CUSTOMER CHURN PREDICTION USING MACHINE LEARNING

Jaswanth Sirigiri
Dept of Computer Science
Texas A&M University-Corpus Christi
jsirigiri@islander.tamucc.edu

Koushik Reddy Kambham
Dept of Computer Science
Texas A&M University-Corpus Christi
kkambham@islander.tamucc.edu

Raghuvamsi Mallampalli
Dept of Computer Science
Texas A&M University-Corpus Christi
rmallampalli@islander.tamucc.edu

Naresh Vemula
Dept of Computer Science
Texas A&M University-Corpus Christi
nvemula@islander.tamucc.edu

Abstract

Customer churn is one of the principal issues in the telecommunications Industry. Clients massively change their specialist co-ops within the limited ability to focus time. Client Churn implies the loss of the entire or part of the administration from the client by any association. Decision makers and business analysts emphasized that attaining new customers is costlier than retaining the existing ones. Due to increasing demand, customers try to switch the network connections frequently. Business analysts and customer relationship management (CRM) analyzers need to know the reasons for churn customers, as well as behavior patterns from the existing churn customers' data. In this, we will discuss the fundamental issue What makes a client remain, and what influences them to go? We have utilized the telecommunications market to break down the stirring issue and have taken the IBM Watson Analysis Dataset for our case study. In this, we have used a Random Forest Classifier (RF), Support Vector Machine (SVM), Decision Tree (DT), and K-Nearest Neighbours Classifier (KNN) which are Machine Learning techniques that contribute highly to the advancement of churn prediction systems providing better performance and cost-effective solutions by evaluation of various metrics like accuracy, precision, recall etc. This project mainly helps to minimize the churn of the customers in the Telecommunications sector. By knowing the significant churn factors from customers' data, Customer Relationship Management (CRM) can improve productivity, recommend relevant promotions to the group of likely churn customers based on similar behaviour patterns, and excessively improve the marketing campaigns of the company.

Keywords-*Churn prediction, Decision Tree, K-Nearest Neighbours, Random Forest, Support Vector Machine.*

I. Introduction

Customer Churn is an important factor in any business firm more so in the Telecommunications sector as customers in this sector frequently change services, hence we can see a lot of churning out of customers which will harm the companies' business. At the point when the business is in a development period of its life cycle, deals are expanding exponentially, and the number of new clients largely dwarfs the number of churners. On the opposite side, organizations in a developed period of their life cycle set their emphasis on decreasing the rate of customer churn.

The fundamental reasons for customer churn are divided into two groups: accidental and intentional. Accidental churn happens when the conditions are changing so keeps the clients from utilizing the services later, for instance, financial conditions that make benefits unreasonably costly for the client. Intentional churn happens when the clients change to another organization that gives comparable services, like better ideas from rivalry, further developed services, and better cost for a similar service. To deal with this problem, telecom operators must recognize these customers before they churn. In this project, we aimed to investigate the main reasons for churn among customers using Telcom customer data. For this purpose, we gathered and processed the data, and based on these data, we implemented and compared four well-known machine learning algorithms that are RF, SVM, DT, and KNN.

Some of the most important factors that are crucial for the customers to churn are tariff plan, contract, duration (length) of the contract, number of services, number of outgoing calls, etc.

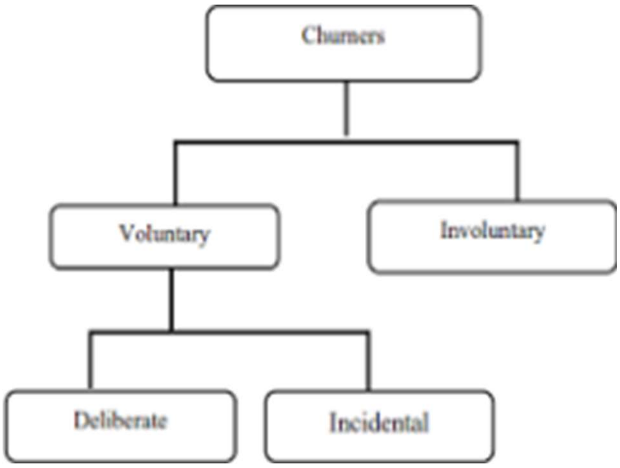


Figure 1. Customer Churn Types

Problem Definition:

The problem statement of this project is to find out the factors for churn and to find out the churning customers in the Telecommunications sector which are the reasons for several losses in the Telecom market. The firm needs to retain existing customers rather than acquire new ones. The main objective behind the problem statement is to build a churn prediction model that uses classification and clustering techniques to identify the churn customers provides the factors behind the churning of customers in the telecom sector and gives insights into the influencing factors of churn. From these above predictions, we can prescribe potential solutions to avoid churn.

Existing System:

It is very important to predict the correct customers of churn to increase the profit. In this Existing system, the LDT and UDT are used the dataset is iterated for these algorithms and takes the best of the iterated samples. After data preprocessing and sample selection, the samples are split for estimation separately by LDT (Lower Distance Zone) and UDT (Upper Distance Zone) techniques. The spilled data is also used for the training process. The results obtained from LDT, UDT, and the training process are given to the classifier. The classifier gives us a separate result based on LDT and UDT techniques. In this, the efficiency is achieved up to 60% to 70% whereas it is suitable for smaller datasets.

Problems in Existing System:

- Suitable for small datasets.
- Less accuracy and precision.
- Poor performance.

Proposed System:

The proposed system (Customer Churn Prediction using Machine Learning) uses Python and Machine Learning Algorithms in the Jupyter Notebook platform. In the existing system algorithms such as the LDT and UDT are used only for the small set of datasets so that it produces an efficiency of 75% and takes much time to train the dataset. So, to overcome this drawback we are using the algorithms called Random Forest Classifier (RF), Support Vector Machine (SVM), Decision Tree (DT), and K-Nearest Neighbors (KNN). We use these algorithms for computational processes and from that we can consider the best algorithm that produces higher efficiency.

To achieve higher efficiency, we are using the "SMOTEENN" method. SMOTE (synthetic minority oversampling technique) is one of the most used oversampling methods to solve the imbalance problem. It aims to balance class distribution by randomly increasing minority class examples by replicating them and ENN's ability to delete some observations from both classes that are identified as having different classes.

Requirements Specification:

Requirement Specifications describe the art-craft of Software Requirements and Hardware Requirements used in this project.

Software Requirements:

Operating System: Windows 10

IDE: Jupyter notebook

Language: Python 3.5 and above

Packages required: numpy, pandas, sklearn, matplotlib, seaborn, pickle.

Hardware Requirements:

RAM: 8GB or more

Hard Disk: 1TB

Processor: Any Intel/Ryzen processor

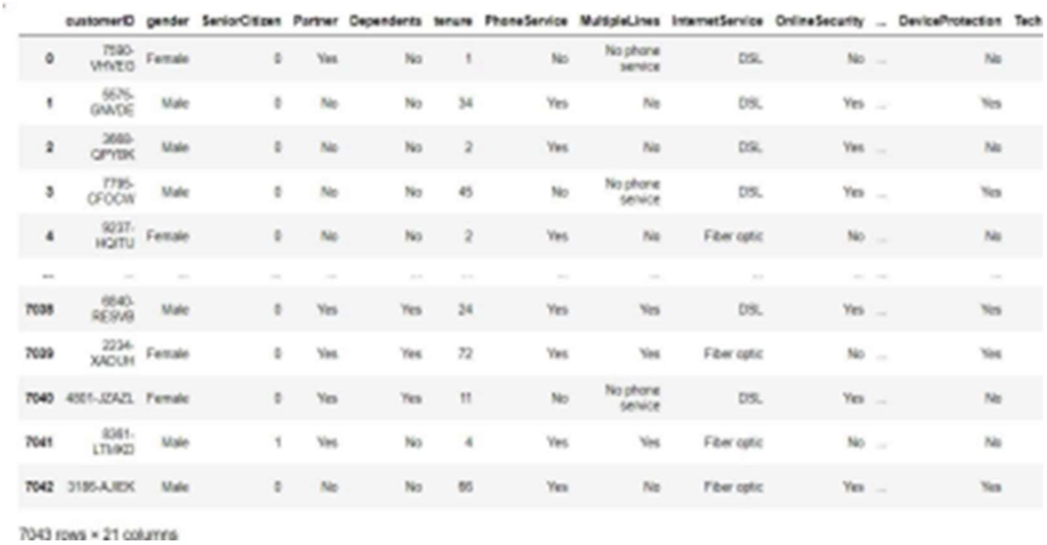
II. Related Work

1) **“Research on Customer Churn Intelligent Prediction Model based on Borderline SMOTE and Random Forest”** by Linmao Feng. This paper designs a borderline-SMOTE random forest prediction model for unbalanced data such as bank customers. Combined with the oversampling algorithm, it can better solve the unbalanced data. AUC, Precision, Recall, and F-mean are used as the evaluation indicators of the model, and KNN, decision tree, and Naive Bayes are used as comparisons. The experimental results show that the Borderline SMOTE-random forest prediction model has about 4% better performance compared with other models.
2) **“Telecommunication Customers Churn Prediction using Machine Learning”** by Nur Abdul Razak, Muhammad Hazim Wahid. This paper presents customer churn prediction based on the usage pattern using a machine learning prediction model, namely linear regression, random forest, Support Vector Machine (SVM), K-nearest neighbor (KNN), and decision tree. The prediction performance of these algorithms is evaluated in terms of accuracy, recall, precision, and F1 score. The results showed that the random model performed the best for the dataset used, compared to other models, with 95.5% accuracy.
3) **“Customer Churn Prediction In Telecommunication Industry Using Random Forest Classifier”** by V. Geetha, A.Punitha, A. Nandhini, S.Shakila, R.Sushmitha, T. Nandhini. In this paper, they propose a system with efficient algorithms known as Random Forest Classifier and Support Vector Machine which selects the important attribute that increases the performance of the system and by implementing these two algorithms we can achieve an efficiency of about 95 percent.
4) **“Customer Churn Reasoning in Telecommunication Domain”** by S. Stehani, N. Karunya, Dr. J. B. Ranjan, Sagara Sumathipala, T. C. Sandanayake. Due to increasing demand, customers try to switch the network connections frequently. This research analyses the churn customer details and predicts the probability of churn in the future using modern machine-learning techniques. It is also important to identify the reasons for the churn to take action to the retention of the customer.
5) **“Churn Prediction: A Comparative Study Using KNN and Decision Trees”** by Mohammad A. Hassonah, Ali Rodan, Abdel-Karim Al-Tamimi, Jamal Alsakran. In this paper, we are conducting a comparison study of the performance towards churn prediction between two of the most powerful machine learning algorithms which are Decision Tree and K-Nearest Neighbor algorithms.

III. Dataset and Features

The case study in the project is a telecommunications company. We have collected the data from the Kaggle website- “TELECOM CHURN DATASET (IBM WATSON ANALYTICS)”.

These tables have historical information but for this exercise, we will be considering the latest values of all the attributes. A data extract from these tables is obtained to further cleanse, process, and create a new feature from the existing attributes. There are in total 7043 records with 21 features.



	customerID	gender	SeniorCitizen	Partner	Dependents	tenure	PhoneService	MultipleLines	InternetService	OnlineSecurity	...	DeviceProtection	Tech
0	7590-VHVEG	Female	0	Yes	No	1	No	No phone service	DSL	No	...	No	
1	5575-GNVDE	Male	0	No	No	34	Yes	No	DSL	Yes	...	Yes	
2	3668-QPYBK	Male	0	No	No	2	Yes	No	DSL	Yes	...	No	
3	7795-CFOCW	Male	0	No	No	45	No	No phone service	DSL	Yes	...	Yes	
4	9237-HQITU	Female	0	No	No	2	Yes	No	Fiber optic	No	...	No	
...	...	...	...	...	...	...	...	...	...	...	...	...	
7038	6840-RESVB	Male	0	Yes	Yes	24	Yes	Yes	DSL	Yes	...	Yes	
7039	2234-XADUH	Female	0	Yes	Yes	72	Yes	Yes	Fiber optic	No	...	Yes	
7040	4801-JZAZL	Female	0	Yes	Yes	11	No	No phone service	DSL	Yes	...	No	
7041	8361-LTMKD	Male	1	Yes	No	4	Yes	Yes	Fiber optic	No	...	No	
7042	3186-AJIEK	Male	0	No	No	66	Yes	No	Fiber optic	Yes	...	Yes	

7043 rows × 21 columns

Figure 1. Screenshot of the Dataset

Data Preprocessing:

Since the churn Dataset has no processing. Data preprocessing is only applied. Data preprocessing is a data mining technique that transforms the raw data into a useful and efficient format. In the Real world, data are generally incomplete, lacking attribute values, certain attributes of interest, or containing only aggregate data. Noisy that is containing errors or outliers and Inconsis tent that is containing discrepancies in codes or names. The dataset has been cleaned to get better results and accuracy. Here, the values are replaced, and the relevance of attributes is checked i.e., some unrelated data features are not considered as they may reduce the model's accuracy. Fig 1 shows the steps of the Data Preprocessing technique for Churn Dataset.

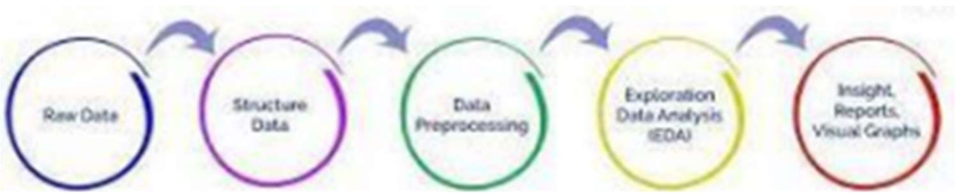

Raw Data
Structure Data
Data Preprocessing
Exploration Data Analysis (EDA)
Insight, Reports, Visual Graphs

Figure 2. Data Preparation for Customer Churn Prediction

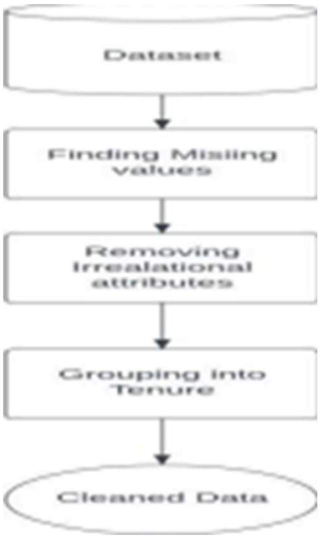


Figure 3. Data Preprocessing of Churn Dataset

System Architecture: Design is the foundation for creating any software or model. Software design is a mechanism of preparing a plan, a layout for structuring the code of a software application. The design of the project helps us to keep in check the following important points: Modularity, Maintainability, Flow of functionality and performance, and Portability and tractability.

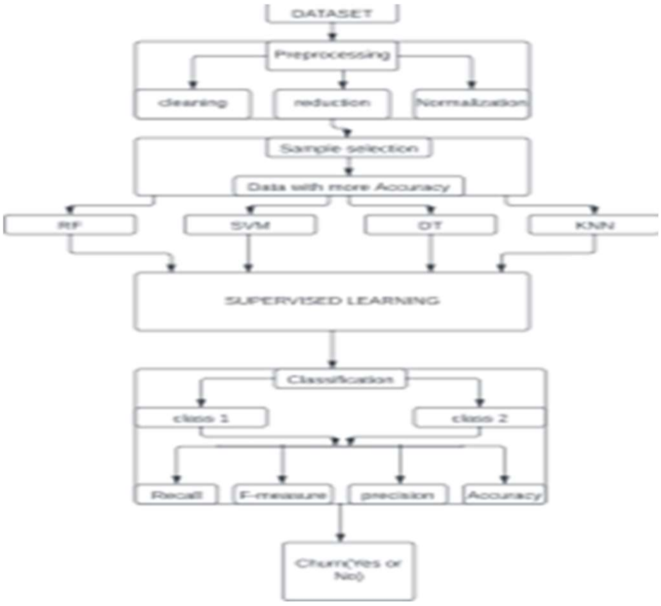


Figure 4. System Architecture of Customer Churn Prediction Model

IV. Methods

Random Forest Classifier (RF):

Random Forest is a popular machine-learning algorithm. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning**.** Random Forest is a classifier that contains several decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset. The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

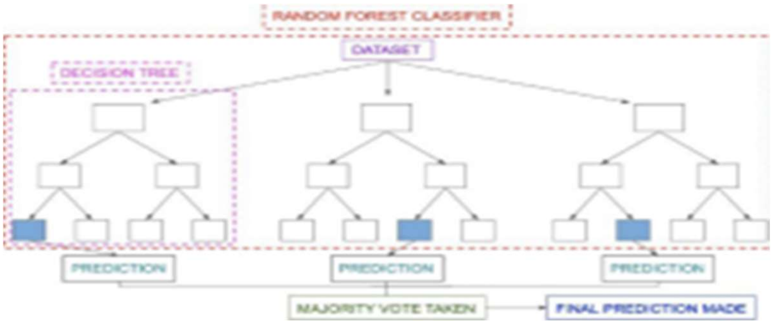


Figure 5. Random Forest Classifier

Support Vector Machine (SVM):

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms. However, primarily, it is used for Classification problems in Machine Learning. The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane. SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called support vectors, and hence algorithm is termed a Support Vector Machine.

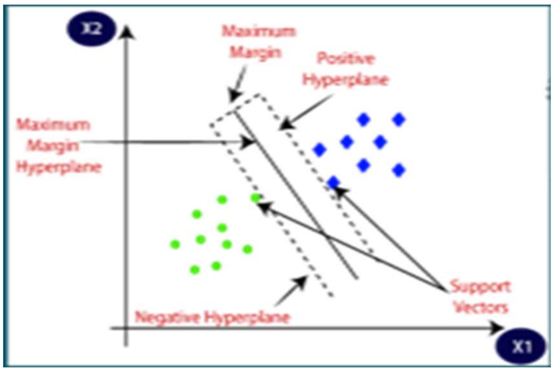


Figure 6. Support Vector Machine

Decision Tree (DT):

Decision Tree is a Supervised learning technique that can be used for both classification and Regression problems, but mostly it is preferred for solving Classification problems. It is a tree-structured classifier. In a Decision tree, there are two nodes, which are the Decision Node and the Leaf Node. Decision nodes are used to make any decision and have multiple branches, whereas Leaf nodes are the output of those decisions and do not contain any further branches. It is a graphical representation for getting all the possible solutions to a problem/decision based on given conditions.

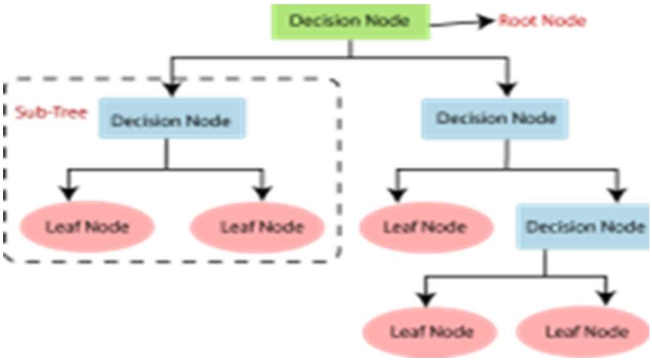


Figure 7. Decision Tree Classifier

K-Nearest Neighbors Classifier (KNN):

K-Nearest Neighbor is one of the simplest Machine Learning algorithms based on Supervised Learning technique. The k-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well-suited category by using the K-NN algorithm. K-NN is a non-parametricalgorithm, which means it does not make any assumption on underlying data. It is also called a lazy learner algorithm because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset. It is used for not just binary classification but for multilevel classifications too. It is a non-parametric machine learning method implying that it doesn'*t* make any assumptions about the data. It also doesn't make any generalizations, and simply checks the neighboring data points to determine the classification of unknown or uncategorized data points. K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for Classification problems. K-NN with SMOTE-ENN has the highest accuracy in predicting customer churn.

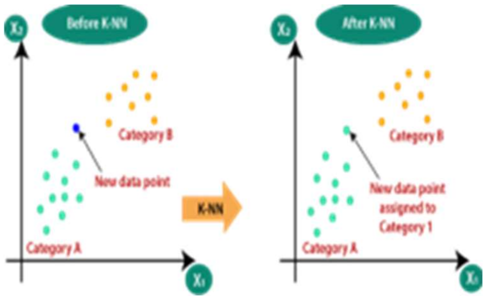


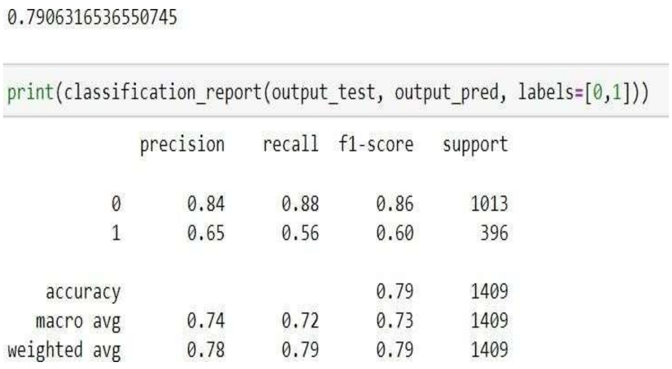
Figure 8. K-Nearest Neighbors

V. Experiments/ Results/ Discussions

The Churn Dataset is used for both training and testing for the models. It is used in an 80%-20% ratio for training and testing respectively. In the model building for Customer Churn Prediction, weare using four well-known Machining Learning Algorithms which are Random Forest Classifier, Decision Tree, Support vector Machine, and K-Nearest Neighbors.

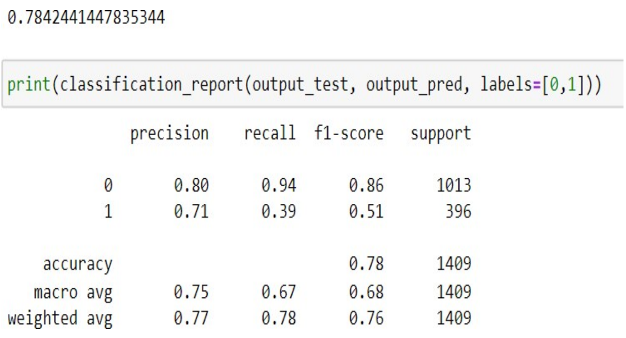
Before SMOTE-ENN:

The Decision Tree Algorithm, Random Forest Classifier, Support Vector Machine, and K-nearest neighbors for Customer Churn Prediction are indicated as mentioned below.



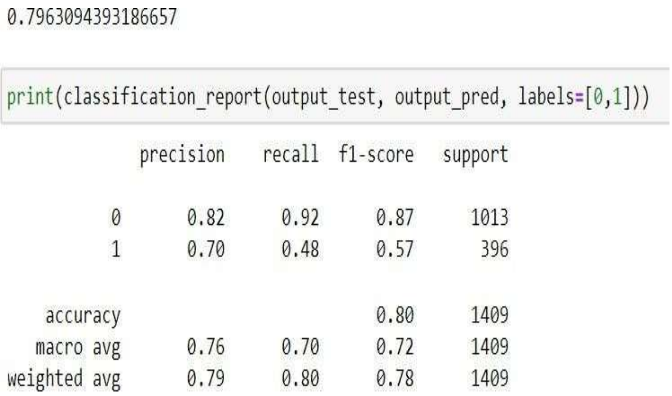
```
0.7906316536550745

print(classification_report(output_test, output_pred, labels=[0,1]))

              precision    recall  f1-score   support

           0       0.84      0.88      0.86      1013
           1       0.65      0.56      0.60       396

    accuracy                           0.79      1409
   macro avg       0.74      0.72      0.73      1409
weighted avg       0.78      0.79      0.79      1409
```

Figure 9. Decision Tree Figure



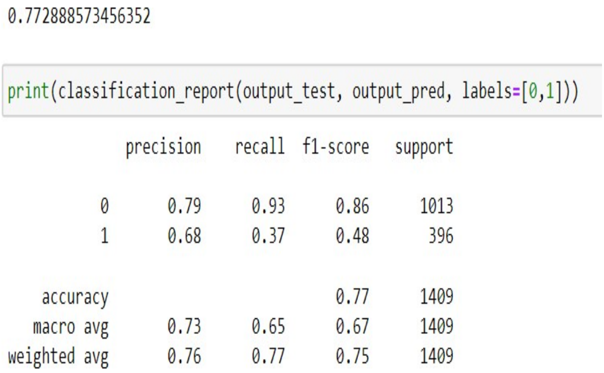
```
0.7842441447835344

print(classification_report(output_test, output_pred, labels=[0,1]))

              precision    recall  f1-score   support

           0       0.80      0.94      0.86      1013
           1       0.71      0.39      0.51       396

    accuracy                           0.78      1409
   macro avg       0.75      0.67      0.68      1409
weighted avg       0.77      0.78      0.76      1409
```

Figure: 10. Random Forest Classifier



```
0.7963094393186657

print(classification_report(output_test, output_pred, labels=[0,1]))

              precision    recall  f1-score   support

           0       0.82      0.92      0.87      1013
           1       0.70      0.48      0.57       396

    accuracy                           0.80      1409
   macro avg       0.76      0.70      0.72      1409
weighted avg       0.79      0.80      0.78      1409
```

Figure 11. Support Vector Machine

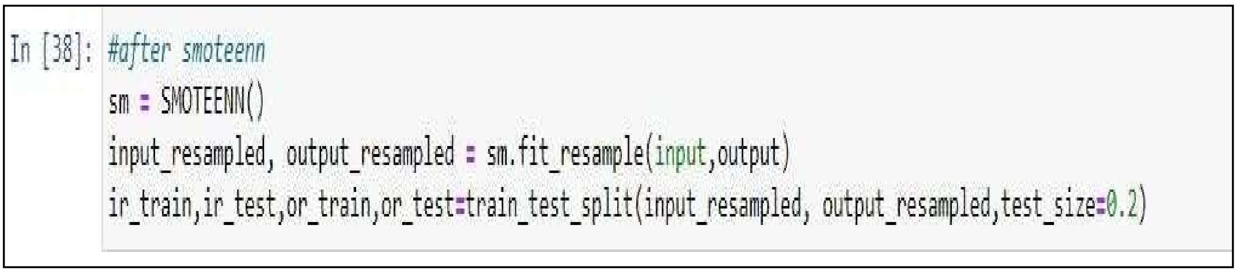


```
0.772888573456352

print(classification_report(output_test, output_pred, labels=[0,1]))

              precision    recall  f1-score   support

           0       0.79      0.93      0.86      1013
           1       0.68      0.37      0.48       396

    accuracy                           0.77      1409
   macro avg       0.73      0.65      0.67      1409
weighted avg       0.76      0.77      0.75      1409
```

Figure 12: K-Nearest Neighbors

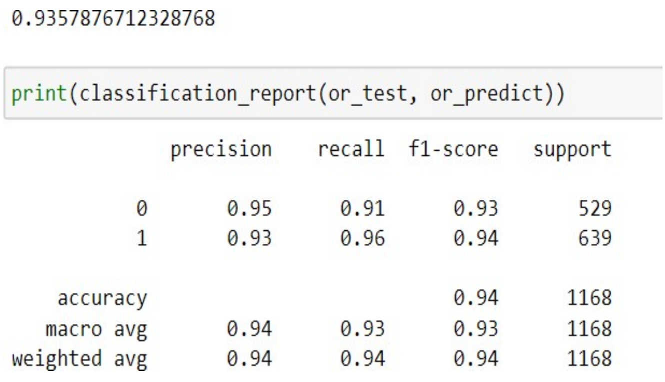
After SMOTE-ENN:

Since the dataset is imbalanced. We are trying to balance the dataset to achieve higher efficiency and accuracy metrics. We are using an oversampling technique called SMOTE-ENN.



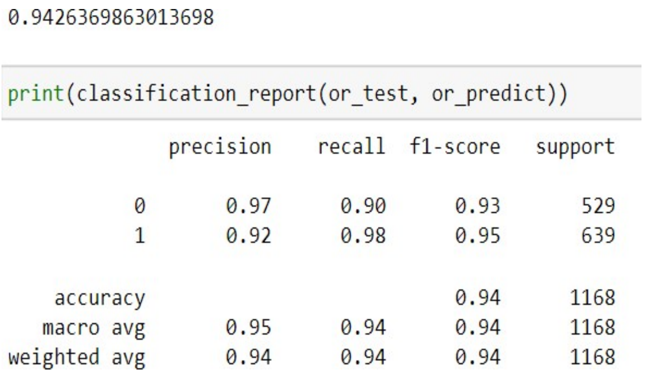
```
In [38]: #after smoteenn
         sm = SMOTEENN()
         input_resampled, output_resampled = sm.fit_resample(input,output)
         ir_train,ir_test,or_train,or_test=train_test_split(input_resampled, output_resampled,test_size=0.2)
```

After applying SMOTE-ENN, The Decision Tree Algorithm, Random Forest Classifier, Support Vector Machine, and K-Nearest Neighbours for Customer Churn Prediction are indicated as mentioned below.



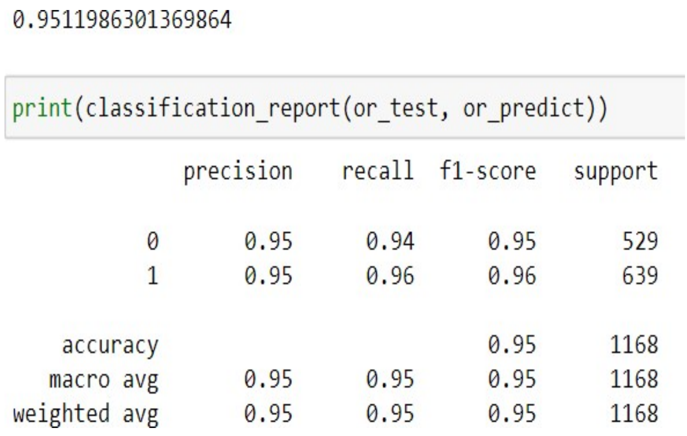
```
0.9357876712328768

print(classification_report(or_test, or_predict))

              precision    recall  f1-score   support

           0       0.95      0.91      0.93       529
           1       0.93      0.96      0.94       639

    accuracy                           0.94      1168
   macro avg       0.94      0.93      0.93      1168
weighted avg       0.94      0.94      0.94      1168
```

Figure 13: Decision Tree after Smote-enn



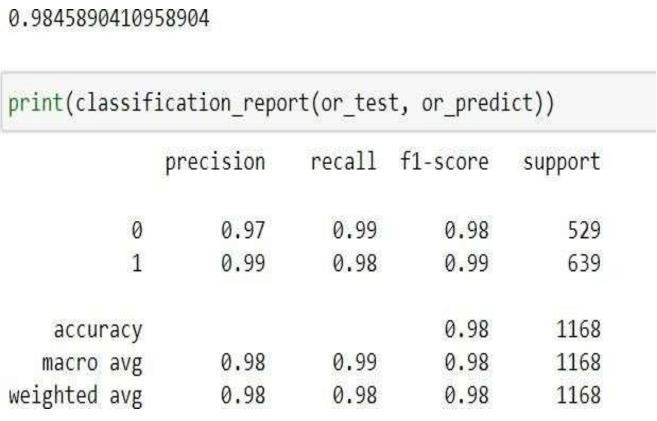
```
0.9426369863013698

print(classification_report(or_test, or_predict))

              precision    recall  f1-score   support

           0       0.97      0.90      0.93       529
           1       0.92      0.98      0.95       639

    accuracy                           0.94      1168
   macro avg       0.95      0.94      0.94      1168
weighted avg       0.94      0.94      0.94      1168
```

Figure 14: Random Forest Classifier after Smote-enn



```
0.9511986301369864

print(classification_report(or_test, or_predict))

              precision    recall  f1-score   support

           0       0.95      0.94      0.95       529
           1       0.95      0.96      0.96       639

    accuracy                           0.95      1168
   macro avg       0.95      0.95      0.95      1168
weighted avg       0.95      0.95      0.95      1168
```

Figure 15: Support Vector Machine after Smote-enn



```
0.9845890410958904

print(classification_report(or_test, or_predict))

              precision    recall  f1-score   support

           0       0.97      0.99      0.98       529
           1       0.99      0.98      0.99       639

    accuracy                           0.98      1168
   macro avg       0.98      0.99      0.98      1168
weighted avg       0.98      0.98      0.98      1168
```

Figure 16: K-Nearest Neighbors after Smote-enn

Comparison of Model Building Mechanisms Before and After SMOTEENN

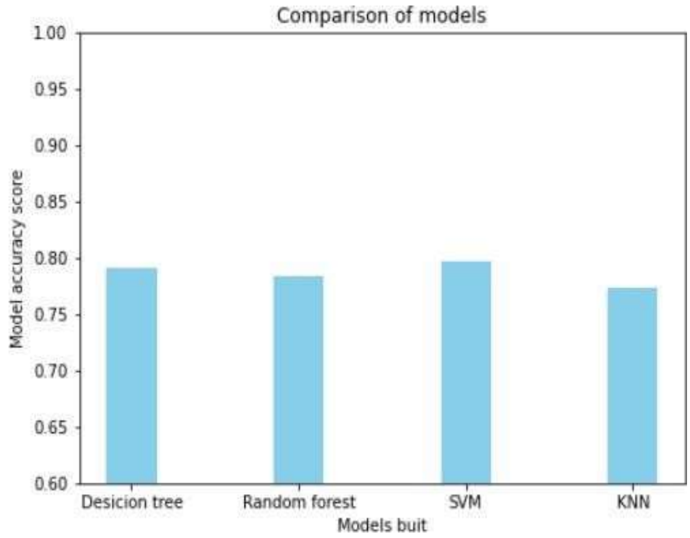


Figure 17

Fig 17 indicates the comparison between the above Machine learning Algorithms before the Smote-enn technique. From the above graph, we can say that Support Vector Machine has the highest efficiency of 79.6%.

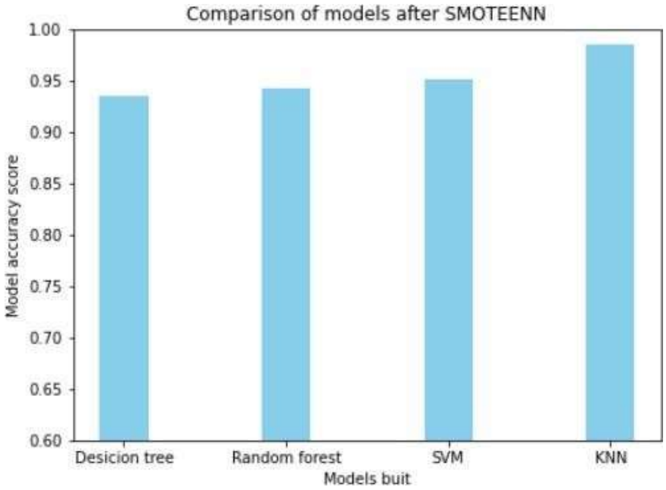


Figure 18

Fig 18 indicates the comparison between the above Machine Learning Algorithms after Smote-enn technique. We can clearly see the increase in the efficiency of each of the models where KNN with Smote-enn has the highest of 98.4%.

VI. Conclusion/Future Work

Conclusion:

The insights from the ML model enable the company to identify potential customer departures and service issues. Relationship Managers (RMs) receive daily updates on customers at risk of churning and the underlying factors. This facilitates proactive engagement with customers to address their concerns before churn occurs, enhancing customer retention and lifetime value. Furthermore, by mitigating customer churn, the company can reduce the volume of dispute calls, leading to cost savings and increased operational efficiency in call centers.

Future work:

The model can run regularly to monitor customer behavior changes, empowering relationship managers to respond effectively. Gathering additional customer data like reviews and demographics aids in understanding reasons for churn. Analyzing geographic and demographic patterns alongside competitors' data informs market share insights. Tracking total customer costs enhances profitability understanding. Integrating promotional data enables tailored pricing strategies. Deploying comprehensive data systems allows for holistic customer profiling, reducing churn and driving revenue growth.

VII. Team Contributions

Member 1 (Naresh):

Oversaw the project's entire development lifecycle in an executive capacity. Executed preprocessing, feature engineering, and model integration. Performed thorough coding to validate the functionality of machine learning classifiers. Played a key role in selecting and implementing machine learning methodologies.

Member 2(Raghu):

Helped gather and prepare data by obtaining client churn datasets from Kaggle. worked together to implement the random forest and decision tree for classification. helped plan and carry out experiments to assess the performance of the classifier. Participates in the creation of visualizations and the interpretation of results.

Member 3(Jaswanth):

Assisted in implementing supervised learning techniques, notably the SVM and KNN classifier, into practice. carried out a comparative study of classifier performance and added knowledge to the section on results. took part in the production of heat maps and comparison graphs, among other visualizations.

Member 4(Koushik):

Had a major influence on the development and application of the SMOTE-EEN function. helped assess the performance of the classifier using metrics for accuracy, recall, and precision. helped create the visualizations that improved the interpretation of the results. contributed qualitative insights while taking part in the misclassification analysis.

Every team member played a pivotal role in different facets of the project, fostering a cooperative and comprehensive endeavor.

References

[1] Dataset: https://www.kaggle.com/datasets/zagarsuren/telecom-churn-dataset-ibm-watson-analytics/code

[2] S. Stehani, N. Karunya, Dr. J.B.Ranjan, Sagara Sumathipala, T.C. Sandanayake, “Customer Churn Reasoning in Telecommunication Domain”, March 2020.

DOI: https://doi.org/10.1109/ICIP48927.2020.9367342

[3] Linmao Feng, “Research on Customer Churn Intelligent Prediction Model based on Borderline-SMOTE and Random Forest”, July 2022.

DOI: https://doi.org/10.1109/ICPICS55264.2022.9873702

[4] Nur Abdul Razak, Muhammad Hazim Wahid, “Telecommunication Customer Churn Prediction using Machine Learning”, December 2021.

DOI: https://doi.org/10.1109/MICC53484.2021.9642137

[5] V. Geetha, A. Punitha, A. Nandhini, S. Shakila, R.Sushmitha, T. Nandhini, “Customer Churn Prediction in Telecommunication Industry using Random Forest Classifier”, November 2020.

DOI: https://doi.org/10.1109/ICSCAN49426.2020.9262288

Libraries used:
NumPy, Pandas, Scikit-learn, Matplotlib, Seaborn, Plotly Express