International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

Customer Churn Prediction in Telecom Industry Using Deep Learning Techniques

[1]Vinston Raja R, [2]Deepak Kumar A, [3]Prabu Sankar N, [4]Gnanavel R, [5]Krishnaraj M, [6]Irin Sherly S
[1]Assistant Professor, Information Technology,
Panimalar Engineering College, Chennai, India.
rvinstonraja@gmail.com
[2]Assistant Professor, Computer Science and Engineering,
St. Joseph's Institute of Technology, Chennai, India.
deepakkumar@stjosephstechnology.ac.in
[3]Assistant Professor, Department of Information Technology,
Panimalar Engineering College, Chennai, India
n.prabusankar81@gmail.com
[4]Assistant Professor
Department of computer science and engineering
Sri Venkateswara College of Engineering, Sriperumbudur,
rgvelu22@gmail.com
[5]Assistant Professor, Information Technology,
Panimalar Engineering College, Chennai, India.
monykrishnaraj@gmail.com
[6]Assistant Professor, Information Technology,
Panimalar Engineering College, Chennai, India.
irinkutty@gmail.com

Abstract— In recent times, the telecommunications request has beenveritably competitive. The cost of retaining telecom guests is lower thanattracting new guests. A telecom company must understand client churn through client relationship management. Therefore, CRM analyzersare demanded to prognosticate which guests will change. In this design, aclient abandonment vaticination model was proposed that uses the Decision Treealgorithm, Random timber algorithm, and Deep literacy algorithm to identify thechurn guests. In Deep Literacy, a Multilayer Perceptron Neural Network has beenused to produce the vaticination model. The performance of all the algorithmswill be compared and estimated in terms of delicacy. The advanced performancevaticination model can be used in the telecom sphere for prognosticatingwhether the guests will churn or not. By knowing the significant churn factorsfrom client data, CRM can ameliorate productivity, recommend applicableelevations to the group of likely churn guests grounded on analogous getspatterns, and exorbitantly ameliorate marketing juggernauts of the company.

Keywords- Customer, Churn Prediction, Telecom Industry, Deep Learning, Random Forest, Multilayer Perceptron

I. INTRODUCTION

Data Science is one of the utmost motivating and vital areas of exploration with the end of rooting information from the tremendous quantum of accumulated data sets. Going on is a period of simplifying nearly all complicated workshops using computers. Homemade processing makes the process slow and other problems are similar to inconsistency and nebulosity on operations.

In the present world, a huge volume of data is being generated by telecom companies at an exceedingly fast rate. There's a range of telecom service providers contending in the request to increase their customer share. Guests have multiple options in the form of better and less precious services. The ultimate thing of telecom companies is to maximize their profit and stay alive in a competitive marketplace. Owing to fierce competition among telecom companies, client churn is ineluctable. Client churn is the act of a client ending a subscription to a service provider and choosing the services of another company.

Companies must reduce client churn because it weakens the company. A check showed that the periodic churn rate in the telecom assiduity ranges from 20 to 40, and the cost of retaining guests is 5 – 10 times lower than the cost of carrying new guests. The cost of prognosticating churn guests is 16 times lower than that for carrying new guests. Dwindling the churn rate by 5 increases the profit from 25 to 85. This shows that client-churn vaticination is important for the telecom sector. Telecom companies consider client relationship operation (CRM) an important factor in retaining guests and precluding client churn. To retain guests, CRM analyzers must prognosticate which guests will churn and dissect the reasons for client churn.

The thing of this design is to give results to client churn using a data science algorithm. For vaticination, Decision Tree algorithms, Random timber algorithm, and Multilayer Perceptron is used to prognosticate the client churn using factors, similar as voice_mail_plan, number_vmail_messages, total_day_minutes, total_day_calls, total_day_ charge, total_eve_ minutes, total_eve_calls,

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

total_eve_charge, total_night_minutes, total_night_calls, total_night_charge, total_intl_minutes, total_intl_calls,total_ intl_charge, number_customer_service_callsetc. The performance of Decision Tree, Random Forest, and Multilayer Perceptron algorithms have been compared in terms of delicacy. The stylish model will be proved to be largely salutary to the telecom assiduity. Once the at-threat guests are linked, the company must perform marketing juggernauts for churn guests to maximize churn-client retention.

II. SYSTEM ANALYSIS

A. Existing System

The existing system has the problem of Overfitting. Because training data will provide accurate results, for testing data it does not produce an accurate result. But using data science and deep learning techniques, the Overfitting problems will be avoided and it generates accurate results for training and testing data. The existing concept deals with Qualitative observations and simple statistical analysis. The qualitative observations deal with the data that can be observed through human senses. They do not involve measurements or numbers. When the data set has missing values, the existing system will not generate accurate results. If the input data is imbalanced data, that is more datasets under one classification and fewer datasets under another classification, then the existing system will not generate accurate results.

B. Proposed System

Churn prediction is vital in the telecom sector as telecom operators have to retain their valuable customers. The Telecom customer churn dataset is collected and analyzed to predict customer churn. The motive of this project is to design a model that can prognosticate the churn customers with maximum accuracy. For classification, Data Science techniques of Decision Tree, Random Forest algorithm, and Deep Learning techniques of Multilayer Perceptron algorithms have been taken and the performance of the algorithms have been compared. Data Science is an interdisciplinary field that incorporates computer science, mathematics, statistics, and domain knowledge. Data Science is a process of doing analysis such as frequent pattern mining, Association rule mining, classification, clustering, regression, etc. to find the intelligent patterns hidden in a huge number of datasets. Using Data Science, more accurate classification results will be obtained. The Data Science algorithm is important because, it can analyze bigger, more complex data and deliver faster more accurate results even on a larger scale and also because so many different industries are starting to rely on Data Science. The proposed system takes care of various issues in the missing values. In the imbalanced data set, the performance of the algorithm is more accurate. It is more flexible and easier to understand and debug. Using Deep Learning algorithms, incredible accuracy will be obtained, because, till the model reaches the accuracy, the error will be back propagated into the network and once again for the new value of weights and bias, the target value will be calculated. It automates repetitive learning; it will learn the information from the dataset automatically. It analyses more and deeper data using a neural network. The proposed system can handle large amounts of data.

III. SYSTEM ARCHITECTURE

The figure below shows the architectural illustration of this design defines the inflow of data for prognosticating the client churn. There a colorful ways for prognosticating whether the client will churn or not similar to, data importing, data preprocessing, bracket using the Decision Tree algorithm, Bracket Using Random Forest Algorithm, and bracket using Multilayer Perceptron. The performance of the Decision Tree, Random Forest, and Multilayer Perceptron have been compared.

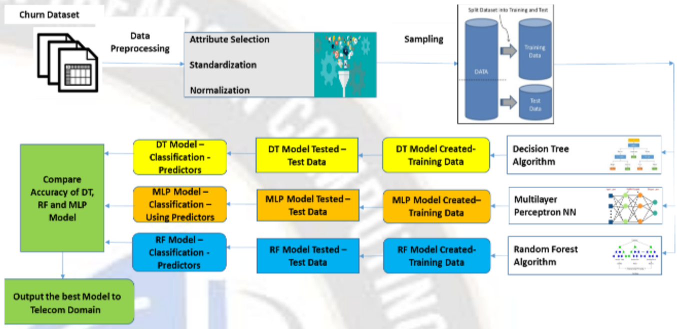


Figure 1. Architecture diagram

The first step is data importing, Data have to be loaded into the Python terrain for analysis. The churn data set is loaded into the terrain for vaticination operation. Also, the packages necessary for Decision Tree, Random Forest, and Multilayer Perceptron algorithm have to be loaded into Python terrain. For Decision Tree algorithm, Decision Tree Classifier modules from sklearn package, Random Forest algorithm, Random Forest Classifier modules from sklearn package, and for Multilayer Perceptron, TensorFlow, keras models, keras. layers loaded into Python terrain. In data pre-processing, trait selection, standardization, and normalization functions will be applied. In standardization, raw data is converted into a common, accessible format. In trait selection, hold only the attributes which is affecting the analysis and it isn't necessary to hold all the attributes for doing the analysis. In Normalization, the mean of the trait will be 0, and standard divagation will be 1. In the churn dataset, the variables are on different scales. This difference in scale can begets problems in training our neural network, as variables with larger scales tend to dominate variables with lower scales. The dataset has regularized and the preprocessed data is given to slice, In slice, the dataset is divided into 2 sets, a training dataset and a testing dataset with the probability of 80 and 20. The training data set is passed onto the Decision Tree algorithm, the DT model is created, also the DT model is tested using test data, when only the predictors are given to the DT model it'll induce the target variable. The training data set is passed onto the Multilayer Perceptron algorithm, also the MLP model have been created, also the MLP model is tested using test data, when only the predictors are given to the MLP model it'll induce the target variable. The training data set is passed onto the Decision Tree algorithm, then the DT model have been created, and then the DT model is tested using test data, when

ISSN: 2321-8169 Volume: 11 Issue: 9
Article Received: 25 July 2023 Revised: 12 September 2023 Accepted: 30 September 2023

only the predictors are given to the DT model it will generate the target variable. The performance of the Decision Tree, Random Forest, and Multilayer Perceptron algorithm is compared in terms of accuracy. The confusion matrix is created to find the accuracy of the model. The best model will be given to the telecom industry to predict whether the customer will churn or not.

A. LEVEL 0 for choosing dataset labels

In figure 2, Data is imported into a python environment for analysis. The input dataset is the churn dataset, which is in CSV format and has to be loaded into the program to start the analysis. Package and modules necessary for Decision Tree, Random Forest algorithm, and Multilayer perceptron algorithm have to be installed and loaded into the program.

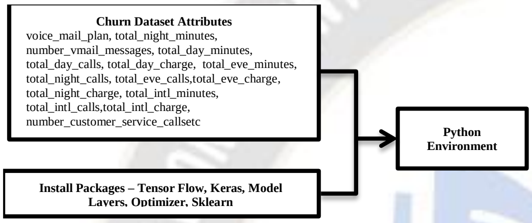


Figure2. Churn Data Importing DFD

Figure 3 shows the Sampling of the dataset. The churn dataset is divided into two sets of samples, one for training data and the other for testing data with the probability of 80% and 20%. The training data is passed on to the Decision Tree, Random Forest, and Multilayer Perceptron algorithm to create the model. Using test data, the model will be validated.

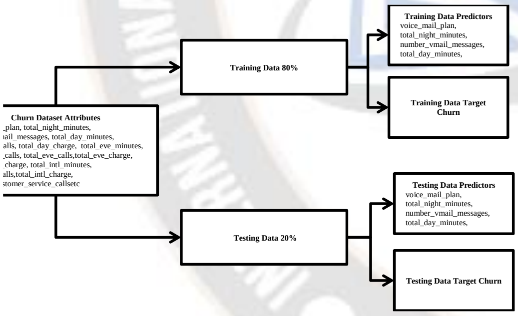


Figure 3. Data Sampling DFD

LEVEL 1

In figure4, preprocessed data is given as input to the Decision Tree algorithm and a Decision Tree model has been created, which is used for the classification and prediction of customer churn.

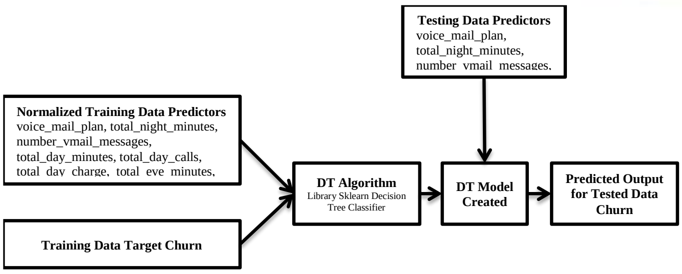


Figure 4. Decision Tree algorithm

B. LEVEL 2

In Figure 5, Preprocessed data is given as input to the Random Forest algorithm and an RF model has been created, which is used for the prediction of customer churn.

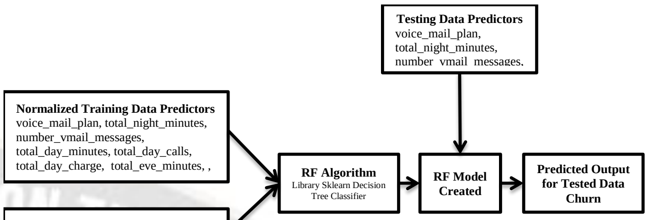


Figure 5. Random Forest algorithm

C. LEVEL 3

In figure 6, preprocessed data is given as input to the Multilayer Perceptron algorithm and an MLP model has been created, which is used for classification and prediction.

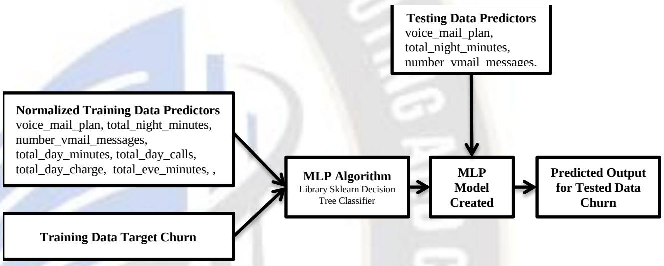


Figure 6. MLP Algorithm DFD

D. LEVEL 4

In Figure 6, the performance of Decision Tree, Random Forest, and Multilayer Perceptron is compared using the accuracy of the classification model. Decision Tree Algorithm predicted the customer churn with the accuracy of 84%, Random Forest with an accuracy of 93%, and Multilayer Perceptron with an accuracy of 89%. The experimental results show that the performance of Random Forest is more accurate when compared to Decision Tree and Multilayer Perceptron algorithm for this churn dataset.

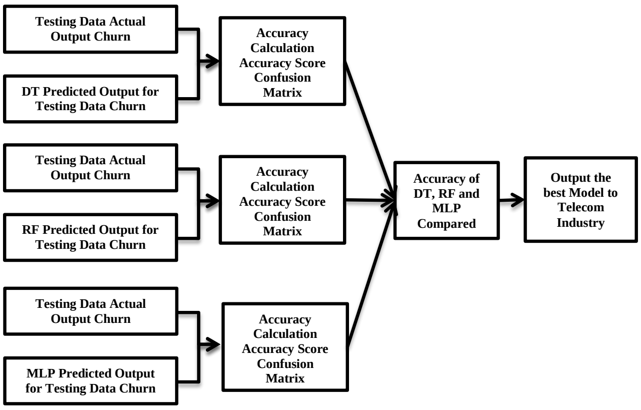


Figure 7. Performance Comparison of DT, RF and MLP

Overall working process of Churn Prediction

Figure 7 shows that the INPUT churns dataset into Python Environment. After giving the dataset to the system, a data sampling process will take place, with Training data at 80% and Test data at 20% as input to the system. The next process is Normalization; reducing the redundancy values and other missing contents in the preprocessing stage. Apply AI, MLP & RF algorithms separately to get the predicted values are separate. Finally, compare all three algorithms using their performance. Final outcome will be taken from the best prediction among the three algorithms.

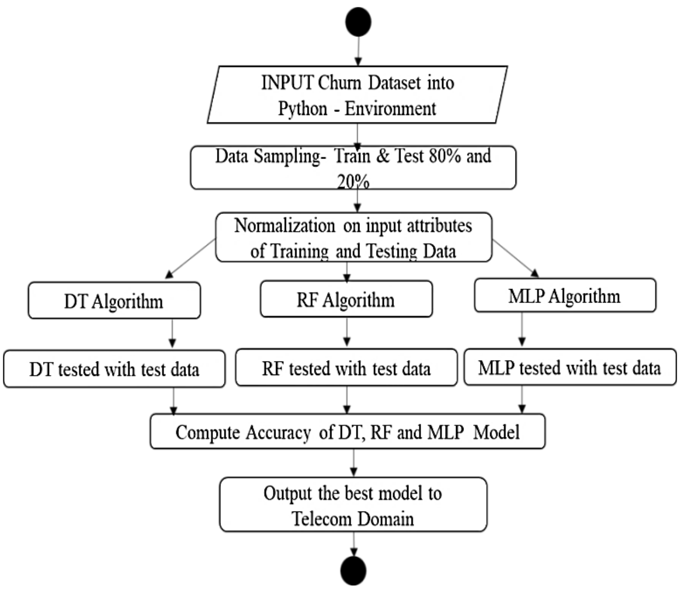


Figure8. Overall working process of Churn Prediction

System specification

Python is a vastly used high position programming language for general- purpose programming, An interpreted language, Python has a design gospel which emphasizes law readability, and a syntax which allows programmers to express generalities in lower lines of law than possible in languages similar as C or Java. Python features a dynamic type system and automatic memory operation and supports multiple programming paradigms including object- acquainted, imperative, functional programming, and procedural styles.

Modules for creating graphical stoner interfaces, connecting to relational databases, generating pseudorandom figures, computation with arbitrary perfection numbers, manipulating regular expressions, and doing unit testing are also included. The Python Package Index, the sanctioned depository containing third- party software for Python, contains over packages offering a wide range of functionality, including, Graphical stoner interfaces, web fabrics, multimedia, databases, networking and dispatches, test fabrics and web scraping, attestation tools, system administration, Scientific computing, textbook processing.

IV. MODULES

There are five Modules

- Data Importing and Preprocessing
- Classification Model Using Decision Tree Algorithm
- Classification Model Using Multilayer Perceptron Neural Network
- Classification Model Using Random Forest Algorithm

Performance Comparison of DT, RF and MLP model

A. Data importing and preprocessing

Data is available in any train in the format of CSV. Data have to be loaded into python terrain for analysis. Once data have been uprooted from the train it should be stored in a data frame. Libraries necessary for the bracket algorithm – Decision Tree, Random Forest, and Multilayer Perceptron algorithm have to be installed into the Python terrain. For the Decision Tree algorithm, the Decision Tree Classifier module, for the Random Forest algorithm, the Random Forest Classifier, and the Multilayer Perceptron, TensorFlow, Keras, and Keras. Models, keras. Layers etc. have to bein stalled and loaded into Python terrain. Data preprocessing is the data mining fashion that involves transubstantiating raw data into an accessible format. The raw data is largely susceptible to noise, missing values, and inconsistency. Real-world data is frequently deficient, and inconsistent, and is likely to contain crimes. The missing Values problem has to be answered in simple statistical ways. Data preprocessing is a proven system for resolving similar issues. In order to ameliorate the quality of the data accordingly, the mining results of raw data is preprocessed so the effectiveness process is bettered. It isn't necessary to hold all the attributes for doing the analysis; we can hold only the attributes which is affecting the analysis. Data standardization is the process by which analogous data is collected in colorful formats is converted to a common format that enhances the comparison process and allows for cooperative exploration and large-scale analytics. Normalization is a scaling fashion in which values are shifted and rescaled so that they end up ranging between 0 and 1. It's also known as Min-Max scaling. After normalization, the mean of the trait is 0, and standard divagation is 1. Each and every trait is treated inversely; no single trait will dominate during model creation. Normalization is used to gauge the data of a trait so that it falls in a lower range, similar to as-1.0 or to 1.0 or0.0 to1.0. It's generally useful for bracket algorithms. when multiple attributes are there but attributes have values on different scales, this may lead to poor data models while performing data mining operations. So they're regularized to bring all the attributes on the same scale.

B. Classification model using Decision Tree algorithm

Classification is a method where we classify data into a given number of classes. Classification is to find out the extent to which a thing will be or will not be a part of a group or type. Classification can be work on either structured or unstructured data. The main goal of a classification problem is to identify the category/class to which new data will fall.

Decision trees are more important and popular tools for classification and prediction. J. Ross Quinlan originally developed ID3 at the University of Sydney. ID3 stands for

Iterative Dichotomizer 3 and is named because the algorithm iteratively (repeatedly) dichotomizes (divides) features into two or more groups at each step.ID3 uses Entropy and Information Gain to build a decision tree. A decision tree is one of the classifiers in the form of a tree structure. The root node represents the entire population or sample and this further gets divided into two or more homogeneous sets.

Splitting is a process of dividing a node into two or more sub-nodes. When a sub-node splits into further sub-nodes, then it is called a decision node. Nodes that do not split is called Leaf or Terminal node. When we remove sub-nodes of a root node, this process is called pruning. It is the opposite process of splitting. A sub-section of the entire tree is called a branch or sub-tree.

Algorithmic steps of Decision Tree algorithm

Step 1: Calculate entropy of the target.

Step 2: The dataset is then split on the different attributes. The entropy for each branch is calculated. Then it is added proportionally, to get total entropy for the split. The resulting entropy is subtracted from the entropy before the split. The result is the Information Gain

Step 3: Choose attribute with the largest information gain as the decision node, divide the dataset by its branches and repeat the same process on every branch. Outlook is having max gain, so tree has to start with outlook.

Step4a: A branch with entropy of 0 is a leaf node.
Step4b: A branch with entropy more than 0 needs splitting
The Previous steps have to be repeated till all the data has been classified.

A node, which is divided into sub-nodes is called parent node of sub-nodes whereas sub-nodes are the child of parent node. The entropy is a measure of the uncertainty or impurity associated with a random variable. Values range from 0 – 1 to represent the entropy of information

$$\mathrm{Entropy}(s) = \sum_{i=1}^{c} -p_i \log_2 P_i$$

Information gain is a measure of the efficacy of an attribute in classifying the training data. Information gain is used as an attribute selection measure. Select the attribute which has the highest Information Gain among others.

$$Gain(S,A) = Entropy(S) - \sum_{v \in Value(A)} \frac{|S_v|}{|S|} Entropy(S_v)$$

Decision tree algorithm is faster than other traditional statistical techniques. It is very flexible, easy to implement and debug. It handles both continuous and discrete set of data. It is used for both binary and multiclass classification. Once the pre-processed data is applied to the Decision tree algorithm, Decision tree Model have been created, which is used to predict the customer churn. Decision tree model output can be visualized by classification rules and binary classification tree.

V. CLASSIFICATION MODEL USING RF ALGORITHM

Random Forest algorithms can be used for both Classifications as well as Regression algorithms. This algorithm was first developed by Leo Brieman and Adele Cutler in 2001. It creates many numbers of decision trees. Every new observation is fed into all the trees and acquires multiple results from a single input. Random forest will use the superiority of votes from all the decision trees to divide data or use an average output for regression.

Random forest is a tree-based algorithm that involves building several trees (decision trees), and then combining their output to improve the generalization ability of the model. The method of merging trees is known as an ensemble method. Ensembling is nothing but a combination of weak learners (individual trees) who perform slightly better than random guessing to produce a strong learner who has arbitrarily good accuracy. In the regression algorithm, the measured variable is continuous. In classification problems, the predicted variable is categorical.

In regression trees, the splitting decision is based on minimizing RSS. The variable that leads to the greatest possible reduction in RSS is chosen as the root node. The tree splitting takes a top-down greedy approach, also known as recursive binary splitting.

This algorithm is called "greedy" because the algorithm cares about making the best split at the current step rather than saving a split for better results on future nodes. It takes less training time as compared to the other algorithms. It predicts output with higher accuracy, and even runs efficiently for the large dataset. It can also maintain accuracy when a large portion of data is lost and thus intercept the overfitting issue. In classification trees, the splitting decision is based on the Gini Index - It is a measure of node purity. If the Gini index takes on a smaller value, it suggests that the node is clear. For a splitting of the tree, the Gini index for a child node should be less than that for the parent node.

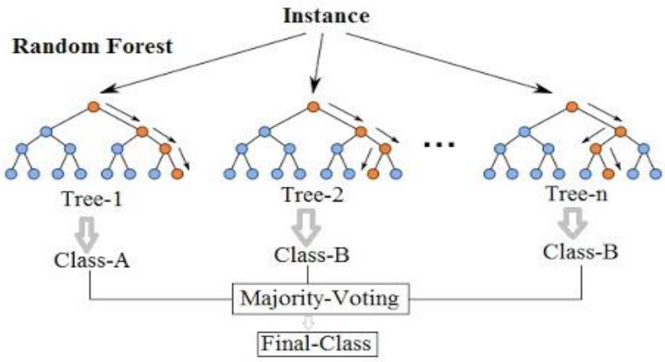


Figure 10. Tree formation by Random Forest algorithm

Algorithmic steps of the Random Forest Algorithm

1. First, start with the selection of random samples from a given dataset.

2. Random Forest algorithm constructs a decision tree for every sample. Then it will get the prediction result from every decision tree.
3. Voting will be performed for every predicted result.
4. At last, select the most voted prediction result as the final prediction result.

The mean decrease in Gini coefficient is a measure of how each variable contributes to the homogeneity of the nodes and leaves in the resulting random forest. Same as information gain. Variable with higher MDG index is the most important variable. Entropy - Entropy is a measure of node impropriety

Entropy = - p(a)*log(p(a)) - p(b)*log(p(b))

Random Forest is best in accuracy among present algorithms and runs effectively on large databases. It can handle thousands of input variables without any variable deletion and gives an estimation of which variables are important in the classification. The created forests can be saved for future use on other data. Prototypes are computed that give information about the relation between the variables and the classification.

It can perform both regression and classification tasks. A random forest produces good predictions that can be understood easily. It can handle large datasets efficiently. The random forest algorithm provides a higher level of accuracy in predicting outcomes than the decision tree algorithm. When using a random forest, more resources are required for computation. It consumes more time compared to a decision tree algorithm.

Random forest regression is not ideal in the prediction of data. Unlike linear regression, which uses obtained study to estimate values beyond the observation range? This explains why most applications of random forest relate to classification. Random forest does not produce good results when the data is very scattered. In this case, the subset of features and the rebooting sample will produce a uniform space. This will lead to poor splits, which will affect the outcome. Once the pre-processed data is applied to the Random Forest algorithm, the Random Forest Model has been created using the training dataset, and the RF model is evaluated using test data. Then the predictors are passed to the model, it will give the output of whether the customer will churn or not.

VI. CLASSIFICATION MODEL USING MULTILAYER PERCEPTRON

Deep Learning is implemented through Neural Networks and the motivations behind Neural Network is the biological Neurons. Neural Networks are modeled as of neurons that are connected in an acyclic graph. Neural Network models are often organized into distinct layers of neurons. A perceptron is a neural network unit (an artificial neuron) that does certain functions to detect features or business intelligence in the input data. The capacity of neural networks comes from their ability to learn the representation in your training data and how to best relate it to the output variable that you want to predict.

A multilayer perceptron (MLP) is a perception that join with added perceptron's, stacked in several layers, to solve difficult problems. The diagram below shows an MLP with four layers. The building block for neural networks is artificial neurons. These are simple computational units that have weighted input signals and produce an output signal using an activation function. Weights are often initialized to small random values, such as values in the range 0 to 0.3, although more complex initialization schemes can be used.

Weight is the parameter in a neural network that converts input data inside the network's hidden layers. Weights are connected with each neuron. A weight is a real number; it can be either positive or negative. The weighted inputs are summed up and passed through an activation function, sometimes called a transfer function.

$$z=\sum_{i=1}^{m} w_i x_i + b$$

An activation function is a simple depiction of added weighted input to the output of the neuron. It is called an activation function because it commands the threshold at which the neuron is activated and strength of the output signal. The activation function is a non-linear transformation that we do over the input before sending it to the next layer of neurons or finalizing it as output. The output of the neuron network is finally established

The activation function compresses the output of the net input function (z) into a new output range depending on the choice of activation function. Multilayer Perceptron is a fully connected multi-layer neural network. The first layer is the input layer and last will be output layer and the in between middle layers will be called as hidden layer. Feed the input data into the input layer and take the output from the output layer. The number of hidden layers can be increased as much as we needed, to make the model more complex as per the task taken.

When working with neural networks, the two main topics are: forward propagation and backpropagation. Forward propagation is the process through which data passes through layers of neurons in a neural network from the input layer all the way to the output layer. Training is done in a supervised learning setting, where each data point has a corresponding label or ground truth. Training is required when the predicted value by the neural network clearly does not match the ground truth for a given input. The training process starts by calculating the error E between the predicted values and the ground truth labels.

The next step would be to propagate this error back into the network and use it to perform gradient descent on the different weights and biases in the network. This epoch of the training process is mention in the figure10. To shows the multilayer perceptron, with input layer, output layer and two hidden layers.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

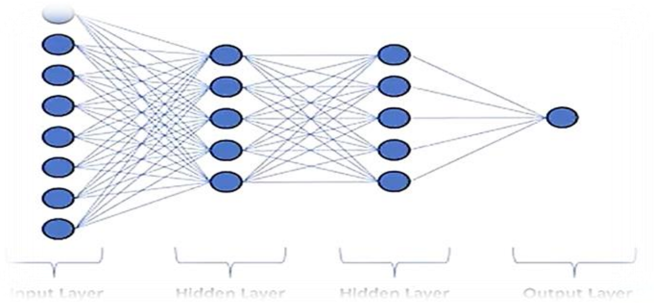


Figure11. Multilayer Perceptron

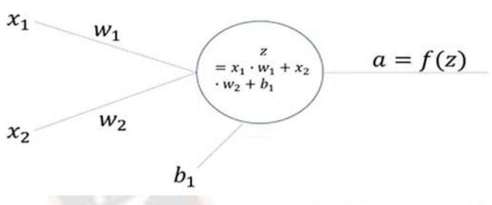
Algorithmic Steps of Multilayer Perceptron

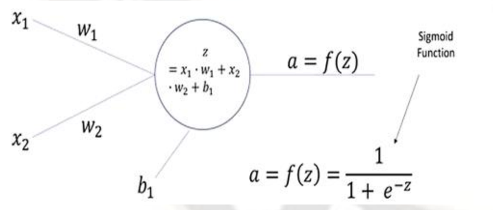
Initialize the weights and biases to random values.

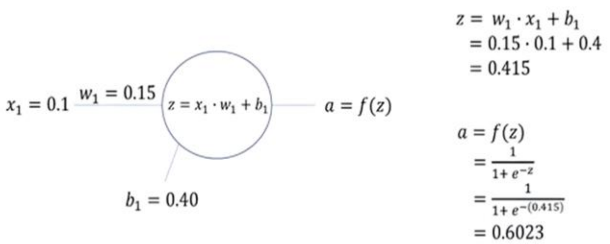
Iteratively repeat the following steps:

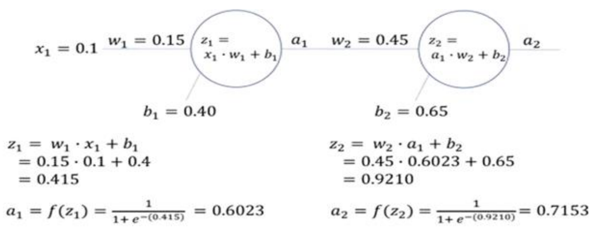
1. Calculate the network output using forward propagation.
2. Calculate the error between the ground truth and the estimated or predicted output of the network.
3. Update the weights and the biases through back propagation.
4. Repeat the above three steps until the number of iterations or epochs is reached or the error between the ground truth and the predicted output is below a predefined threshold.

Forward propagation is the process through which data passes through layers of neurons in a neural network from the input layer all the way to the output layer.





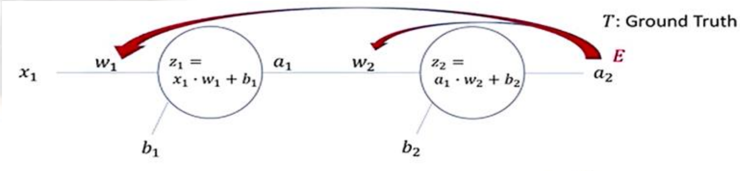




In Backpropagation, Calculate the error between the ground truth and the estimated output. Let's denote the error by E. Propagate the error back into the network and update each weight and bias as per the following equation

$$w_i \to w_i - \eta \cdot \frac{\partial E}{\partial w_i}$$

$$b_i \to b_i - \eta \cdot \frac{\partial E}{\partial b_i}$$



1. Calculate the error: $E = \frac{1}{2}(T - a_2)^2$ $\qquad E = \frac{1}{2m}\sum_{i=1}^{m}(T_i - a_{2,i})^2$
2. Update w_2, b_2, w_1, and b_1

Backpropagation – Updating W2

$$w_2 \to w_2 - \eta \cdot \frac{\partial E}{\partial w_2}$$

$$E = \frac{1}{2}(T - a_2)^2 \longrightarrow \frac{\partial E}{\partial a_2}$$

$$a_2 = f(z_2) = \frac{1}{1+e^{-z_2}} \longrightarrow \frac{\partial a_2}{\partial z_2}$$

$$z_2 = a_1 \cdot w_2 + b_2 \longrightarrow \frac{\partial z_2}{\partial w_2}$$

$$\frac{\partial E}{\partial w_2} = \frac{\partial E}{\partial a_2} \cdot \frac{\partial a_2}{\partial z_2} \cdot \frac{\partial z_2}{\partial w_2}$$

$$= \big(-(T - a_2)\big) \cdot \big(a_2(1 - a_2)\big) \cdot (a_1)$$

$$w_2 \to w_2 - \eta \cdot \big(-(T - a_2)\big) \cdot \big(a_2(1 - a_2)\big) \cdot (a_1)$$

Eqn # 1: $z_1 = x_1 \cdot w_1 + b_1$

Eqn # 2: $a_1 = f(z_1) = \frac{1}{1+e^{-z_1}}$

Eqn # 3: $z_2 = a_1 \cdot w_2 + b_2$

Eqn # 4: $a_2 = f(z_2) = \frac{1}{1+e^{-z_2}}$

Eqn # 5: $E = \frac{1}{2}(T - a_2)^2$

$$\frac{\partial E}{\partial w_2} = \frac{\partial E}{\partial a_2} \cdot \frac{\partial a_2}{\partial z_2} \cdot \frac{\partial z_2}{\partial w_2}$$

- $E = \frac{1}{2}(T - a_2)^2$
  $\frac{\partial E}{\partial a_2} = 2 \cdot \frac{1}{2}(T - a_2) \cdot (-1) = -(T - a_2)$
- $a_2 = \frac{1}{1+e^{-z_2}} = (1 + e^{-z_2})^{-1}$
  $\frac{\partial a_2}{\partial z_2} = -1 \cdot (1 + e^{-z_2})^{-2} \cdot e^{-z_2} \cdot (-1)$
  $= \frac{e^{-z_2}}{(1 + e^{-z_2})^{2}} = (a_2)^2 \frac{1 - a_2}{a_2} = a_2(1 - a_2)$
- $z_2 = a_1 \cdot w_2 + b_2$
  $\frac{\partial z_2}{\partial w_2} = a_1$

$$\frac{\partial E}{\partial w_2} = -(T - a_2) \cdot a_2(1 - a_2) \cdot a_1$$

Backpropagation – Updating b_2

$$b_2 \rightarrow b_2 - \eta \cdot \frac{\partial E}{\partial b_2}$$

$$\frac{\partial E}{\partial b_2} = \frac{\partial E}{\partial a_2} \cdot \frac{\partial a_2}{\partial z_2} \cdot \frac{\partial z_2}{\partial b_2}$$

$$= \left(-(T - a_2)\right) \cdot \left(a_2(1 - a_2)\right) \cdot 1$$

$$b_2 \rightarrow b_2 - \eta \cdot \left(-(T - a_2)\right) \cdot \left(a_2(1 - a_2)\right) \cdot 1$$

Eqn # 1: $z_1 = x_1 \cdot w_1 + b_1$

Eqn # 2: $a_1 = f(z_1) = \frac{1}{1 + e^{-z_1}}$

Eqn # 3: $z_2 = a_1 \cdot w_2 + b_2$

Eqn # 4: $a_2 = f(z_2) = \frac{1}{1 + e^{-z_2}}$

Eqn # 5: $E = \frac{1}{2}(T - a_2)^2$

$$\frac{\partial E}{\partial b_2} = \frac{\partial E}{\partial a_2} \cdot \frac{\partial a_2}{\partial z_2} \cdot \frac{\partial z_2}{\partial b_2}$$

- $E = \frac{1}{2}(T - a_2)^2$
  $$\frac{\partial E}{\partial a_2} = 2 \cdot \frac{1}{2}(T - a_2) \cdot (-1) = -(T - a_2)$$
- $a_2 = \frac{1}{1 + e^{-z_2}} = (1 + e^{-z_2})^{-1}$
  $$\frac{\partial a_2}{\partial z_2} = -1 \cdot (1 + e^{-z_2})^{-2} \cdot e^{-z_2} \cdot (-1)$$
  $$= \frac{e^{-z_2}}{(1 + e^{-z_2})^{2}} = (a_2)^2 \cdot \frac{1 - a_2}{a_2} = a_2(1 - a_2)$$
- $z_2 = a_1 \cdot w_2 + b_2$
  $$\frac{\partial z_2}{\partial b_2} = 1$$

$$\frac{\partial E}{\partial b_2} = -(T - a_2) \cdot a_2(1 - a_2) \cdot 1$$

Once MLP model have been created, By using testing data, model can be evaluated. By giving only predictor variables, target variable can be predicted.

VII. PERFORMANCE COMPARISON OF DT, RF AND MLP

Once the classification model is completed, it is necessary to evaluate its performance. The performance of the Decision Tree, Random Forest and Multilayer Perceptron models are compared in terms of accuracy. The confusion matrix is created to calculate the accuracy of the classification model. The confusion matrix provides us a matrix/table as output and describes the performance of the model. It is also known as the error matrix. The matrix consists of predictions result in a summarized form, which has a total number of correct predictions and incorrect predictions. The matrix looks like as below table:

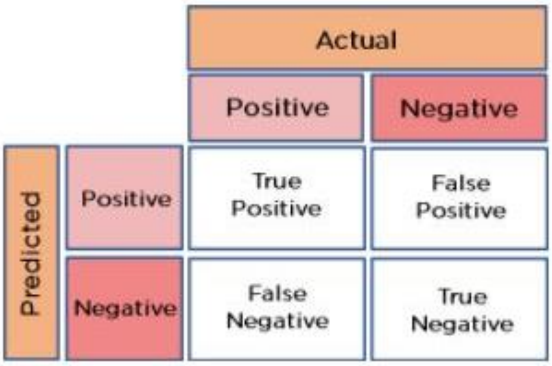


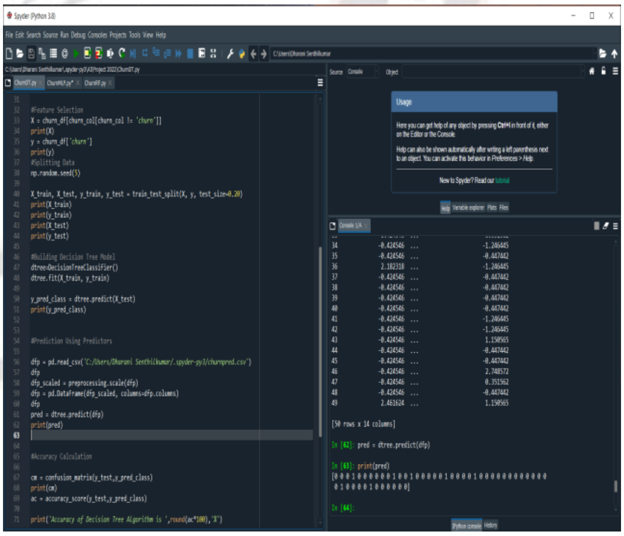
Figure 12. Confusion Matrix

Accuracy = (TP+TN)/(TP+FP+FN+TN)

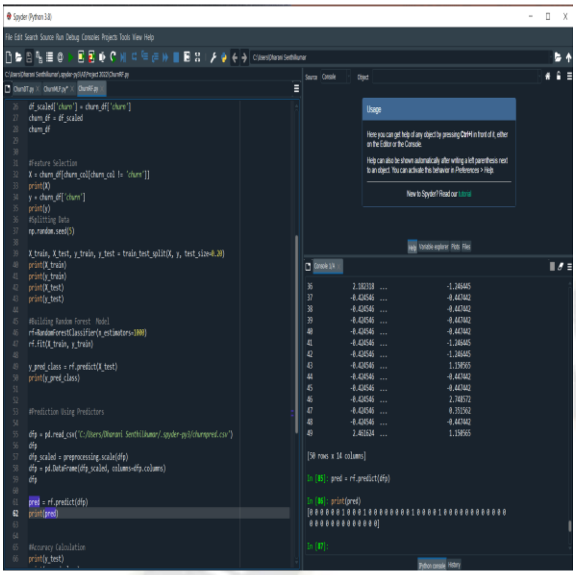
Decision Tree, Random Forest algorithms and MLP algorithm is used to classify and predict the customer churn using factors, such as voice_mail_plan, number_vmail_messages, total_day_minutes, total_day_calls, total_day_charge total_eve_minutes, total_eve_calls, total_intl_minutes, total_intl_calls, total_intl_charge, number_customer_service_calls etc. The performance of both DT, RF and MLP Algorithms will be compared in terms of accuracy.

The Decision Tree model created with 84% accuracy, Random Forest with the accuracy of 93% and Multilayer perceptron model created with the accuracy of 89%. So, the Random Forest algorithm is performing better when compared to Decision Tree algorithm and Multilayer Perceptron for this dataset. This best model will be used in telecom industry for predicting customer churn.

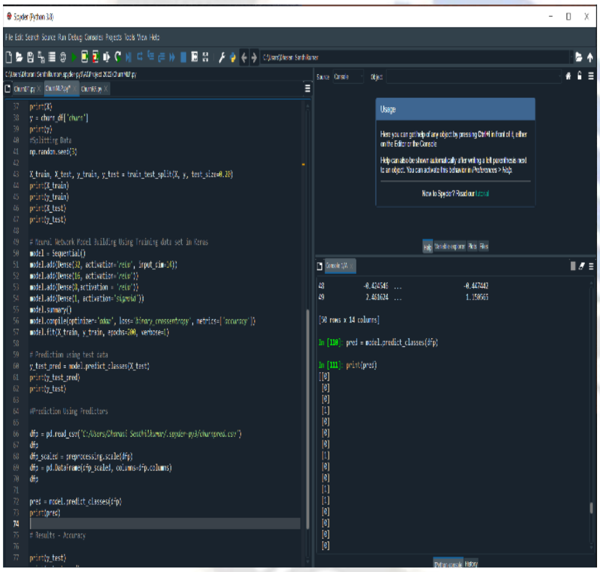
A. DT model Classification Using Predictors



B. RF model Classification Using Predictors



C. MLP model Classification Using Predictors



VIII. SOURCE CODE

```
# Churn Prediction Using Decision Tree Algorithm
# Load in the data set \n",
churn_df = pd.read_csv('C:/Users/Dharani
Senthilkumar/.spyder-py3/churn.csv')
churn_df
print(churn_df.shape)
churn_df.sample(10)

churn_col = churn_df.columns
print(churn_col)
churn_df.isnull().any()

# Data Preprocessing
df_scaled = preprocessing.scale(churn_df)
df_scaled
df_scaled = pd.DataFrame(df_scaled,
columns=churn_df.columns)
df_scaled['churn'] = churn_df['churn']
churn_df = df_scaled
churn_df

#Feature Selection
X = churn_df[churn_col[churn_col != 'churn']]
print(X)
y = churn_df['churn']
print(y)
#Splitting Data
np.random.seed(5)
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.20)
print(X_train)
print(y_train)
print(X_test)
print(y_test)

#Building Decision Tree Model
dtree=DecisionTreeClassifier()
dtree.fit(X_train, y_train)
y_pred_class = dtree.predict(X_test)
print(y_pred_class)

#Prediction Using Predictors
dfp = pd.read_csv('C:/Users/Dharani Senthilkumar/.spyder-
py3/churnpred.csv')
dfp
dfp_scaled = preprocessing.scale(dfp)
dfp = pd.DataFrame(dfp_scaled, columns=dfp.columns)
dfp
pred = dtree.predict(dfp)
print(pred)

#Accuracy Calculation
cm = confusion_matrix(y_test,y_pred_class)
print(cm)
ac = accuracy_score(y_test,y_pred_class)
print('Accuracy of Decision Tree Algorithm is
',round(ac*100),'%')

## Churn Prediction Using Multilayer Perceptron
import pandas as pd
from sklearn.model_selection import train_test_split
from keras.models import Sequential
from keras.layers import Dense
from sklearn import preprocessing
import numpy as np
churn_df = pd.read_csv('C:/Users/Dharani
Senthilkumar/.spyder-py3/churn.csv')
churn_df
print(churn_df.shape)
churn_df.sample()
churn_col = churn_df.columns
print(churn_col)
churn_df.isnull().any()
```

```
# Data Preprocessing
df_scaled = preprocessing.scale(churn_df)
df_scaled
df_scaled = pd.DataFrame(df_scaled,
columns=churn_df.columns)
df_scaled['churn'] = churn_df['churn']
churn_df = df_scaled
churn_df


#Feature Selection
X = churn_df[churn_col[churn_col != 'churn']]
print(X)
y = churn_df['churn']
print(y)
#Splitting Data
np.random.seed(3)
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.20)
print(X_train)
print(y_train)
print(X_test)
print(y_test)
# Neural Network Model Building Using Training data set in
Keras
model = Sequential()
model.add(Dense(32, activation='relu', input_dim=14))
model.add(Dense(16, activation='relu'))
model.add(Dense(8,activation = 'relu'))
model.add(Dense(1, activation='sigmoid'))
model.summary()
model.compile(optimizer='adam', loss='binary_crossentropy',
metrics=['accuracy'])
model.fit(X_train, y_train, epochs=200, verbose=1)

# Prediction using test data
y_test_pred = model.predict_classes(X_test)
print(y_test_pred)
print(y_test)

#Prediction Using Predictors
dfp = pd.read_csv('C:/Users/Dharani Senthilkumar/.spyder-
py3/churnpred.csv')
dfp
dfp_scaled = preprocessing.scale(dfp)
dfp = pd.DataFrame(dfp_scaled, columns=dfp.columns)
dfp
pred = model.predict_classes(dfp)
print(pred)

# Results - Accuracy
print(y_test)
print(y_test_pred)
scores = model.evaluate(X_test, y_test)
print("Accuracy of Multilayer Perceptron: %.0f%%\n" %
(scores[1]*100))

# Churn Prediction Using Random Forest Algorithm
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix, accuracy_score
from sklearn.ensemble import RandomForestClassifier
from sklearn import preprocessing

import numpy as np
# Load in the data set \n",
churn_df = pd.read_csv('C:/Users/Dharani
Senthilkumar/.spyder-py3/churn.csv')
churn_df
print(churn_df.shape)
churn_df.sample(10)
churn_col = churn_df.columns
print(churn_col)
churn_df.isnull().any()

# Data Preprocessing
df_scaled = preprocessing.scale(churn_df)
df_scaled
df_scaled = pd.DataFrame(df_scaled,
columns=churn_df.columns)
df_scaled['churn'] = churn_df['churn']
churn_df = df_scaled
churn_df

#Feature Selection
X = churn_df[churn_col[churn_col != 'churn']]
print(X)
y = churn_df['churn']
print(y)
#Splitting Data
np.random.seed(10)
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.20)
print(X_train)
print(y_train)
print(X_test)
print(y_test)
#Building Random Forest  Model
rf=RandomForestClassifier(n_estimators=1000)
rf.fit(X_train, y_train)
y_pred_class = rf.predict(X_test)
print(y_pred_class)
#Prediction Using Predictors
dfp = pd.read_csv('C:/Users/Dharani Senthilkumar/.spyder-
py3/churnpred.csv')
dfp
dfp_scaled = preprocessing.scale(dfp)
dfp = pd.DataFrame(dfp_scaled, columns=dfp.columns)
dfp
pred = rf.predict(dfp)
print(pred)

#Accuracy Calculation
print(y_test)
print(y_pred_class)
cm = confusion_matrix(y_test,y_pred_class)
print(cm)
ac = accuracy_score(y_test,y_pred_class)
```

```
print(ac)
print('Accuracy of Random Forest ALgorithm  is ',round(ac*100),'%')
```

IX. CONCLUSION AND FUTURE ENHANCEMENT

In the present competitive market of telecom domain, churn prediction is a significant issue of the CRM to retain valuable customers by identifying a similar group of customers and providing competitive offers/services to the respective groups. The goal of this project is to predict whether the customers are going to churn from particular telecom industry or not Classification model built using Decision tree algorithm, Random Forest algorithm and Multilayer Perceptron algorithm. The Random Forest is performed with the accuracy of 93%, Decision Tree with the accuracy of 84% and Multilayer Perceptron with the accuracy of 89%. A comparison between the simulated results obtained from these three models demonstrates that the random forest model yields good results. This model helps to retain customers and solve the problems of CRM and decision maker of a company. In future, the designed system of data science classification algorithms can be used in other applications to predict or diagnose the diseases, bank loan approval etc. The work can be extended and improved for the automation of customer churn analysis by including some more data science and Deep learning algorithms.

REFERENCES

[1] Anurag Bhatnagar, et. al., " A Robust Model for Churn Prediction using Supervised Machine Learning", IEEE 9th International Conference on Advanced Computing (IACC), 2019.

[2] Pushkar Bhuse, "Machine Learning Based Telecom-Customer Churn Prediction", IEEE International Conference on Intelligent Sustainable Systems, 2021

[3] Manas Rahman, "Machine Learning Based Customer Churn Prediction", IEEE International Conference on Electronics, Communication and Aerospace Technology, 2020.

[4] Laurie Butgereit, "Work Towards Using Micro-services to Build a Data Pipeline for Machine Learning Applications: A Case Study in Predicting Customer Churn", IEEE International Conference on Innovative Trends in Communication and Computer Engineering, 2020.

[5] Ishpreet Kaur, "Customer Churn Analysis and Prediction in Banking Industry using Machine Learning", IEEE International Conference on Parallel, Distributed and Grid Computing, 2021.

[6] Ahmet Tugrul Bayrak, Asmin Alev Aktas, "Personalized Customer Churn Analysis with Long Short-Term Memory", IEEE International Conference on Big Data and Smart Computing, 2021.

[7] Jas Semrl; Alexandru Matei, "Churn prediction model for effective gym customer retention", IEEE International Conference on Behavioural, Economic, Socio-cultural Computing, 2017

[8] Laurie Butgereit, "Big Data and Machine Learning for Forestalling Customer Churn Using Hybrid Software", IEEE Conference on Information Communications Technology and Society, 2020

[9] B.Prabadevi, R.Shalini, B.R. Kavitha,Customer churning analysis using machine learning algorithms,International Journal of Intelligent Networks, Volume 4,2023,Pages 145-154,ISSN 2666-6030,https://doi.org/10.1016/j.ijin.2023.05.005.

[10] Hemlata Jain, Ajay Khunteta, Sumit Srivastava,Churn Prediction in Telecommunication using Logistic Regression and Logit Boost, Procedia Computer Science,Volume 167,2020,Pages 101-112,ISSN 1877-0509,https://doi.org/10.1016/j.procs.2020.03.187.

[11] Ebrah, K. and Elnasir, S. (2019) Churn Prediction Using Machine Learning and Recommendations Plans for Telecoms. Journal of Computer and Communications, 7, 33-53. doi: 10.4236/jcc.2019.711003.

[12] Khulood Ebrah1*, Selma Elnasir, Churn Prediction Using Machine Learning and Recommendations Plans for TelecomsJournal of Computer and Communications, 2019, 7, 33-53https://www.scirp.org/journal/jcc ISSN Online: 2327-5227 ISSN Print: 2327-5219

[13] Ahmad, A.K., Jafar, A. & Aljoumaa, K. Customer churn prediction in telecom using machine learning in big data platform. J Big Data 6, 28 (2019). https://doi.org/10.1186/s40537-019-0191-6

[14] Suh, Y. Machine learning based customer churn prediction in home appliance rental business. J Big Data 10, 41 (2023). https://doi.org/10.1186/s40537-023-00721-8

[15] Weilong Li1 and Chujin Zhou Published under licence by IOP Publishing Ltd IOP Conference Series: Materials Science and Engineering, Volume 768, Information TechnologyCitation Weilong Li and Chujin Zhou 2020 IOP Conf. Ser.: Mater. Sci. Eng. 768 052070DOI 10.1088/1757-899X/768/5/052070

[16] Ming Zhao , Qingjun Zeng , Ming Chang , Qian Tong , and Jiafu Su, A Prediction Model of Customer Churn considering Customer Value: An Empirical Research of Telecom Industry in China Hindawi Discrete Dynamics in Nature and Society Volume 2021, Article ID 7160527, 12 pages https://doi.org/10.1155/2021/7160527

[17] Lalwani, P., Mishra, M.K., Chadha, J.S. et al. Customer churn prediction system: a machine learning approach. Computing 104, 271–294 (2022). https://doi.org/10.1007/s00607-021-00908-y

[18] Faritha Banu J, Neelakandan S, Geetha BT, Selvalakshmi V, Umadevi A, Martinson EO. Artificial Intelligence Based Customer Churn Prediction Model for Business Markets. Comput Intell Neurosci. 2022 Sep 29;2022:1703696. doi: 10.1155/2022/1703696. PMID: 36238670; PMCID: PMC9552693.

[19] Saha, L.; Tripathy, H.K.; Gaber, T.; El-Gohary, H.; El-kenawy, E.-S.M. Deep Churn Prediction Method for Telecommunication Industry. *Sustainability* 2023, *15*, 4543. https://doi.org/10.3390/su15054543

[20] Mustafa N, Sook Ling L and Abdul Razak SF. Customer churn prediction for telecommunication industry: A Malaysian Case Study [version 1; peer review: 2 approved]. F1000Research 2021, 10:1274

[21] Khattak, A., Mehak, Z., Ahmad, H. et al. Customer churn prediction using composite deep learning technique. Sci Rep 13, 17294 (2023). https://doi.org/10.1038/s41598-023-44396-w

[22] Panimalar, S. A., & Krishnakumar, A. (2023). A review of churn prediction models using different machine learning and deep learning approaches in cloud environment. Journal of Current Science and Technology, 13(1), 136–161. Retrieved from https://ph04.tci-thaijo.org/index.php/JCST/article/view/211

[23] Panimalar, S. A., & Krishnakumar, A. (2023). A review of churn prediction models using different machine learning and deep learning approaches in cloud environment. Journal of Current Science and Technology, 13(1), 136–161. Retrieved from https://ph04.tci-thaijo.org/index.php/JCST/article/view/211

[24] E. Jamalian and R. Foukerdi, "A Hybrid Data Mining Method for Customer Churn Prediction", Eng. Technol. Appl. Sci. Res., vol. 8, no. 3, pp. 2991–2997, Jun. 2018.

[25] Rani, K. Sandhya and ., Shaik Thaslima and ., N.G.L. Prasanna and ., R.Vindhya and ., P. Srilakshmi, Analysis of Customer Churn Prediction in Telecom Industry Using Logistic Regression (JUNE 10, 2021). International Journal of Innovative Research in Computer Science & Technology (IJIRCST) ISSN: 2347-5552, Volume-9, Issue-4, July 2021, https://doi.org/10.21276/ijircst.2021.9.4.6, Available at SSRN: https://ssrn.com/abstract=3902033

[26] A. Mishra and U. S. Reddy, "A Novel Approach for Churn Prediction Using Deep Learning," *2017 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, Coimbatore, India, 2017, pp. 1-4, doi: 10.1109/ICCIC.2017.8524551.