TELECOM CUSTOMER RETENTION PREDICTION USING DEEP LEARNING

Sheshi Kiran Reddy Mandla,
School of Computer Science & Engineering,
Vellore Institute of Technology ,
Chennai, INDIA.
mandlasheshi.kiran2020@vitstudent.ac.in

Uppanapalli Lakshmi Sowjanya,
School of Computer Science & Engineering,
Vellore Institute of Technology ,
Chennai, INDIA.
lakshmi.sowjanya2020@vitstudent.ac.in

Amaravadi Dheeraj,
School of Computer Science & Engineering,
Vellore Institute of Technology,
Chennai, INDIA.
amaravadi.dheeraj2020@vitstudent.ac.in

Praveen Joe I R
School of Computer Science & Engineering,
Vellore Institute of Technology ,
Chennai, INDIA.
praveen.joe@vit.ac.in

***Abstract*—In order for a business to grow, it must both acquire new customers and retain existing ones. However, it is normal for businesses to experience some level of customer turnover, also known as churn. Churn is the percentage of customers who cancel their subscriptions within a certain time frame. One of the biggest issues facing large companies is customer turnover, also known as churn. This can have a significant impact on a company's revenue, particularly in the telecommunications industry. To combat this problem, companies are working to predict which customers are most likely to leave, so that they can take action to retain them. A related metric is growth rate, which is the percentage of new customers who sign up for a company's services during a specific period. The goal of our research is to create a dashboard or areport for key recommendations which helps us to create a model that can predict which customers are at the highest risk of churning, helping telecommunications providers to better retain their customers. Using some business intelligence tools such as power BI, tableau, maven & excel we are going to perform the data cleaning, data exploration, data visualization, suggesting key recommendations and building a model. We aregoing to also do some preliminary research on some topics which effect the churn and different types of churns such as the "Tariff Plan Churn", "Service Churn", "Product Churn", "Usage Churn" and the key drivers for subscriber loyalty and also key drivers which influence churn. In this paper we mainly concentrated on the sampling techniques such as SMOTE and k fold stratified sampling process and concentrated on balancing of the used dataset and the effect of accuracy on it and we used the deep learning method and CNN optimizers such as the Adam, Adagrad and RMSProp.**

Keywords—Customer churn, Growth Rate, Business Intelligence, Analysis, Data Cleaning, Data Exploration, Data Visualization, Key Recommendations, Model Building, Subscriber Loyalty.

I. INTRODUCTION

Customer retention is an important metric for any business, particularly in the telecommunications industry. It is crucial for companies to understand why customers leave and to predict which customers are most likely to leave in the future in order to take action to improve customer satisfaction and reduce customer churn.

A. OBJECTIVE

The objective of this project is to analyse customer retention in a telecom company and to predict which customers are most likely to leave in the future. By understanding the factors that contribute to customer churn, the company can take steps to improve customer satisfaction and reduce the number of customers who leave. The main objective of this project is to gain a deeper understanding of customer retention in a telecom company. This will involve analysing customer behaviour, demographic information, and customer service interactions. The goal is to identify the factors that contribute to customer churn. By understanding these factors, the company can take steps to improve customer satisfaction and reduce customer churn.

B. SCOPE

The scope of this project includes gathering data on customer behaviour, demographic information, and customer service interactions. This data will be analysed to understand the factors that contribute to customer churn, and machine learning algorithms will be used to predict which customers are most likely to leave in the future.

The scope of this project includes the following steps:

- **Data collection:** Collecting data on customer behaviour, demographic information, and customer service interactions.
- **Data analysis:** Analysing the collected data to understand the factors that contribute to customer churn.
- **Predictive modelling:** Using machine learning algorithms to predict which customers are most likely to leave in the future.

C. FEASIBILITY

This project is feasible due to the availability of data on customer behaviour and demographic information, as well as the availability of machine learning algorithms to make predictions. The accuracy of the predictions will depend on the quality and quantity of the data available, as well as the choice of algorithm and the way in which it is trained.

D. EXPECTED OUTCOME

The outcome of this project will be a better understanding of customer retention in the telecom company and predictions about which customers are most likely to leave in the future. This information will be valuable for the company in taking steps to improve customer satisfaction and reduce customer churn. The company may also be able to use the results of this project to inform future marketing and customer service strategies. For example, the company may identify certain customer segments that are at a higher risk of churning and target these segments with targeted marketing campaigns or special offers. In addition, this project may also lead to the development of a customer retention prediction model that can be used by the company on an ongoing basis to monitor and predict customer churn. This will allow the company to proactively address potential issues before they result in customers leaving the company.

II. RELATED WORK

The research in [1] talks about the importance of the customer churn analysis and how these predictive measures help in bringing huge profits for the organizations.So, in [1] 3 different variants of boosting have been used which are AdaBoost, Gentle AdaBoost, Modest AdaBoost. Top decile left is used as the primary assessment criterion.

In [2] , the research is carried out to develop a combined prediction model of customer churn using the prediction results of the decision tree model and the neural network model. The research shows us that the combined model works better than individual.

The algorithms used in [3] Logistic Regression and also SVM and also they tried to tackle the imbalance in the data by using the stratified sampling method.The experimental results are useful to predict the effective churn analysis.

In [4] , the researchers have emphasized on the fact that the customer churn analysis is important for better customer relationship management and also its very important in banking sector.They used the whale optimization algorithm on the SVM and different variants f SVM are compared such as the GA-SVM, SVM and also compared with the multilayer perceptron and the logistic regression methods.Finally they have achieved the best model for predicting customer churn in banks.

The algorithms used in the [5] is ANN & CNN and they have used the data from the national retail supermarket chain store(27 month data) and they have found out that CNN is performing better than any other algorithm used.

The paper [6] focuses on predicting customer churn in digital banking platforms, which is crucial for customer relationship management and increasing customer satisfaction. A classifier-based model is proposed in [6] using various supervised classification algorithms such as ANN, Logistic Regression, AdaBoost, Gradient_Boosting and Random Forest, with a unified voting approach for improved prediction accuracy. Hyperparameter optimization and feature ranking with random forest are also applied to improve model performance. The experimental case study with a dataset of 10,000 customers shows a promising accuracy of 87% for the proposed model, indicating its potential effectiveness in predicting customer churn behavior on digital banking platforms.

The paper [7] proposes a qualitative simulation model based on service exploration theory and qualitative Kuiperspsila simulation to describe and predict the changing process of customer churn. The model uses a causality graph, decision variables, state variables, and sequential causality constraint to capture the factors influencing customer churn. The approach in [7] involves multiple rounds of qualitative simulation and random behavior extraction to improve reasoning. The experimental simulation of the envisaged model shows promising results in terms of predicting and explaining the changing process of customer churn, providing qualitative insights to companies to implement effective strategies.

The paper [8] proposes a new approach to predicting the churn rate in the banking sector using the LSTM model and the SMOTE data preprocessing technique to address the data imbalance problem. The goal in [8] is to identify potential churners and non-churners among customers to increase profitability and customer retention. The proposed system shows promising results with an accuracy of 88%, outperforming the system without the SMOTE technique, indicating the potential value for organizations in identifying customers with a higher probability of churn.

The paper [9] proposes an unbalanced data transfer ensemble model (TEMID) that combines transfer learning, sampling and multiple classifications to predict customer churn. [9] focuses on augmenting the training set and balancing it using transfer learning and sampling techniques. The performance of TEMID is compared to existing transfer learning algorithms on two unbalanced datasets, and the results indicate that TEMID improves the performance of customer churn predictions.

The work in [10] highlights the importance of churn prediction as a key indicator of a company's long-term success or failure. Both machine learning and deep learning techniques are explored in this study to predict telecom customer churn. Traditional techniques such as Random Forest Classifier and SVMs are compared in [10] with newer architectures such as XGBoost and Deep Neural Networks and their effectiveness evaluated in more detail using Grid Search. The conclusion of the experiment suggests that the Random Forest model performs better in this particular use case, with a prediction accuracy of 90.96% for pre-grid search test data.

The paper [11] points out that although extensive research has been conducted to develop new data mining algorithms to predict customer churn, there has been limited focus on selecting effective input variables. for the model. The paper [11] proposes a step-by-step procedure for the selection of input variables related to telecom churn prediction and validates its effectiveness through comparative experiments using data from a telecom provider. This research contributes to the literature by addressing the important aspect of the choice of input variables in customer churn prediction models.

The paper [12] focuses on the development of a deep learning model to predict customer churn in e-commerce, which is difficult due to the high proportion of one-time customers. Despite the inherent inaccuracy of the predictions due to company specificity, the results show promising measures of accuracy, precision, and recall. The paper [12] fills a gap in the research and contributes to the existing literature by using deep learning tools and incorporating unsubscribe and transaction history to develop a methodology to predict churn rate. retailers. This research has the potential to improve overall business performance by improving customer retention strategies.

The paper [13] highlights the importance of customer retention in the fast-growing market across all industries, as acquiring new customers becomes more costly and competitive. It [13] emphasizes the need for service providers to avoid churn, which means customers leave the company's service. The paper focuses on examining popular machine learning algorithms used to predict churn, not only in banking, but also in other industries that rely heavily on customer retention.

The study of [14] proposes a semi-supervised ensemble model based on Metacost that combines semi-supervised learning, cost-sensitive learning and random subspace ensemble methods to predict the rate of customer unsubscribe. The model of [14] addresses the challenge of limited labeled samples by modifying the class labels of the initially labeled training set, selectively labeling unlabeled samples and training multiple classifiers. Empirical results show that the proposed model outperforms commonly used supervised and semi-supervised ensemble models in predicting customer churn, highlighting the effectiveness of the proposed approach in solving the data scarcity problem. in predicting customer churn.

The paper [15] proposes a churn prediction model for the telecommunications industry that uses classification and clustering techniques to identify churn customers and understand the reasons for customer churn. The model of [15] performs well with an accuracy of 88.63% using the random forest algorithm for classification. The paper suggests that by identifying important churn factors and using customer profiling techniques such as k-means clustering, companies can improve customer retention policies, recommend relevant promotions, and improve campaigns. marketing, thereby improving customer relationship management (CRM) and reducing churn.

The [16] gives us a overview idea about the best neural network architectures used for the data analysis and also it is helpful in selecting proper deep learning architecture mainly on the ANN.

The [17] gave us an insight about the swarm based algorithms used for multi layer clustering and the importance of these swarm based algorithms.

The TABLE-1 will gives us a inference on the literature survey we have done which consists of the title of the paper , dataset used , algorithms used and the results or accuracies achieved.

Table 1- Literature Survey

Paper	Dataset	Algorithm used	Accuracy achieved
[1] The Application of AdaBoost in Customer Churn Prediction	Anonymous Bank Data	AdaBoost, Gentle AdaBoost, Modest AdaBoost	Top Decile Left is usedas metric and it predicted that there are nearly 80% of the total potential churners in the predicted 10% riskiest customers
[2] Research on a Customer Churn Combination Prediction Model Based on Decision Tree and Neural Network	Customer information of a supermarket from June 2018 to April 2019	Decision Tree & Neural Network model	Decision Trees- 93.47% Neural Network model - 96.42% Combined Model - 98.87%
[3] Telecom Customer Churn Prediction Method Based on Cluster Stratified Sampling Logistic Regression	Dataset from Orange Telecom & UCI	Binary Logistic Regression Model (LRM)	SVM - 0.9013 LRM-0.9147
[4]Bank Customer Churn Prediction Analysis Based on Improved WOA-SVM	Data from Domestic Commercial Bank	WOA-SVM , GA-SVM, SVM, Multilayer Perceptron, Logistic Regression	Evaluation based on the correct rate, hit rate, coverage rate and boost factor
[5] Customer Churn Prediction Using Ordinary Artificial Neural Network and Convolutional Neural Network Algorithms: A Comparative Performance Assessment	Data from the national retail chain store	CNN, ANN, Logistic Regression, Linear SVM, Cubic SVM, Fine Gaussian SVM, Medium Gaussian SVM, Coarse Gaussian SVM, Fine KNN, Medium KNN, Coarse KNN, Cosine KNN, Cubic KNN, Weighted KNN.	1) Logistic Regression- 93.3% 2) Linear SVM- 95.9% 3) Cubic SVM-96% 4) Fine Gaussian SVM-65.5% 5) Medium Gaussian SVM- 94.7% 6) Coarse Gaussian SVM- 93% 7) Fine KNN- 73.5% 8) Medium KNN- 70.5% 9) Coarse KNN- 67.8% 10) Cosine KNN- 83.9% 11) Cubic KNN- 71.8% 12) Weighted KNN- 73.5% 13)ANN - 96% **14)CNN-97.62%**

[6]Enhancing Customer Churn Prediction in Digital Banking using Ensemble Modeling	Electronic Banking transactions - 10K transactions records	KNN (k-Nearest Neighbors), Logistic Regression, AdaBoost, Gradient_Boosting and Random Forest	Accuracy Of 87%.
[7] Study on the Qualitative Simulation-Based Customer Churn Prediction	Not Used	QSIM Algorithm	"Combination EXPLOSION" of QSIM algorithm is eliminated for CCP , gives out a churn probability of customer in each step
[8] LSTM Model to Predict Customer Churn in Banking Sector with SMOTE Data Preprocessing	Banking data	SMOTE Technique & LSTM	88% accuracy
[9]Transfer ensemble model for customer churn prediction with imbalanced class distribution	UCI - Churn Dataset & "CBC" dataset (Churn of Bank in Chongqing) of a commercial bank credit card customer database in Chongqing	Tr-SVM ; TrAdaBoosting ; TrBagging ; TEMID	**CHURN DATASET :** Tr-SVM : 0.8239 TrAdaBoosting : 0.8386 TrBagging : 0.8017 TEMID : 0.8124 **CBC DATASET :** Tr-SVM : 0.9165 TrAdaBoosting : 0.9156 TrBagging : 0.9257 TEMID : 0.9142
[10]Machine Learning Based Telecom-Customer Churn Prediction	Churn data of a Telecom company and is the data of the customers who relinquished the company's service a month back , it also contains the data of the services provided by the operator	Ridge Classification , Random Forest , XGBoost , KNN , SVC , Deep Neural Network	Highest Accuarcy for Random Forest before grid search on the test data which is 91.26%(test dataset)
[11] A novel and convenient variable selection method for choosing effective input variables for telecommunication customer churn prediction model	Customer basic information, customer bill information, customer calling detailed record information and customer called detailed record information. (2000 customers - 246-churned & 1754 normal)	Decision Tree	Tree60 - 0.87 & Tree216 - 0.90
[12]Deep learning for customer churn prediction in e-commerce decision support	Ecommerce data with 626275 rows ad 131 columns	Different deep learning network topologies	Global Avaerage Accuracy is 73.6%
[13] A Survey on Customer Churn Prediction using Machine Learning Techniques	Research on methods used for the customer churn prediction	Recommends SVM combined with boosting algorithms	Not Known
[14] Cost-sensitive semi-supervised ensemble model for customer churn prediction	"CBC" dataset (Churn of Bank in Chongqing) of a commercial bank credit card customer database in Chongqing & the churn dataset	SSEM, RASCO, DRSCO, Semi-bagging, Bagging RSS, IBRF	"CHONGQING" DATA SET - (g=1:3) - Semi bagging - 0.9513 - AUC value "CHURN" DATA SET - ((g=1:3)) - 0.8611 - Semi-bagging - AUC Value
[15]	Their own churn dataset & also churn-bigml datset	Random Forest , Attribute selected classifier , J48, Random Tree, Decision Stump, AdaBoostM1, Classifier+Decision Stump+Bagging+Random Tree, Naive Bayes,Multi Layer Perceptron , Logistic Regression,IBK,LWL	Highest Accuracy for Attribute selected classifier & J48 which is 91.91%

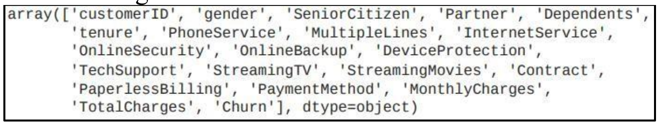
III. PROPOSED ARCHITECTURE

A. DATASET

Data set chosen for this project is related to telecommunication company 1-month customer data of originally of size of 7043 rows and 21 columns.

And it contains attributes as listed in below figure. There are totally 21 attributes listed here as shown in Figure-1.

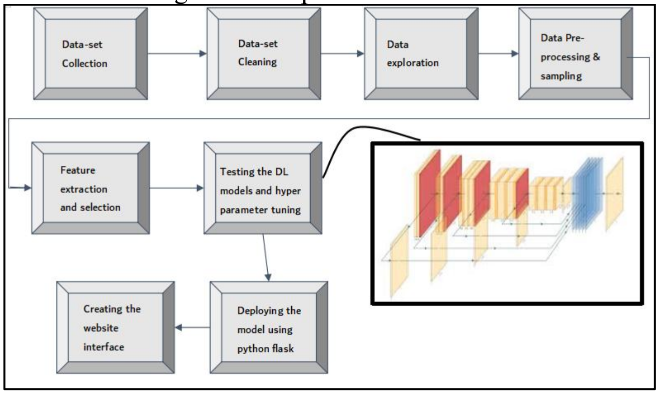
Figure 1 - Initial attributes in the dataset



```
array(['customerID', 'gender', 'SeniorCitizen', 'Partner', 'Dependents',
       'tenure', 'PhoneService', 'MultipleLines', 'InternetService',
       'OnlineSecurity', 'OnlineBackup', 'DeviceProtection',
       'TechSupport', 'StreamingTV', 'StreamingMovies', 'Contract',
       'PaperlessBilling', 'PaymentMethod', 'MonthlyCharges',
       'TotalCharges', 'Churn'], dtype=object)
```

The proposed architecture is shown in the Figure 2.The steps include the dataset collection , dataset cleaning , data exploration , data pre-processing & sampling , feature extraction and selection & testing the deep learningl models and hyper parameter tuning is done at the last for best results.

Figure 2 - Proposed Architecture


Data-set Collection
Data-set Cleaning
Data exploration
Data Pre-processing & sampling
Feature extraction and selection
Testing the DL models and hyper parameter tuning
Creating the website interface
Deploying the model using python flask

B. EXPLOROTARY DATA ANALYSIS & DATA PRE-PROCESSING

We first treated the missing values and then we found out that there are 11 null values in the "Total Charges" but we deleted those records as they does not show any effect on the data rather than formatting them.

Then we have done the "Uni-variate" analysis and also "Bi-variate" analysis.

Insights from the data analysis:

1)There are totally 5174 non-churners whereas 1869 customers are in an idea to change the telecom operator.

2)Churn rate is high in the customers who are not provided with services like online security(Figure 3) ,technical support(Figure 4), online-backup(Figure 5), device protection(Figure 6) and churned customers are mostly having month-to month contract(Figure 7) and whose connection is of fibre optic type(Figure 8) and also most churned customers are having "Electronic Check" payment method(Figure 9).

Figure 3 - Distribution of churners for types of online security mentioned

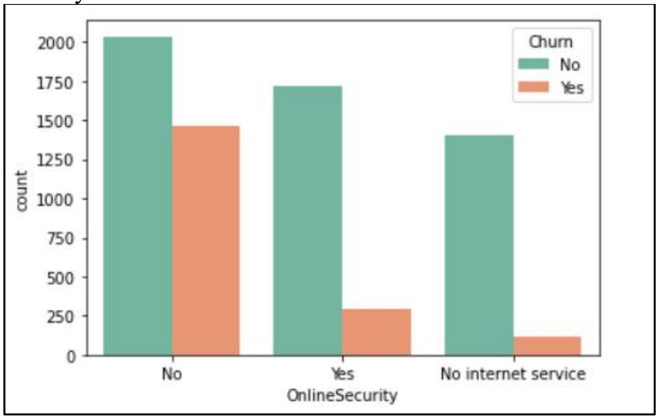


Figure 4 - Distribution of churners for types of Tech Support mentioned

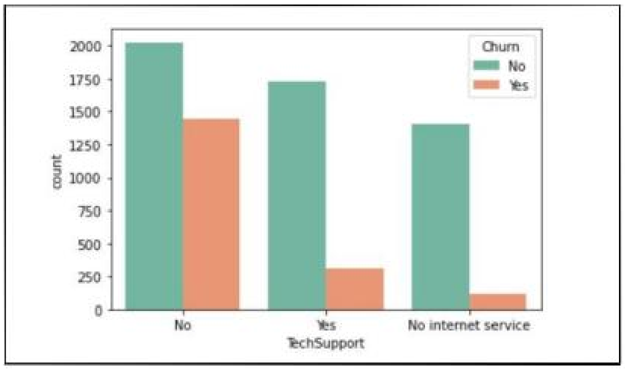


Figure 5 - Distribution of churners for types of Online Backup mentioned

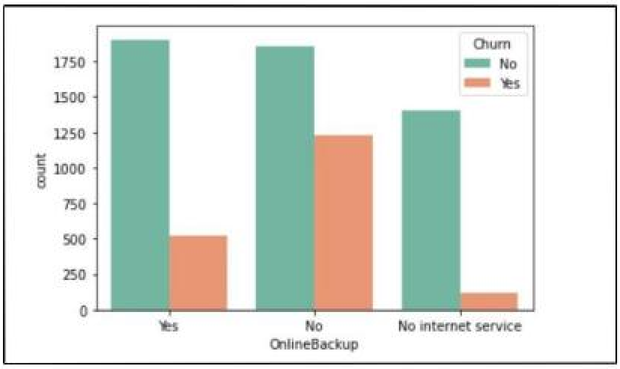


Figure 6 - Distribution of churners for types of Device Protection mentioned

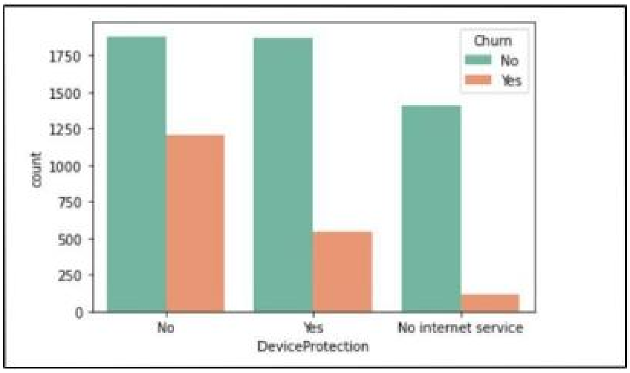


Figure 7- Distribution of churners for types of contract mentioned

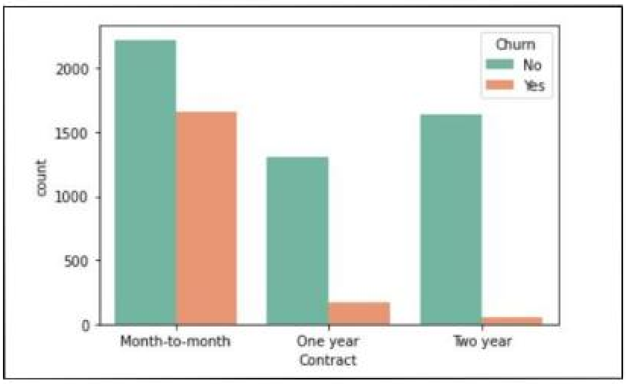
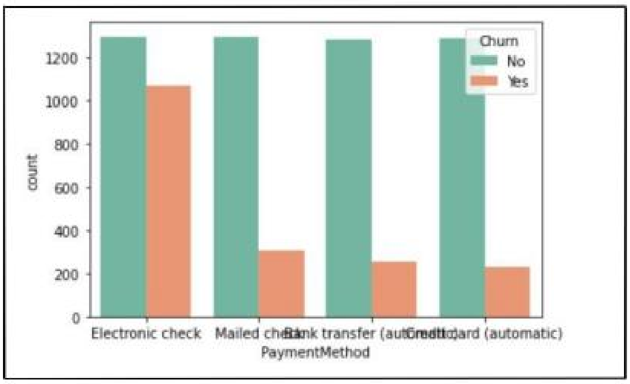


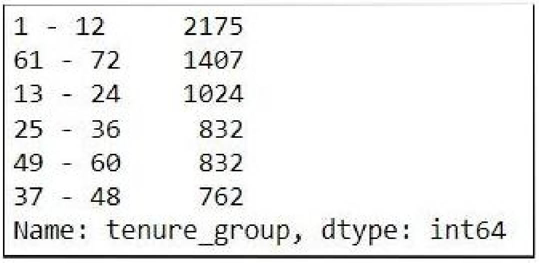
Figure 8- Distribution of churners for types of Payment Method mentioned



The values in the x-axis in figure 8 are "Electronic Check","Mailed Check", "Bank Transfer (automatic)","Credit Card (automatic)" .

3) We can understand that company is unable to build their trust in the customers in the first year itself or at the start , so there are many churners in the tenure of 1-12 months but they slowly build their trust in the customers which we can see it in the Figure 9.

Figure 9- Distribution of customers according to tenure in months

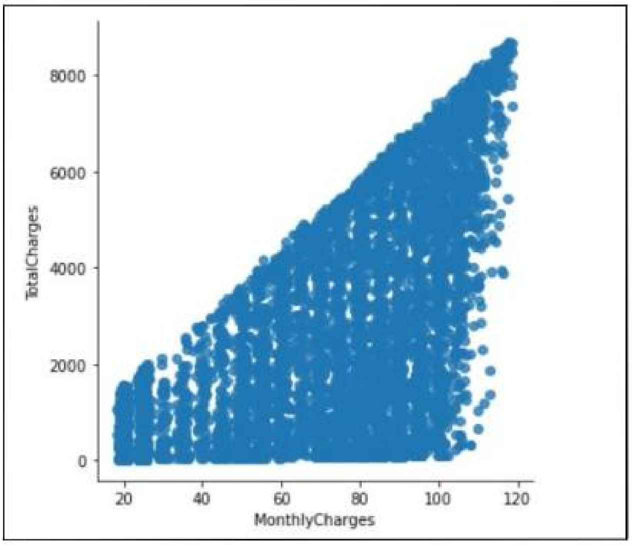


```
1 - 12     2175
61 - 72    1407
13 - 24    1024
25 - 36     832
49 - 60     832
37 - 48     762
Name: tenure_group, dtype: int64
```

4)So for better analysis we divide the tenure attribute into 6 attributes using the auto encoding process and we will do the same for the other categorical attributes.

5)Apart from that we can see that there is a direct linear relationship between the monthly charges and the total charges in Figure 10.

Figure 10- Relationship between the total charges and the monthly charges



But if we observe the graphs below , we can see that if the monthly charges are high then there are more churners(Figure 11) whereas if there are less total charges then there are more churners(Figure 12).This is contradicting the relationship we found between the monthly charges and the total charges.

Figure 11 - Distribution of Churners and the non churners based on the monthly charges using kde-plot

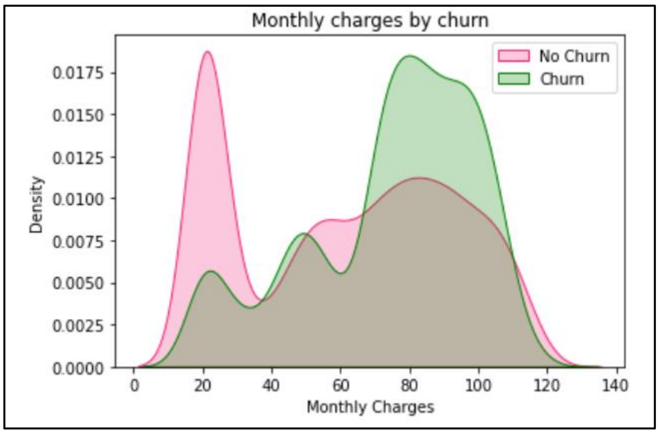
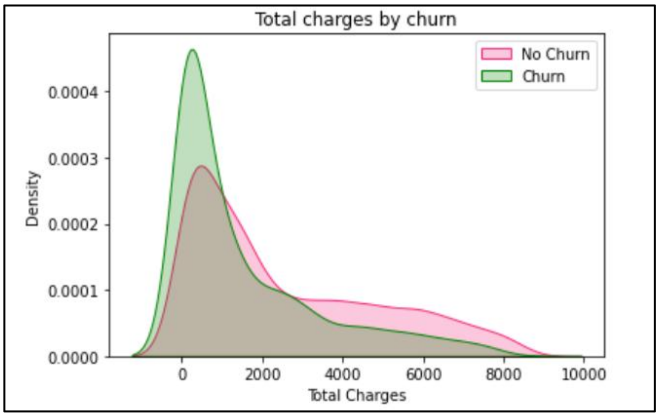


Figure 12 - Distribution of Churners and the non churners based on the total charges using kde-plot

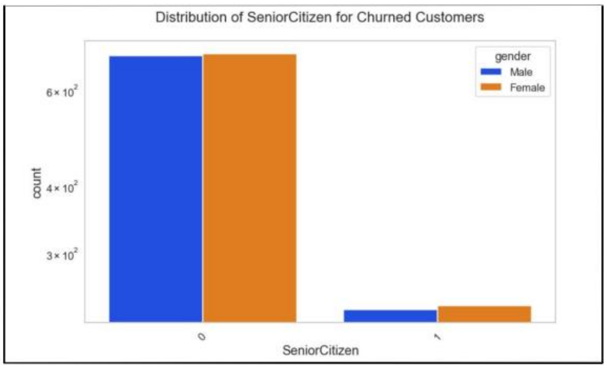


That actually says that the people whose tenure is between 1- 12 months will be having low total charges compared to the people whose tenure is high , so their total charges are low compared to these set of people but the churners experienced the high monthly charges.

6)We also used the StandardScaler() function to normalize the values of the monthly charges and the total charges.

7)From Bi-variate analysis we found that senior citizen are mostly churners as shown in the Figure 13.

Figure 13 - Distribution Of number senior citizens depicting the number of churners and non-churners based on the gender



8)Therefore , finally after data pre-processing we got totally 51 columns after auto encoding as you can see in the Table 2.

Table 2 - Newly added attributes after data pre-processing

S.No	Attribute Name
1	SeniorCitizen
2	Churn
3	gender_Female
4	gender_Male
5	Partner_No
6	Partner_Yes
7	Dependents_No
8	Dependents_Yes
9	PhoneService_No
10	PhoneService_Yes
11	MultipleLines_No
12	MultipleLines_No phone service
13	MultipleLines_Yes
14	InternetService_DSL
15	InternetService_Fiber optic
16	InternetService_No
17	OnlineSecurity_No
18	OnlineSecurity_No internet service
19	OnlineSecurity_Yes
20	OnlineBackup_No
21	OnlineBackup_No internet service
22	OnlineBackup_Yes
23	DeviceProtection_No
24	DeviceProtection_No internet service
25	DeviceProtection_Yes
26	TechSupport_No
27	TechSupport_No internet service
28	TechSupport_Yes
29	StreamingTV_No
30	StreamingTV_No internet service
31	StreamingTV_Yes
32	StreamingMovies_No
33	StreamingMovies_No internet service
34	StreamingMovies_Yes
35	Contract_Month-to-month
36	Contract_One year
37	Contract_Two year
38	PaperlessBilling_No
39	PaperlessBilling_Yes
40	PaymentMethod_Bank transfer (automatic)
41	PaymentMethod_Credit card (automatic)
42	PaymentMethod_Electronic check
43	PaymentMethod_Mailed check
44	tenure_group_1 - 12
45	tenure_group_13 - 24
46	tenure_group_25 - 36
47	tenure_group_37 - 48
48	tenure_group_49 - 60
49	tenure_group_61 - 72
50	MonthlyCharges_Normalized
51	TotalCharges_Normalized

C. *CNN ARCHITECTURE*

A convolutional neural network (CNN) is a type of deep learning model that is commonly used in image and video processing tasks, but can also be applied to other types of data. The basic idea behind CNNs is to learn and extract features from raw input data such as images or text by applying multiple convolution filters and combining layers. When predicting the results of a data set where every attribute is of equal importance, a CNN model can be used to automatically learn the most important features of the input data. To do this, the input data is first converted into a suitable form that can be fed into the network, such as a series of image sets or vector sequences. The CNN model then applies a series of convolution filters to the input data that search the input data for specific features or patterns. These filters can be thought of as little windows that read the input and calculate the dot product between the filter weights and the input data values at each location. The resulting output is a functional map showing the presence or absence of a filter feature or pattern anywhere in the input data. After the convolutional filters, levels of aggregation are used to sample the feature maps and reduce their size, making the model more computationally efficient and less prone to overfitting. Finally, the resulting feature maps are smoothed into a vector and fed into one or more fully connected layers that perform a series of linear operations to produce the final model result. Based on the results of the CNN model, various outcomes, such as binary classification or regression, can be predicted. When predicting outcomes from a dataset where every attribute is equally important, a CNN model can learn to extract key characteristics from input data and use them to make accurate predictions.

The model used here is the CNN architecture consists of the following layers:

1. **Convolutional layer (Convolution1D):** This layer has 64 filters (also called kernels) with a size of 3 and applies the Rectified Linear Unit (ReLU) activation function. The padding parameter is set to "same", which means that the input is padded with zeros to ensure that the output has the same width and height as the input. The input_shape parameter is set to (50, 1), which means that the input data has 50 time steps and 1 channel (or function).

2. **Maximum pooling level (MaxPooling1D):** This level performs maximum pooling with a pool size of 2, reducing the spatial size of the input by taking the maximum value within each pool. This helps reduce the number of parameters and computations in the network.

3. **Flat layer (flatten):** This layer flattens the input tensor into a required one-dimensional array before passing it to a fully connected layer.Fully connected layer (dense): This layer has 128 units and applies the ReLU activation function. It is a fully connected layer, which means that every neuron in this layer is connected to every neuron in the previous layer.

4. **Dropout Layer:** This layer applies dropout regularization at a rate of 0.5, which randomizes 50% of the input units to 0 during training to avoid overfitting.

5. **Output layer (dense):** This layer has 2 units and applies the sigmoid activation function used for binary classification problems. The output of this level represents the predicted probabilities of the two classes.Overall, this CNN architecture is often used for binary classification tasks where the input data has 50 time steps and 1 channel.

6. The total trainable parameters in the CNN architecture provided in the code are calculated as follows:

I. Convolutional Layer (Convolution1D):

i. Number of filters: **64**

ii. Filter size: **3**

iii. Number of input channels: **1**

iv. Number of output channels (units): **64**

v. Number of trainable parameters: (filter size * number of input channels + 1) * number of output channels = (3 * 1 + 1) * 64 = **256**

II. Fully Connected Layer (Dense):

i. Number of input units: **128 (output from the previous layer)**

ii. Number of output units: **128**

iii. Number of trainable parameters: **(number of input units + 1)** * number of output units = **(128 + 1) * 128 = 16512**

III. Dropout Layer (Dropout):

i. Dropout rate: **0.5 (50% of the input units are randomly set to 0 during training)**

ii. Number of trainable parameters: **0 (dropout layer has no trainable parameters)**

IV. Output Layer (Dense):

i. Number of input units: **128 (output from the previous layer)**

ii. Number of output units: **2 (binary classification problem with 2 classes)**

iii. Number of trainable parameters: **(number of input units + 1) * number of output units =** **(128 + 1) * 2 = 258**

iv. Total trainable parameters = **Sum of trainable parameters from all layers = 256 + 16512 + 0 + 258 = 17026**

7. So, the total number of trainable parameters in the CNN architecture provided in the code is **17026.**

D. STRATIFIED K-FOLD CROSS VALIDATION

Stratified K-Fold Cross-Validation is a technique used to estimate the performance of the machine learning model. It's exactly like a Stratified K Fold Sampling, but rather than use it for training and testing purposes, it is being used to assess the model's performance. In order to ensure that each fold has almost the same amount of samples from each class, the data set is divided into K equalized folds in the Stratified K Fold Cross Validation. The model shall be trained on K1 folds and assessed in the remaining ones.

It will be repeated K times, using all of the folds for evaluation on a single occasion. For an estimation of model performance, the results shall be averaged across all K folds. The advantages of synthesized K Fold Cross Validation are that it provides a more precise estimate of the model's performance compared to other methods such as simplified K Fold Cross Validation.

For this reason, it ensures that every fold is distributed in line with the same classes to avoid discrimination and oversimplification. When dealing with imbalanced datasets in which individual classes do not have equal number of samples, the use of Stratified KDLLF Cross Validation is particularly useful. In such cases, using a representative sample of data can help to ensure an accurate assessment of the model and may lead to more accurate prediction by use of K Stratified XML Cross Fold Validation. One of the strongest techniques for evaluating machine learning models' performance, with particularly in view of balancing datasets, is systematic K.Fold cross validation.

E. ADAM

Adam is another optimization algorithm that can be used in CNN models to efficiently train them and improve their performance in predicting outcomes from a dataset where every attribute is equally important. Here are some key points about Adam's uniqueness and how it works:

- This means that it not only uses a moving average of past gradients like momentum, but also adapts the learning rates of each weight parameter based on the statistics of the gradients.
- The learning rates in Adam are adapted by computing a separate adaptive learning rate for each weight parameter based on estimates of the first and second moments of the gradients.
- The first moment estimate is a moving average of the gradients, which serves as an estimate of the gradient mean.
- The second moment estimate is a moving average of the squared gradients, which serves as an estimate of the gradient variance.
- By using these estimates, Adam is able to automatically adapt the learning rates for each weight parameter in a way that is sensitive to the magnitude and direction of the gradients.
- Overall, Adam is a powerful optimization algorithm that can be used to train CNN models more efficiently and effectively, especially when every attribute of the input data is equally important.
- Its ability to adapt the learning rates of each weight parameter based on estimates of the gradients can help the model to converge more quickly and produce more accurate results.

F. Adagrad

Adagrad is an optimization algorithm that can be used with CNN models to train them more effectively and efficiently. By predicting results from a dataset where each attribute is equally important, Adagrad can help ensure that the model learns the most important features of the input data and converges to a good solution. Here are some key points about the uniqueness of Adagrad and how it works:

- Adagrad is a gradient-based optimization algorithm, i.e. it works by calculating the gradients of the loss function with respect to the model parameters and adjusting them in a direction that minimizes the loss.
- This means that parameters with large gradients have a lower learning capacity, while parameters with small gradients have a higher learning capacity.
- Learning rate matching in Adagrad is done using a parameter specific learning rate, calculated by dividing the initial learning rate by the sum of squares of the historical gradients for that parameter.
- This means that parameters with large historical gradients have a slower learning rate, while parameters with small historical gradients have a higher learning rate.
- This is because the parameter-specific learning rate can be adjusted separately for each feature, so the model learns the most important features faster and more accurately.
- Instead, the learning rate is automatically adjusted for each parameter based on its historical gradients.

G. RMSProp

RMSProp is an optimization algorithm that can be used in CNN models to improve the efficiency and effectiveness of the training process. When predicting outcomes from a dataset where each attribute is equally important, RMSProp can help the model learn the most important features and converge to a good solution. Here are some key points about RMSProp and its uniqueness:

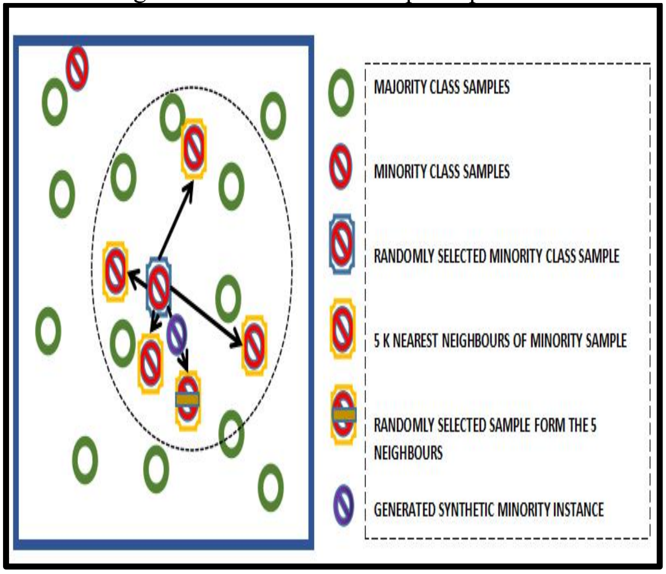
- Unlike Adagrad, RMSProp uses a moving average of squared gradients to normalize the learning rate, which helps prevent the learning rate from becoming too large or too small as training progresses.
- One of the main differences between RMSProp and other optimization algorithms is that it uses a parameter-specific learning rate that adapts based on the moving average of the squared gradients of each parameter.
- This is because parametric learning rates can help the model avoid getting stuck in narrow valleys of the loss function.
- One potential disadvantage of RMSProp is that it can be sensitive to the choice of hyper-parameters, such as the initial learning rate and the decay rate of the moving average.
- Overall, RMSProp is a powerful optimization algorithm that helps CNN models learn more efficiently and effectively, especially when dealing with large datasets or datasets with high-dimensional inputs.
- Its parametric learning rate adaptation, moving average of squared slopes, and robustness to local optima make it a popular choice for many deep learning applications.

H. SMOTE TECHNIQUE

The SMOTE algorithm is a technique used in machine learning to address imbalanced classification problems, where one class has significantly fewer samples than the other. It generates synthetic samples for the minority class by creating new ones that are linear combinations of existing samples. To apply SMOTE, a sample is selected from the minority class, and its k nearest neighbors (where k is a user-defined parameter) are identified. One of these neighbors is randomly chosen, and a new sample is generated by interpolating between the selected sample and the chosen neighbor. This process is repeated until the desired number of synthetic samples has been created.

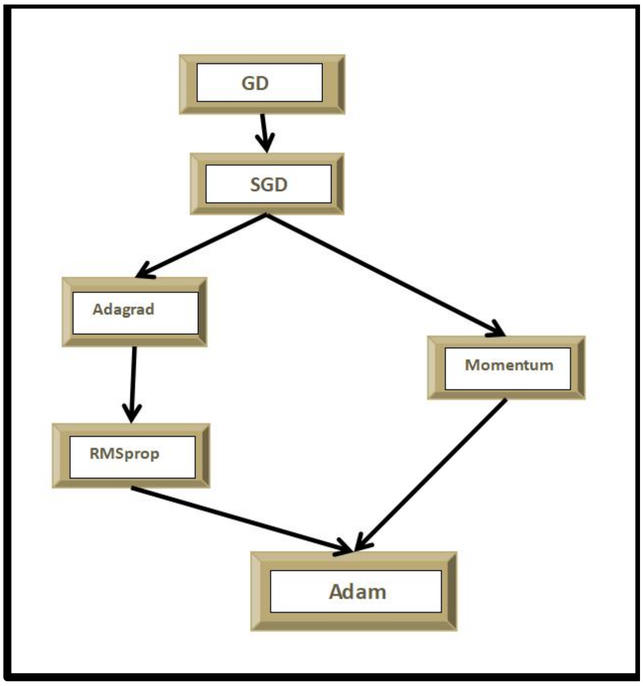
SMOTE is effective in increasing the number of minority class samples and improving classifier performance for imbalanced classification tasks. However, in some cases, it can produce noisy or redundant samples, which may negatively impact classification performance. Therefore, it's crucial to carefully determine the appropriate value for the k parameter and evaluate classifier performance with and without SMOTE oversampling to determine its usefulness for a given data set and task. The illustration of the algorithms is shown in the Figure 14.

Figure 14 - SMOTE technique depiction


MAJORITY CLASS SAMPLES
MINORITY CLASS SAMPLES
RANDOMLY SELECTED MINORITY CLASS SAMPLE
5 K NEAREST NEIGHBOURS OF MINORITY SAMPLE
RANDOMLY SELECTED SAMPLE FORM THE 5 NEIGHBOURS
GENERATED SYNTHETIC MINORITY INSTANCE

IV. PROPOSED METHODOLOGY

Figure 15 - Hierarchy of optimizers of Deep Learning



The Figure 15 shows the hierarchy of the deep learning optimizers.

A. Points that support Adagrad and RMSProp and Adam model's high accuracies.

1) Adagrad:

- Adagrad is a proven optimization algorithm that has been used successfully in many deep learning applications.
- Adagrad is well-suited to deal with sparse gradients, which can be a common issue in deep learning models with large numbers of parameters.
- Adagrad adjusts the learning rate for each parameter based on its past gradients, which can help the model learn efficiently and effectively.
- Adagrad is simple to implement and does not require much hyper parameter tuning, making it an attractive option for many deep learning applications.
- One potential drawback of Adagrad is that it may accumulate too much gradient history over time, leading to a diminishing learning rate and slow convergence.

2) RMSProp:

- RMSProp is an adaptive optimization algorithm that can help deep learning models learn more efficiently and effectively.
- RMSProp uses a moving average of squared gradients to adjust the learning rate for each parameter, which can help prevent the learning rate from becoming too large or too small.
- RMSProp is less prone to getting stuck in local optima than some other optimization algorithms, such as stochastic gradient descent.
- RMSProp can help the model learn more quickly by adapting the learning rate based on the gradients of each parameter.
- One potential drawback of RMSProp is that it can be sensitive to the choice of hyper parameters, such as the initial learning rate and the decay rate of the moving average.

3) Adam:

- Adam is an advanced optimization algorithm that combines the benefits of Adagrad and RMSProp.
- Adam uses both a moving average of past gradients and a moving average of past squared gradients to adapt the learning rate for each parameter.
- Adam is well-suited to deal with sparse gradients, making it a good choice for deep learning models with large numbers of parameters.
- Adam is computationally efficient and requires relatively little hyper parameter tuning, making it an attractive option for many deep learning applications.
- One potential drawback of Adam is that it can be sensitive to the choice of hyper parameters, such as the initial learning rate and the decay rates of the moving averages.

B. DATA PRE-PROCESSING

Since the data set had significantly more records for the non-churners we applied the smote resampling technique to avoid the imbalance in the data.

Total number of records totally added to the churn type is 775 (change in percentage is 18.3%) and for the non - churn type 1921 rows are deleted which is 18.3%).

Table 3 - Datasets records changed after SMOTE technique

Sample Type	Before SMOTE RESAMPLING	After SMOTE RESAMPLING
Non Churn	5174 (73.46%)	3253(55.16%)
Churn	1869(26.53%)	2644(44.83%)
Total	7043	5897

So 18.3% is deleted from the non churn type and it is added to the churn type(the numbers are shown in the Table 3).

C. MODEL BUILDING

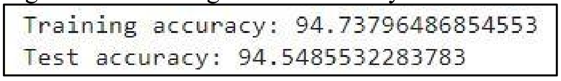
In the CNN model, we added two layers - Convolution1D and MaxPooling1D - and used different optimizers such as Adam(200 Epochs), RMSProp(200 Epochs), and Adagrad(200 Epochs). We applied 2-fold cross-validation and binary cross-entropy and 10-fold cross validation and cross entropy as the metric for loss. Additionally, we performed hyperparameter tuning for parameters such as number of hidden units, number of stacked units, loss function, value of dropouts, number of epochs, folds variations, and choice of metrics, in order to obtain a final model with improved accuracy.

To evaluate the performance of the model, we used various performance metrics such as precision, recall, F1-score, and support, and also plotted the ROC curve to calculate the area under the curve (AUC).

V. RESULTS & FIGURES

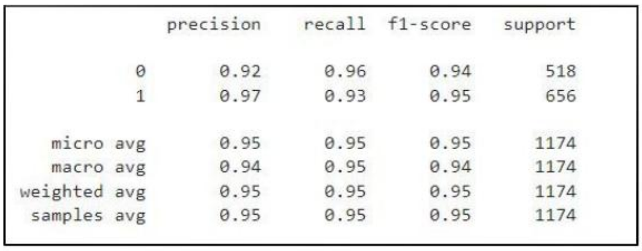
a) **ADAM (200 EPOCHS)**

Figure 16 - Training & Test accuracy of ADAM



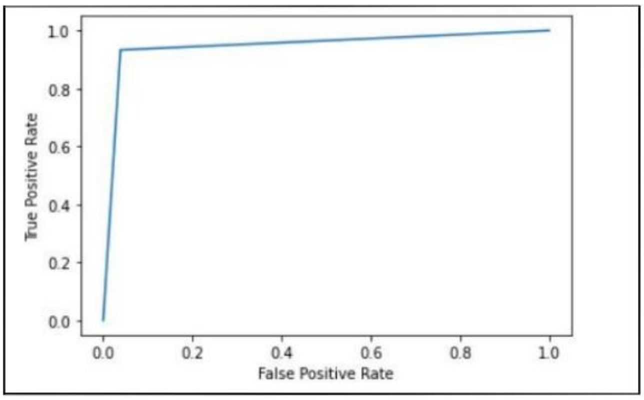
```
Training accuracy: 94.73796486854553
Test accuracy: 94.5485532283783
```

Figure 17 - Performance metrics report of ADAM



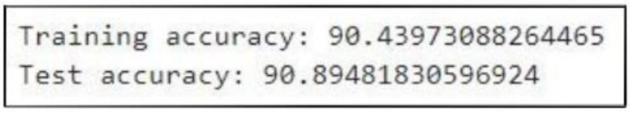
```
              precision    recall  f1-score   support

           0       0.92      0.96      0.94       518
           1       0.97      0.93      0.95       656

   micro avg       0.95      0.95      0.95      1174
   macro avg       0.94      0.95      0.94      1174
weighted avg       0.95      0.95      0.95      1174
 samples avg       0.95      0.95      0.95      1174
```

Figure 18 - ROC curve for the ADAM technique



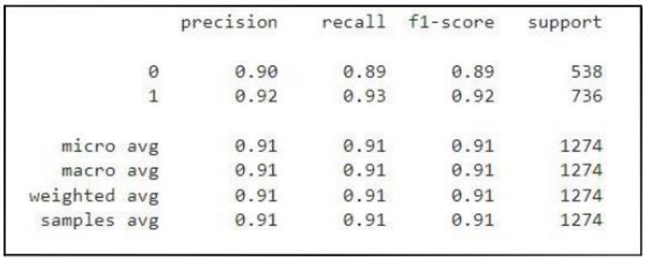
b) **ADAGRAD (200 EPOCHS)**

Figure 19 - Training & Test accuracy of ADAGRAD



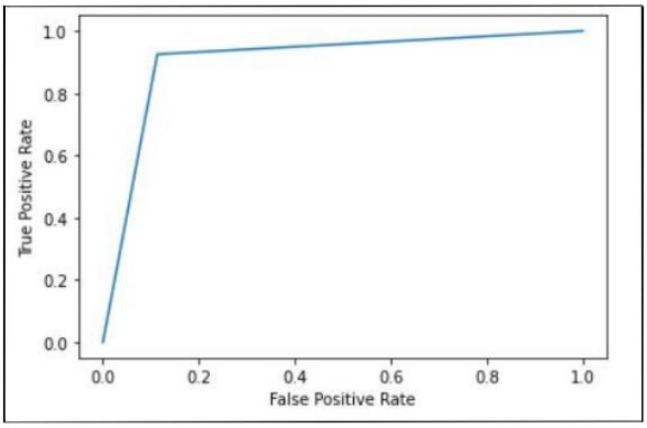
```
Training accuracy: 90.43973088264465
Test accuracy: 90.89481830596924
```

Figure 20 - Performance metrics report of ADAGRAD



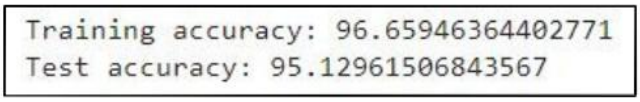
```
              precision    recall  f1-score   support

           0       0.90      0.89      0.89       538
           1       0.92      0.93      0.92       736

   micro avg       0.91      0.91      0.91      1274
   macro avg       0.91      0.91      0.91      1274
weighted avg       0.91      0.91      0.91      1274
 samples avg       0.91      0.91      0.91      1274
```

Figure 21 - ROC curve for the ADAGRAD technique



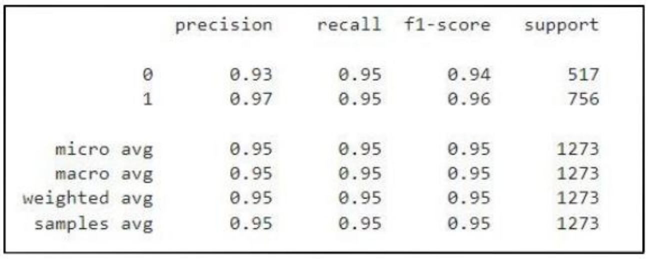
c) **RMSPROP (200 EPOCHS)**

Figure 22 - Training & Test accuracy of RMSPROP



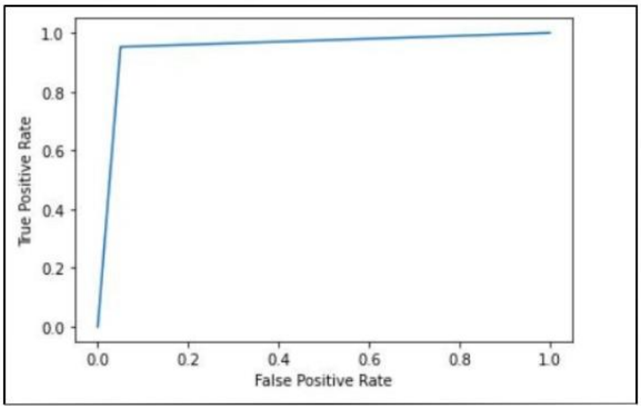
```
Training accuracy: 96.65946364402771
Test accuracy: 95.12961506843567
```

Figure 23 - Performance metrics report of RMSPROP



```
              precision    recall  f1-score   support

           0       0.93      0.95      0.94       517
           1       0.97      0.95      0.96       756

   micro avg       0.95      0.95      0.95      1273
   macro avg       0.95      0.95      0.95      1273
weighted avg       0.95      0.95      0.95      1273
 samples avg       0.95      0.95      0.95      1273
```

Figure 24 - ROC curve for the RMSPROP technique



We have selected the "RMSPROP - 200 epochs" model because it gives us the high accuracy and also the the f-score value near 1 (~0.96) proves that the model is best trained, neither under-fitted or over-fitted.

The above Figures from Figure-16 to Figure-24 are respectively the training & testing accuracy , AUC score and the ROC curve respectively of Adam , Adagrad and RMSProp optimizers.

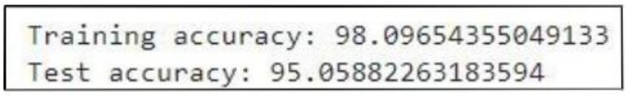
Table 4 - Accuracy Results Of the Used Models

OPTIMIZER	NO OF EPOCHS	TRAINING ACCURACY (%)	TEST ACCURACY (%)
Adam	200	94.737964865	94.548553223
Adagrad	200	90.439730882	90.894818396
RMSprop	200	96.659463644	95.129615068

Since we got highest train or test accuracy for RMSPROP we also wanted to do a 10-fold stratified sampling and cross validation on the above method rather than the 2-fold sampling on the RMSPROP and we achieved a better training accuracy than mentioned till now but the test accuracy is not improved.

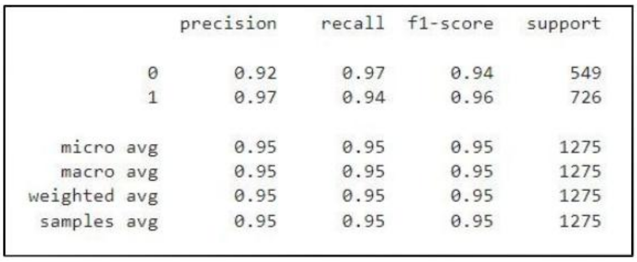
d) RMSPROP (200 EPOCHS) - 10 fold stratified sampling & cross validation

Figure 25 - Training & Test accuracy of RMSPROP (10 fold stratified sampling and cross validation).



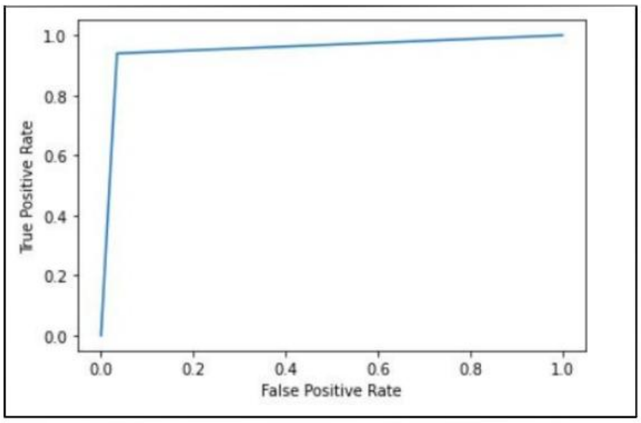
```
Training accuracy: 98.09654355049133
Test accuracy: 95.05882263183594
```

Figure 26- Performance metrics report of RMSPROP (10 fold stratified sampling and cross validation).



```
              precision    recall  f1-score   support

           0       0.92      0.97      0.94       549
           1       0.97      0.94      0.96       726

   micro avg       0.95      0.95      0.95      1275
   macro avg       0.95      0.95      0.95      1275
weighted avg       0.95      0.95      0.95      1275
 samples avg       0.95      0.95      0.95      1275
```

Figure 27 - ROC curve for the RMSPROP technique (10 fold stratified sampling and cross validation)



VI. CONCLUSION

A deep neural network consisting of stacked Convolution1D and MaxPooling1D layers, together with appropriate changes in optimizers and epochs, has been successfully created. In order to avoid bias in the model & to feed the data to the algorithm for training, appropriate preprocessing & sampling has been carried out. For improved accuracy, the model has been tested by using a two fold cross validation method and tuning with Hyperparameters.

On the same data set of 200 epochs & 2 fold stratified sampling and cross validation, RMSProp optimizer showed better results than we could have expected that is it showed a considerable increase in accuracy which was almost 98% (RMSPROP - 200 Epochs - 10 fold stratified sampling and cross validation) where as test accuracy is high for (RMSPROP - 200 Epochs - 2 fold stratified sampling) which is nearly 95%.

REFERENCES

[1] S. Jinbo, Li Xiu and L. Wenhuang, "The Application ofAdaBoost in Customer Churn Prediction," 2007 International Conference on Service Systems and Service Management, Chengdu, China, 2007, pp. 1-6, doi: 10.1109/ICSSSM.2007.4280172.

[2] X. Hu, Y. Yang, L. Chen and S. Zhu, "Research on a Customer Churn Combination Prediction Model Based on Decision Tree and Neural Network," 2020 IEEE 5th International Conference on Cloud Computing and Big Data Analytics (ICCCBDA), Chengdu, China, 2020, pp. 129-132, doi: 10.1109/ICCCBDA49378.2020.9095611.

[3] Peng Li, Siben Li, Tingting Bi and Yang Liu, "Telecom customer churn prediction method based on cluster stratified sampling logistic regression," International Conference on Software Intelligence Technologies and Applications & International Conference on Frontiers of Internet of Things 2014, Hsinchu, 2014, pp. 282-287, doi: 10.1049/cp.2014.1576.

[4] H. He, "Bank Customer Churn Prediction Analysis Based on Improved WOA-SVM," 2022 3rd International Conference on Electronic Communication and Artificial Intelligence (IWECAI), Zhuhai, China, 2022, pp. 451-455, doi: 10.1109/IWECAI55315.2022.00093.

[5] SEYMEN, Omer Faruk, Emre ÖLMEZ, Onur DOĞAN, E. R. Orhan, and Kadir HIZIROĞLU. "Customer Churn Prediction Using Ordinary Artificial Neural Network and Convolutional Neural Network Algorithms: A Comparative Performance Assessment." Gazi University Journal of Science.

[6] R. Alenezi and S. A. Ludwig, "Classifying DNS Tunneling Tools For Malicious DoH Traffic," 2021 IEEE Symposium Series on Computational Intelligence (SSCI), Orlando, FL, USA, 2021, pp. 1-9, doi: 10.1109/SSCI50451.2021.9660136.

[7] M. Galal, S. Rady and M. Aref, "Enhancing Customer Churn Prediction in Digital Banking using Ensemble Modeling," 2022 4th Novel Intelligent and Leading Emerging Sciences Conference (NILES), Giza, Egypt, 2022, pp. 21-25, doi: 10.1109/NILES56402.2022.9942408.

[8] H. Wang and W. -j. Li, "Study on the Qualitative Simulation-Based Customer Churn Prediction," 2009 International Symposium on Information Engineering and Electronic Commerce, Ternopil, Ukraine, 2009, pp. 528-532, doi: 10.1109/IEEC.2009.117.

[9] J. Latheef and S. Vineetha, "LSTM Model to Predict Customer Churn in Banking Sector with SMOTE Data Preprocessing," 2021 2nd International Conference on Advances in Computing, Communication, Embedded and Secure Systems (ACCESS), Ernakulam, India, 2021, pp. 86-90, doi: 10.1109/ACCESS51619.2021.9563347.

[10] Y. Wang and J. Xiao, "Transfer Ensemble Model for Customer Churn Prediction with Imbalanced Class Distribution," 2011 International Conference of Information Technology, Computer Engineering and Management Sciences, Nanjing, China, 2011, pp. 177-181, doi: 10.1109/ICM.2011.397.

[11] P. Bhuse, A. Gandhi, P. Meswani, R. Muni and N. Katre, "Machine Learning Based Telecom-Customer Churn Prediction," 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS), Thoothukudi, India, 2020, pp. 1297-1301, doi: 10.1109/ICISS49785.2020.9315951.

[12] Pondel, Maciej & Wuczyński, Maciej & Gryncewicz, Wieslawa & Łysik, Łukasz & Hernes, Marcin & Rot, Artur & Kozina, Agata. (2021). Deep Learning for Customer Churn Prediction in E-Commerce Decision Support. Business Information Systems. 3-12. 10.52825/bis.v1i.42.

[13] A., Saran & D., Chandrakala. (2016). A Survey on Customer Churn Prediction using Machine Learning Techniques. International Journal of Computer Applications. 154. 13-16. 10.5120/ijca2016912237.

[14] A., Saran & D., Chandrakala. (2016). A Survey on Customer Churn Prediction using Machine Learning Techniques. International Journal of Computer Applications. 154. 13-16. 10.5120/ijca2016912237.

[15] I. Ullah, B. Raza, A. K. Malik, M. Imran, S. U. Islam and S. W. Kim, "A Churn Prediction Model Using Random Forest: Analysis of Machine Learning Techniques for Churn Prediction and Factor Identification in Telecom Sector," in IEEE Access, vol. 7, pp. 60134-60149, 2019, doi: 10.1109/ACCESS.2019.2914999.

[16] Joe, Praveen. "IR and Varalakshmi, P„„A Survey on Neural Network Models for Data Analysis‟." ARPN Journal of Engineering and Applied Sciences 10.11 (2015): 4872-4876.

[17] I. R. Praveen Joe & P. Varalakshmi (2019) A Multilayered Clustering Framework to build a Service Portfolio using Swarm-based algorithms , Automatika, 60:3, 294-304, DOI: 10.1080/00051144.2019.1590951.

[18] Joe, IR Praveen, and P. Varalakshmi. "A Two Phase Approach for Efficient Clustering of Web Services." Computational Intelligence, Cyber Security and Computational Models. Springer,Singapore, 2016. 165-170.