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

PREDICTION OF TELECOM CUSTOMER CHURN

Project

Submitted to Department of Computer Science

KAMARAJAR GOVERNMENT ARTS COLLEGE,SURANDAI

(Manonmaniam Sundaranar University,Thirunelveli)

Done By

Team Id:NM2023TMID18446

Team Leader:

ARAVINDRAJA.R -20201061506205

Team Members:

ABIGANESH.P -20201061506202

PAVUNKARTHIKEYAN.R -20201061506235

PAVITH KANCHAN.S -20201061506233

Guided By

Dr.SANTHANA DEVI

Department of Computer Science

KAMARAJAR GOVERNMENT ARTS COLLEGE

SURANDAI-627859

APRIL 2023

CONTENTS:

1.INTROCTION

Customer churn prediction is a critical task in the field of telecommunications. It refers to the process of identifying customers who are likely to terminate their subscription or discontinue their services with a telecom service provider. Churn prediction is important for telecom companies as it allows them to proactively take measures to retain their existing customers and reduce customer turnover.

The telecommunications industry is highly competitive, with multiple service providers vying for customers. Acquiring new customers can be expensive, and retaining existing customers is often more cost-effective. By predicting customer churn, telecom companies can take proactive actions, such as targeted marketing campaigns, personalized offers, and improved customer service, to retain customers and prevent them from switching to a competitor.

Customer churn prediction involves analyzing various data points, such as customer demographics, usage patterns, billing information, customer service interactions, and historical churn data, to build predictive models. These models can identify patterns and trends that indicate a customer's likelihood to churn. Machine learning and data analytics techniques are commonly used in customer churn prediction to analyze large amounts of data and generate accurate predictions.

1.1 Overview

The project involves collecting and analyzing large amounts of historical data from various sources, such as customer demographics, usage patterns, service plans, call records, and customer complaints. Data preprocessing techniques, including data cleaning, feature engineering, and data integration, are applied to prepare the data for analysis. Exploratory data analysis (EDA) techniques are used to gain insights into the data, identify trends, and detect patterns related to customer churn.

Several machine learning algorithms, such as logistic regression, decision trees, random forests, support vector machines, and neural networks, are employed to build predictive models. These models are trained on a subset of the data and evaluated using various performance metrics, such as accuracy, precision, recall, and F1-score, to assess their predictive performance. Model tuning techniques, such as cross-validation and hyper parameter optimization, are used to optimize model performance.

Once the best-performing model is selected, it is deployed to a production environment to predict customer churn in real-time or near real-time. The insights generated from the analysis and predictions are used to develop retention strategies, targeted marketing campaigns, and proactive customer engagement initiatives to reduce customer churn and improve customer retention rates. The results of the project are communicated through reports, visualizations, and presentations to stakeholders, providing valuable insights for decision-making and strategic planning in the telecom industry.

1.2 Purpose

Retaining Customers: One of the main objectives of churn prediction is to identify customers who are at risk of leaving the telecom service provider and take proactive measures to retain them. By analyzing customer data and identifying patterns that indicate potential churn, telecom companies can implement targeted retention strategies such as personalized offers, discounts, or loyalty programs to incentivize customers to stay.

Improving Customer Experience: Churn prediction can help telecom companies identify pain points in their customer experience journey that may be leading to customer dissatisfaction and churn. By analyzing customer feedback, service usage patterns, and other relevant data, telecom companies can make data-driven improvements to their services, products, and customer support, leading to higher customer satisfaction and loyalty.

Enhancing Marketing Strategies: Churn prediction can provide insights into the characteristics and behaviors of customers who are more likely to churn. This information can be used to improve marketing strategies by targeting specific customer segments with tailored marketing campaigns, promotions, and communication channels. Telecom companies can optimize their marketing efforts, allocate resources more efficiently, and increase the effectiveness of their marketing campaigns.

Cost Reduction: Acquiring new customers is generally more expensive than retaining existing ones. Churn prediction can help telecom companies identify customers who are likely to churn and take preventive measures to reduce churn rates.

2.PROBLEM DEFINITION &DESIGN THINKING

Empathy maps & Ideation & Brainstorming Map are useful tools for understanding and empathizing with customers, in this case, telecom customers who have churned or canceled their services.

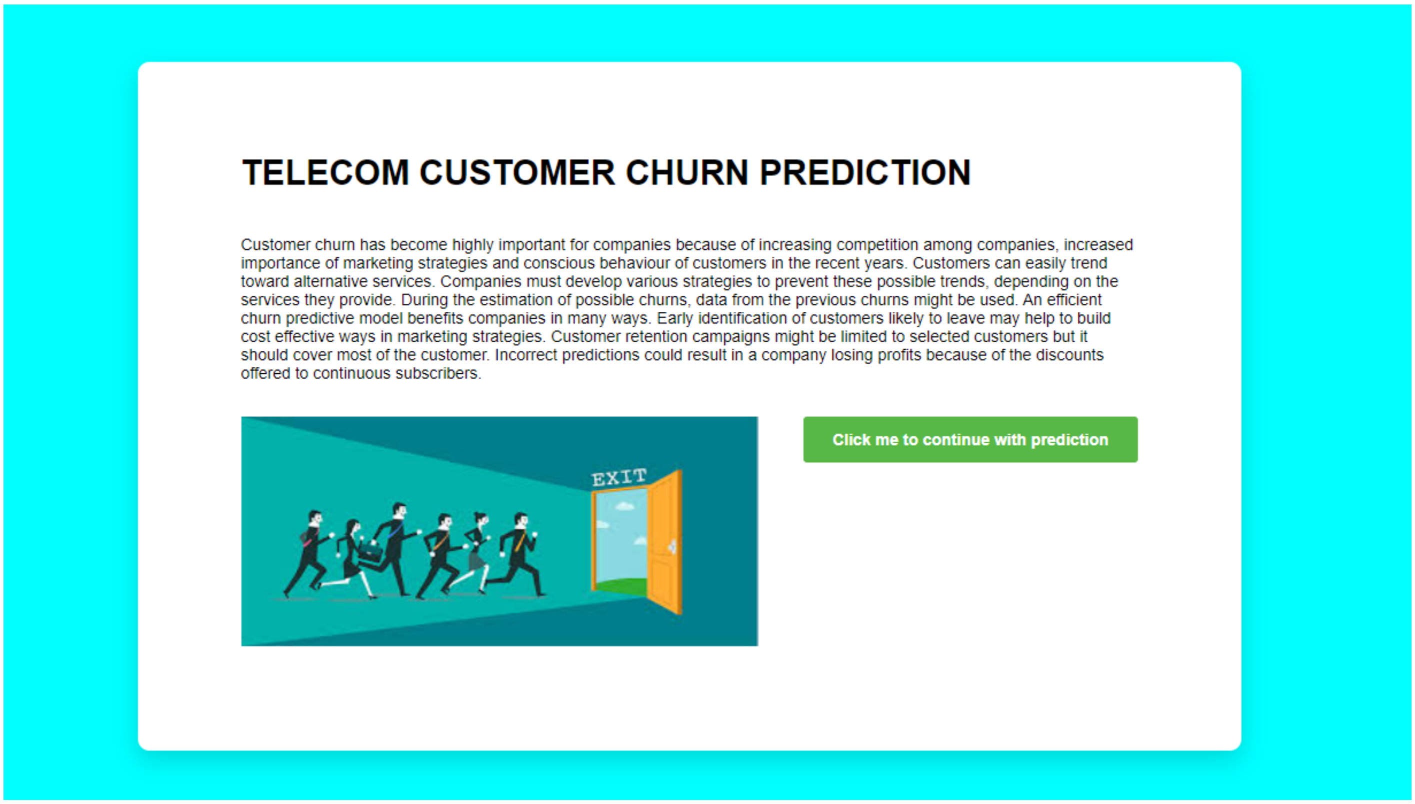
2.1 Empathy Map

2.2 Ideation & Brainstorming Map

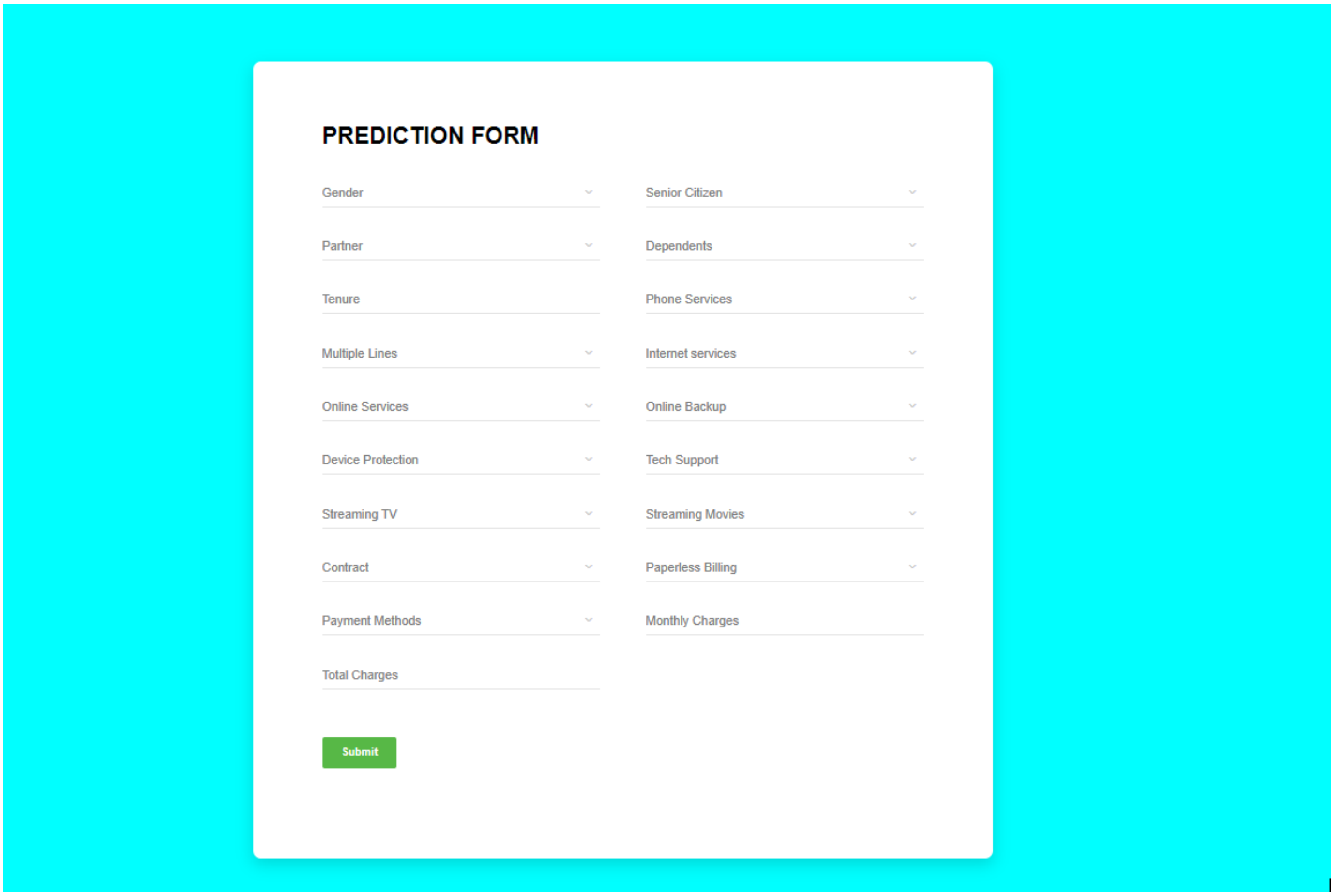
3.RESULT

Go to web browser and write the localhost URL (http://127.0.0.1:5000) to get the below result

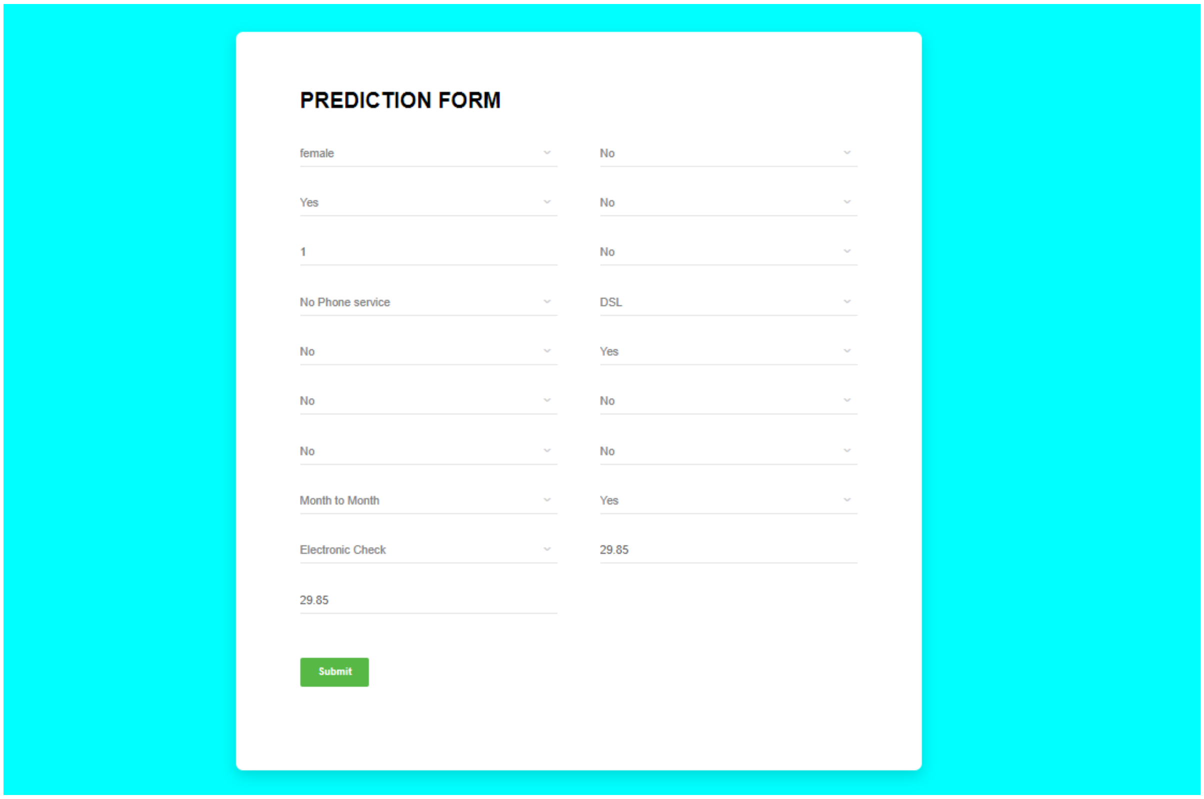
Home page



Prediction Form page

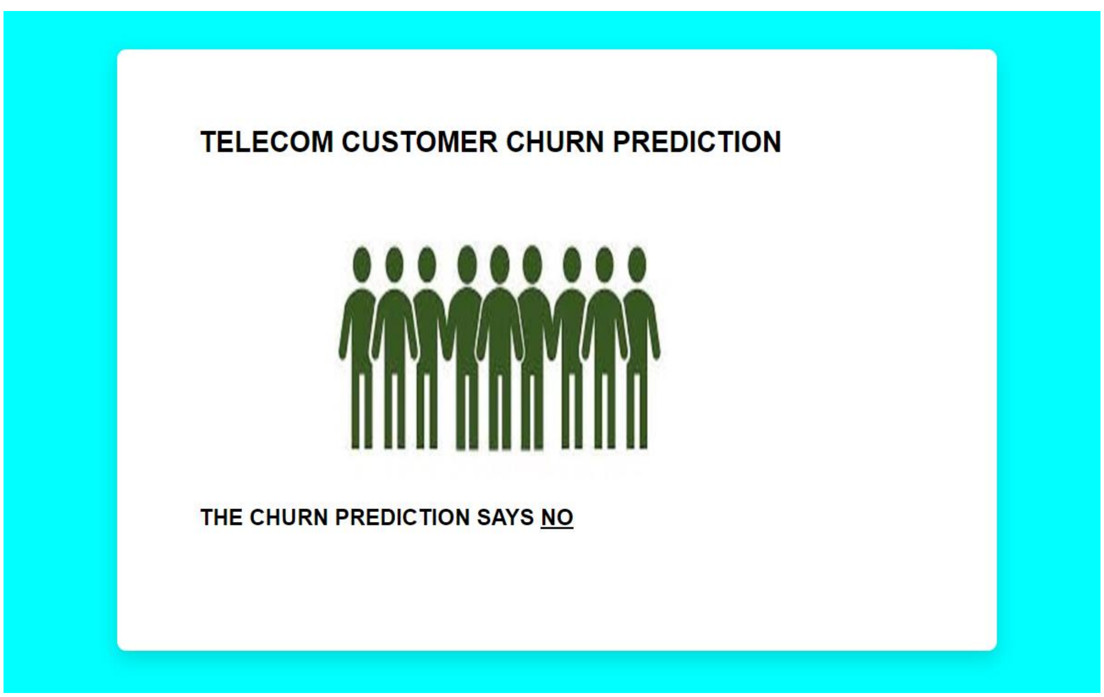


The Churn Prediction Says (NO)

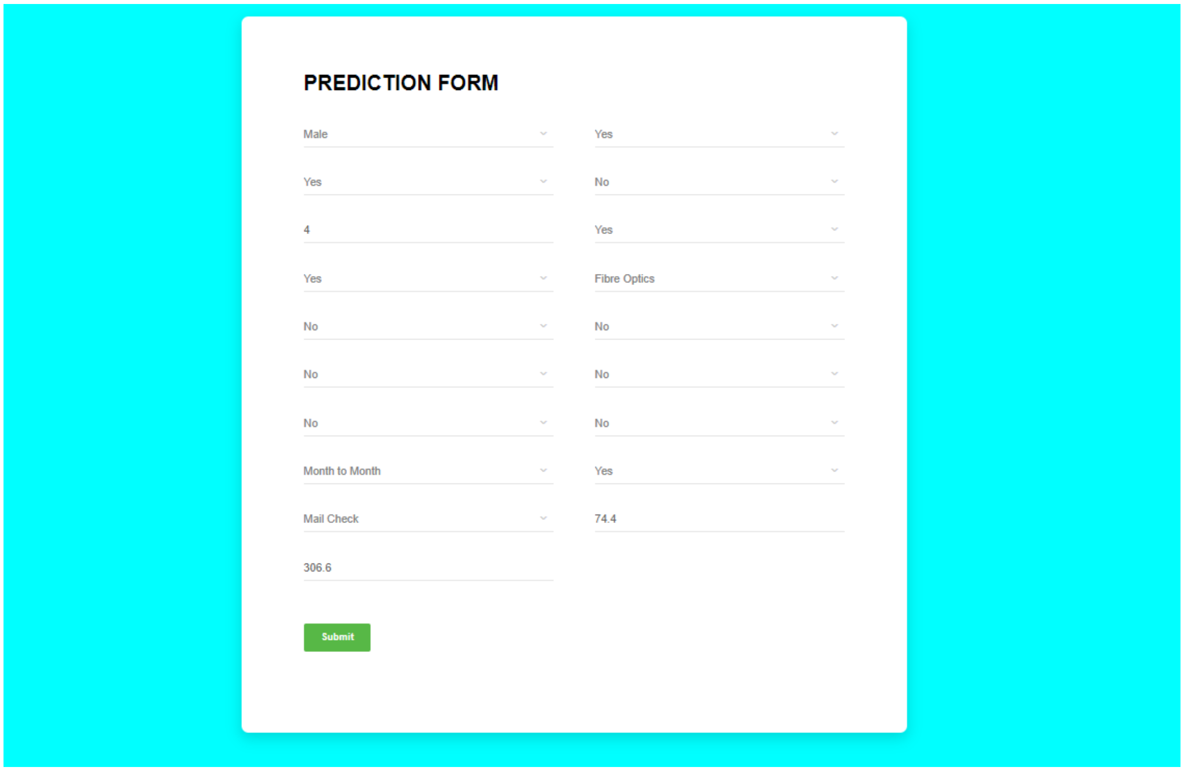


Input the predicted values:

example (Female,No,Yes,No,1,No,No PhoneServices,DSL,No,Yes,No,No, No,No,Month to Month,Yes,Electronic Check,29.85,29.85)

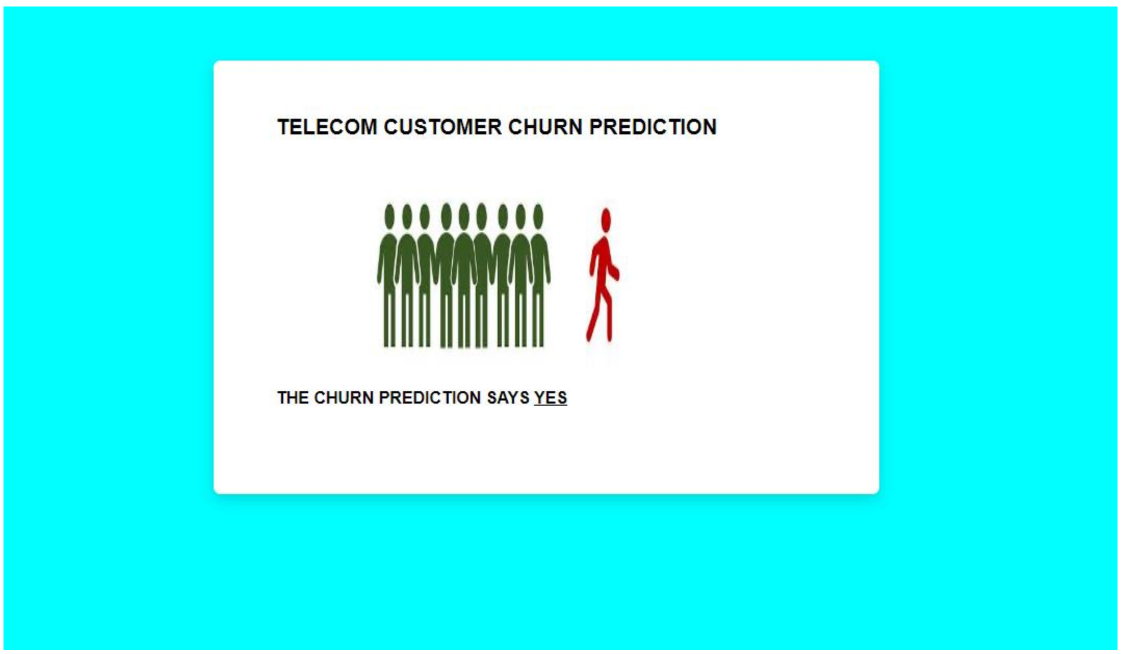


The Churn Prediction Says(YES)



Input the predicted values:

Example (Male,Yes,Yes,No,4,Yes,Yes,Fiber Optics,No,No,No,No,No,No,Month to Month,yes,mail check,74.4,306.6)



4.ADVANTAGES & DISADVANTAGES

4.1 Advantages:

Improved Customer Retention: By accurately predicting customer churn, telecom companies can take proactive measures to retain customers who are at risk of leaving. This may include targeted offers, personalized promotions, or tailored retention strategies, which can help improve customer retention rates and reduce customer churn.

Cost Savings: Acquiring new customers is typically more expensive than retaining existing ones. By predicting customer churn and taking preventive actions, telecom companies can potentially save on customer acquisition costs and reduce the need to spend resources on acquiring new customers to replace the ones that churned.

Enhanced Customer Experience: By leveraging customer churn prediction, telecom companies can identify the factors that drive customer churn and take corrective actions to address them. This can result in an improved customer experience, as companies can proactively address customer concerns and pain points, leading to higher customer satisfaction and loyalty.

Data-Driven Decision Making: Customer churn prediction relies on data analysis and machine learning algorithms, which enable telecom companies to make data-driven decisions. This can result in more effective and efficient marketing strategies, targeted campaigns, and resource allocation, as decisions are based on insights derived from customer data.

4.2 Disadvantages:

Data Privacy Concerns: Customer churn prediction requires access to large amounts of customer data, including personal and sensitive information. This can raise concerns about data privacy and security, and telecom companies need to ensure that they comply with relevant data protection regulations and implement appropriate data security measures to protect customer information.

Accuracy and Reliability: Customer churn prediction models are not always 100% accurate and reliable. False positives (predicting a customer will churn when they won't) and false negatives (predicting a customer won't churn when they will) can occur, leading to potential misinterpretation of results and incorrect decision-making. Ensuring the accuracy and reliability of churn prediction models may require continuous monitoring, validation, and refinement.

Implementation Challenges: Implementing a churn prediction solution may require significant investments in technology infrastructure, data integration, and skilled resources to develop and maintain the predictive models. This can be challenging for smaller telecom companies with limited resources or technical expertise.

Ethical Considerations: The use of customer churn prediction raises ethical concerns, such as fairness, bias, and discrimination. It is important to ensure that the models used for churn prediction do not perpetuate or exacerbate existing biases, and that decision-making based on churn prediction is fair and transparent.

5.APPLICATION

Retention Strategies: Telecom companies can use churn prediction models to identify customers who are at high risk of churning and implement targeted retention strategies. This can involve offering personalized discounts, promotions, or loyalty rewards to incentivize customers to stay with the telecom provider.

Customer Segmentation: Churn prediction models can help telecom companies segment their customer base based on churn risk. This can enable companies to allocate resources more effectively by focusing their retention efforts on high-value customers who are at high risk of churning, rather than applying a one-size-fits-all approach to all customers.

Service Improvement: Churn prediction models can identify the main reasons why customers are leaving, such as poor network quality, billing issues, or customer service problems. This information can help telecom companies identify areas of improvement in their services and take proactive steps to address these issues, ultimately reducing customer churn.

Marketing Campaigns: Churn prediction models can also be used to optimize marketing campaigns by targeting promotions, discounts, and offers to customers who are at high risk of churning. This can help telecom companies improve customer retention by providing tailored incentives to customers who are most likely to leave.

Product Development: Churn prediction models can provide insights into customer preferences and behavior, which can be used to inform product development strategies. By understanding the needs and preferences of customers who are at high risk of churning, telecom companies can develop new products or features that meet their needs and increase customer satisfaction.

6.CONCLUSION

In conclusion, telecom customer churn prediction is a valuable tool that can be applied across different areas within the telecom industry. By leveraging advanced analytics and machine learning techniques, churn prediction models can help telecom companies identify customers who are at high risk of churning and take proactive measures to retain them. This can include implementing targeted retention strategies, optimizing marketing campaigns, improving services, and enhancing the overall customer experience. The application of churn prediction models can lead to increased customer retention, improved customer satisfaction, and ultimately, better business performance for telecom companies.

7.SCOPE OF FUTURE ENHANCEMENT

7.1 Scope

the scope of telecom customer churn prediction encompasses various stages, from data collection and preparation to model development, evaluation, deployment, monitoring, and maintenance, with a focus on generating actionable insights and recommendations to reduce customer churn and improve business performance. Ethical and legal considerations are also important aspects of the scope of churn prediction.

7.2 Future Enhancement

future enhancements in telecom customer churn prediction could involve leveraging advanced machine learning techniques, big data, and real-time data processing, personalization and contextualization, integration of multiple data sources, explainable AI, proactive churn prevention strategies, continuous model monitoring and maintenance, and ethical and responsible AI practices. These enhancements can contribute to more accurate, effective, and ethical churn prediction models that help telecom companies improve customer retention and business performance.

8.APPENDIX

8.1 Source Code

PREDICTION OF TELECOM CUSTOMER CHURN

PREDICTION OF TELECOM CUSTOMER CHURN

[1]

```
#import necessary libraries

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.preprocessing import LabelEncoder
from sklearn.preprocessing import OneHotEncoder
from imblearn.over_sampling import SMOTE
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.svm import SVC
from sklearn.metrics import accuracy_score
from sklearn.metrics import confusion_matrix
import sklearn.metrics as metrics
import matplotlib.pyplot as plt
import pickle
```

[2]

```
#Read the dataset

data=pd.read_csv('WA_Fn-UseC_-Telco-Customer-Churn.csv')
```

[3]

```
data.head()
```

	customerID	gender	SeniorCitizen	Partner	Dependents	tenure	PhoneService	MultipleLines	InternetService	OnlineSecurity	...	DeviceProtection	TechSu
0	7590-VHVEG	Female	0	Yes	No	1	No	No phone service	DSL	No	...	No	No
1	5575-GNVDE	Male	0	No	No	34	Yes	No	DSL	Yes	...	Yes	No
2	3668-QPYBK	Male	0	No	No	2	Yes	No	DSL	Yes	...	No	No
3	7795-CFOCW	Male	0	No	No	45	No	No phone service	DSL	Yes	...	Yes	Yes
4	9237-HQITU	Female	0	No	No	2	Yes	No	Fiber optic	No	...	No	No

5 rows × 21 columns

```python
#checking the null values

data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7043 entries, 0 to 7042
Data columns (total 21 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   customerID        7043 non-null   object
 1   gender            7043 non-null   object
 2   SeniorCitizen     7043 non-null   int64
 3   Partner           7043 non-null   object
 4   Dependents        7043 non-null   object
 5   tenure            7043 non-null   int64
 6   PhoneService      7043 non-null   object
 7   MultipleLines     7043 non-null   object
 8   InternetService   7043 non-null   object
 9   OnlineSecurity    7043 non-null   object
 10  OnlineBackup      7043 non-null   object
 11  DeviceProtection  7043 non-null   object
 12  TechSupport       7043 non-null   object
 13  StreamingTV       7043 non-null   object
 14  StreamingMovies   7043 non-null   object
 15  Contract          7043 non-null   object
 16  PaperlessBilling  7043 non-null   object
 17  PaymentMethod     7043 non-null   object
 18  MonthlyCharges    7043 non-null   float64
 19  TotalCharges      7043 non-null   object
 20  Churn             7043 non-null   object
```

```python
data.isnull().any()
```

```
customerID          False
gender              False
SeniorCitizen       False
Partner             False
Dependents          False
tenure              False
PhoneService        False
MultipleLines       False
InternetService     False
OnlineSecurity      False
OnlineBackup        False
DeviceProtection    False
TechSupport         False
StreamingTV         False
StreamingMovies     False
Contract            False
PaperlessBilling    False
PaymentMethod       False
MonthlyCharges      False
TotalCharges        False
Churn               False
dtype: bool
```

[6]
```python
data['TotalCharges']=pd.to_numeric(data['TotalCharges'],errors='coerce')
```

```python
data.head()
```

	customerID	gender	SeniorCitizen	Partner	Dependents	tenure	PhoneService	MultipleLines	InternetService	OnlineSecurity	...	DeviceProtection	TechSupp
0	7590-VHVEG	Female	0	Yes	No	1	No	No phone service	DSL	No	...	No	No
1	5575-GNVDE	Male	0	No	No	34	Yes	No	DSL	Yes	...	Yes	No
2	3668-QPYBK	Male	0	No	No	2	Yes	No	DSL	Yes	...	No	No
3	7795-CFOCW	Male	0	No	No	45	No	No phone service	DSL	Yes	...	Yes	Yes
4	9237-HQITU	Female	0	No	No	2	Yes	No	Fiber optic	No	...	No	No

5 rows × 21 columns

[8]

```python
#Descriptive statistical
data.describe()
```

	SeniorCitizen	tenure	MonthlyCharges	TotalCharges
count	7043.000000	7043.000000	7043.000000	7032.000000
mean	0.162147	32.371149	64.761692	2283.300441
std	0.368612	24.559481	30.090047	2266.771362
min	0.000000	0.000000	18.250000	18.800000
25%	0.000000	9.000000	35.500000	401.450000
50%	0.000000	29.000000	70.350000	1397.475000
75%	0.000000	55.000000	89.850000	3794.737500
max	1.000000	72.000000	118.750000	8684.800000

[9]

```python
data['TotalCharges'].fillna(data['TotalCharges'].median(),inplace=True)
```

[10]

```python
data.isnull().any()
```

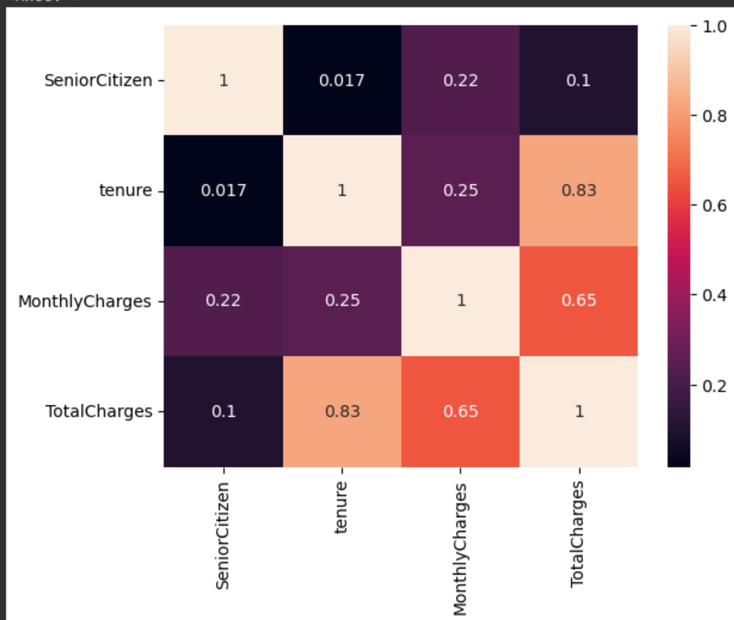
```
customerID          False
gender              False
SeniorCitizen       False
Partner             False
Dependents          False
tenure              False
PhoneService        False
MultipleLines       False
InternetService     False
OnlineSecurity      False
OnlineBackup        False
DeviceProtection    False
TechSupport         False
StreamingTV         False
StreamingMovies     False
Contract            False
PaperlessBilling    False
PaymentMethod       False
MonthlyCharges      False
TotalCharges        False
Churn               False
dtype: bool
```

Multivariate analysis

[11]

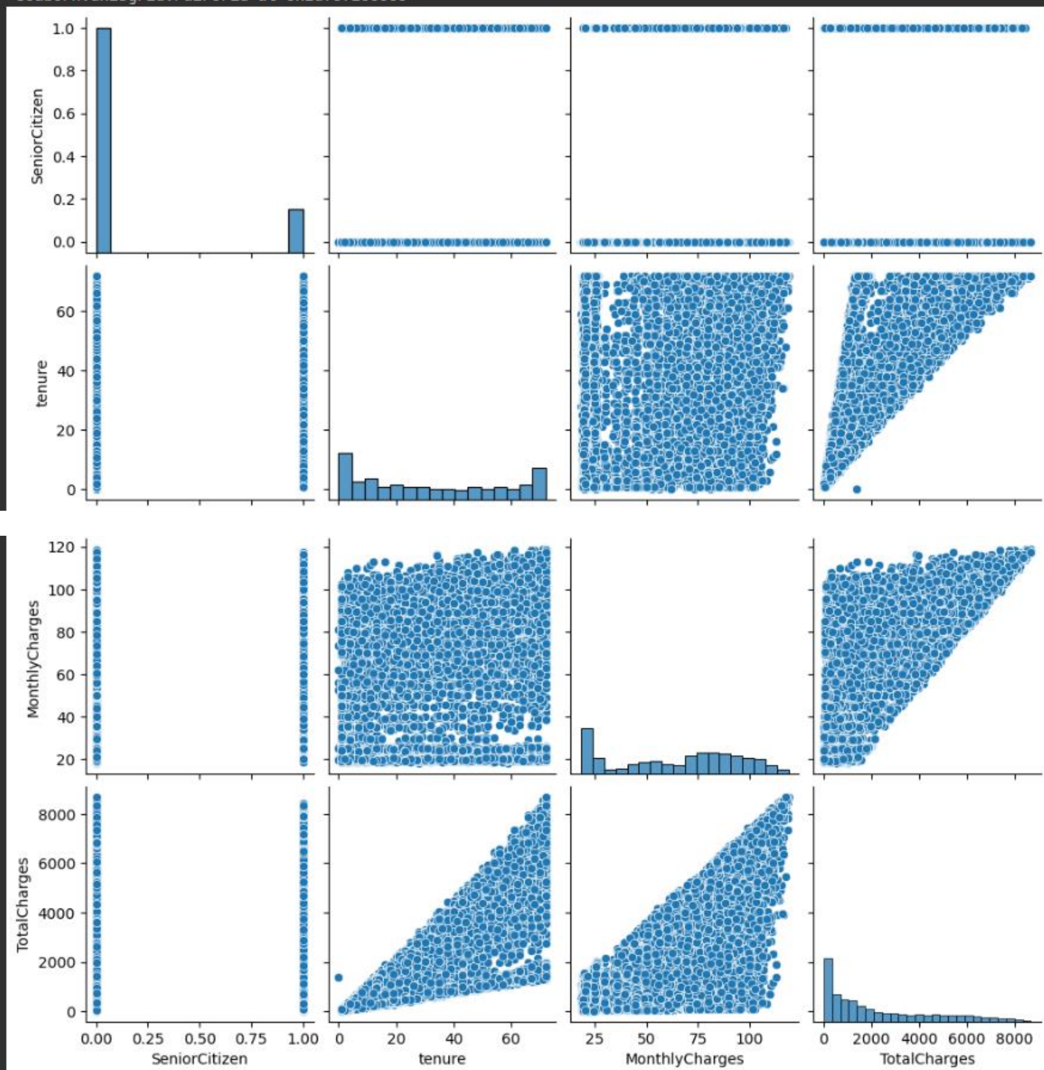
```
sns.heatmap(data.corr(),annot=True)
```

<Axes: >



```
sns.pairplot(data=data,markers=["^","v"],palette="inferno")
```

```
<seaborn.axisgrid.PairGrid at 0x1d7891356c0>
```



[13]

```
#checking correlation

data.corr()
```

```
C:\Users\HP\AppData\Local\Temp\ipykernel_21124\17054435.py:3: FutureWarning: The default value of numeric_only i
  data.corr()
```

	SeniorCitizen	tenure	MonthlyCharges	TotalCharges
SeniorCitizen	1.000000	0.016567	0.220173	0.102652
tenure	0.016567	1.000000	0.247900	0.825464
MonthlyCharges	0.220173	0.247900	1.000000	0.650864
TotalCharges	0.102652	0.825464	0.650864	1.000000

[14]

```
data.shape
```

```
(7043, 21)
```

Unique values

[15]
```
data['gender'].unique()
```
```
array(['Female', 'Male'], dtype=object)
```

[16]
```
data['Partner'].unique()
```
```
array(['Yes', 'No'], dtype=object)
```

[17]
```
data['Dependents'].unique()
```
```
array(['No', 'Yes'], dtype=object)
```

[18]
```
data['PhoneService'].unique()
```
```
array(['No', 'Yes'], dtype=object)
```

[19]
```
data['MultipleLines'].unique()
```

```
data['Contract'].unique()
```
```
array(['Month-to-month', 'One year', 'Two year'], dtype=object)
```

[28]
```
data['PaperlessBilling'].unique()
```
```
array(['Yes', 'No'], dtype=object)
```

[29]
```
data['PaymentMethod'].unique()
```
```
array(['Electronic check', 'Mailed check', 'Bank transfer (automatic)',
       'Credit card (automatic)'], dtype=object)
```

[30]
```
data['MonthlyCharges'].unique()
```
```
array([29.85, 56.95, 53.85, ..., 63.1 , 44.2 , 78.7 ])
```

[31]
```
data['Churn'].unique()
```
```
array(['No', 'Yes'], dtype=object)
```

[32]

```
#Label Encoding
le=LabelEncoder()
```

[33]

```
data['gender']=le.fit_transform(data['gender'])
data['Partner']=le.fit_transform(data['Partner'])
```

[34]

```
data['Dependents']=le.fit_transform(data['Dependents'])
data['MultipleLines']=le.fit_transform(data['MultipleLines'])
data['PhoneService']=le.fit_transform(data['PhoneService'])
data['InternetService']=le.fit_transform(data['InternetService'])
data['OnlineSecurity']=le.fit_transform(data['OnlineSecurity'])
data['OnlineBackup']=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_transform(data['Churn'])
```

[35]

```
# Data after label encoding
data
```

	customerID	gender	SeniorCitizen	Partner	Dependents	tenure	PhoneService	MultipleLines	InternetService	OnlineSecurity	...	DeviceProtection	TechS
0	7590-VHVEG	0	0	1	0	1	0	1	0	0	...	0	0
1	5575-GNVDE	1	0	0	0	34	1	0	0	2	...	2	0
2	3668-QPYBK	1	0	0	0	2	1	0	0	2	...	0	0
3	7795-CFOCW	1	0	0	0	45	0	1	0	2	...	2	2
4	9237-HQITU	0	0	0	0	2	1	0	1	0	...	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...
7038	6840-RESVB	1	0	1	1	24	1	2	0	2	...	2	2
7039	2234-XADUH	0	0	1	1	72	1	2	1	0	...	2	0
7040	4801-JZAZL	0	0	1	1	11	0	1	0	2	...	0	0
7041	8361-LTMKD	1	1	1	0	4	1	2	1	0	...	0	0
7042	3186-AJIEK	1	0	0	0	66	1	0	1	2	...	2	2

7043 rows × 21 columns

[36]

```
#drop the 'customer id'
data.drop(columns='customerID', inplace=True)
```

splitting the dataset into dependent & independent variable

[37]
```
#independent variable

x=data.iloc[:,:19].values
```

[38]
```
data['Churn'].value_counts()
```
```
0    5174
1    1869
Name: Churn, dtype: int64
```

[39]
```
x.shape
```
```
(7043, 19)
```

```
# dependent variable

y=data.iloc[:,19:20].values
```

[41]
```
y
```
```
array([[0],
       [0],
       [1],
       ...,
       [0],
       [1],
       [0]])
```

[42]
```
#OneHot Encoding
one=OneHotEncoder()
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,g,h,i,j,x), axis=1)
```

[43]
```
x.shape
```
```
(7043, 40)
```

Handling imbalance data

[44]
```
#handling the imbalance data using SMOTE
smt = SMOTE()

x_resample, y_resample = smt.fit_resample(x,y)
```

[45]
```
x_resample.shape
```
```
(10348, 40)
```

[46]
```
y_resample
```
```
array([0, 0, 1, ..., 1, 1, 1])
```

[47]

```
x_resample.shape
```

```
(10348, 40)
```

splitting data into train and test

[48]

```
x_train,x_test,y_train,y_test = train_test_split(x_resample,y_resample,test_size = 0.2, random_state=0)
```

[49]

```
print(x_train.shape)
print(x_test.shape)
print(y_train.shape)
print(y_test.shape)
```

```
(8278, 40)
(2070, 40)
(8278,)
(2070,)
```

Model Building

Support Vector Machine

[50]

```
sc = StandardScaler()
x_train = sc.fit_transform(x_train)
x_test = sc.fit_transform(x_test)
```

[51]

```
x_train
```

```
array([[-0.98053789, -0.32159712,  1.17248385, ..., -1.40758562,
         1.3144004 ,  0.35279623],
       [ 1.09450139, -0.32159712, -0.91081472, ..., -1.40758562,
         0.41957265,  0.44867181],
       [-0.98053789, -0.32159712,  1.17248385, ..., -1.40758562,
        -1.54019694, -0.13923035],
       ...,
       [ 1.09450139, -0.32159712, -0.91081472, ...,  0.76674364,
         0.79584841, -0.50290366],
       [ 1.09450139, -0.32159712, -0.91081472, ...,  0.76674364,
         0.10251587, -0.9087194 ],
       [ 1.09450139, -0.32159712, -0.91081472, ..., -1.40758562,
        -1.66736421, -0.89813258]])
```

[52]

```
x_test
```

```
array([[-0.97438469, -0.32996565,  1.17256325, ...,  0.76518784,
        -0.28012765, -0.91485212],
       [ 1.10362921, -0.32996565, -0.9111073 , ...,  0.76518784,
        -1.70896718, -0.78181417],
       [-0.97438469, -0.32996565,  1.17256325, ...,  0.76518784,
         0.85368197, -0.34087105],
       ...,
       [-0.97438469, -0.32996565,  1.17256325, ..., -1.41121977,
         0.43516443,  0.82601279],
       [ 1.10362921, -0.32996565, -0.9111073 , ...,  0.76518784,
        -1.6758195 , -0.85327931],
       [-0.97438469, -0.32996565,  1.17256325, ...,  0.76518784,
         1.28277441,  0.433946  ]])
```

[53]

```
svm=SVC(kernel='linear')
svm.fit(x_train,y_train)
```

```
SVC
SVC(kernel='linear')
```

[54]

```
svm_pred=svm.predict(x_test)
svm_pred
```

```
array([1, 0, 1, ..., 1, 0, 1])
```

[55]

```
#accuracy score
svm_acc=accuracy_score(svm_pred,y_test)
svm_acc
```

```
0.7541062801932367
```

[56]

```
#confution matrix
svm_cm=confusion_matrix(svm_pred,y_test)
svm_cm
```

```
array([[703, 179],
       [330, 858]], dtype=int64)
```

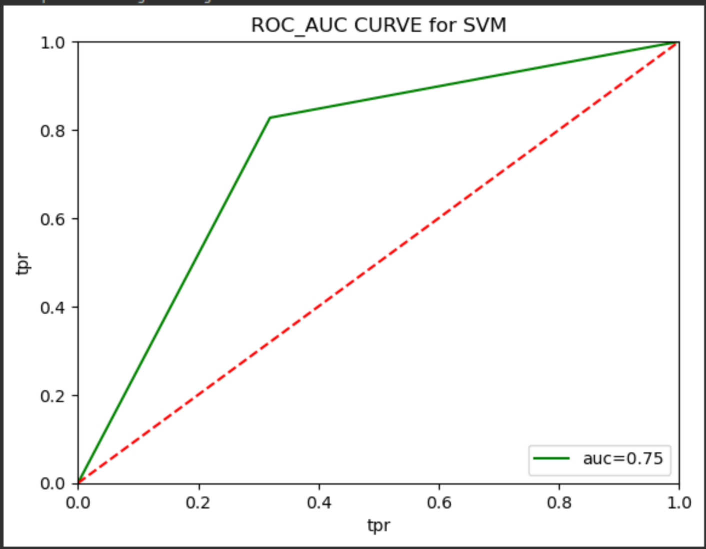
[57]

```
#Visual the svm model
fpr,tpr,threshold=metrics.roc_curve(y_test,svm_pred)
roc_auc=metrics.auc(fpr,tpr)
```

[58]

```
plt.title("ROC_AUC CURVE for SVM")
plt.plot(fpr,tpr,'g',label='auc=%0.2f'%roc_auc)
plt.plot([0,1],[0,1],'r--')
plt.xlim([0,1])
plt.ylim([0,1])
plt.xlabel('tpr')
plt.ylabel('tpr')
plt.legend(loc='lower right')
```

```
<matplotlib.legend.Legend at 0x1d78a86ace0>
```



save the pickle file

[59]

```
pickle.dump(svm,open('telecom-churn.pkl','wb'))
```