models called training set and the other for evaluating the performance of the trained model called test set.

Step 6: Feature Scaling Since most ML algorithms use Euclidian Distance between two data points in their computations, features possessing high variance in magnitudes lead to inconsistency in calculations. Hence, all the features are scaled using z-score normalisation technique.

C. MACHINE LEARNING ALGORITHMS

In the current study, variants of decision data have been implemented.

1. Decision Tree

A decision tree is a classification scheme which generates a flow chart like structure where an internal node represents a test on an attribute, each branch represents outcome of the test and leaf node represents classes. Decision tree partitions the input space into cells where each cell belongs to one class.The decision tree is developed into two phases: building and pruning. In the building phase, data set partitioning is done till the records in a single partition contain identical values. On the other hand, in the second phase branches containing noisy data are removed .

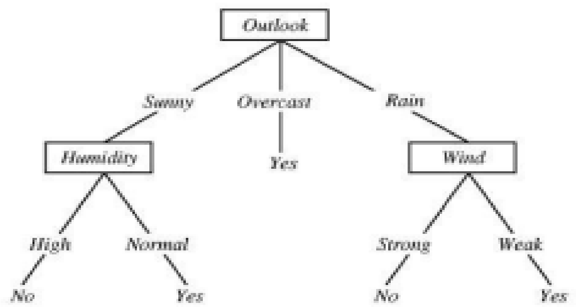

Figure 2: A Decision Tree

2. Random Forest

Random forest classifiers are a type of ensemble-based learning method, which is a big group. They are easy to set up, work quickly, and be very successful in many different areas. The key idea behind the random forest approach is that in the training stage, many "simple" decision trees are made, and in the classification stage, the majority vote (mode) across all of them is used. Among other things, this voting strategy fixes the fact that decision trees tend to overfit training data, which is not a good thing. During the training stage, random forests use a general method called "bagging" on each tree in the group. Bagging takes random samples from the training set and fits trees to these samples over and over again. Each tree grows without being cut back.

The number of trees in the ensemble is a free parameter that can be easily learned automatically using a technique called "out-of-bag error." Random forests are popular in part because, like naive Bayes and k-nearest neighbor-based algorithms, they are easy to understand and work well most of the time. But unlike the first two methods, random forests make it hard to predict how the structure of the final trained model will look. This is a natural result of the fact that building a tree is a random process.

3. ANN Classifier

Artificial Neural Networks (ANN), often referred to simply as neural networks, are a class of machine learning algorithms inspired by the structure and functioning of the human brain. They are a fundamental component of deep learning, a subset of machine learning that has gained tremendous popularity and success in various domains. Artificial Neural Networks have proven to be highly effective for a wide range of tasks, and their ability to learn from large datasets and capture complex patterns in data has contributed to their widespread use in the field of machine learning and artificial intelligence.

4. Gradient Boost Algorithm

Gradient Boosting is a powerful and widely used machine learning algorithm that belongs to the ensemble learning family. It is designed for both regression and classification tasks. The fundamental idea behind Gradient Boosting is to combine the predictions of multiple weaker models (typically decision trees) to create a stronger, more accurate model. Gradient Boosting is a versatile and effective algorithm, but it can be computationally intensive and may require careful tuning of hyperparameters. When used appropriately, it is a powerful tool for a wide range of machine learning tasks, including churn prediction, as mentioned in your original question.

D. PERFORMANCE METRICS

The performance of various ML classifiers was evaluated by computing their confusion matrix, and thereby determining the values of True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) are calculated.

The various performance parameters used were:

1. *Accuracy:* The classification accuracy of an algorithm is directly proportional to the number of correctly classified samples (true positives and true negatives).

$$Accuracy = \frac{TP+TN}{N} \quad \ldots\ldots\ldots (1)$$

Where

$$N = TP + TN + FP + FN$$

Here “TN” stands for True Negative, “TP” stands for True Positive, “FN” stands for False Negative and “FP” stands for False Positive. TP Rate is also known as sensitivity. It tells us what portion of the data is correctly classified as positive.

2. *Precision:* Precision measures the ratio of correctly classified positive samples to all the classified positive samples.

$$P = \frac{TP}{(TP+FP)} \ldots\ldots\ldots\ldots (2)$$

3. *Recall:* The recall is another measure for completeness i.e. the true hit of the algorithm. It is the probability that all the relevant instances are selected by the system. The low value of recall means many false negatives. It is calculated by using

$$Recall = \frac{TP}{(TP+FN)} \ldots\ldots\ldots\ldots (3)$$

4. *F1-Score:* It is defined as the harmonic mean (average) of precision and specificity/recall value.

$$F1_{score} = \frac{2*P*Specificity}{P+Specificity} \ldots\ldots\ldots (4)$$

V. RESULTS AND DISCUSSIONS

A confusion matrix is a powerful tool used in machine learning to evaluate the performance of a classification model. It provides a clear representation of the model's predictions in terms of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). Analyzing the confusion matrix before and after hyperparameter tuning offers insights into how the model's predictive capabilities have improved. Let's break down the confusion matrix components:

A. BEFORE HYPER TUNNING

• *True Positives (TP):* The number of instances where the model correctly predicted customer churn.

• *True Negatives (TN):* The number of instances where the model correctly predicted non-churn.

• *False Positives (FP):* The number of instances where the model predicted churn, but the customer did not actually churn (Type I error).

• *False Negatives (FN):* The number of instances where the model predicted non-churn, but the customer actually churned (Type II error).

The confusion matrix before hyperparameter tuning provides a baseline understanding of the model's strengths and weaknesses shown in fig.3 to fig.6. It helps identify areas for improvement, such as reducing false positives or false negatives, to enhance the overall predictive performance.

Decision Tree

```
dt_model=DecisionTreeClassifier( )
dt_model.fit(X_train,y_train)
```

```
▾ DecisionTreeClassifier
DecisionTreeClassifier()
```

```
evaluate_model_performance(dt_model,X_test)
```

```
Validation Accurary : 0.78 %
Precision Score : 0.46 %
Recall Score : 0.46 %
F1 Score : 0.46 %
              precision    recall  f1-score   support

           0       0.86      0.86      0.86      1582
           1       0.46      0.46      0.46       416

    accuracy                           0.78      1998
   macro avg       0.66      0.66      0.66      1998
weighted avg       0.78      0.78      0.78      1998
```

Figure 3: Confusion matrix of Decision Tree before hyper tunning

Random Forest Classifier

```
[53]: rf = RandomForestClassifier()
      rf.fit(X_train,y_train)
```

```
[53]: ▾ RandomForestClassifier
      RandomForestClassifier()
```

```
[54]: evaluate_model_performance(rf,X_test)
```

```
Validation Accurary : 0.84 %
Precision Score : 0.68 %
Recall Score : 0.41 %
F1 Score : 0.51 %
              precision    recall  f1-score   support

           0       0.86      0.95      0.90      1582
           1       0.68      0.41      0.51       416

    accuracy                           0.84      1998
   macro avg       0.77      0.68      0.71      1998
weighted avg       0.82      0.84      0.82      1998
```

Figure 4 : Confusion matrix of Random Forest before hyper tunning

Gradient Boosting Classifier

```
gbc = GradientBoostingClassifier()
gbc.fit(X_train,y_train)
```

```
▾ GradientBoostingClassifier
GradientBoostingClassifier()
```

```
evaluate_model_performance(gbc,X_test)
```

```
Validation Accurary : 0.84 %
Precision Score : 0.71 %
Recall Score : 0.40 %
F1 Score : 0.51 %
              precision    recall  f1-score   support

           0       0.86      0.96      0.91      1582
           1       0.71      0.40      0.51       416

    accuracy                           0.84      1998
   macro avg       0.79      0.68      0.71      1998
weighted avg       0.83      0.84      0.82      1998
```

Figure 5: Confusion matrix of Gradient Boost hyper tunning

Build a model (ANN) in tensorflow/keras

```
model = keras.Sequential([
    keras.layers.Dense(26, input_shape=(9,), activation='relu'),
    keras.layers.Dense(15, activation='relu'),
    keras.layers.Dense(1, activation='sigmoid')
])

# opt = keras.optimizers.Adam(learning_rate=0.01)

model.compile(optimizer='adam',
              loss='binary_crossentropy',
              metrics=['accuracy'])

model.fit(X_train, y_train, epochs=100)
```

•••

```
model.evaluate(X_test, y_test)
```

```
63/63 [==============================] - 0s 1ms/step - loss: 0.4151 - accuracy: 0.8258
[0.4151197373867035, 0.8258258104324341]
```

Figure 6 : Confusion matrix of ANN Before hyper tunning

B. AFTER HYPER TUNNING

We performed several experiments on the proposed churn model using machine learning algorithms on the dataset after hyper parameter tunning. In Fig.7 and Fig.10, we can observe the results obtained after hyper parameter tunning . Comparing the confusion matrices before and after hyperparameter tuning allows for a more detailed assessment of model refinement. A successful tuning process should ideally result in a reduction of false positives and false negatives, leading to an improved balance between precision and recall.

```
#evaluate final DecisionTreeClassifier Performance
evaluate_model_performances(dt_final_model,x_test)

Validation Accurary : 0.83 %
Precision Score : 0.82 %
Recall Score : 0.85 %
F1 Score : 0.84 %
              precision    recall  f1-score   support

           0       0.83      0.79      0.81       614
           1       0.82      0.85      0.84       696

    accuracy                           0.83      1310
   macro avg       0.83      0.82      0.82      1310
weighted avg       0.83      0.83      0.82      1310
```

Figure 7: Confusion matrix of Decision Tree after hyper tunning

```
#evaluate Random Forest Classifier
evaluate_model_performances(rf_final_model,x_test)

Validation Accurary : 0.87 %
Precision Score : 0.85 %
Recall Score : 0.91 %
F1 Score : 0.88 %
              precision    recall  f1-score   support

           0       0.89      0.82      0.86       614
           1       0.85      0.91      0.88       696

    accuracy                           0.87      1310
   macro avg       0.87      0.87      0.87      1310
weighted avg       0.87      0.87      0.87      1310
```

Figure 8 : Confusion matrix of Random Forest after hyper tunning

```
#evaluate final GradientBoostingClassifier Performance
evaluate_model_performances(gb_final_model,x_test)

Validation Accurary : 0.85 %
Precision Score : 0.85 %
Recall Score : 0.88 %
F1 Score : 0.86 %
              precision    recall  f1-score   support

           0       0.86      0.82      0.84       614
           1       0.85      0.88      0.86       696

    accuracy                           0.85      1310
   macro avg       0.85      0.85      0.85      1310
weighted avg       0.85      0.85      0.85      1310
```

Figure 9: Confusion matrix of GBC Classifier after hyper tunning

```
[99]: from sklearn.metrics import confusion_matrix , classification_report

      print(classification_report(Y_test,y_pred))
              precision    recall  f1-score   support

           0       0.84      0.78      0.81       614
           1       0.82      0.87      0.84       696

    accuracy                           0.83      1310
   macro avg       0.83      0.82      0.83      1310
weighted avg       0.83      0.83      0.83      1310
```

Figure 10: Confusion matrix of ANN Classifier after hyper tunning

TABLE I: PERFORMANCE COMPARISON OF DIFFERENT MACHINE LEARNING ALGORITHMS (BEFORE HYPER TUNING)

ML Algorithms	**Accuracy (%)**	**Precession (%)**	**Recall (%)**	**F1Score (%)**
Decision Tree	0.78	0.46	0.46	0.46
Random Forest	0.84	0.68	0.41	0.51
ANN	0.82	0.61	0.44	0.51
Gradient Boost	0.84	0.71	0.40	0.51

Before hyper parameter tuning, all models, Random Forest and Gradient Boosting achieved the highest accuracy among the models. There is a trade-off between precision and recall in these models, emphasizing the need for a balanced approach. Decision Tree and ANN exhibit potential areas for improvement in both precision and recall. After hyper parameter tuning, there was a significant improvement in all metrics for all algorithms.

TABLE II: PERFORMANCE COMPARISON OF DIFFERENT MACHINE LEARNING ALGORITHMS (AFTER HYPER TUNING)

ML Algorithms	Accuracy (%)	Precession (%)	Recall (%)	F1Score (%)
Decision Tree	0.83	0.82	0.85	0.84
Random Forest	0.87	0.85	0.91	0.88
ANN	0.84	0.82	0.87	0.84
Gradient Boost	0.85	0.85	0.88	0.86

Hyper parameter tuning led to significant improvements in the performance All models exhibit improvements in accuracy, precision, recall, and F1-Score after hyperparameter tuning. Random Forest and ANN continue to offer competitive performance, with well-balanced precision and recall. Decision Tree and Gradient Boosting show significant enhancements, addressing the trade-offs observed in the initial analysis.This post-hyperparameter tuning analysis demonstrates the effectiveness of optimization in enhancing the predictive capabilities of machine learning models for customer churn analysis in the banking sector. The refined models are better equipped to support decision-making processes and strategic initiatives for customer retention.

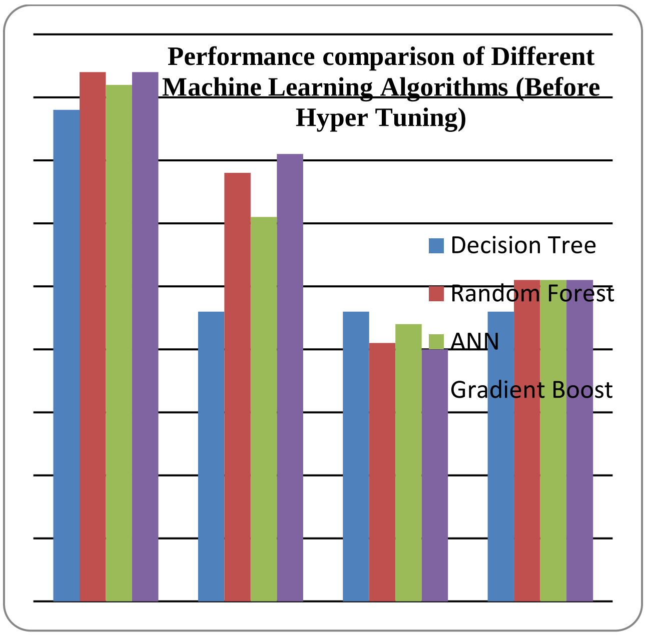

Figure 11: Comparison Graph of different Machine Learning Algorithm (before Hyper Tuning)

The initial performance of all algorithms was relatively poor. Random Forest and Gradient Boost outperform the other algorithms in terms of accuracy, but their precision and recall values vary. Decision Tree and ANN exhibit balanced recall and precision but with a lower overall accuracy compared to Random Forest and Gradient Boost. The F1-Score is consistent across Random Forest, ANN, and Gradient Boost, indicating a reasonable balance between precision and recall for these models.

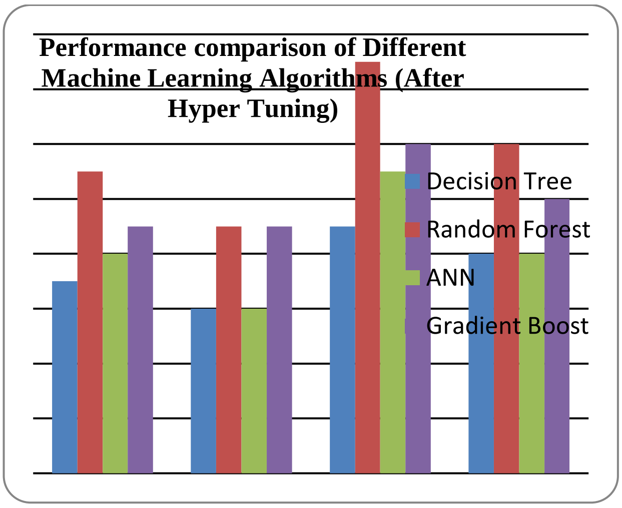

Figure 12: Comparison Graph of different Machine Learning Algorithm (After Hyper Tuning)

Hyper parameter tuning significantly improved the performance of all the algorithms. All models show significant improvements in accuracy, precision, recall, and F1-Score after hyper parameter tuning. Random Forest maintains high performance across all metrics, achieving the highest recall among the models. Decision Tree and ANN models, while improved, still exhibit a slightly lower recall compared to Random Forest and Gradient Boost.

VI. CONCLUSION AND FUTURE SCOPE

The integration of machine learning algorithms for Customer Churn Analysis in the banking sector presents a promising avenue for customer retention strategies. Ongoing advancements in model development, interpretability, and real-time capabilities will continue to refine and optimize the effectiveness of these approaches in addressing the dynamic challenges of customer churn in the banking industry.

FUTURE SCOPE

Explore advanced feature engineering techniques to extract more meaningful information from the dataset, potentially improving the models' predictive performance.

VII. REFERENCES

[1]. Amal M. Almana, Mehmet Sabih Aksoy, Rasheed Alzaharni, “A Survey on Data Mining Techniques in Customer Churn Analysis for Telecom Industry”, International Journal of Engineering Research and Applications, Vol. 4, Issue 5, ISSN: 2248-9622, May 2014, Pp. 165-171.

[2]. Clement Kirui, Li Hong, Wilson Cheruiyot, Hillary Kirui, “Predicting Customer Churn in Mobile Telephony Industry Using Probabilistic Classifiers in Data Mining”, International Journal of Computer Science Issues, Vol. 10, Issue 2, No 1, ISSN: 1694- 0814, March 2013, Pp. 165-172.

[3]. Wai-Ho Au, Keith C. C. Chan, Xin Yao, “A Novel Evolutionary Data Mining Algorithm With Applications to Churn Prediction”, IEEE Transactions on Evolutionary Computation, IEEE, Vol. 7, No. 6, December 2013, Pp. 532-545.

[4]. Erfaneh Gharavi, Mohammad. Jafar Tarokh, “Predicting Customers’ Future Demand using Data Mining Analysis: A Case Study of Wireless Communication Customer”, 5th Conference on Information and Knowledge Technology, IEEE, 2013, Pp. 338- 343.

[5]. Wei Yu, Dawn N. Jutla, Shyamala C. Sivakumar, “A Churn Strategy Alignment Model for Managers in Mobile Telecom”, Proceeding of the 3rd Annual Communication Networks and Services Research Conference, IEEE, 2005.

[6]. Chao Zhu, Jiayin Qi, Chen Wang, “An Experimental Study on four Models of Customer Churn Prediction”, Proceedings of the 2009 IEEE International Conference on Systems, Man and Cybernetics, USA, October 2009, Pp. 3199-3204.

[7]. Adnan Idris, Asifullah Khan, “Customer Churn Prediction for Tele communication:Employing various features selectiontechniques and tree based ensemble classifiers”, IEEE, 2012.

[8]. Navid Forhad, Md. Shahriar Hussain, Rashedur M Rahman, “Churn Analysis: Predicting Churners”, IEEE, 2014, Pp. 237-241.

[9]. N.Kamalraj, Dr. A.Malathi, “Applying Data Mining Techniques in Telecom Churn Prediction”, International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 3, Issue 10, ISSN: 2277 128X, October 2013, Pp. 363-370.

[10]. Javed Basiri, Fattaneh Taghiyareh, Behzad Moshiri, “A Hybrid Approach to Predict

Churn", IEEE Asia-Pacific Services Computing Conference, 2010, Pp. 485-491.

[11]. Essam Shaaaban, Yehia Helmy, Ayman Khedr, Mona Nasr, "A Proposed Churn Prediction Model", International Journal of Engineering Research and Applications, Vol. 2, Issue 4, ISSN: 2248-9622, June-July 2012, Pp. 693-697.

[12]. Umman Tuğba Şimşek Gürsoy, "Customer Churn Analysis in Telecommunication Sector", Istanbul University Journal of the School of Business Administration, Vol. 39, No. 1, ISSN: 1303- 1732, 2010, Pp. 35-49.

[13]. V. Umayaparvathi, K. Lyakutti, "Applications of Data Mining in Telecom churn Prediction", International Journal of Computer Applications, Vol. 42, No. 20, ISSN: 0975-8887, March 2012, Pp. 5-9.

[14]. Saad Ahmed Qureshi, Ammar Saleem Rehman, Ali Mustafa Qamar, Aatif Kamal, "Telecommunication Subscribers' Churn Prediction Model Using Machine Learning", IEEE, 2013, Pp. 131-136.

[15]. Ning Lu, Hua Lin, Jie Lu, "A Customer Churn Prediction Model in Telecom Industry Using Boosting", IEEE Transactions on Industrial Informatics, Vol. 10, No. 2, May 2014, Pp. 1659-1665.

[16]. Nikita Jain, Vishal Srivastav, "Data Mining Techniques: A Survey Paper", International Journal of Research in Engineering and Technology, Vol. 2, Issue 11, ISSN: 2321-7308, Nov 2013, Pp. 116-119.

[17]. Vladislav Lazarov, Marius Capota, "Churn Prediction", Technische Universität München.

[18]. http://10.35.2.6:50624/help/index.jsp?topic=%2Fcom.ibm.spss.sta tistics.algorithms%2Falg_tree-chaid.htm

[19]. Rahman Mansouri, Mohamad Saraee, RasoulAmirfattahi, "Applications of Data Mining in Predicting Cell Phones Subscribers Behavior Employing the Contact Pattern", International Conference on Data Storage and Data Engineering, IEEE 2010.

[20]. Khalida, Sunarti, Norazrina, Faizin, "Data Mining in Churn Analysis Model for Telecommunication Industry" Journal of Statistical Modeling and Analytics, Vol. 1, No.19-27,ISSN: 2180- 3102, 2010, Pp. 19-27.

[21]. Amal M. Almana, Mehmet Sabih Aksoy, Rasheed Alzaharni, "A Survey on Data Mining Techniques in Customer Churn Analysis for Telecom Industry", International Journal of Engineering Research and Applications, Vol. 4, Issue 5, ISSN: 2248-9622, May 2014, Pp. 165-171.

Cite this article as :

Dharani Pitchala, Dr. D. Vivekananda Reddy, "Enhancing Customer Churn Analysis in the Banking Sector through Machine Learning Algorithms and Hyper Parameter Tuning", International Journal of Scientific Research in Science and Technology (IJSRST), Online ISSN : 2395-602X, Print ISSN : 2395-6011, Volume 10 Issue 6, pp. 172-182, November-December 2023.
Journal URL : https://ijsrst.com/IJSRST52310625