International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

# Telecom Churn Prediction Using an Ensemble Approach with Feature Engineering and Importance

**Hemlata Jain a[1]*, Ajay Khunteta[2], Sumit Srivastava[3]**

**Submitted: 22/07/2022 Accepted: 25/09/2022**

***Abstract –*** In Telecommunication industry, churn prediction is loss of customers and faces fierce competition to retain customers. Churn is the phenomena of a customer leaving a business, and in this context, churn prediction refers to predicting the client's intention to leave. In order to retain customers company needs a good churn prediction model. For a churn prediction model, company needs to predict why customer have churned in past and which factor is most important to predict customers who are near churn. This paper primarily focused on the feature importance and feature engineering for churn prediction model. For classification phase two ensemble models, Random forest and Gradient boosted trees were used. This paper also emphasised on why feature importance and feature engineering are important prediction. where, this paper includes various data pre-processing steps that played an important role in this model. This model uses Cell2Cell dataset of size 3333 subscribers and 57 features. This study presented a very good comparison between the model developed in the study with old models. The implementation part has been done using python and apache spark, that are very good platform for data analysis using machine learning and data mining. For improved performance and effective outcomes Hyper parameter optimization using a grid is used. Prediction performance is evaluated for accuracy, Confusion matrix before and after grid based hyper parameter optimisation. The model out performed and achieved 95% accuracy using Random Forest and 97% accuracy using gradient boosted trees.

***Keywords****:* *Telecommunications, Churn Prediction, Random Forest, Gradient Boosted Tree, Feature Importance, Feature Engineering;*

## 1. Intoduction

Smart phones, the Internet, video games, video calls, and regular phone calls have all become commonplace in this new century. Customers that are price sensitive suffer from high phone rates and expensive internet plans, which leads them to search for less expensive alternatives and switch providers. Churn is the process of changing service providers. Customer churn in the telecom sector is calculated as a loss to the business as a result of the customer leaving the business. Business experts and decision-makers stressed that acquiring new clients is more expensive than keeping the ones you already have [3]. Customer turnover has grown to be a serious problem for telecom firms that requires immediate attention. Companies invest in solutions for predicting the customers who are going to churn in the near future so that retention solutions can be given to the dissatisfied customers while not losing revenues on offering appealing retention plans for customers who were already not going to churn. Customer churn results in direct revenue loss for the company. These Churn Prediction methods employ a variety of technologies and are improving as researchers develop the prediction models. Churn Prediction Model makes predictions using a telecom database. It examines consumer behaviour to forecast future churners.

With terabytes and petabytes of data and a high number of features, telecom databases are extremely complicated and need the development of advanced data science models.

There is a huge advancement on in the field of big data and machine learning. Due to that many models have been developed widely. Researchers proved different methods and used different techniques for churn prediction. [1] used customer usage and other related information for churn prediction and used random forest, decision tree and gradient boosted tress for prediction. For better results grid based hyper parameter has been applied. [2] used newer way of feature engineering and selection. For prediction this research used decision tree, random forest, XGBoost classifier and generated good results. Research [3] used classification as well as clustering technique to classify churner and to know their behaviour patterns. For prediction random forest technique was used. [4] used 10 techniques for churn prediction and compared their performance. Techniques are random forest, decision tree, Discriminant analysis, instance-based learning, Support Vector Machine,

*Hemlata Jain*
*Computer Science School of Basic and Applied Sciences, Poornima University Jaipur-303905, India,*
*E-mail: mailhemajain@gmail.com, 0000-0003-0679-0835*

*Ajay Khunteta*
*Professor at Poornima University, Jaipur-303905, India*
*E-mail: khutetaajay@poornima.org, 0000-0002-5335-9434*

*Sumit Srivastava*
*Professor, Department of Information Technology, Manipal University Jaipur, Jaipur-303007, India*
*E-mail: sumit.310879@gmail.com*

Logistic Regression, Native Bayesian, Multilayer perceptron, ada boosting and stochastic boosting. Research [5] used Logistic Regression and Decision tree for churn prediction. In research [6] some features selection and extraction techniques were used and compared like PCA, sequential forward floating searching, LVW Algorithm, Genetic Algorithm. This research used two phase feature selection model and proved it better method as compared to once. Research [7] used few steps for feature selection using AUC score of features. For prediction Decision tree was used and showed the performance by using ROC curve and computation time recourse consumption.

In literature, some of researchers focused on prediction techniques and some focused on different feature extraction techniques. Feature extraction and selection is very important for a churn prediction model as well as prediction technique. Random Forest itself provide a very good feature analysis techniques that is feature importance. This technique assigns all the features their importance weights. That shows the importance of each feature for developing the prediction model. This paper performed different data pre-processing task, feature engineering task such as Auto encoding the categorical variables, normalized continuous variables and created some new features based on already existing features. Later, this model used the feature importance and based on that removed and added some features. Later for prediction random forest and gradient boosted trees were used. In this study grid based hyper parameter optimisation were used that is giving very good results in term of accuracy and making model more efficient.

The sections of this study are as follows: Section III provides a brief description of the methodology used in this study. Section II: Literature review summarising previous research. The proposed work and database are described in Section IV, while the results and discussion are covered in Section V, and the author's accomplishments and future plans are mentioned in Section VI, which concludes this paper.

## 2. Literature Review

In the past researches were done on some of the traditional machine learning and Deep Leaning models. These models are discussed in this section.

In the research [8] hybrid two phase feature selection techniques were used for reducing dimensionality and improve performance. This research used both traditional expertise approach in first phase and Markov Blanket Discovery technique in second phase. The dataset was taken from a branch if Chinese wireless telecom company. Company provided the dataset of 12000 subscribers which have 3146 churners. For prediction C5.0 and K2 techniques were used. The experiment was performed on weka 3.6.1 environment. Results showed that features after two phase selection process gave good results as 82.56 hit rate where in single phase it gave 74.32 hit rate. The research [9] performed some features engineering task with churn prediction techniques. Feature engineering process aggregated the value of columns per month as per average, count, sum, max and min. and the count of distinct valuer for categorical columns. Other types of features were calculated based on social activities of customers through SMS and call and spark engine. Decision tree, random forest and XGBoost algorithm was used to classify churners. The AUC score achieved by XGBoost was 84, random forest 79.1 and by decision tree was 76.

The research [11] defined some specific rules on features based on customers churning behaviour. Based on that rule decision tree was trained and then tested on test dataset. The dataset was acquired from PAKDD datamining competition. That achieved the accuracy of 98%. This model achieved. By defining specific rules sometimes leads to overfit the model. In study [12] logistic Regression and Decision tree was used for prediction. This study used the dataset of 7000 instances and selected 10 important attributes out of 17. Study showed both models performed well and achieved 91 % accuracy. This study limits with no use of feature selection and engineering techniques. Some of researches are listed below which are related to this research:

| *Research* | *Methodology* | *Dataset size and Features* | *Technique used* | *Performance Measures* | *Performance* |
|---|---|---|---|---|---|
| [1] | Used customers behaviour and usage information | Used Orange Dataset of size 3333 and features size is 21 | Decision Tree Random Forest, Gradient Boosted Tree. | Accuracy, sensitivity, Specificity, tree depth | Tec Ac S Sp<br>DT 86 21 96<br>RF 91 47 98<br>GB 91 49 98 |
| [2] | Feature Engineering | Dataset were provided by SyrialTel company. Dataset was over 9-month data. Dataset | Decision Tree, Random Forest, Gradient Boosted Machine Tree “GBM” and | AUC | AUC of XGBoost– 89%<br>GSN(B) – 85.5<br>RF– 83.4<br>DT – 79.1 |

|  |  |  |  |  |  |
|---|---|---|---|---|---|
|  |  |  | Extreme Gradient Boosting “XGBOOST |  |  |
| [3] | Used classification as well as clustering techniques and defined the factors behind the churn | South Asia GSM telecom of size 64107 and features 29, Orange dataset of size 3333 and 16 features | Random Forest | TP rate, FP Rate, Precision, Recall, F_Measure, ROC Area, Accuracy | SA O<br>TP - 0.89 0.90<br>FP- 0.24 0.57<br>Pre – 0.89 0.89<br>R - 0.89 0.90<br>FM – 0.88 0.88<br>ROC–0.94 0.83<br>AC – 89% |
| [4] | Used some feature engineering process and then prediction | - | Random Forest, XGBoost, logistic Regression | Precision, Recall, F_Score, Support, Accuracy | RF XG LR Ac<br>P-62 52 56 80<br>X-58 50 54 78<br>L-57 56 56 79 |

**Table. 1.** Literature Review State of Art Comparison

In literature, number of feature selection and feature engineering techniques were used and have achieved very good results. All the techniques were used for prediction. In most of the research random forest was used with or without feature engineering. However, there was no clear visualisation of feature importance and feature engineering was not done based on the customer past behaviours. If customer past behaviour will be taken in to mind then this will lead to good prediction in terms of every performance measure. In his study all needed steps were taken related to feature selection, feature engineering and have presented the clear visualisation of feature Importance. Based on that visualisation many features were created and removed.

## 3. Proposed Methodology

### 3.1. Random Forest

Leo Bremen invented Random Forests (RF), and RM is focused on the principle of Bagging and randomly selecting features. A random forest is a bagging approach of ensemble learning implementation. RFs are the classifiers which uses Decision tress and populates them can be seen in the figure given below:

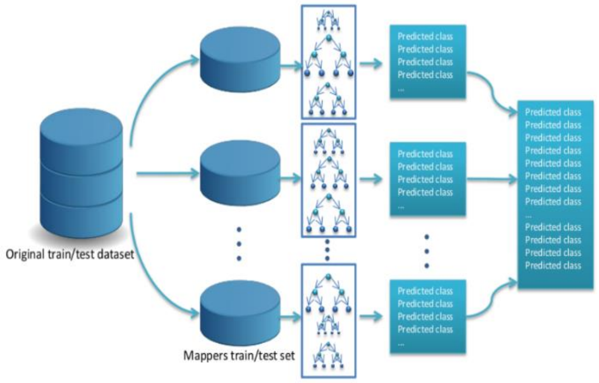

**Fig 1:** Random Forest Structure

In the forest, Decision Trees(DT) were recruited using random bootstrapped replicate of data. Classifiers are selected based on variance and biasing nature of the classifier. Because decision trees for Random forests exhibit low bias and high variance, they are utilized as the base classifier in this experiment. In RM process, forests grows and uses the collection of classification trees or forests. Trees are trained independently in parallel fashion. RM works for increasing the diversity between the individual trees by open up randomness in training process, By substituting each iteration of the original dataset with a subsample, the randomization is introduced. and splitting nodes in each tree using random groups of attributes. We can classify additional objects by majority vote once the forest has grown. The following is a brief description of the pseudo code for the step-by-step approach to use random forests:

Create a bootstrap sample of the training set of size M in step 1, where M stands for the training set's size. The resulting boot sample will suffice as the training set for forest-based base classifiers. Step 2: From the training set, extract 'n' replicate samples with replacement; 'n' is the number of trees in the forest. Step 3: Construct classification trees by growing trees in a forest and picking k out of K attributes at each node to link the splits where k = M or log2 M. Step 4: Grow all trees continuously, without pruning, until the forest has reached its maximum size. Step 5: To predict an unknown variable, average the results of all the trees.

RM have no risk of over fitting, One of the most widely used ensemble approaches for regression and classification is RM. Although there is no maximum on the amount of trees that RM can use, hardware and time limitations may put a limit on it. However, RM can be expand in distributed fashion so easily because it works in parallel fashion and also can be used with large amount of data.

### 3.2. Why not Decision Tree?

Decision tree and Random Forest both are similar tree-based Algorithms but there some differences that makes Random Forest better Algorithm than Decision tree.

- Random Forest is essentially a group of decision trees, with the average or majority vote determined as the model's desired outcome.
- Compared to Decision Tree, the Random Forest model is less prone to overfitting and highly reproducible that are more generic.
- The Random Forest method outperforms the decision tree method in terms of reliability and is commonly considered as more trustworthy.

At the end sometimes Decision tree is not able to take right decisions in that case Random Forest algorithm works better compared to Decision tree. Random Forest takes majority vote from the remaining trees and given the output based on the majority vote.

### 3.3. Gradient Boosted Tree

Friedman first put up the Gradient Boosted Tree in 1999 [12]. The Gradient Boosting Tree is a sophisticated machine learning technique that has demonstrated its effectiveness in a number of real-world applications. In contrast to Random Forests, boosting approaches use a positive ensemble building approach. The primary concept underlying boosting is to continuously incorporate fresh learning models when creating ensembles. In the boosting models, a cumulative error is taken into account, and for each iteration, a new, fundamental weak learner is trained to correspond to the error. To provide more precise and improved result estimation in these models, new models are successively adapted. The new base learners are made to best correspond with the loss or error function's negative gradient. Gradient boosted trees' sequential model fitting method makes it incredibly flexible and configurable for any kind of data requirement. Gradient boosting function is represented in eq(3) when the input feature vector is labelled as (x, y).

$$F(x, \sigma, \gamma) = \sum_{i=1}^{n} \sigma_i h(x, y_i)$$
(1)

The gradient boosting process is carrying out two major duties when h is given as the base weak learners and the summation shows the linear combination of weak learners.

- Calculation of σi provides the weight of a given classifier in the context of ensemble learning.
- The Gradient Boosted Trees are susceptible to over-fitting when trained with more base classifiers; consequently, validation is carried out while training the model employing Computation of i-th weak classifier, yi.

### 3.4. Random Forests versus Gradient-Boosted Trees

- Unlike GBTs, which grow serially, random forests grow coextensively, letting them to educate multiple trees at once. Training Gradient Boosted Trees takes a bit longer.
- Random Forest saves the model from overfitting by increasing the number of trees. Where in GBT, increasing the number of tree doesn’t wort so it is prone to overfitting.
- Random forests works in parallel fashion therefore RF can easily be established distributive manner GBT works in trial after trial manner so it can not be.
- Random forest performs best with deep trees, although boosting performs best with shallow trees.
- Random forests are simple to tune; performance in random forests increases indirectly as the number of trees in the forest increases, however performance in gradient-boosted trees drops.

The fundamental process that will be used for the suggested methodology is shown in Figure 7. The first and most important phase is gathering the data. Since the data are usually noisy and filled with errors, the first step in pre-processing the data is to eliminate the noise and clean the data as necessary. Data normalisation, a phase in the data pre-processing procedure that makes the data all fall within the same range, is another crucial stage in this study. The next phase is feature selection, which comes after data preparation. Utilizing feature importance, feature selection is carried out in this study. The most important features are ordered by importance. Additional to feature significance For feature selection, features engineering is also carried out. Based on the most pertinent features that were previously extracted, features engineering is carried out. In this work, the feature selection also assisted in boosting the classifier's overall efficiency in terms of processing and memory use. In contrast to dimensionality reduction, which enables the production of added features as needed, feature selection demands that attributes either be selected or not selected. The data will be sampled in the following sequence. To divide the data into two distinct subsets, sampling is used. The sampling process is carried out to avoid our model becoming overfitted. When our model fits to noise rather than for strong signals, a model anomaly called over fitting occurs. One of the simplest ways to ensure that our model doesn't overfit is to use sampling. The classifiers utilised in this work, Random Forest and Gradient boosted trees, are trained using the training object set in the subsequent phase. A full description of each classifier is provided separately in various subsections. The prediction model has also been created and is prepared for testing. Utilizing the testing object set created during the sampling process, testing is conducted. Test items are supplied to the prediction model as input, and its performance and optimization potential are assessed. If the classifier's performance is not adequate, optimization is carried out. The goal of the optimization step is to improve the parameters of the Random Forest and Gradient Boosted Trees. Additional performance-related optimizations, such as caching, execution time, tuning, etc., are also carried out. The performance and accuracy evaluation comes last. Utilizing the accuracy, confusion matrix, and Receiver Operating Characteristic (or ROC) curve, the performance of all three classifiers was assessed.

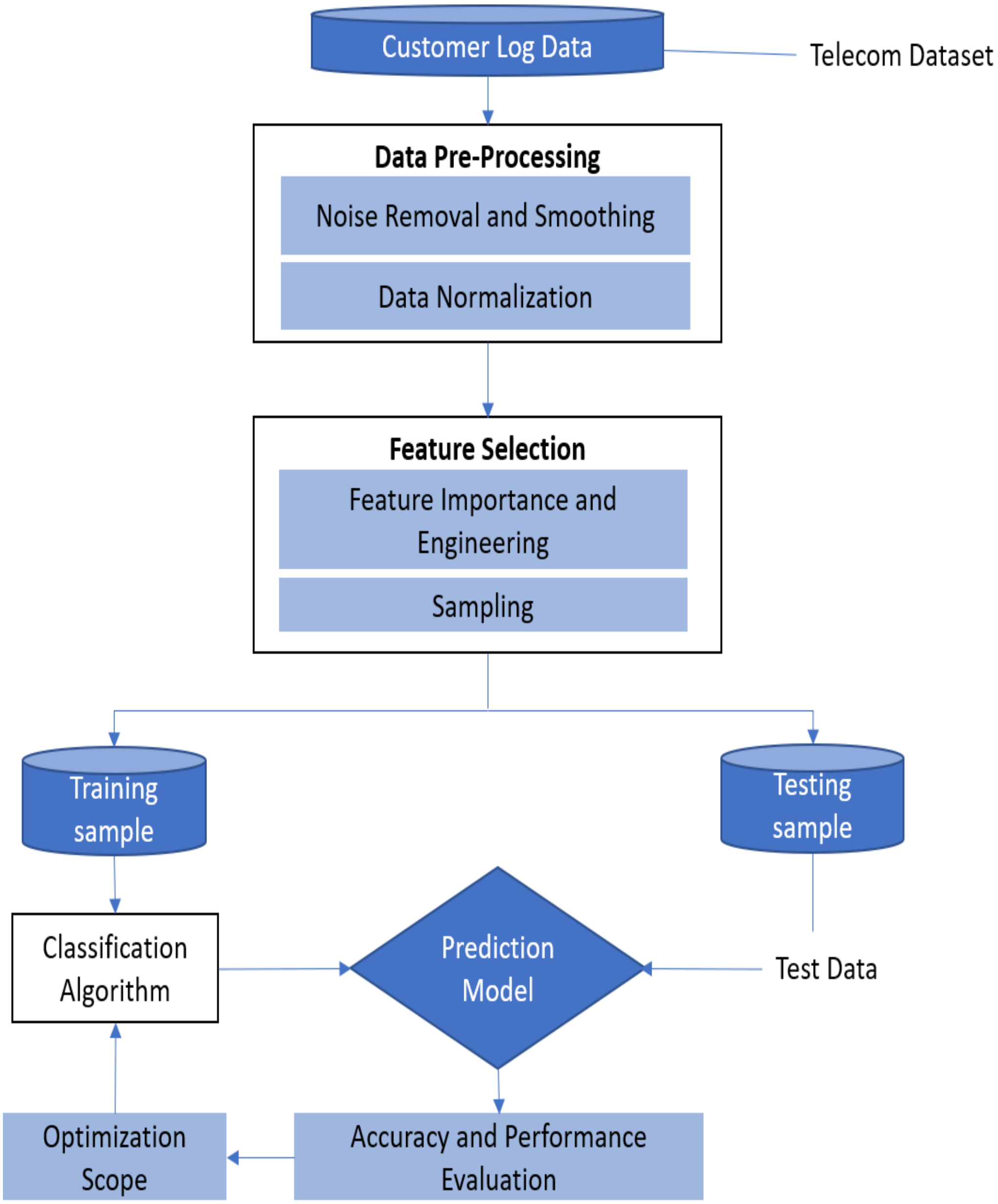

**Fig2. Workflow of Proposed Methodology**.

## 4. Dataset

The experiment was performed in Python. The dataset is taken from an American Telecom company Cell2Cell. It consists of 3333 subscriber's entry and 57 features representing the customer details and plans details. The dataset was downloaded from kaggel.com. dataset have 2852 churners and 481 non churners entry. The churn rate of Cell2Cell is 28% for two years. The class distribution is heavily skewed towards the non-churn class due to the extremely low ratio of churners to non-churners, which promotes a class imbalance. The most common approach adopted by plenty of researchers to tackle the concerns of class imbalance during the data pre-processing phase is sampling [13]. On the dataset some data prepossessing and feature selection task will be performed in this study. As stated in table features are categorised in 4 different categories and in each category, features are further divided in three types. Numerous numerical properties exist, and each feature's value range varies. Where dataset also have some categorical features. Machine learning algorithm does not support categorical features.

**Table. 2:** Cell2Cell dataset Attributes Description.

| *Sr. No.* | *Category* | *Columns* | *Numeric* | *Binary* |
|---|---|---|---|---|
| | | *Categorical* | | |
| 1 | Demographic | MaritalStatus | AgeHH1, AgeHH2 | |
| 2 | Relational | CreditRating | CustomerID,MonthlyRevenue, MonthlyMinutes, TotalRecurringCharge, DirectorAssistedCalls OverageMinutes, RoamingCalls, PercChangeMinutes, PercChangeRevenues, DroppedCalls, BlockedCalls, UnansweredCalls CustomerCareCalls, ThreewayCalls, ReceivedCalls, OutboundCalls, InboundCalls, PeakCallsI0ut OffPeakCallsI0ut, DroppedBlockedCalls CallForwardingCalls, CallWaitingCalls UniqueSubs, ActiveSubs, RetentionCalls, RetentionOffersAccepted, ReferralsMadeBySubscriber AdjustmentsToCreditRating MadeCallToRetentionTeam | RespondsToMailOffers<br>OptOutMailings<br>Churn |
| 3 | Customer's characterstics | PrizmCode<br>Occupation<br>ServiceArea | Handsets,<br>HandsetModels<br>IncomeGroup<br>HandsetPrice | ChildrenInHH,<br>HandsetRefurbished<br>HandsetWebCapable<br>TruckOwner, RVOwner<br>Homeownership<br>BuysViaMailOrder<br>NonUSTravel,<br>OwnsComputer<br>HasCreditCard<br>NotNewCellphoneUser<br>OwnsMotorcycle |
| 4 | Duration | | MonthsInService<br>CurrentEquipmentDays | |

### 4.1. Data Pre-Processing

Database is the collection consists of number of rows and features. Dataset have different types of values and ranges. There may be null values and missing values stored. These values may lead to inaccuracy of the model. There data pre-processing is taken as very important task for any machine learning model. In this study different data pre-processing task were performed, that are discussed below:

- Filled all NA values with Column Mean.
- All NAN fields were initialized with 0.
- All fields with 'Yes' and 'No' values were converted to '1' and '0' respectively.
- All unknown field were initialized with Mean value of the column. There are small found 'Unknown' there it was field with Mean. In case of large unknown values, it is not good way to fill with column.
- <u>**Data Normalisation**</u>**:** There are 37 numerical features, as shown in Table 2.1, and each feature has a unique value range. As a result, normalising every number field is crucial. Machine learning algorithms have difficulties when given values that fall between distinct ranges. In essence, normalisation is a scaling technique that scales feature values between 0 and 1. that operate on a scalar min-max. It scales value in accordance with the feature's maximum and minimum values. Normalization operates as follows:

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}}$$
(2)

The feature's maximum and minimum values are indicated by the variables X max and X min. If the value of X is the feature's minimum value, then the numerator will be 0 and the value of X' will be 0; alternatively, if the value of X is the feature's maximum value, then the numerator will be equal to the denominator, where the value of the field will be 1; otherwise, the value will fall between 0 and 1.

### 4.2. Feature Importance

Finding the cause of customer churn is one of the churn prediction model's primary goals. A very effective strategy for visualising the relevance of each feature in the dataset is feature importance. A feature of Random Forest called feature significance has previously been identified as being highly useful. The dataset is used to apply the importance of this study's feature. The weighted decrease in node impurity divided by the likelihood of reaching that node is used to calculate the relevance of a feature. The number of samples that reach the node is divided by the total number of samples to determine the probability of the node. The importance will increase with the computed value. It can be interpreted as follows:

$$nj_i = w_i c_i - w_{left(i)} c_{left(i)} - w_{right(i)} c_{right(i)}$$

(3)

Where right(i) and left(i) are the right split child node and left split child node, respectively, and nj i is the node significance, w i is the weighted number of samples that have reached the node, and c i is the node impurity value. This is how the feature relevance for each feature is determined.

$$fi_j = \frac{\sum i{:}node\ i\ splits\ on\ feature\ j\ nj_i}{\sum n{\in}all\ nodes\ nj_k}$$

(4)

After determining the importance of each feature, the values are normalised in the range of 0 to 1 in the manner described below. where fi j denotes the importance of feature j and nj i denotes the importance of node i.

$$normfi_j = \frac{fi_j}{\sum i \in all\ features fj_j}$$

(5)

By computing a feature's average across all of the trees, the final relevance of a feature at the Random Forest level is determined. The total number of trees is divided by the sum of the feature importance on each tree.

$$RFfi_j = \frac{\sum i \in all\ trees\ normsfi_{ji}}{K}$$

(6)

Here, RFfi j stands for final feature importance, normsfi ji for the feature f in tree I's normalised feature importance, and K for the overall number of trees. Below is a visualisation of the study's Feature importance:

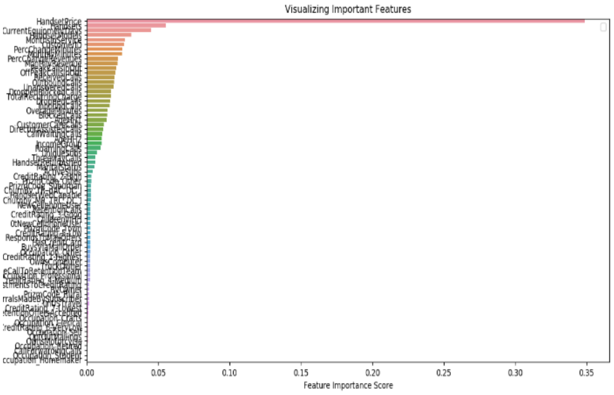

**Fig. 3:** Initial Feature Importance

The number of features is large therefore visualisation is not clear. But it can be seen only one features "HandsetPrice" is having max importance score. Then Handset and then some of features have average importance some are very less and some have null importance. Later in this study few steps were taken to deal with these features.

### 4.3. Feature Engineering

Feature Engineering corresponds to the specific set of action and data manipulation that is done for capturing the similar information and that gives the right direction to the model to achieve the target goal. Here the domain knowledge plays the important role. In this study few steps are performed in feature engineering. Some features are engineered before feature importance and some are after feature importance. This are discussed below:

#### 4.3.1. Encoding Categorical features

as stated in the Table 2.1 there are 5 categorical features. Machine learning does not support categorical features therefore these features cannot be feed into model. Removing these features is not a good way because every feature has its own importance for the model. Therefore, every feature is encoded such that the category of each feature is converted into a column. If a feature has 4 category values then 4 new columns is added to dataset and the value of that resulted column is set to 0 or 1. If the subscriber belongs to that category then it is set to 1 else 0. Features "Maritalstatus", "CreditRating", "PrimeCode' and Occupation are encoded in such way.

There is one more categorical feature named "ServiceArea" which as 740 different categories. If this feature will be encoded then it will create 740 columns in the dataset that will create much noise in dataset therefore removing this feature is good idea.

#### 4.3.2. CustomerID

the feature Customer ID is removed from the dataset because customer id is unique for every customer it does not make any sense for prediction model

#### 4.3.3. Manually created feature based on usage detail

some features are created manually that got much importance in feature importance. In this phase all features are analysed manually and got important facts about some features that are given below:

- customer having "MonthlyRevenue" <=100 have more chances to churn
- customer having "TotalRecurringCharges" <100 have more chances to churn
- customers having number of "Dropped calls" <=50 have more chances to churn
- customers having "Income" <=6 have more chances to churn

based on these facts or rules one feature is created "Churnby_MR_TRC_DC_I"

if all above conditions are true then ChurnBy_MR_TRC_DC_I is set to 1 else 0.

If

- Customer having Total_Requering <=100 then have more chances to churn

- Customer having Director_AssistedCall <= 20 then have more chances to churn
- Customer having Dropped_calls<=20 then have more chances to churn
- customers having “Income” <=6 have more chances to churn

Based on these facts one more feature is created “ChurnBy_TR_dAC_DC_I”

If all above conditions are true then

Set “ChurnBy_TR_dAC_DC_I” =1 else 0.

As stated in feature Importance figure these two new features got importance as follows

| | |
|---|---|
| ChurnBy_TR_dAC_DC_I | 0.002131 |
| Churnby_MR_TRC_DC_I | 0.001871 |

### 4.3.4. Feature Engineering After Feature Importance

Feature importance gave a deep insight on all features. Based on that some features are removed and some features are created. Some features in the dataset are having null or very less importance that are “Occupation”, “CallForwardingCalls”, “OwnMoterCycle”, “RetentionOfferAccepted”, “OptOutMailing”, “CreditRating”. In feature importance figure there are two features “HandsetPrice” and ‘Handset” having high importance as 0.165279 and 0.098864 respectively. Therefore, based on these feature two features are created. After analysing the features some facts are found that are

- Customer having handset price between 90 to 200 having are 17185 with number of churners 10798 out of 14711.
- Customers having 1 Handset are 8428 and all have churned.
- Customer comes in the Handset feature category 1,2,3,4,8,9 is more likely to churn.

Feature 1: ChurnBy_HP is set to 1 when HandsetPrice is between 90 to 200 else 0.

Feature 2: ChurnBy_HP_H is set to 1 when HandsetPrice is between 90 to 200 and Customer having Handset 1,2,3,4,8,9 else 0.

Here is the feature importance graph after Feature engineering process.

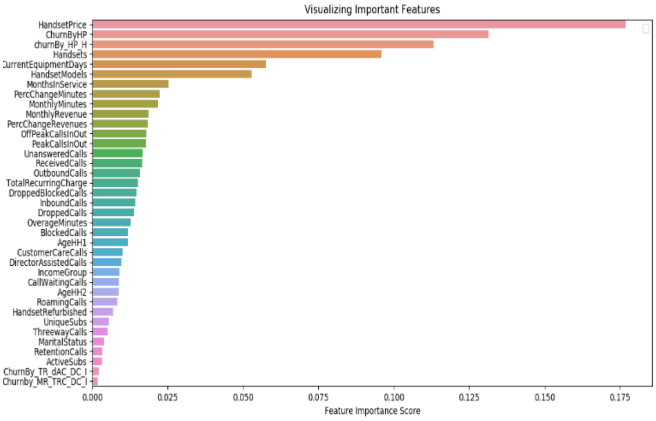

**Fig 4:** Feature Importance After Feature Engineering

It is clearly seen that newly created features in feature engineering process got higher feature importance score. Now there are 4 features with higher importance score. And new features created before feature importance also have importance score better than the features removed. Increasing the features with higher importance defiantly works on model performance. In this study Cell2Cell dataset used for churn prediction model. Dataset have 3333 subscriber’s entry and initially 57. Features. Out of that 1” Churn” feature is Target feature. After data pre-processing, feature importance and feature engineering task dataset come out with 11 new features. 7 features from encoding categorical features and 4 new created features. 8 features removed after feature importance process. Dataset now have 59 independent meaning full features and 1 dependent feature. For classification two ensemble models Random Forest and gradient boosted tress are used. All experiment is done on python and apache spark platform.

### 4.4. Random Forest Classifier

The first ensemble model used to develop a prediction model was Random Forest. By majority vote, the Random forests were averaged. When node K receives Z as input, the following information is present:

$$I(K) = |X|H(Z) - |XL|H(XL) - |XR|H(XR)$$
(7)

Where |X| denotes the input sample size, which in our case is 75% of the initial dataset. The Shannon entropy H(X) and the sizes of the right and left subclasses of Z are denoted by |XR| and |XL|, respectively.

### 4.4.1. Parameters

the model used the following parameter thatare used in the spark ML Library.

- Feature subset strategy (fsStrategy): At every internal node the fraction of total number of features are splitted. This candidate fration is given as parameter and this process is called feature subset strategy.

- Maximum depth (maxDepth): Tree depth is a parameter that is supplied when the tree model is initialised. In addition to making trees more strong and expensive, increasing the tree depth value also causes overfitting. The deeper trees require more time to train.
- Total Number of Trees Grown in Forest (numTrees): This parameter counts all of the trees that have been planted in the forest. The number of trees is not set; more trees in a forest will reduce variance and improve results, while execution may need more time due to the longer training period. In our studies, we maintained a total of 14 trees.
- Sub sampling rate (ssRate): The suggested value for this parameter is 1, but it can be changed to accelerate training. It determines how much of the dataset is used to train each tree in the forest.

### 4.4.2. Model Development

The maximum depth was set at 6, the subsampling approach was fixed at 1.0, and the feature utilised for splitting at each node was 9. The random forest classifier was constructed by filling 22 trees in the forest. 75 percent of the dataset was utilised for tree learning, while the remaining 25 percent was used to test the model.

### 4.4.3. Gradient Boosted Tree:

Gradient boosted trees (GBTs), another extremely effective ensemble method, are utilised to build classifiers for churn prediction. GBTs' primary base function is to iteratively train trees while attempting to reduce the loss function. The loss function for GBTs is the "Logarithmic loss" (log loss), as it is the least biased for binary classification. If there are M instances, then the following equation is used to determine the logarithmic loss:

$$logloss = 2\sum i = 1 \quad M\, log(1 + \exp\,(-2\; liF(xi))) \tag{8}$$

Here, the labels and features of instance I are designated as li and xi, respectively. For example, F(xi) predicted is label for i. (by the model). The gradient boosting method is extremely straightforward. Iteratively, a linear collection of predictors is created; then, to enhance the performance and weighted average of the predictors, an incremental classifier—referred to as the final predictor—is inserted at each step. The detailed procedure is provided below:

Step 1. All Decision trees or base(week) learners are initialised in this step.

Step 2. at each iteration, inputs are re-weighted 'up-weighting' this is done for the inputs which are not classified by existing forest.

Step 3. A new classifier "hi" is constructed for residuals.

Step 4. Again new weight "γi" is computed for new base classifier.

Step 5. Add the values ( γi ,hi ) to the existing forest.

Step 6. The function will Return the final tree from the forest.

### 4.1.1. Parameters

The basic parameters are given and tuned to the gradient boosted tree are tree depth, number of bins etc. where some other parameters are also given below:

- Loss function: in this, the loss function is defined. The results are dependent on different loss function, in this study log loss function is defined.
- Number of iterations: this parameter is given as the number of trees I the forest that is also taken as number of iterations. Increasing the number may lead to over fitting and larger training time.

## 5. Result and Discussion

In this section the results is discussed using both models random forest and gradient boosted trees. Later grid based hyper parameter optimisation is performed on both models. First basic results are discussed with obtained without hyper parameter optimisation and in next sub sequent section Hyper parameter optimisation is performed. Later all performance measures are discussed after hyper parameter optimisation and evaluated using confusion matrix.

Model performance is evaluated and compared with the existing models listed in literature review. In literature review Random Forest was used with or without feature selection and engineering. This study used the feature importance techniques and visualized the feature importance for better understanding the features. That came out with very informative analysis results. Based on this model created new feature which have got importance. This model is able gave dataset new features with high importance for churn prediction. This model performed equally good or better in case of some performance measures. The results are discussed below:

### 5.1. Confusion Matrix

### 5.1.1. Gradient Boosted

This study first used gradient boosted trees ensemble model that gave the classification accuracy as 92% that is very good accuracy achieved. The dataset was splited in test and training in 70:30 ratio. The max depth were chosen was 7 and number of iteration was 9. The confusion matrix achieved after this implementation is shown below in table: 3

Table 3: Gradient Boosted Confusion Matrix.

| | *No* | *Yes* |
|---|---|---|
| *No* | 2816 | 34 |
| *Yes* | 244 | 239 |

### 5.1.2. Random Forest

Random forest gave the classification accuracy as 91%. In The experiment the number of trees was taken in forest was 10 and given depth of each tree was 5 and the number of bins was 20. After the experiment the achieved confusion matrix is shown below in table 4:

Table 4: Random Forest confusion Matrix.

| | *No* | *Yes* |
|---|---|---|
| *No* | 2820 | 30 |
| *Yes* | 255 | 228 |

Table 5 : Other Performance Measures.

| *Classifier Parameter* | *Gradien Boosted* | *Random Forest* |
|---|---|---|
| *Accuracy* | 92% | 91% |
| *Sensitivity* | 0.5 | 0.5 |
| *Specificity* | 0.99 | 0.99 |

## 5.2. Result Optimisation

Results optimisation is to given a specific input to the objective function which results in a maximum or minimum evaluator. Results optimisation is very big problem in machine learning algorithms and needed to consider for better results optimisations.

In this study the models were evaluated using primary results and not specific to dataset analysis. Later in this study Grid based hyper parameter result optimisation is performed to get better results. In grid based hyperparameter optimisation, parameters for classification algorithm were chosen using grid search approach that works in the following steps:

1 **step**. Based on the theoretical knowledge, a list of possible values is prepared for every input parameter k.
2 **step**. A grid is prepared using all possible values and cost is calculated for each grid set.
3 **step**. Out of all grid, grid having lowest cost is selected that grid is also called hyper parameter grid.

### 5.2.1. Gradient Boosted Trees

In gradient Boosted Trees, some parameter can be optimised that are Number of Iteration, Maximum Depth, Training and testing sampling rate that were kept in the ratio 70:30. In the figure 5 the accuracy variation can be seen based on the number of iterations. Where tree depth was set to 7 and iteration parameter grid was set as (1,2,3,5,7,10,13,15,20,25).

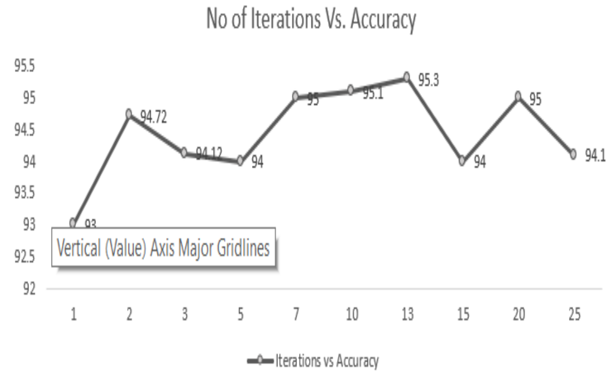

**Fig5**: Number of Iterations vs Accuracy Chart.

In the figure the accuracy change can be seen based on number of iterations. Here the maxium accuracy has been seen from number of iteration 10 to 13. The optimal accuracy was for 13th iteration. Again the grid with maximum dept was set to (1,2,3,5,7,10,13,15,20) and the cost graph can be seen in the figure 6 given below:

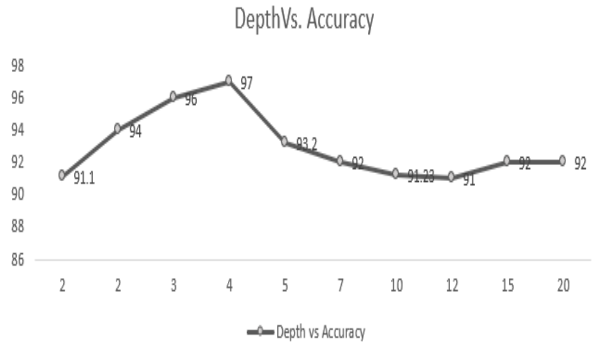

**Fig 6**: Depth vs Accuracy Chart.

The maximum prediction accuracy is 97% and is observed at depth=4, Therefore, the optimal parameter were chosen were depth having value 4 and iteration number having value 13 by observing both parameters vs. accuracy plots. After hyper parameter optimisation the corresponding confusion matrix for GBTs is given in Table 6.

*Table 6: Gradient Boosted Confusion Matrix After Result Optimisation.*

| | *No* | *Yes* |
|---|---|---|
| *No* | 2828 | 22 |
| *Yes* | 85 | 398 |

### 5.2.2. Random Forest

In Random Forest, the parameters can be optimised that are Number of trees, Tree Depth, and number of bins splitting the nodes. Training and testing set were set in the ratio70:30. In the figure 7 the accuracy variation can be seen based on the depth parameter. Where number of bins set to 16 and depth parameter were set as (4,5,6,7,8,9,10,12,15).

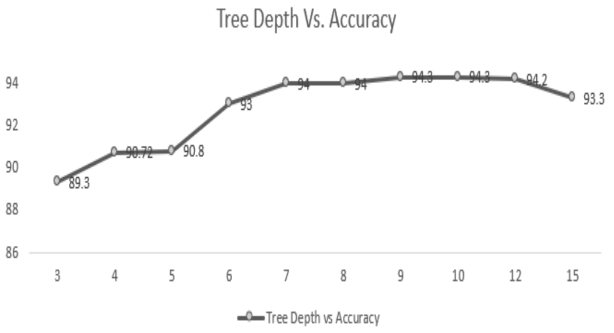

**Fig7**: Tree Depth vs Accuracy Chart.

In the figure7 the maximum performance can be seen at the depth value 9 or 10, therefore dept value is now set to 9 and the parameter number of trees is changed to find best hyper parameter set for the algorithm Random Forest. After final set the Accuracy variation can be seen in the figure 8 given below.

The maximum accuracy is captured is 95.32% that is observed at depth value 10, number of trees 20 and number of bin value 16. The confusion matrix is given below after hyper parameter optimisation for Random Forest in table 7.

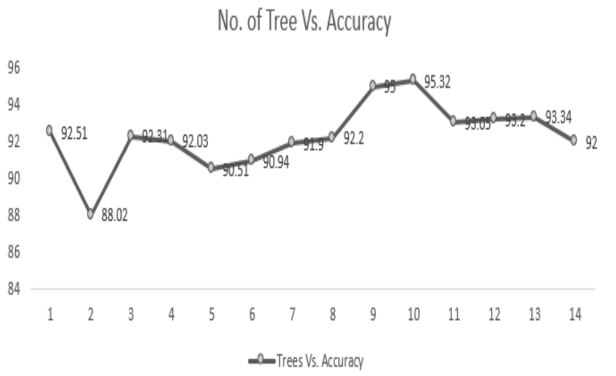

**Fig8**: Number of Trees vs Accuracy Chart.

*Table 7: Random Forest Confusion Matrix After Result Optimisation.*

| | *No* | *Yes* |
|---|---|---|
| *No* | 2834 | 16 |
| *Yes* | 140 | 343 |

The optimized results for all the classifiers are summarized in Table 8.

*Table 8: Other Performance Measures After Result Optimisation.*

| *Classifier Parameter* | *Gradient Boosted* | *Random Forest* |
|---|---|---|
| *Accuracy* | 97% | 95% |
| *Sensitivity* | 0.82 | 0.71 |
| *Specificity* | 0.99 | 0.99 |

## 6. Conclusion

This model presents the efficient methodology to anticipate churn prediction in telecommunications. Bootstrap based ensemble model has been used for churn prediction. The ensemble model used in the study is Random Forest(Bagging) and Gradient Boosted Tree(Boosting). The ensembles approaches used in the study outperformed specially residual feedback improvement based approaches. Study showed the importance of feature engineering and feature importance for a churn prediction model. For feature engineering task this study worked on encoding categorical features and created new features that achieved high importance for churn prediction model. This study showed the effect of feature importance that can be very useful for a churn prediction model. Gradient Boosted Tree performed better in terms of accuracy and sensitivity. This study also used result optimisation for better refinery results. The methodology is tested on an American telecom company cell2cell of size 3333 and 56 features initially. Feature importance and engineering task gave 59 meaningful features.

In future, number of classification techniques can be used and a compared the results. This study used subscribers past data, a churn prediction model can be tested on a real time data and can present a good use of IOT technology. This model is restricted to one dataset, feature engineering task can be improved that can work for any dataset.

### References

[1]. Pretam Jayaswal+, Bakshi Rohit Prasad*, Divya Tomar!, and Sonali Agarwal#, “An Ensemble Approach for Efficient Churn Prediction in Telecom Industry”, International Journal of Database Theory and Application, Vol.9, No.8 (2016), pp.211-232 http://dx.doi.org/10.14257/ijdta.2016.9.8.21

[2]. Vanitha, D. D. . (2022). Comparative Analysis of Power switches MOFET and IGBT Used in Power Applications. International Journal on Recent Technologies in Mechanical and Electrical Engineering, 9(5), 01–09. https://doi.org/10.17762/ijrmee.v9i5.368

[3]. Abdelrahim Kasem Ahmad* , Assef Jafar and Kadan Aljoumaa, “Customer churn prediction in telecom using machine learning in big data platform”, Journal of big data, Ahmad *et al. J Big Data (2019) 6:28* https://doi.org/10.1186/s40537-019-0191-6

[4]. IRFAN ULLAH1, BASIT RAZA 1, AHMAD KAMRAN MALIK 1, MUHAMMAD IMRAN1, SAIF UL ISLAM 2, AND SUNG WON KIM 3, “A Churn Prediction Model Using Random Forest:

Analysis of Machine Learning Techniques for Churn Prediction and Factor Identification in Telecom Sector", IEEE Access, May 6, 2019. Volume 7.

[5]. L. N. Balai, G. K. J. A. K. S. (2022). Investigations on PAPR and SER Performance Analysis of OFDMA and SCFDMA under Different Channels. International Journal on Recent Technologies in Mechanical and Electrical Engineering, 9(5), 28–35. https://doi.org/10.17762/ijrmee.v9i5.371

[6]. Sahar F. Sabbeh, "Machine-Learning Techniques for Customer Retention: A Comparative Study", (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 9, No. 2, 2018.

[7]. Freddie Mathews Kau, Hlaudi Daniel Masethe and Craven Klaas Lepota, " Service Provider Churn Prediction for Telecoms Company using Data Analytics", Proceedings of the World Congress on Engineering and Computer Science 2017 Vol I WCECS 2017, October 25-27, 2017, San Francisco, USA.

[8]. Ravita, R., & Rathi, S. (2022). Inductive Learning Approach in Job Recommendation. International Journal of Intelligent Systems and Applications in Engineering, 10(2), 242–251. Retrieved from https://ijisae.org/index.php/IJISAE/article/view/1829

[9]. Malla, S., M. J. . Meena, O. . Reddy. R, V. . Mahalakshmi, and A. . Balobaid. "A Study on Fish Classification Techniques Using Convolutional Neural Networks on Highly Challenged Underwater Images". International Journal on Recent and Innovation Trends in Computing and Communication, vol. 10, no. 4, Apr. 2022, pp. 01-09, doi:10.17762/ijritcc.v10i4.5524.

[10]. Yin Wu, Jiayin Qi, "The Study on Feature Selection in Customer Churn Prediction Modeling', Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics San Antonio, TX, USA - October 2009.

[11]. Jiayin Qi, Yuanquan Li, "A novel and convenient variable selection method for choosing effective input variables for telecommunication customer churn prediction model", Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics San Antonio, TX, USA - October 2009.

[12]. Xu Hong, Zhang Zigang, Zhang Yishi*," Churn Prediction in Telecom Using a Hybrid Two-phase Feature Selection Method", 2009 Third International Symposium on Intelligent Information Technology Application. IEEE DOI 10.1109/IITA.2009.392.

[13]. Pepsi M, B. B. ., V. . S, and A. . A. "Tree Based Boosting Algorithm to Tackle the Overfitting in Healthcare Data". International Journal on Recent and Innovation Trends in Computing and Communication, vol. 10, no. 5, May 2022, pp. 41-47, doi:10.17762/ijritcc.v10i5.5552.

[14]. V. Kavitha, S. V Mohan Kumar, G. Hemanth Kumar, M. Harish," Churn Prediction of Customer in Telecom Industry using machine Learning Algorithms", International Journal of Engineering Research & Technology (IJERT), ISSN: 2278-0181, Vol. 9 Issue 05, May-2020.

[15]. DR. M.BALASUBRAMANIAN *, M.SELVARANI **, "CHURN PREDICTION IN MOBILE TELECOM SYSTEM USING DATA MINING TECHNIQUES", International Journal of Scientific and Research Publications, Volume 4, Issue 4, April 2014 1 ISSN 2250-3153.

[16]. J. Pamina, J. Beschi Raja, S. Sathya Bama, S. Soundarya, M.S. Sruthi, S. Kiruthika, V.J. Aiswaryadevi, G. Priyanka, " An Effective Classifier for Predicting Churn in Telecommunication", Jour of Adv Research in Dynamical & Control Systems, Vol. 11, 01-Special Issue, 2019.

[17]. J. H. Friedman, "Greedy function approximation: A gradient boosting machine", Annals of Statistics vol. 29, (2001), pp. 1189-1232.

[18]. Chaudhary, D. S. . (2022). Analysis of Concept of Big Data Process, Strategies, Adoption and Implementation. International Journal on Future Revolution in Computer Science &Amp; Communication Engineering, 8(1), 05–08. https://doi.org/10.17762/ijfrcsce.v8i1.2065

[19]. G. M. Weiss, "Mining with rarity: A unifying framework", ACM SIGKDD Explorations Newslett., vol. 6, no. 1, (2004), pp. 7-19.

[20]. Agarwal, D. A. . (2022). Advancing Privacy and Security of Internet of Things to Find Integrated Solutions. International Journal on Future Revolution in Computer Science &Amp; Communication Engineering, 8(2), 05–08. https://doi.org/10.17762/ijfrcsce.v8i2.2067