**CUSTOMER ATTRITION PREDICTION USING MACHINE LEARNING**

A

Report Submitted in partial fulfilment of the

Requirements for the award of the Degree of

**BACHELOR OF ENGINEERING**

**IN INFORMATION TECHNOLOGY**

By

**M VANDITH REDDY<1602-19-737-056>**

**KVS SIDDHARTHA<1602-19-737-058>**

**K DEEPAK SRINIVAS<1602-19-737-009>**



Under the Guidance of

Dr. K Ram Mohan Rao

B. Leelavathy

Department of Information Technology

Vasavi College of Engineering (Autonomous)

(Affiliated to Osmania University)

Ibrahim Bagh, Hyderabad-31

2021

**BONAFIDE CERTIFICATE**

This is to certify that the project report titled “Customer Attrition Prediction Using Machine Learning” project work of Mr. M. Vandith Reddy, KVS Siddhartha, K Deepak Srinivas bearing Roll.no:1602-19-737-056,058,009 who carried out this project under my supervision in the V semester for the academic year 2021- 2022.

Signature Signature

External Examiner Internal Examiner

**ABSTRACT**

Do you know how much it costs to get a new customer? Did you ever think of which would be the best out of keeping your old customers or getting new? What will happen if you do not consider the attrition rate of your Service?

Customer Attrition Prediction will help the services know how their service is being received by calculating the attrition rate. Attrition Rate is the metric that will say how many customers are leaving your service. Attrition Rate helps the services know how many customers would not buy your goods/ use your service which would stand as a parameter for their further improvement in supplying goods/ providing services.

The goal of our project is to predict this rate using machine learning algorithms such as RandomForest, SVM, Naive Bayes.

# Introduction

## Description in general

In general, our problem statement says that it could calculate the possibility of the customer leaving the service or being in it/ continuing the service, this is attrition rate.

So, for the calculation of this attrition rate, we take a sample dataset consisting of parameters that are appropriate for the chosen field of dataset.

We generally use Machine Learning algorithms in order to predict this of rate using a given set, by either building a web-based app or normal mobile app, we create an interface that takes in some values for the given displayed parameters and then calculate the attrition rate by using the model that we have built by training it using Machine Learning Algorithms.

We generally decide what are the algorithms to adopt by testing all of them using the data set and compare their accuracies and choose them.

In general, the steps that are included in Machine Learning are:

* Importing the Data, which is the input to our model
* Cleaning the Data, as data should not be duplicated, incomplete, incompatible or ambiguous because they result bad patterns for that dataset and hence could not predict the correct result.
* Split the Data into a training and testing sets, where training set is used as input to model while testing set is used for testing the designed model
* Creating a model, which involves selection of algorithm that suits to our project or that is appropriate to our model.
* Training a model, where model looks for the patterns in the data provided
* Making Predictions based on the patterns developed and using test dataset
* Evaluate the model and improve based on what is not correct

## Description of that is specific to our Project

* We prepare the data that has all the necessary attributes that are used in predicting the Customer Attrition including the Target Variable (Attrition Rate)
* We create a model using Sequential Keras Modelling, execute and train it then evaluate and determine the training and testing accuracies (using ROC AUC score)
* Then we use several Machine Learning Models such as Naive Bayes, K-Neighbors, Cat Boost, Gradient Boosting, RandomForest and calculate their ROC-AUC, Accuracy Mean and select high accurate Model for Data Segmentation
* In the next Step, we categorize customers into various classes based on their churn rate, like very low attrition, low attrition, medium attrition, high attrition, very high attrition rate.
* During the Data Segmentation Customers with attrition rate greater than 60 are chosen and we see if they are profitable. For this Customer Lifetime value is calculated to know which customer to Retain.
* Corresponding company takes necessary actions to retain that set of customers.

As of now, we have developed a User Interface that has 5 choices to select clicking on which display the functionality of them by taking the values of the data given by the user and the dataset that is used for the project in some use cases.

# Proposed Work

## Use Cases:

1. ***Exploratory Data Analysis***:

This use case would be used in order to explore the data set that we have chosen.

It has option like

**Show Dataset**: which shows the entire dataset opted for the project

**Head**: this displays the first five data values for every parameter in the dataset.

**Tail**: this displays the last five data values for every parameter in the dataset.

**Shape**: this displays the total number of columns and rows that there in our dataset.

**Describe**: it is the option that displays the result of the mathematical function upon the numerical valued data in the dataset, these mathematical functions include average, maximum value, minimum value, standard deviation, count of them, percentage of people with some value.

**Missing Value**: it lists out the number of missing values for the given dataset parameters.

**Column Names**: it displays all the columns that are present in the dataset.

**Select Column to show:** which takes the choice of the user and display the values of that column.

1. ***Data Visualization:***

This is option is used when the user wants to know like what the amount of people is who opt some specific value for that column in figures like histogram, line chart, bar chart.

This includes two options:

**Numerical Variables**: which have got some numerical value, and has columns listed upon clicking which results the charts, graphs that would better describe their need.

**Categorical Variables**: these are characterized based on the parameter and has options listed displays the appropriate graphs upon clicking.

1. ***Prediction:***

This is the core use-case of our project which would describe the attrition possibility.

This option asks the users what all they must give in order to predict the result using the model trained, as of now we have just provided a way to take the values upon clicking, and these would be made work after training our model which involves the knowledge of machine learning. It displays whether he/she can be retained or not.

1. ***Model Statistics***:

This is the use-case that would be used to know how accurate the model is to predict the result and show the report of what is generated.

## 2) UI Prototype/ Use-case diagram:

Diagram

Description automatically generated

## 3) Architecture and Design:

Diagram

Description automatically generated

## 4. Technologies Used:

### Software:

**Libraries:** Streamlit library for Frontend and Backend, Numpy, MatPlotLib, Pandas, Seaborn are the other libraries.

**IDE**: Visual Studio Code

**Data Set**: from Kaggle of Telcom service (You can get it in reference)

### Hardware:

Computer with windows 10 OS and i3 processor.

## 5. Implementation:

**For EDT**:

def EDA():

st.header("Exploratory Data Analysis")

method\_names=["Show dataset","Head","Tail","Shape","Describe","Missing value","Columns Names"]

method\_operation=[data,data.head(),data.tail(),data.shape,data.describe(),data.isnull().sum(),data.columns]

for i in range(len(method\_names)):

if st.checkbox(method\_names[i]):

st.write(method\_operation[i])

all\_columns=list(data.columns)

if st.checkbox("Select columns to show"):

selected\_columns=st.multiselect("Select column",all\_columns)

new\_df=data[selected\_columns]

st.dataframe(new\_df)

**For Data Visualization**:

def DataVisualization():

st.header("Data Visualization")

if st.checkbox("Numerical variable"):

column\_name=st.selectbox("",("Select column","MonthlyCharges","SeniorCitizen","TotalCharges","Age"))

if column\_name=="MonthlyCharges":

plt.figure(figsize=(5,3))

st.set\_option('deprecation.showPyplotGlobalUse', False)

st.write(plt.hist(data[column\_name],color="skyblue",edgecolor="black"))

st.pyplot()

st.write(sns.distplot(data[column\_name]))

st.pyplot()

elif column\_name=="SeniorCitizen":

plt.figure(figsize=(5,3))

st.set\_option('deprecation.showPyplotGlobalUse', False)

st.write(plt.hist(data[column\_name],color="skyblue",edgecolor="black"))

st.pyplot()

st.write(sns.distplot(data[column\_name]))

st.pyplot()

elif column\_name=="TotalCharges":

plt.figure(figsize=(5,3))

st.set\_option('deprecation.showPyplotGlobalUse', False)

st.write(plt.hist(data[column\_name],color="skyblue",edgecolor="black"))

st.pyplot()

st.write(sns.distplot(data[column\_name]))

st.pyplot()

if st.checkbox("Categorical variable"):

column\_name=st.selectbox("",("Select column","Contract","PhoneService","tenure","InternetService"))

if column\_name=="Contract":

plt.figure(figsize=(5,3))

st.set\_option('deprecation.showPyplotGlobalUse', False)

st.write(plt.hist(data[column\_name],color="skyblue",edgecolor="black"))

st.pyplot()

elif column\_name=="PhoneService":

st.set\_option('deprecation.showPyplotGlobalUse', False)

st.write(plt.hist(data[column\_name],color="skyblue",edgecolor="black"))

st.pyplot()

elif column\_name=="tenure":

st.set\_option('deprecation.showPyplotGlobalUse', False)

st.write(plt.hist(data[column\_name],color="skyblue",edgecolor="black"))

st.pyplot()

elif column\_name=="InternetService":

st.set\_option('deprecation.showPyplotGlobalUse', False)

st.write(plt.hist(data[column\_name],color="skyblue",edgecolor="black"))

st.pyplot()

**For Prediction:**

def prediction():

st.header("Prediction")

col\_list=[0]\*features

gen\_col=["Female","Male"]

gen\_num=list(range(len(gen\_col)))

gen=st.radio("Select gender",gen\_num,format\_func=lambda x:gen\_col[x])

col\_list[0]=gen

age\_col=[0,1]

age\_num=list(range(len(age\_col)))

age=st.radio("Senior Citizen?",age\_num,format\_func=lambda x:age\_col[x])

col\_list[1]=age\_col

partner\_col=["Yes","No"]

partner\_num=list(range(len(partner\_col)))

partner=st.radio("Partner?",partner\_num,format\_func=lambda x:partner\_col[x])

col\_list[2]=partner

dpdt\_col=["Yes","No"]

dpdt\_num=list(range(len(dpdt\_col)))

dpdt=st.radio("Dependant?",dpdt\_num,format\_func=lambda x:dpdt\_col[x])

col\_list[3]=dpdt

Tenure\_col=st.slider("Enter tenure : ",0,72)

col\_list[4]=Tenure\_col

ps\_col=["Yes","No"]

ps\_num=list(range(len(ps\_col)))

ps=st.radio("Phone Service?",ps\_num,format\_func=lambda x:ps\_col[x])

col\_list[5]=ps

ml\_col=["Yes","No"]

ml\_num=list(range(len(ml\_col)))

ml=st.radio("Multiple Lines?",ml\_num,format\_func=lambda x:ml\_col[x])

col\_list[6]=ml

InternetService=["InternetService","Fiber Optic","DSL","No internet service"]

InternetS\_option=st.selectbox("",InternetService)

if InternetS\_option=="Fiber Optic":

col\_list[7]="Fiber Optic"

# col\_list[10]=0

elif InternetS\_option=="DSL":

col\_list[7]="DSL"

# col\_list[10]=1

elif InternetS\_option=="No internet service":

col\_list[7]="No internet service"

# col\_list[10]=0

os\_col=["Yes","No"]

os\_num=list(range(len(os\_col)))

os=st.radio("Online Security?",os\_num,format\_func=lambda x:os\_col[x])

col\_list[8]=os

ob\_col=["Yes","No"]

ob\_num=list(range(len(ob\_col)))

ob=st.radio("Online Backup?",ob\_num,format\_func=lambda x:ob\_col[x])

col\_list[9]=ob

dp\_col=["Yes","No"]

dp\_num=list(range(len(dp\_col)))

dp=st.radio("Device Protection?",dp\_num,format\_func=lambda x:dp\_col[x])

col\_list[10]=dp

ts\_col=["Yes","No"]

ts\_num=list(range(len(ts\_col)))

ts=st.radio("Tech Support?",ts\_num,format\_func=lambda x:ts\_col[x])

col\_list[11]=ts

stv\_col=["Yes","No"]

stv\_num=list(range(len(stv\_col)))

stv=st.radio("Streaming TV?",stv\_num,format\_func=lambda x:stv\_col[x])

col\_list[12]=stv

sm\_col=["Yes","No"]

sm\_num=list(range(len(sm\_col)))

sm=st.radio("Streaming Movie?",sm\_num,format\_func=lambda x:sm\_col[x])

col\_list[13]=sm

Contract=["Contract","One year","month-to-month"]

Contract\_option=st.selectbox("",Contract)

if Contract\_option=="One year":

col\_list[14]="One year"

elif InternetS\_option=="month-to-month":

col\_list[14]="month-to-month"

pbl\_col=["Yes","No"]

pbl\_num=list(range(len(pbl\_col)))

pbl=st.radio("Paper Billing?",pbl\_num,format\_func=lambda x:sm\_col[x])

col\_list[15]=pbl

paymentmethod=["Payment Method","Electronic check","Mail check","Credit card (automatic)","Bank transfer (automatic)"]

pm\_option=st.selectbox("",paymentmethod)

if pm\_option=="Electronic check":

col\_list[16]="Electronic check"

elif pm\_option=="Mail check":

col\_list[16]="Mail check"

elif pm\_option=="Credit card (automatic)":

col\_list[16]="Credit card (automatic)"

elif pm\_option=="Bank transfer (automatic)":

col\_list[16]="Bank transfer (automatic)"

Monthlycharges=st.number\_input("Enter Monthly charges: ",step=0.01)

col\_list[17]=Monthlycharges

Totalcharges=st.number\_input("Enter Total charges:",step=0.01)

col\_list[18]=Totalcharges

if st.checkbox("Your entries"):

d={}

feature=["gender","SeniorCitizen","Partner","Dependents","tenure","PhoneService","MultipleLines","InternetService","OnlineSecurity","OnlineBackup","DeviceProtection","TechSupport","StreamingTV","StreamingMovies","Contract","PaperlessBilling","PaymentMethod","MonthlyCharges","TotalCharges"]

for i in range(len(feature)):

if i<19:

d[feature[i]]=col\_list[i]

else:

d[feature[i]]=[col\_list[i],col\_list[i+1]]

st.write(d)

if st.button("Predict"):

pass

## 6) Result:

**For EDT:**

Graphical user interface, application

Description automatically generated

Table

Description automatically generated

Graphical user interface, table

Description automatically generated

Graphical user interface, table

Description automatically generated

**For Data Visualization:**

Graphical user interface, application

Description automatically generated

Chart

Description automatically generated

Chart, histogram

Description automatically generated

Graphical user interface, application

Description automatically generated

Chart

Description automatically generated

**For Prediction:**

Graphical user interface, application, Teams

Description automatically generated

Graphical user interface, application, Teams

Description automatically generated

Graphical user interface, application, Teams

Description automatically generated

## 7) Github link:

<https://github.com/VandithReddy/Customer-Attrition-Prediction>

## 8) Discussion and Future Work:

As we have done the UI part in this project, we will include the machine learning part in the next extension of this project, where we will be doing data cleaning, pre-processing, data splitting, model creation, evaluation and data segmentation which can predict like whom can we retain.

## 9) References:

* [🔎🔎 Customer Churn Prediction - 95% accuracy | Kaggle](https://www.kaggle.com/sibelius5/customer-churn-prediction-95-accuracy)
* <https://towardsdatascience.com/churn-prediction-using-neural-networks-and-ml-models-c817aadb7057>
* [Customer Retention Analysis & Churn Prediction - Case Studies | Addepto](https://addepto.com/customer-retention-analysis-churn-prediction/)
* [Telecom-Customer-Churn-Prediction | Kaggle](https://www.kaggle.com/anirbansarkar823/telecom-customer-churn-prediction)
* https://www.profitwell.com/recur/all/churn-prediction#:~:text=Churn%20prediction%20helps%20prevent%20churn%20in%20the%20first,business%20insight%20Get%20insight%20on%20churn%20trends.%20