INTELLIGENT CUSTOMER RETENSION:USING MACHINE LEARNING FOR ENHANCED PREDICTION OF TELECOM CUSTOMER CHURN

The project submitted to Smart Internz

By

KALAIVANAN D (2022K2002)

GOWTHAM E (2022K1999)

JAGAN KUMAR C (2022K2000)

JANANI A (2022K2001)

Department of Computer Science

GOVERNMENT ARTS AND SCIENCE COLLEGE

MODAKURICHI

(Affiliated to Bharathiar University)

ERODE - 638104

INDEX

1 INTRODUCTION

Overview

Customer churn is often referred to as customer attrition, or customer defection which is the rate at which the customers are lost. Customer churn is a major problem and one of the most important concerns for large companies. Due to the direct effect on the revenues of the companies, especially in the telecom field, companies are seeking to develop means to predict potential customer to churn. Looking at churn, different reasons trigger customers to terminate their contracts, for example better price offers, more interesting packages, bad service experiences or change of customers' personal situations.

Customer churn has become highly important for companies because of increasing competition amongcompanies, increased importance of marketing strategies and conscious behaviour of customers in the recent years. Customers can easily trend toward alternative services. Companies must develop various strategies to prevent these possible trends, depending on the services they provide. During the estimation of possible churns, data from the previous churns might be used. An efficient churn predictive model benefits companies in many ways. Early identification of customers likely to leave may help to build cost effective ways in marketing strategies. Customer retention campaigns might be limited to selected customers but it should cover most of the customer. Incorrect predictions could result in a company losing profits because of the discounts offered to continuous subscribers.

Telecommunication industry always suffers from a very high churn rates when one industry offers a better plan than the previous there is a high possibility of the customer churning from the present due to a better plan in such a scenario it is very difficult to avoid losses but through prediction we can keep it to a minimal level.

Telecom companies often use customer churn as a key business metrics to predict the number of customers that will leave a telecom service provider. A machine learning model can be used to identity the probable churn customers and then makes the necessary business decisions.

Purpose

As mentioned above, due to the digitalization and the access to information in large proportions, companies are under constant threats of customer churning, in particular subscription-based service providers that rely on such incomes. Unlike the business-to-customer domain, business to business characteristics, such as higher transactional value of each customer, constitutes a greater negative impact on the revenue stream when customers churn . Hence, it is important to study churn prediction and to support companies in this problem. The purpose of our study is to contribute knowledge in the field of churn prediction in a B2B subscription based service context. In this study we employ different machine learning techniques, in order to investigate the effects on the performance of churn prediction models in the B2B subscription-based service context. In particular, the aim is to propose and evaluate different approaches and techniques of how machine learning can be used to predict churn; hence creating value for businesses offering subscription-based services. The objectives of this study are to:

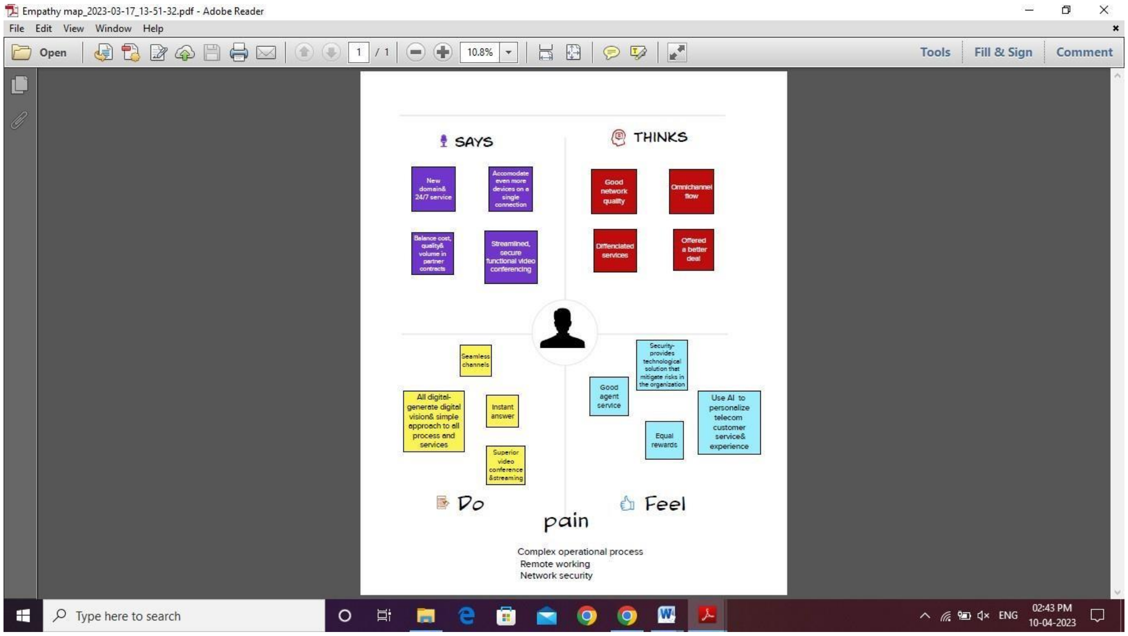
i) construct a binary churn prediction model,

ii) compare different types of algorithms for the built model, and

iii) study the effects of balancing the training dataset for the considered model

2 Problem Definition & Design Thinking

Empathy map



SAYS

New domain& 24/7 service

Accomodate even more devices on a single connection

Balance cost, quality& volume in partner contracts

Streamlined, secure functional video conferencing

THINKS

Good network quality

Omnichannel flow

Differciated services

Offered a better deal

Seamless channels

All digital-generate digital vision& simple approach to all process and services

Instant answer

Superior video conference &streaming

Do

Security-provides technological solution that mitigate risks in the organization

Good agent service

Use AI to personalize telecom customer service& experience

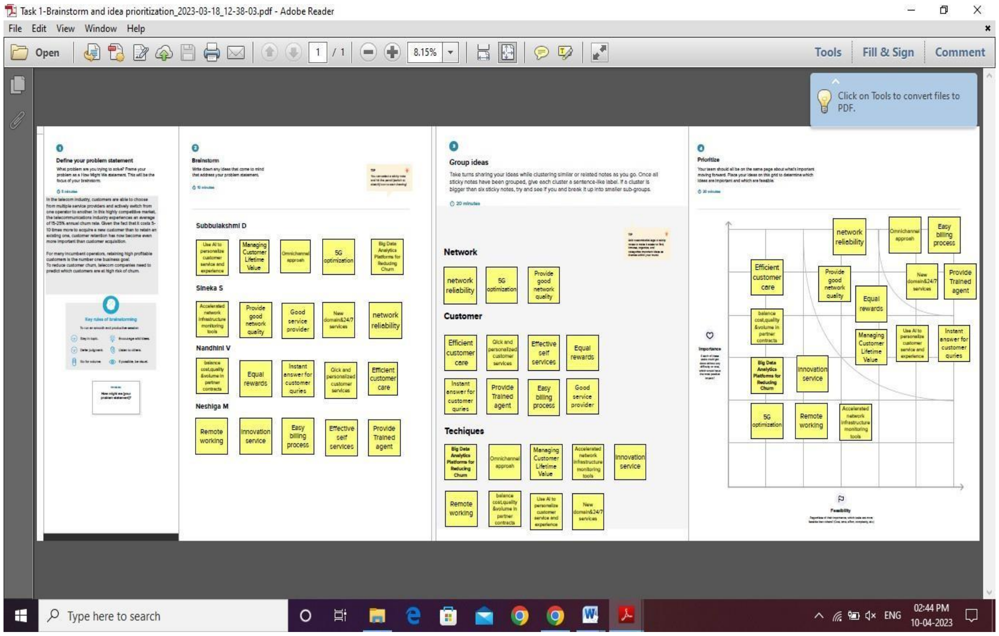
Equal rewards

Feel

pain

Complex operational process
Remote working
Network security

Ideation & Brainstorming Map



Task 1-Brainstorm and idea prioritization_2023-03-18_12-38-03.pdf - Adobe Reader
File
Edit
View
Window
Help
Open
1
/ 1
8.15%
Tools
Fill & Sign
Comment
Click on Tools to convert files to PDF.
Type here to search
ENG
02:44 PM
10-04-2023

1

Define your problem statement

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

In the telecom industry, customers are able to choose from multiple service providers and actively switch from one operator to another. In this highly competitive market, the telecommunications industry experiences an average of 15-25% annual churn rate. Given the fact that it costs 5-10 times more to acquire a new customer than to retain an existing one, customer retention has now become even more important than customer acquisition.

For many incumbent operators, retaining high profitable customers is the number one business goal.
To reduce customer churn, telecom companies need to predict which customers are at high risk of churn.

Key rules of brainstorming

2

Brainstorm

Write down any ideas that come to mind that address your problem statement.

Subbulakshmi D

Use AI to personalize customer service and experience | Managing Customer Lifetime Value | Omnichannel approah | 5G optimization | Big Data Analytics Platforms for Reducing Churn

Sineka S

Accelerated network infrastructure monitoring tools | Provide good network quality | Good service provider | New domain&24/7 services | network reliability

Nandhini V

balance cost,quality &volume in partner contracts | Equal rewards | Instant answer for customer quries | Qick and personalized customer services | Efficient customer care

Neshiga M

Remote working | Innovation service | Easy billing process | Effective self services | Provide Trained agent

3

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

Network

network reliability | 5G optimization | Provide good network quality

Customer

Efficient customer care | Qick and personalized customer services | Effective self services | Equal rewards | Instant answer for customer quries | Provide Trained agent | Easy billing process | Good service provider

Techiques

Big Data Analytics Platforms for Reducing Churn | Omnichannel approah | Managing Customer Lifetime Value | Accelerated network infrastructure monitoring tools | Innovation service | Remote working | balance cost,quality &volume in partner contracts | Use AI to personalize customer service and experience | New domain&24/7 services

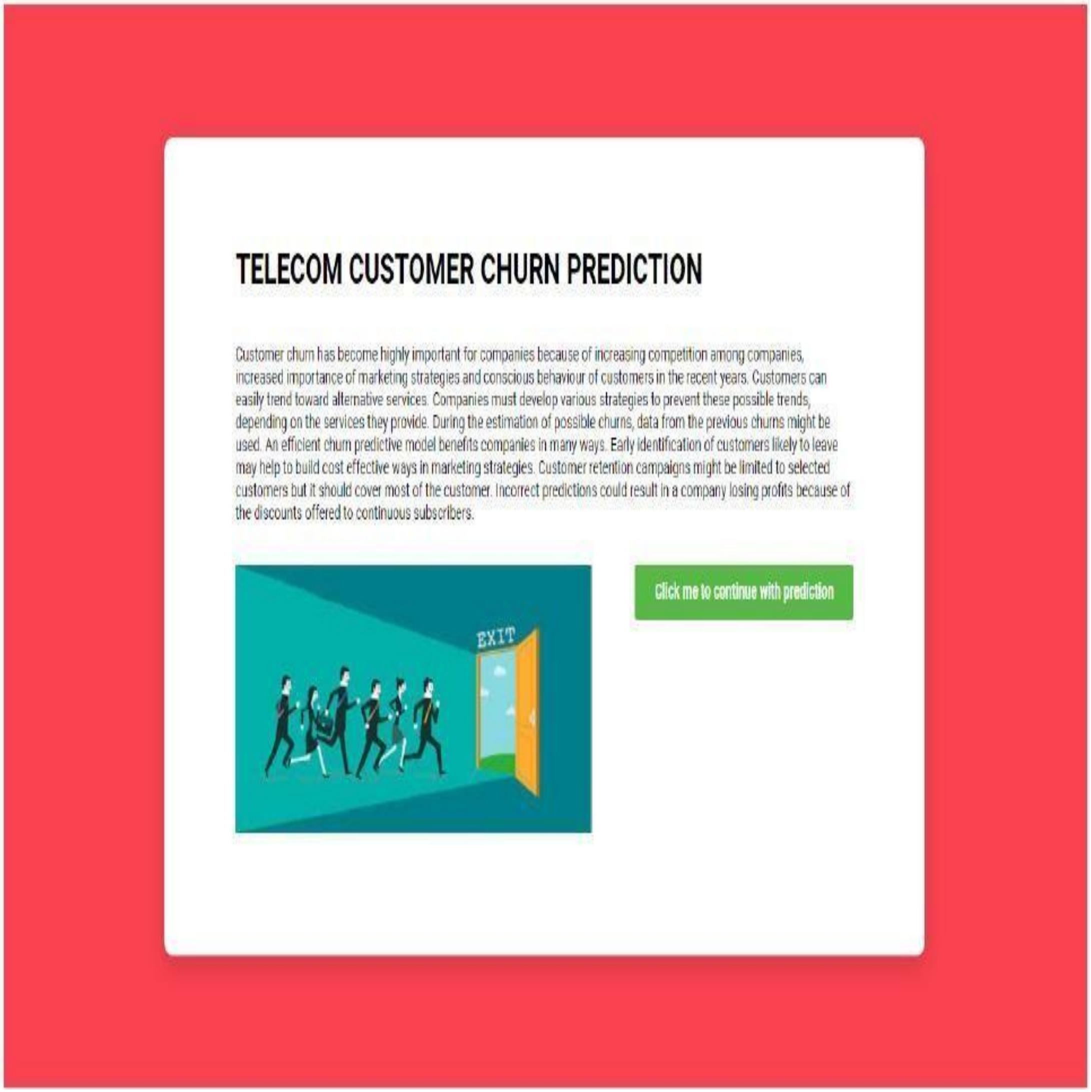
4

Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

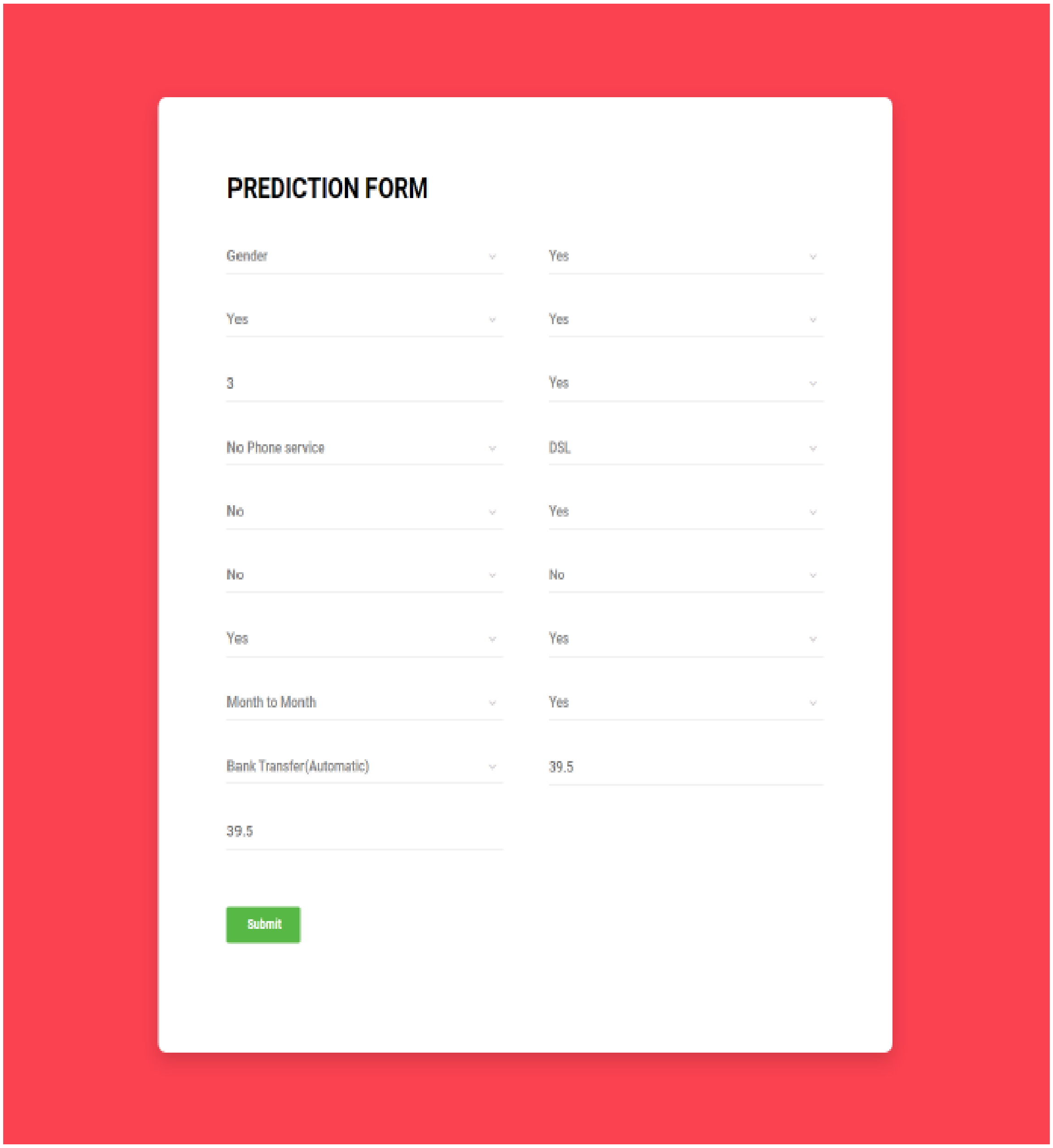
Importance
Feasibility
network reliability
Omnichannel approah
Easy billing process
Efficient customer care
Provide good network quality
Equal rewards
New domain&24/7 services
Provide Trained agent
balance cost,quality &volume in partner contracts
Managing Customer Lifetime Value
Use AI to personalize customer service and experience
Instant answer for customer quries
Big Data Analytics Platforms for Reducing Churn
Innovation service
5G optimization
Remote working
Accelerated network infrastructure monitoring tools

3 RESULT



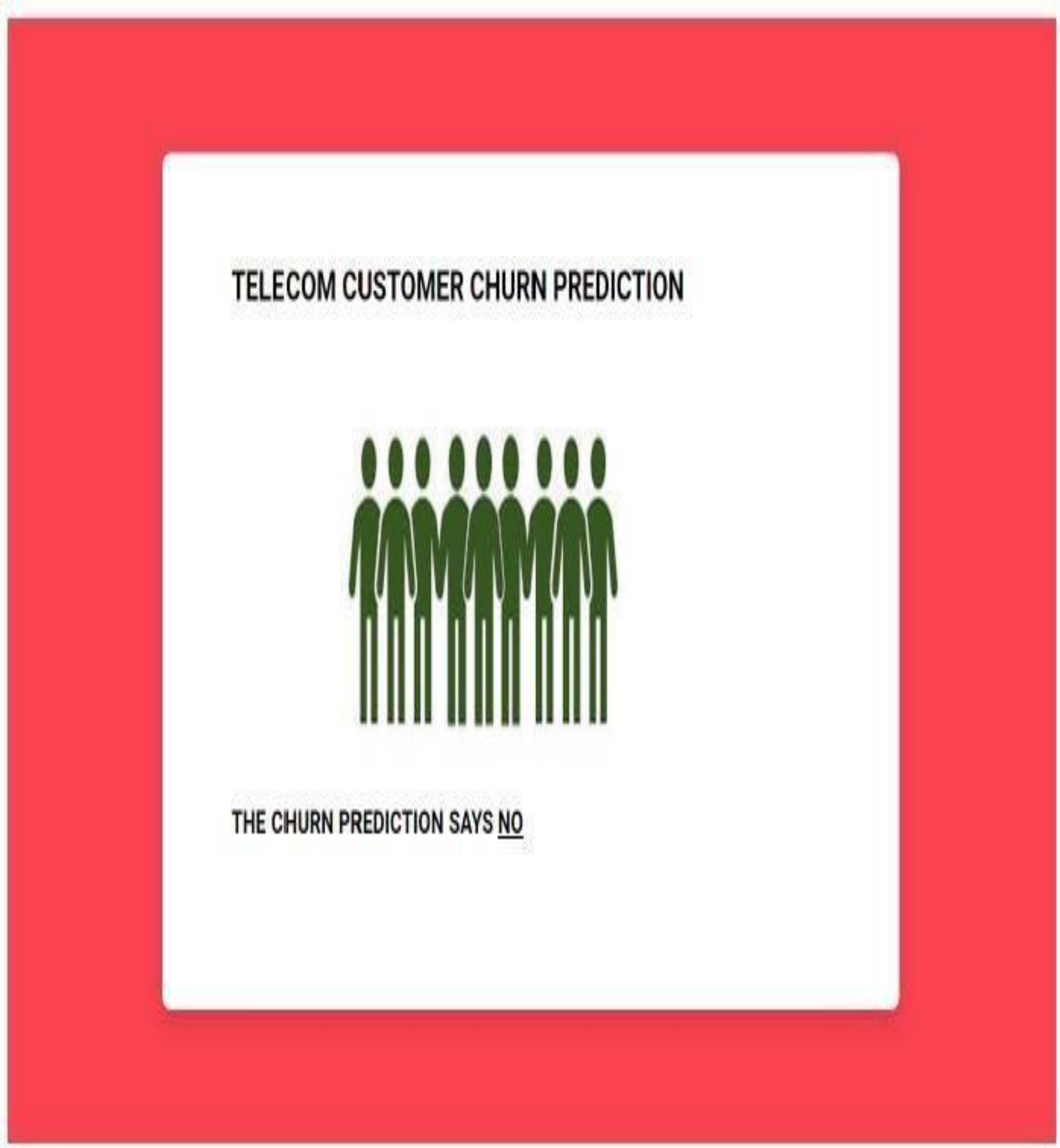
TELECOM CUSTOMER CHURN PREDICTION

Customer churn has become highly important for companies because of increasing competition among companies, increased importance of marketing strategies and conscious behaviour of customers in the recent years. Customers can easily trend toward alternative services. Companies must develop various strategies to prevent these possible trends, depending on the services they provide. During the estimation of possible churns, data from the previous churns might be used. An efficient churn predictive model benefits companies in many ways. Early identification of customers likely to leave may help to build cost effective ways in marketing strategies. Customer retention campaigns might be limited to selected customers but it should cover most of the customer. Incorrect predictions could result in a company losing profits because of the discounts offered to continuous subscribers.



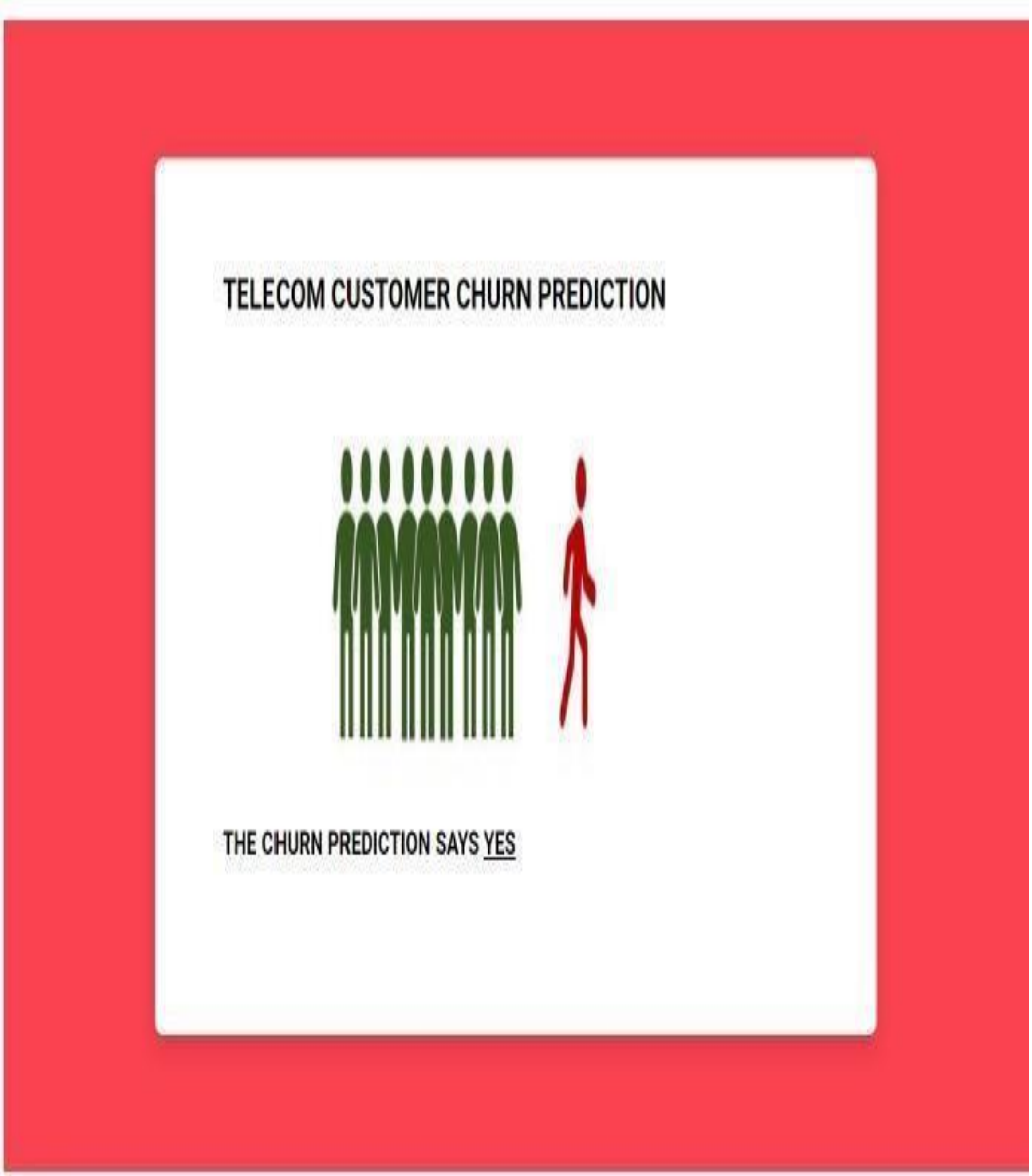
PREDICTION FORM

Gender	Yes
Yes	Yes
3	Yes
No Phone service	DSL
No	Yes
No	No
Yes	Yes
Month to Month	Yes
Bank Transfer(Automatic)	39.5
39.5	



TELECOM CUSTOMER CHURN PREDICTION

THE CHURN PREDICTION SAYS NO



TELECOM CUSTOMER CHURN PREDICTION

THE CHURN PREDICTION SAYS YES

4 ADVANTAGES & DISADVANTAGES

"Communications technology is a crucial part of both **today's** *and tomorrow's society."*

In this system, we use various algorithms like Random Forest, XGBoost & Logistic Regression to find accurate values and which helps us to predict the churn of the customer. Here we implement the model by having a dataset that is trained and tested, which makes us have maximum correct values. In the Initial step, data preprocessing is performed in which we do filtering data and convert data into a similar form, and then we make feature selection.

In the further step prediction and classification is done using the algorithms like Random Forest, XGBoost, Logistic Regression(LR). Training and testing the model with the data set, we observe the behavior of the customer and analyze them. In the final step, we do analysis based on the results obtained and predict the customer churn.

Advantages of Telecom customer churn prediction :

- ✓ Quick and accessible communication
- ✓ Lack of time period
- ✓ Saves time
- ✓ Saves gasoline (do not need to drive distance)
- ✓ More than two people can communicate with at least one another at an equivalent time
- ✓ Next "best thing" to being there
- ✓ Easy to exchange ideas and knowledge via phone and/or fax
- ✓ Worldwide access
- ✓ Easy access to the people you would like to contact.
- ✓ Less effort in using transportation just to satisfy a private personally.
- ✓ You can just occupy your home and use a telephone or a cellphone if you would like to speak to someone.

- ✓ Enable end-users to speak electronically and share hardware, software, and data resources.
- ✓ This make corporation to do the transaction at the point only and in a very fast way from many remote locations, exchange business documents electronically with customers and suppliers, or remotely monitor and control production processes.
- ✓ Interconnect the pc systems of a business so their computing power is often shared by end-users throughout an enterprise.
- ✓ Make the organization work with collaboration and communication among the staff inside and out of doors a corporation.
- ✓ Speed
- ✓ Develops new products and inventions

Disadvantages of Telecom customer churn prediction:

- ✓ Cultural Barrier
- ✓ Misunderstanding
- ✓ Prank calls
- ✓ Sometimes expensive
- ✓ High electric bills
- ✓ Remote areas don’t have access
- ✓ Remote areas might not be ready to afford the necessary equipment
- ✓ Cannot see whom you’re speaking with
- ✓ Cannot see facial expressions, therefore results in misunderstandings
- ✓ Cultural barriers
- ✓ Poor connections or downed power lines during/after storms

5 APPLICATIONS

Customer churn is a common problem across businesses in many sectors. If you want to grow as a company, **you have to invest in acquiring new clients**. Every time a client leaves, it represents a significant investment lost. Both time and effort need to be channelled into replacing them. Being able to predict when a client is likely to leave, and offer them incentives to stay, can offer huge savings to a business. As a result, understanding what keeps customers engaged is extremely valuable knowledge, as it can help you to develop your retention strategies, and to roll out operational practices aimed at keeping customers from walking out the door. Predicting churn is a fact of life for any subscription business, and even slight fluctuations in churn can have a significant impact on your bottom line.

6 CONCLUSION

The importance of this type of research in the telecom market is to help companies make more profit. It has become known that predicting churn is one of the most important sources of income to telecom companies. Hence, this research aimed to build a system that predicts the churn of customers in SyriaTel telecom company. These prediction models need to achieve high AUC values. To test and train the model, the sample data is divided into 70% for training and 30% for testing. We chose to perform cross-validation with 10-folds for validation and hyperparameter optimization. We have applied feature engineering, effective feature transformation and selection approach to make the features ready for machine learning algorithms. In addition, we encountered another problem: the data was not balanced. Only about 5% of the entries represent customers' churn. This problem was solved by under sampling or using trees algorithms not affected by this problem. Four tree based algorithms were chosen because of their diversity and applicability in this type of prediction. These algorithms are Decision Tree, Random Forest, GBM tree algorithm, and XGBOOST algorithm. The method of preparation and selection of features and entering the mobile social network features had the biggest impact on the success of this model, since the value of AUC in SyriaTel reached 93.301%. XGBOOST tree model achieved the best results in all measurements. The AUC value was 93.301%. The GBM algorithm comes in the second place and the random forest and Decision Tree came third and fourth regarding AUC values. We have evaluated the models by fitting a new dataset related to different periods and without any proactive action from marketing, XGBOOST also gave the best result with 89% AUC. The decrease in result could be due to the non-stationary data model phenomenon, so the model needs training each period of time.

7 FUTURE SCOPE

Some features such as Contract ID, MSISDN and other unique features for all customers were removed. They are not used in the training process because they have a direct correlation with the target output (specific to the customer itself). We deleted features with identical values or missing values, deleted duplicated features, and features that have few numeric values. We found that more than half of the features have more than 98% of missing values. We tried to delete all features that have at least one null value, but this method gave bad results.

Finally, we filled out the missing values with other values derived from either the same features or other features. This method is preferable so that it enables us to use the information in most features for the training process. We applied the following:

- ✓ Records that contain more than 90% of missing features were deleted.
- ✓ Features that have more than 70% of missing values were deleted.
- ✓ For the missing categories in categorical features, they were replaced by a new category called ‘Other’.
- ✓ The missing numerical values were replaced with the average of the feature.
- ✓ The number of categorical features were 78, the first 31 most frequent categories were chosen and the remaining categories were replaced with a new category, so the total number is 32 categories.
- ✓ There are some other features with a numeric character but they contain only a limited number of duplicate values in more than one record. This indicates that they are categorical so we have dealt with them as categorical features, but the experiment shows that they perform worse with the model, so that they have been deleted.
- ✓ We have also calculated the correlation between numerical features using Pearson and removed the correlated features. This removal had no effect on the final result. Many other methods were tested, but this applied approach gave the best performance of the four algorithms. The number of features after this operation exceeded 2000 features at the end.

8 APPENDIX

```
#import necessary libraries
import pandas as pd
import  numpy as np
import pickle
import matplotlib. pyplot as plt
%matplotlib inline
import seaborn as sns
import sklearn
from sklearn.preprocessing import LabelEncoder, OneHotEncoder
from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.svm import SVC
from sklearn.model_selection import RandomizedSearchCV
import imblearn
from imblearn.over_sampling import SMOTE
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import accuracy_score,classification_report, confusion_matrix,
f1_score

data=pd.read_csv(r"/Churn_Modelling.csv")
data

#checking for null values
data.TotalCharges= pd.to_numeric(data.TotalCharges,errors= 'coerce')
data.isnull().any ()
```

```
 data["TotalCharges"].fillna(data["TotalCharges"].median() ,inplace =True)
 data.isnull().sum()

 from sklearn.preprocessing import LabelEncoder
 le = LabelEncoder()
 data["gender"] = le.fit_transform(data["gender"])
 data["Partner"] = le.fit_transform(data["Partner"])
 data["Dependents"] = le.fit_transform(data["Dependents"])
 data["PhoneService"] = le.fit_transform(data[ "PhoneService"])
 data["MultipleLines"] = le.fit_transform(data["Multiplelines"])
 data["InternetService"] = le.fit_transform(data["InternetService"])
 data["OnlineSecurity"] = le.fit_transform(data["OnlineSecurity"])
 data["onl ineBackup"] = le.fit_transform (data["OnlineBackup"])
 data["DeviceProtection"] = le.fit_transform(data["DeviceProtection"])
 data["TechSupport"] = le.fit_transform(data["TechSupport"])
 data["StreamingTV"] = le.fit_transform(data["StreamingTV"])
 data["StreamingMovies"] = le.fit_transform(data["StreamingMovies"])
 data["Contract"] = le.fit_transform(data["Contract"])
 data["PaperlessBilling"] = le.fit_transform(data["PaperlessBilling"])
 data["PaymentMethod"] = le.fit_transform(data["PaymentMethod"])
 data["Churn"] = le.fit_transfora (data["Churn"])

 data.head()

 x=data.iloc[:,0:19].values
 y= data.iloc[:,19:20].values
from sklearn.preprocessing import OnetHotEncoder
one = OnetHotEncoder()
 a= one.fit_transform(x[:,6:7]).toarray()
 b= one. fit_transform(x[:,7:8]) .toarray()
 c= one.fit_transform(x[:,8:9]).toarray()
```

```
d= one.fit_transform(x[:,9:10]).toarray ()
e=one.fit_transform(x[:,10:11]). toarray()
f=one.fit_transform(x[:,11:12]).toarray()
g=one.fit_transform(x[:,12:13]).toarray()
h=one.fit_transform(x[:,13:14]). toarray()
i=one. fit_transform(x[:,14:15]).toarray()
j=one.fit_transform(x[:,16:17]).toarray()
x=np.delete (x, [6,7, 8, 9, 10, 11, 12, 13, 14,16], axis=1)
x=np.concatenate((a,b, c,d, e, f,8, h, i,j,x), axis=1)

from imblearn.over_sampling import SMOTE
smt=SMOTE()
x_resample,y_resample=smt.fit_resample(x,y)
x_resample
y_resample
x.shape,x_resample.shape
y.shape,y_resample.shape

data.describe()

plt.figure (figsize=(12,5))
plt.subplot (1,2,1)
sns.distplot (data["tenure"])
plt.subplot (1,2,2)
sns.distplot (data["MonthlyCharges"])

plt.figure (figsize=(12,5))
plt.subplot(1,2,1)
sns.countplot(data["gender"])
plt.subplot(1,2,2)
sns.countplot(data["Dependents"])
sns.barplot(x="Churn",y="MonthlyCharges",data=data
sns.heatmap(data.corr(),annot=True)
sns.pairplot(data=data,markers=["^","v"],palette="inferno")

from sklearn.model_selection import train_test_spit
x_train,x_test,y_train,y_test=train_test_split(x_resample,y_resample,test_size=0.2,random_state=0)

from sklearn.preprocessing import StandardScaler
sc=StandardScaler()
x_train=sc.fit_transform(x_train)
x_test=sc.fit_transform(x_test)

#importing and building the Decision tree model
def logreg(x_train,x_test,y_train,y_test):
lr=LogisticRegression(random_state=0)
lr.fit(x_train,y_train)
y_lr_tr= lr.predict(x_train)
print(accuracy_score(y_lr_ tr,y_train))
yPred_lr=lr.predict(x_test)
print(accuracy_score(yPred_lr,y_test))
print("***Logistic Regression***")
print("Confusion_Matrix")
print(confusion_matrix(y_test,yPred_lr))
print("Classification Report")
print(classification_report(y_test,yPred_lr))

logreg(x_train,x_test,y_train,y_test)
```

```
#importing and building the Decision tree model
def decisionTree(x_train,x_test,y_train,y_test) :
  dtc=DecisionTreeClassifier(criterion="entropy", random_state=0)
  dtc.fit(x_train,y_train)
  y_dt_tr=dtc.predict(x_train)
  print(accuracy_score(y_dt_tr,y_train))
  yPred_dt=dtc.predict(x _test)
  print(accuracy_score(yPred_dt,y_test))
  print ("***Decision Tree***")
  print ("Confusion_Matrix")
  print (confusion_matrix(y_test,yPred_dt))
  print("classification Report")
  print(classification_report(y_test,yPred_dt))

#printing the train accuracy and test accuracy respectively
decisionrTree(x_train,x_test,y_train,y_test)

#importing and bui lding the random forest model
def RandomForest (x_train, x_test, y_train,y_test):
  rf=RandomForestclassifier(criterion="entropy",n_estimators=10,random_state=0)
  rf.fit(x_train,y_train)
  y_rf_tr = rf.predict(x_train)
  print(accuracy_score(y_rf_tr,y_train))
  yPred_rf = rf.predict(x_test)
  print (accuracy_score(yPred_rf, y_test))
  print("***Random Forest***")
  print("Confusion_Matrix")
  print(confusion_matrix(y_test, yPred_rf))
  print("Classification Report")
  print(classification_report(y_test,yPred_rf))
  #printing the train accuracy and test accuracy respectively
  #importing and building the KNN modeldef
  KNN(x_train, x_test,y_train,y_test) :knn =
  KNeighborsClassifier() knn.fit(x_train,y_train)
  y_knn_tr = knn.predict(x_train)
  print(accuracy_score (y_knn_tr,y_train) )
  yPred_knn = knn.predict(x_test)
  print(accuracy_score (yPred_knn,y_test))
  print("***KNN***")
  print("Confusion_Matrix")
  print(confusion_matrix(y_test, yPred_knn))
  print("Classification Report")
  print(classification_report(y_test,yPred_knn))

#printing the train accuracy and test accuracy respectively
KNN (X_train, x_test,y_train,y_test)

#importing and building the random forest model
def svm(x_tarin,x_test,y_train,y_test):
  svm = SVC(kernel ="linear")
  svm.fit(x_train,y_train)
  y_svm_tr = svm. predict (x_train)
  print(accuracy_score(y_svm_tr,y_train))
  yPred_svm = svm. predict (x_test)
  print(accuracy_score(yPred_svm,y_test))
  print(""***Support Vector Machine***")
  print("Confusion_Matrix")
  print(confusion_matrix(y_test,yPred_svm))
  print("Classification Report")
  print(classification_report(y_test,yPred_svm))
```

```
#printing the train accuracy and test accuracy respectively
svm(x_train, x_test,y_train,y_test)

from flask import flask, render_template, request
import keras
from keras.models import load_model
app = flask(_name_)
model = load_model("telecom_churn.h5")

@app.rute('/') # rendering the html template
def home():
 return render_template('home.html')

@app.route('/')
def helloworld():
 return render render_template("base.html")
@app.route('\assement')
def prediction():
 return render render_template(index.html")
@app.route('/predict', methods=['POST'])
def admin():
a=request.form["gender"]
if(a== 'f'):
    a=0
 if(a== 'm'):
     a=1
 b=request.form["srcitizen"]
 if(b== 'n'):
      b=0
  if(a== 'y'):
     b=1
```

```
   c=request.form[" partner"]
   if(c== 'n'):
            c=0

    if(c== 'y'):
        c=1
    d=request.form[" dependents"]
      if(c== 'n'):
         d=0
     if(d== 'y'):
       d=1
     e=request.form[" tenure"]
     d=request.form["phservices"]
     if(f== 'n'):
          f=0
       if(f== 'y'):
          f=1
       e=request.form["multi"]
       if(g== 'n'):

import keras
from keras.models import sequential
from keras.layers import Dense

classifier=Sequential()
classifier.add(Dence(
q=request.form{“plb”]
if (q==’n’):
q=0
if(q==’y’):
q=1
```

```
r=request.form[‘mcharges”]
s=request.form[“tcharges’]
t=[[int(g1),int(g2),int(g3),int(h1),int(h2),int(h3),int(i),int(i2),int(i3),int(j1),int(j2),int(j3)
print(t)
x=model.predict(t)
print(x[0])
if(x[[0]]<=0.5):

y=’no’
return render_template(“predno.html”,z=y)
if(x[[0]]>=0.5)L
y=”Yes”
return render_template(“predyes.html”,z=y)
l1, 12, 13=1,0,0
if (1 == 'nis'):
l1, 12, 13=0,1,0
if (1 == 'y):
l1,12, 13=0,0,1
m= request.form[ "stv"]
if (m =='n'):
m1, m2, m3=1,0,0
if (m =='nis'):
m1, m2, m3=0,1,o
if (m == 'y'):
m1, m2, m3=0,0,1
request.form["smv"]
if (n == 'n'):
n1, n2, n3=1,0,0
if (n =='nis') :
n1, n2, n3-0,1,0
if (n =='y'):
n1,n2, n3=0,O,1
```

```
O= request.form["contract"]
if (o== 'mtm):
01,02,03=1,0,0
if (o == oyr'):
o1,02 , o3=0,1,0
if (o == 'tyrs):
01,02, 03=0,0,1
p= request.form["pmt"]
if (p == 'ec'):
p1,p2, p3, p4=1,0, 0, 0

if (p == 'mail'):
p1, p2, p3, p4-0,1,0,0
if (p == ‘bt'):
p1, p2, p3, p4-0,0, 1,0
if (p == 'cc'):
p1, p2, p3, p4=0,0,0, 1
q= request. form[ "plb"]
if (g == "n'):
gl,g2,g3=1,0,0
if (g == 'nps'):
g1,g2,g3=0,1,o
if (g == 'y"):
g1,g2,g3-0,0,1
h= request. form[ "is"]
if (h == 'dsl '):
hi,h2, h3=1,0,o
if (h == 'fo'):
hi, h2, h3-0,1,o
if (h == 'n'):
h1,h2, h3-0,0,1
i= request.form[ "os"]
```

```
if (i == 'n'):
i1, i2, i3-1,0,o
if (i == 'nis'):
i1, i2, i3-0,1,0
if (i == 'y):
i1,i2, i3=0, e,1
j= request.form["ob" ]
if (Ở == 'n'):
j1,j2, j3=1,0,e
if (j == 'nis'):
j1,j2,j3-0,1,
if (j == 'y"):
j1,j2,j3-0,0, 1
if (i == 'n'):
i1, i2, i3-1,0,o
if (i == 'nis'):
i1, i2, i3-0,1,0
if (i == 'y):
i1,i2, i3=0, e,1
j= request.form["ob" ]
if (Ở == 'n'):
j1,j2, j3=1,0,e
if (j == 'nis'):
j1,j2,j3-0,1,
if (j == 'y"):
j1,j2,j3-0,0, 1

k= request. form[ "dp"]
if (k == 'n):
k1,k2, k3=1,0,e
if (k == 'nis '):
```

```
k1,k2, k3=0,1,0
if (k == 'y'):
k1, k2, k3=0,0,1
l= request.form[ "ts"]
if (1 == 'n'):
11,12,l3=1,0,0
```