Phase – II

Data Preprocessing

AI-Driven Exploration and Prediction of Company Registration Trends with Registrar of Companies(ROC)

Date	**11 October 2023**
Team ID	**Proj_212170_Team_2**
Project name	**Registrar of Companies(ROC)**
Maximum marks	

#Importing the packages needed for the above given problem

```
import numpy as np
import pandas as pd
```

- Here, you are importing the pandas library with the alias "pd," which is a common practice. However, you also attempted to import pandas with the alias "np," which is usually used for NumPy, another popular Python library. It's better to use "pd" consistently for pandas

```
import matplotlib.pyplot as plt
import seaborn as sns
```

#importing the necessary packages and libraries for the above given problems

```
import numpy as np

from numpy import concatenate

import urllib.request as urllib

from sklearn.preprocessing import StandardScaler, MinMaxScaler,
LabelEncoder, OneHotEncoder from sklearn.model_selection import
train_test_split

from sklearn.metrics import mean_squared_error

from keras.models import Sequential
```

#reading the dataset as csv file

```
df=pd.read_csv(r"C:\Users\91908\Downloads\Data_Gov_Tamil_Nadu.csv")
```

- This code reads data from a CSV file located at the specified path and stores it in a pandas DataFrame called df. The encoding='latin-1' parameter is used to specify the character encoding of the file

#displaying the dataset in default manner

df.head(7)

output:

Out[6]:

	CORPORATE_IDENTIFICATION_NUMBER	COMPANY_NAME	COMPANY_STATUS	COMPANY_CLASS	COMPANY_CATEGORY	COMPANY_SUB_CATEGORY	DATE_
0	F00643	HOCHTIEFF AG,	NAEF	NaN	NaN	NaN	
1	F00721	SUMITOMO CORPORATION (SUMITOMO SHOJI KAISHA LI...	ACTV	NaN	NaN	NaN	
2	F00892	SRILANKAN AIRLINES LIMITED	ACTV	NaN	NaN	NaN	
3	F01208	CALTEX INDIA LIMITED	NAEF	NaN	NaN	NaN	
4	F01218	GE HEALTHCARE BIO-SCIENCES LIMITED	ACTV	NaN	NaN	NaN	
5	F01265	CAIRN ENERGY INDIA PTY. LIMITED	NAEF	NaN	NaN	NaN	
6	F01269	TORIELLI S.R.L	ACTV	NaN	NaN	NaN	

#Training the csv file(dataset)

train_df=pd.read_csv(r"C:\Users\91908\Downloads\Data_Gov_Tamil_Nadu.csv")

testing the csv file

#Testing the csv file(dataset)

test_df=pd.read_csv(r"C:\Users\91908\Downloads\Data_Gov_Tamil_Nadu.csv")

#displaying all the columns

code: **train_df.columns**

output:

```
Index(['CORPORATE_IDENTIFICATION_NUMBER', 'COMPANY_NAME', 'COMPANY_STAT
US',
       'COMPANY_CLASS', 'COMPANY_CATEGORY', 'COMPANY_SUB_CATEGORY',
       'DATE_OF_REGISTRATION', 'REGISTERED_STATE', 'AUTHORIZED_CAP',
       'PAIDUP_CAPITAL', 'INDUSTRIAL_CLASS',
       'PRINCIPAL_BUSINESS_ACTIVITY_AS_PER_CIN', 'REGISTERED_OFFICE_ADD
RESS',
       'REGISTRAR_OF_COMPANIES', 'EMAIL_ADDR', 'LATEST_YEAR_ANNUAL_RETU
RN',
       'LATEST_YEAR_FINANCIAL_STATEMENT'],
      dtype='object')
```

Filling Missing Values:

df.fillna({'COMPANY_CLASS': 'Private', 'COMPANY_CATEGORY': 'Com pany limited by Shares', 'COMPANY_SUB_CATEGORY': 'Non-govt co mpany'}) ¬ This line attempts to fill missing values in specific colum ns ('COMPANY_CLASS

#displaying the dataset present in the bottom

Code:

```
train_df.tail()
```

output:

Out[10]:

	CORPORATE_IDENTIFICATION_NUMBER	COMPANY_NAME	COMPANY_STATUS	COMPANY_CLASS	COMPANY_CATEGORY	COMPANY_SUB_CATEGORY
150866	U74997TN2016PTC112556	QUAD42 MEDIA PRIVATE LIMITED	ACTV	Private	Company limited by Shares	Non-govt company
150867	U74997TN2018PTC121491	IYERAATHU FOODS PRIVATE LIMITED	ACTV	Private	Company limited by Shares	Non-govt company
150868	U74997TZ2016PTC027802	POLYGAR FARM SOLUTIONS PRIVATE LIMITED	STOF	Private	Company limited by Shares	Non-govt company
150869	U74997TZ2018PTC030177	PANDIYA AGRI SOLUTIONS PRIVATE LIMITED	ACTV	Private	Company limited by Shares	Non-govt company
150870	U74997TZ2019PTC032491	NROOT TECHNOLOGIES PRIVATE LIMITED	ACTV	Private	Company limited by Shares	Non-govt company

#Describe all the datasets

train_df.describe()

output:

Out[11]:

	AUTHORIZED_CAP	PAIDUP_CAPITAL
count	1.508710e+05	1.508710e+05
mean	3.522781e+07	2.328823e+07
std	1.408554e+09	1.072457e+09
min	0.000000e+00	0.000000e+00
25%	1.000000e+05	1.000000e+05
50%	8.000000e+05	1.000000e+05
75%	2.000000e+06	6.857450e+05
max	3.000000e+11	2.461230e+11

#showing all the null values present in the dataset

print(df.isnull().sum())

output:

```
CORPORATE_IDENTIFICATION_NUMBER              0
COMPANY_NAME                                 0
COMPANY_STATUS                               0
COMPANY_CLASS                              334
COMPANY_CATEGORY                           334
COMPANY_SUB_CATEGORY                       334
DATE_OF_REGISTRATION                        39
REGISTERED_STATE                             0
AUTHORIZED_CAP                               0
PAIDUP_CAPITAL                               0
INDUSTRIAL_CLASS                           310
PRINCIPAL_BUSINESS_ACTIVITY_AS_PER_CIN       0
REGISTERED_OFFICE_ADDRESS                   90
REGISTRAR_OF_COMPANIES                     174
EMAIL_ADDR                               38129
LATEST_YEAR_ANNUAL_RETURN                75889
LATEST_YEAR_FINANCIAL_STATEMENT          75782
dtype: int64
```

#cleaning the datas which is unwanted for the dataset

Code:

```
df.COMPANY_SUB_CATEGORY   =df.EMAIL_ADDR.fillna("unknown")

print(df.isnull().sum())
```

output:

```
CORPORATE_IDENTIFICATION_NUMBER          0
COMPANY_NAME                             0
COMPANY_STATUS                           0
COMPANY_CLASS                          334
COMPANY_CATEGORY                       334
COMPANY_SUB_CATEGORY                     0
DATE_OF_REGISTRATION                    39
REGISTERED_STATE                         0
AUTHORIZED_CAP                           0
PAIDUP_CAPITAL                           0
INDUSTRIAL_CLASS                       310
PRINCIPAL_BUSINESS_ACTIVITY_AS_PER_CIN     0
REGISTERED_OFFICE_ADDRESS               90
REGISTRAR_OF_COMPANIES                 174
EMAIL_ADDR                           38129
LATEST_YEAR_ANNUAL_RETURN            75889
LATEST_YEAR_FINANCIAL_STATEMENT      75782
dtype: int64
```

#code for identifying the datatype present in the dataset

Code:

```
print(df.shape)
print("\n")
print(df.dtypes)
```

output:

```
(150871, 17)


CORPORATE_IDENTIFICATION_NUMBER            object
COMPANY_NAME                               object
COMPANY_STATUS                             object
COMPANY_CLASS                              object
COMPANY_CATEGORY                           object
COMPANY_SUB_CATEGORY                       object
DATE_OF_REGISTRATION                       object
REGISTERED_STATE                           object
AUTHORIZED_CAP                            float64
PAIDUP_CAPITAL                            float64
INDUSTRIAL_CLASS                           object
PRINCIPAL_BUSINESS_ACTIVITY_AS_PER_CIN     object
REGISTERED_OFFICE_ADDRESS                  object
REGISTRAR_OF_COMPANIES                     object
EMAIL_ADDR                                 object
LATEST_YEAR_ANNUAL_RETURN                  object
LATEST_YEAR_FINANCIAL_STATEMENT            object
dtype: object
```

#printing the information about the dataset

Code:

```
df.info()
print('_'*40)
df.info()
```

output:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150871 entries, 0 to 150870
Data columns (total 17 columns):
 #   Column                                 Non-Null Count   Dtype
---  ------                                 --------------   -----
 0   CORPORATE_IDENTIFICATION_NUMBER        150871 non-null  object
 1   COMPANY_NAME                           150871 non-null  object
 2   COMPANY_STATUS                         150871 non-null  object
 3   COMPANY_CLASS                          150537 non-null  object
 4   COMPANY_CATEGORY                       150537 non-null  object
 5   COMPANY_SUB_CATEGORY                   150871 non-null  object
 6   DATE_OF_REGISTRATION                   150832 non-null  object
 7   REGISTERED_STATE                       150871 non-null  object
 8   AUTHORIZED_CAP                         150871 non-null  float64
 9   PAIDUP_CAPITAL                         150871 non-null  float64
 10  INDUSTRIAL_CLASS                       150561 non-null  object
 11  PRINCIPAL_BUSINESS_ACTIVITY_AS_PER_CIN 150871 non-null  object
 12  REGISTERED_OFFICE_ADDRESS              150781 non-null  object
 13  REGISTRAR_OF_COMPANIES                 150697 non-null  object
 14  EMAIL_ADDR                             112742 non-null  object
 15  LATEST_YEAR_ANNUAL_RETURN              74982 non-null   object
 16  LATEST_YEAR_FINANCIAL_STATEMENT        75089 non-null   object
dtypes: float64(2), object(15)
memory usage: 19.6+ MB

________________________________________
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150871 entries, 0 to 150870
Data columns (total 17 columns):
 #   Column                                 Non-Null Count   Dtype
---  ------                                 --------------   -----
 0   CORPORATE_IDENTIFICATION_NUMBER        150871 non-null  object
 1   COMPANY_NAME                           150871 non-null  object
 2   COMPANY_STATUS                         150871 non-null  object
 3   COMPANY_CLASS                          150537 non-null  object
 4   COMPANY_CATEGORY                       150537 non-null  object
 5   COMPANY_SUB_CATEGORY                   150871 non-null  object
 6   DATE_OF_REGISTRATION                   150832 non-null  object
 7   REGISTERED_STATE                       150871 non-null  object
```

```
 8   AUTHORIZED_CAP                          150871 non-null  float64
 9   PAIDUP_CAPITAL                          150871 non-null  float64
 10  INDUSTRIAL_CLASS                        150561 non-null  object
 11  PRINCIPAL_BUSINESS_ACTIVITY_AS_PER_CIN  150871 non-null  object
 12  REGISTERED_OFFICE_ADDRESS               150781 non-null  object
 13  REGISTRAR_OF_COMPANIES                  150697 non-null  object
 14  EMAIL_ADDR                              112742 non-null  object
 15  LATEST_YEAR_ANNUAL_RETURN               74982 non-null   object
 16  LATEST_YEAR_FINANCIAL_STATEMENT         75089 non-null   object
dtypes: float64(2), object(15)
memory usage: 19.6+ MB
```