```
from google.colab import drive
```

```
drive.mount("/content/drive/")
```

```
Drive already mounted at /content/drive/; to attempt to forcibly remount, call drive.mo
```

```
pip install pyforest
```

```
Requirement already satisfied: pyforest in /usr/local/lib/python3.6/dist-packages (1.0.
```

```
import pyforest
```

```
lazy_imports()
```

```
['import glob',
 'import pydot',
 'from sklearn.ensemble import RandomForestClassifier',
 'import gensim',
 'import spacy',
 'import altair as alt',
 'from sklearn.ensemble import GradientBoostingRegressor',
 'import pandas as pd',
 'import bokeh',
 'import statistics',
 'import re',
 'import dash',
 'import plotly.express as px',
 'import numpy as np',
 'from pyspark import SparkContext',
 'from pathlib import Path',
 'import plotly as py',
 'import plotly.graph_objs as go',
 'from openpyxl import load_workbook',
 'from sklearn.manifold import TSNE',
 'import sys',
 'import pickle',
 'from sklearn.feature_extraction.text import TfidfVectorizer',
 'import seaborn as sns',
 'from sklearn.model_selection import train_test_split',
 'import matplotlib.pyplot as plt',
 'from sklearn.ensemble import GradientBoostingClassifier',
 'import matplotlib as mpl',
 'import os',
 'from sklearn import svm',
 'from dask import dataframe as dd',
 'import tqdm',
 'import tensorflow as tf',
 'import datetime as dt',
 'from sklearn.preprocessing import OneHotEncoder',
 'import keras',
 'import nltk',
 'import sklearn',
 'from sklearn.ensemble import RandomForestRegressor']
```

```
import sqlite3
```

```
con=sqlite3.connect("/content/drive/My Drive/colab folder/Datasets/amazon-fine-food-reviews/
```

```
database=pd.read_sql_query("select * from reviews where score<>3",con)
```

database

	Id	ProductId	UserId	ProfileName	HelpfulnessNumerator	Helpful
0	1	B001E4KFG0	A3SGXH7AUHU8GW	delmartian	1	
1	2	B00813GRG4	A1D87F6ZCVE5NK	dll pa	0	
2	3	B000LQOCH0	ABXLMWJIXXAIN	Natalia Corres "Natalia Corres"	1	
3	4	B000UA0QIQ	A395BORC6FGVXV	Karl	3	
4	5	B006K2ZZ7K	A1UQRSCLF8GW1T	Michael D. Bigham "M. Wassir"	0	
...	...	...	...	...	...	...
525809	568450	B001EO7N10	A28KG5XORO54AY	Lettie D. Carter	0	
525810	568451	B003S1WTCU	A3I8AFVPEE8KI5	R. Sawyer	0	
525811	568452	B004I613EE	A121AA1GQV751Z	pksd "pk_007"	2	
525812	568453	B004I613EE	A3IBEVCTXKNOH	Kathy A. Welch "katwel"	1	
525813	568454	B001LR2CU2	A3LGQPJCZVL9UC	srfell17	0	

525814 rows × 10 columns

```
def partition(x):
  if x>3:
    return 1
  return 0


database.Score=database.Score.map(partition)


database.Score.value_counts()
```

```
1    443777
0     82037
Name: Score, dtype: int64
```

```
database.shape
```

```
(525814, 10)
```

```
database=database[database.HelpfulnessNumerator<=database.HelpfulnessDenominator]


database.shape
```

```
(525812, 10)
```

```
sort=database.sort_values("ProductId")


sort.shape
```

```
(525812, 10)
```

```
final=sort.drop_duplicates(subset={"UserId","ProfileName","Time","Text"}, keep='first', inp
final.shape
```

```
(364171, 10)
```

```
from bs4 import BeautifulSoup
import re


def decontracted(phrase):
    # specific
    phrase = re.sub(r"won't", "will not", phrase)
    phrase = re.sub(r"can\'t", "can not", phrase)

    # general
    phrase = re.sub(r"n\'t", " not", phrase)
    phrase = re.sub(r"\'re", " are", phrase)
    phrase = re.sub(r"\'s", " is", phrase)
    phrase = re.sub(r"\'d", " would", phrase)
    phrase = re.sub(r"\'ll", " will", phrase)
    phrase = re.sub(r"\'t", " not", phrase)
    phrase = re.sub(r"\'ve", " have", phrase)
    phrase = re.sub(r"\'m", " am", phrase)
    return phrase


stopwords= set(['br', 'the', 'i', 'me', 'my', 'myself', 'we', 'our', 'ours', 'ourselves', '
            "you'll", "you'd", 'your', 'yours', 'yourself', 'yourselves', 'he', 'him', 'his
```

```
            'she', "she's", 'her', 'hers', 'herself', 'it', "it's", 'its', 'itself', 'they'
            'theirs', 'themselves', 'what', 'which', 'who', 'whom', 'this', 'that', "that'll
            'am', 'is', 'are', 'was', 'were', 'be', 'been', 'being', 'have', 'has', 'had',
            'did', 'doing', 'a', 'an', 'the', 'and', 'but', 'if', 'or', 'because', 'as', 'un
            'at', 'by', 'for', 'with', 'about', 'against', 'between', 'into', 'through', 'du
            'above', 'below', 'to', 'from', 'up', 'down', 'in', 'out', 'on', 'off', 'over',
            'then', 'once', 'here', 'there', 'when', 'where', 'why', 'how', 'all', 'any', 'b
            'most', 'other', 'some', 'such', 'only', 'own', 'same', 'so', 'than', 'too', 've
            's', 't', 'can', 'will', 'just', 'don', "don't", 'should', "should've", 'now',
            've', 'y', 'ain', 'aren', "aren't", 'couldn', "couldn't", 'didn', "didn't", 'doe
            "hadn't", 'hasn', "hasn't", 'haven', "haven't", 'isn', "isn't", 'ma', 'mightn',
            "mustn't", 'needn', "needn't", 'shan', "shan't", 'shouldn', "shouldn't", 'wasn'
            'won', "won't", 'wouldn', "wouldn't"])


%%time
from tqdm import tqdm
preprocessed_reviews = []
# tqdm is for printing the status bar
for sentance in tqdm(final['Text'].values):
    sentance = re.sub(r"http\S+", "", sentance)
    sentance = BeautifulSoup(sentance, 'lxml').get_text()
    sentance = decontracted(sentance)
    sentance = re.sub("\S*\d\S*", "", sentance).strip()
    sentance = re.sub('[^A-Za-z]+', ' ', sentance)
    # https://gist.github.com/sebleier/554280
    sentance = ' '.join(e.lower() for e in sentance.split() if e.lower() not in stopwords)
    preprocessed_reviews.append(sentance.strip())

 100%|██████████| 364171/364171 [02:18<00:00, 2629.95it/s]CPU times: user 2min 13s, sys:
    Wall time: 2min 18s


from tqdm import tqdm
preprocessed_summary = []
# tqdm is for printing the status bar
for sentance in tqdm(final['Summary'].values):
    sentance = re.sub(r"http\S+", "", sentance)
    sentance = BeautifulSoup(sentance, 'lxml').get_text()
    sentance = decontracted(sentance)
    sentance = re.sub("\S*\d\S*", "", sentance).strip()
    sentance = re.sub('[^A-Za-z]+', ' ', sentance)
    # https://gist.github.com/sebleier/554280
    sentance = ' '.join(e.lower() for e in sentance.split() if e.lower() not in stopwords)
    preprocessed_summary.append(sentance.strip())

  6%|▌         | 22297/364171 [00:05<01:21, 4215.56it/s]/usr/local/lib/python3.6/dist-p
    ' Beautiful Soup.' % markup)
100%|██████████| 364171/364171 [01:27<00:00, 4151.67it/s]


final["Cleaned_text"]=preprocessed_reviews
final["Cleaned_summary"]=preprocessed_summary
```

```
/usr/local/lib/python3.6/dist-packages/ipykernel_launcher.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user
  """Entry point for launching an IPython kernel.
/usr/local/lib/python3.6/dist-packages/ipykernel_launcher.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user
```

```
final
```

	Id	ProductId	UserId	ProfileName	HelpfulnessNumerator	Helpful
138693	150511	0006641040	A1C9K534BCI9GO	Laura Purdie Salas	0	
138708	150526	0006641040	A3E9QZFE9KXH8J	R. Mitchell	11	
138707	150525	0006641040	A2QID6VCFTY51R	Rick	1	
138706	150524	0006641040	ACITT7DI6IDDL	shari zychinski	0	
138705	150523	0006641040	A2P4F2UO0UMP8C	Elizabeth A. Curry "Lovely Librarian"	0	
...	...	...	...	...	...	
178136	193165	B009RSR8HO	A1I08MP3H92U6R	Thomas	1	
173675	188389	B009SF0TN6	A1L0GWGRK4BYPT	Bety Robinson	0	
204727	221795	B009SR4OQ2	A32A6X5KCP7ARG	sicamar	1	
5259	5703	B009WSNWC4	AMP7K1O84DH1T	ESTY	0	
302474	327601	B009WVB40S	A3ME78KVX31T21	K'la	0	

364171 rows × 12 columns

```
final["Combined_text"]=final.Cleaned_text.values+" "+final.Cleaned_summary.values
```

```
/usr/local/lib/python3.6/dist-packages/ipykernel_launcher.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user
  """Entry point for launching an IPython kernel.
```

```
time_sorted_data=final.sort_values("Time")
```

```
time_sorted_data
```

	Id	ProductId	UserId	ProfileName	HelpfulnessNumerator	Helpful
138706	150524	0006641040	ACITT7DI6IDDL	shari zychinski	0	
138683	150501	0006641040	AJ46FKXOVC7NR	Nicholas A Mesiano	2	
417839	451856	B00004CXX9	AIUWLEQ1ADEG5	Elizabeth Medina	0	
346055	374359	B00004CI84	A344SMIA5JECGM	Vincent P. Ross	1	
417838	451855	B00004CXX9	AJH6LUC1UT1ON	The Phantom of the Opera	0	
...	...	...	...	...	...	...
418130	452168	B000FIWIWA	A1TED4G0PWZPQV	Stellavera S. Kilcher	0	
234951	254899	B0034WSXIC	A2EOLZ2F8PG87D	xtreme dude	0	
520425	562645	B005C7R8HE	A3MAVAV6Q37XT5	Marilyn Graper	0	
416667	450607	B004MBJPV8	AG7EPW4BU9SG4	2 Toddlers' Mom "2 Toddlers' Mom"	0	
106073	115167	B0001HAEKI	A289SYWE4BHCF	akilah	0	

364171 rows × 13 columns

```
final_data=time_sorted_data.take(np.random.permutation(len(final)))[:100000]
print(final_data.shape)
```

```
(100000, 13)
```

```
final_data
```

	Id	ProductId	UserId	ProfileName	HelpfulnessNumerator	Helpful
128097	138991	B0041CIR62	A2DB720I9XRX7K	K. Draper	0	
482199	521389	B007JFXWRC	AEBR0PZSRE0YL	Zensaba	0	
88785	96633	B008KZ5KZ2	A25ICPW744JKJ	Joyce Ann Miller	2	
181217	196532	B004K78HTA	A2W4BJLCVWAKT8	Daniel Moyer	0	
291997	316307	B000NBYPFM	A1SDD92YHPVZRM	Harold S. Freeman "stutac2"	2	
...	...	...	...	...	...	
346088	374393	B00004CI84	A3J0V5L5APJ61R	S. Sawin "Sgrosvenor"	0	
258399	280114	B0019FR0WQ	A3S459EHZONQUI	pleinelune	2	
512578	554200	B000RH4FG6	A2TFHIJBZY2H2H	K. Friend	0	
3599	3912	B00449NWW6	AD6BI0XB666BK	Abix	2	

130512	141644	B004NPUTSQ	A3AJNQ0DNZ31ZV	In DC	1

100000 rows × 13 columns

```
x=np.array([1,2,3,4,5,6])
t=np.array([0,1,1,0,1,1])
y=x.take(np.random.permutation(len(x)))
y
```

```
array([1, 4, 2, 5, 3, 6])
```

```
from sklearn.model_selection import TimeSeriesSplit
from sklearn.model_selection import GridSearchCV


X=final_data.Combined_text
Y=final_data.Score


x_train,x_test,y_train,y_test=train_test_split(X,Y,test_size=0.25,random_state=101)
```

```

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

```
(75000,)
(25000,)
(75000,)
(25000,)
```

```
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.preprocessing import StandardScaler
from sklearn.preprocessing import Normalizer


count_vec=CountVectorizer(min_df=10,max_features=50000,dtype="float")


count_vec.fit(x_train)
```

```
CountVectorizer(analyzer='word', binary=False, decode_error='strict',
                dtype='float', encoding='utf-8', input='content',
                lowercase=True, max_df=1.0, max_features=50000, min_df=10,
                ngram_range=(1, 1), preprocessor=None, stop_words=None,
                strip_accents=None, token_pattern='(?u)\\b\\w\\w+\\b',
                tokenizer=None, vocabulary=None)
```

```
bow_x_train=count_vec.transform(x_train)
bow_x_test=count_vec.transform(x_test)


print(bow_x_train.shape)
print(bow_x_test.shape)
```

```
(75000, 11182)
(25000, 11182)
```

```
nor=Normalizer()
```

```
nor
```

```
Normalizer(copy=True, norm='l2')
```

```
"""bow_x_train=bow_x_train.toarray()
bow_x_test=bow_x_test.toarray()"""
```

```
'bow_x_train=bow_x_train.toarray()\nbow_x_test=bow_x_test.toarray()'
```

```
type(bow_x_train)
```

```
scipy.sparse.csr.csr_matrix
```

```
scalar=StandardScaler()
```

```
nor.fit(bow_x_train)
```

```
Normalizer(copy=True, norm='l2')
```

```
bow_train=nor.transform(bow_x_train)
bow_test=nor.transform(bow_x_test)
```

```
from sklearn.linear_model import LogisticRegression
```

```
lg=LogisticRegression(n_jobs=-1,class_weight="balanced")
parameter={"C":[0.0001,0.001,0.01,0.1,1,10,100,200,300,400,500,600,700,800,900,1000]}
```

```
lg_l1=LogisticRegression(penalty="l1",n_jobs=-1,class_weight="balanced",solver="liblinear")
parameter={"C":[0.0001,0.001,0.01,0.1,1,10,100,200,300,400,500,600,700,800,900,1000]}
```

```
cv_timeseries=TimeSeriesSplit(n_splits=5).split(bow_train)
```

```
%%time
bow_gcv=GridSearchCV(lg,param_grid=parameter,scoring="roc_auc",cv=cv_timeseries,return_trai
bow_gcv.fit(bow_train,y_train)
```

```
CPU times: user 3.7 s, sys: 941 ms, total: 4.64 s
Wall time: 1min 18s
```

```
print(bow_gcv.best_params_)
print(bow_gcv.best_score_)
```

```
{'C': 10}
0.9629978687587677
```

```
%%time
cv_timeseries=TimeSeriesSplit(n_splits=5).split(bow_train)
bow_gcv_l1=GridSearchCV(lg_l1,param_grid=parameter,scoring="roc_auc",cv=cv_timeseries,retur
```

```
bow_gcv_l1=GridSearchCV(lg_l1,param_grid=parameter,scoring='roc_auc',cv=cv_timeseries,retur
bow_gcv_l1.fit(bow_train,y_train)
```

```
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
```

```
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
```

```
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
```

```
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
```

```
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/_logistic.py:1539: UserWarn
'n_jobs' > 1 does not have any effect when 'solver' is set to 'liblinear'. Got 'n_jobs'
CPU times: user 4min 56s, sys: 761 ms, total: 4min 57s
Wall time: 4min 57s
```

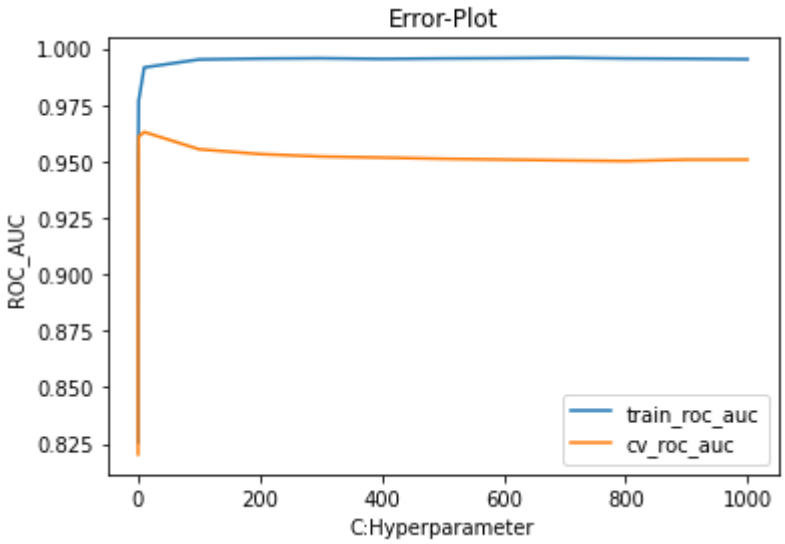
```
print(bow_gcv_l1.best_params_)
```

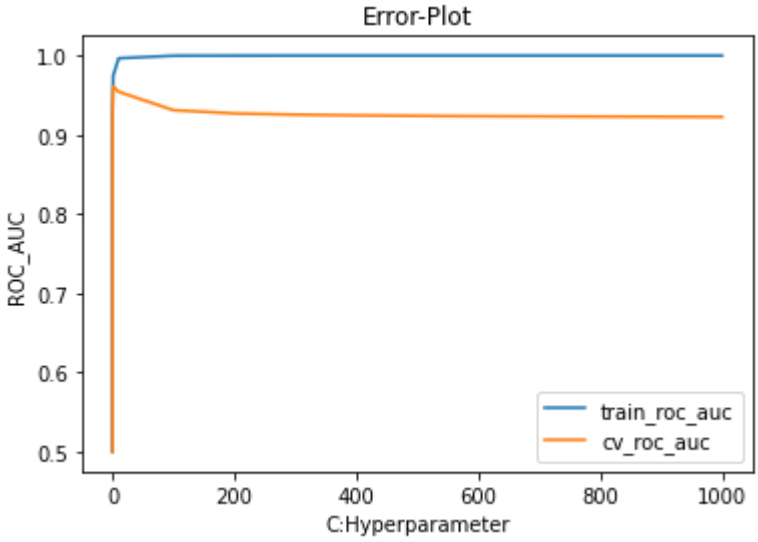
```
print(bow_gcv_l1.best_score_)
```

```
{'C': 1}
0.9612046893916439
```

```
def plot_cv_results(gcv,c):
  train_roc_auc=gcv.cv_results_["mean_train_score"]
  cv_roc_auc=gcv.cv_results_["mean_test_score"]
  plt.plot(c,train_roc_auc,label="train_roc_auc")
  plt.plot(c,cv_roc_auc,label="cv_roc_auc")
  plt.xlabel("C:Hyperparameter")
  plt.ylabel("ROC_AUC")
  plt.legend(loc="lower right")
  plt.title("Error-Plot")
  plt.show()
```

```
C=[0.0001,0.001,0.01,0.1,1,10,100,200,300,400,500,600,700,800,900,1000]
plot_cv_results(bow_gcv,C)
plot_cv_results(bow_gcv_l1,C)
```






```
l2_sparsity=bow_gcv.best_estimator_.coef_[0]
print("the features which has value 'zero' after applying l2 regularization in LR:",list(l2_
print(f'Weight Vector with values between 0 and 1 -> {len(l2_sparsity[(l2_sparsity > 0) & (
```

```
the features which has value 'zero' after applying l2 regularization in LR: 0
Weight Vector with values between 0 and 1 -> 3834
```

```
l1_sparsity=bow_gcv_l1.best_estimator_.coef_[0]
print("The features which has value 'zero' after applying l1 regularization in LR:",list(l1_
```

```
The features which has value 'zero' after applying l1 regularization in LR: 9637
```

```
print("The features/weights/dimension which has value between 0 to 1 in l1- regularization
print(f'Weight Vector with values between 0 and 1 -> {len(l1_sparsity[(l1_sparsity > 0) & (
```

```
The features/weights/dimension which has value between 0 to 1 in l1- regularization :--
Weight Vector with values between 0 and 1 -> 320
```

```
l1_sparsity[(l1_sparsity < 0)].shape+l1_sparsity[(l1_sparsity > 0)].shape
```

```
(740, 805)
```

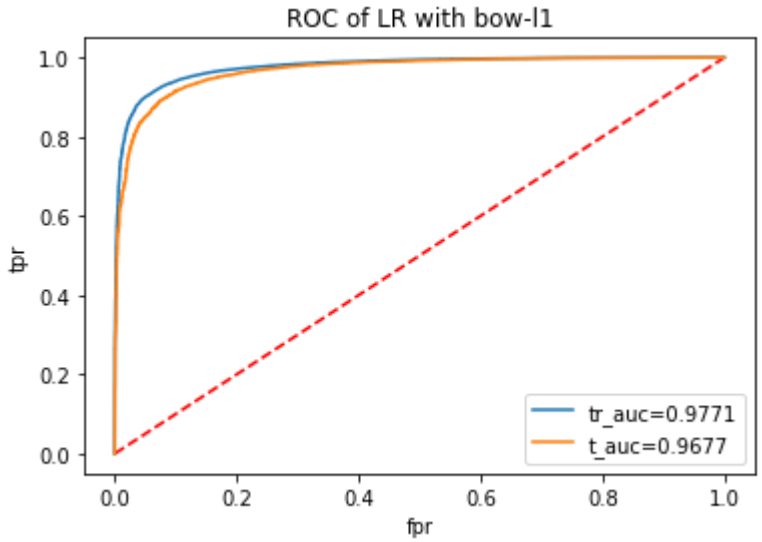
```
740+805
```

```
1545
```

```
from sklearn.metrics import roc_auc_score,roc_curve,classification_report,confusion_matrix,a


def plot_model(title,clf,x_train,x_test,y_train,y_test):
  print(title)
  print("="*50)
  y_tr_pred=clf.predict_proba(x_train)
  tr_fpr,tr_tpr,tr_thre=roc_curve(y_train,y_tr_pred[:,1])
  tr_auc=auc(tr_fpr,tr_tpr)

  y_t_pred=clf.predict_proba(x_test)
  t_fpr,t_tpr,t_thre=roc_curve(y_test,y_t_pred[:,1])
  t_auc=auc(t_fpr,t_tpr)

  plt.plot([0,1],[0,1],"r--")
  plt.plot(tr_fpr,tr_tpr,label="tr_auc=%0.4f"%tr_auc)
  plt.plot(t_fpr,t_tpr,label="t_auc=%0.4f"%t_auc)
  plt.legend(loc="lower right")
  plt.xlabel("fpr")
  plt.ylabel("tpr")
  plt.title("ROC of"+" "+title)
  plt.show()

  y_pred=clf.predict(x_test)
  print("the classification report: \n",classification_report(y_test,y_pred))

  sns.heatmap(confusion_matrix(y_test,y_pred),annot=True,fmt="g")
  plt.xlabel("observed")
  plt.ylabel("predicted")
  plt.title(title)
  plt.show()


plot_model("LR with bow-l1",bow_gcv_l1,bow_train,bow_test,y_train,y_test)
plot_model("LR with bow-l2",bow_gcv,bow_train,bow_test,y_train,y_test)
```

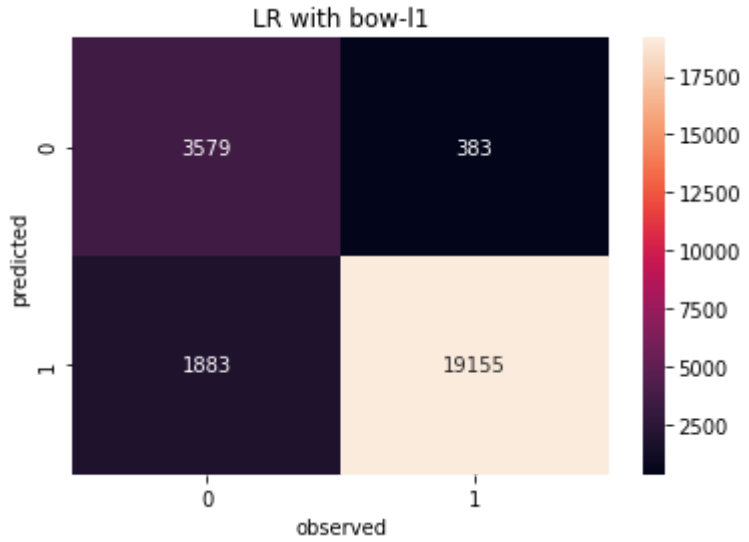
LR with bow-l1

==========



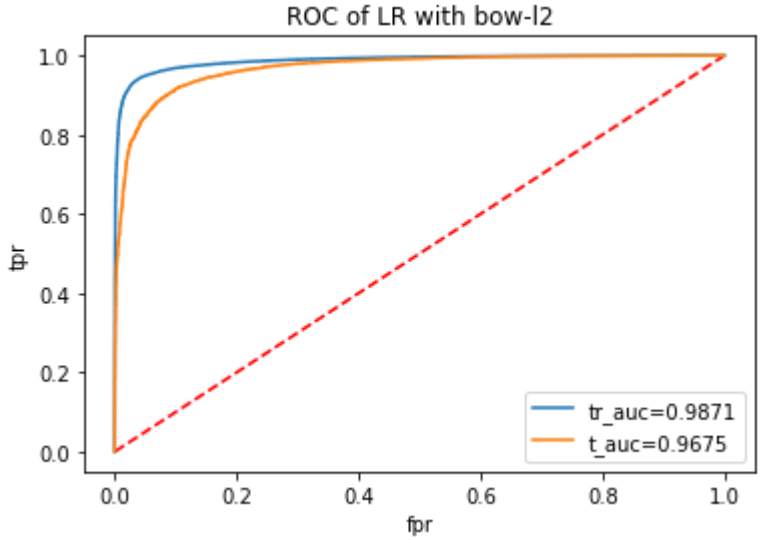
the classification report:

	precision	recall	f1-score	support
0	0.66	0.90	0.76	3962
1	0.98	0.91	0.94	21038
accuracy			0.91	25000
macro avg	0.82	0.91	0.85	25000
weighted avg	0.93	0.91	0.91	25000



LR with bow-l2

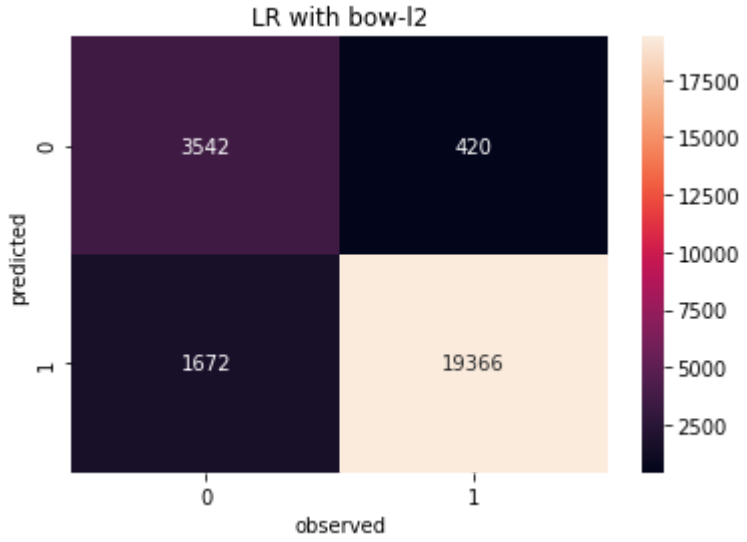
==========



the classification report:

	precision	recall	f1-score	support

```
              0       0.68      0.89      0.77      3962
              1       0.98      0.92      0.95     21038

       accuracy                           0.92     25000
      macro avg       0.83      0.91      0.86     25000
   weighted avg       0.93      0.92      0.92     25000
```




```
bow_feature_names=count_vec.get_feature_names()
```

```
np.count_nonzero(bow_gcv_l1.best_estimator_.coef_)
```

```
1545
```

Pertubation Test:

```
"""target_dims = your_target.shape
noise = np.random.rand(target_dims)
noisy_target = your_target + noise
np.random.(A or B) A=Normal B=Uniform"""
```

```
target_dims=bow_train.shape
target_dims
```

```
(75000, 11182)
```

```
noise=np.random.rand(75000, 11182)
noisy_target=bow_train+noise
```

```

```

```
noisy_target.shape
```

```
(75000, 11182)
```

```
"""%%time
lg=LogisticRegression(n_jobs=-1,class_weight="balanced")
parameter={"C":[0.0001,0.001,0.01,0.1,1,10,20,30,40,50,60,70,80,90,100,200,300,400,500,600,
cv_timeseries=TimeSeriesSplit(n_splits=5).split(noisy_target)
noisy_gcv=GridSearchCV(lg,param_grid=parameter,scoring="roc_auc",cv=cv_timeseries,return_tra
noisy_gcv.fit(noisy_target,y_train)
```

```
noisy_gcv.fit(noisy_target,y_train)
print(noisy_gcv.best_params_)
print(noisy_gcv.best_score_)"""
```

```
'%%time\nlg=LogisticRegression(n_jobs=-1,class_weight="balanced")\nparameter={"C":[0.00
```

```
tf_vec=TfidfVectorizer(max_features=50000,min_df=10,dtype="float")
```

```
tf_vec.fit(x_train)
```

```
/usr/local/lib/python3.6/dist-packages/sklearn/feature_extraction/text.py:1817: UserWar

    Only (<class 'numpy.float64'>, <class 'numpy.float32'>, <class 'numpy.float16'>) 'dtype

    TfidfVectorizer(analyzer='word', binary=False, decode_error='strict',
                    dtype='float', encoding='utf-8', input='content',
                    lowercase=True, max_df=1.0, max_features=50000, min_df=10,
                    ngram_range=(1, 1), norm='l2', preprocessor=None,
                    smooth_idf=True, stop_words=None, strip_accents=None,
                    sublinear_tf=False, token_pattern='(?u)\\b\\w\\w+\\b',
                    tokenizer=None, use_idf=True, vocabulary=None)
```

```
tf_x_train=tf_vec.transform(x_train)
tf_x_test=tf_vec.transform(x_test)
```

```
print(tf_x_train.shape)
print(tf_x_test.shape)
```

```
(75000, 11182)
(25000, 11182)
```

```
nor.fit(tf_x_train)
```

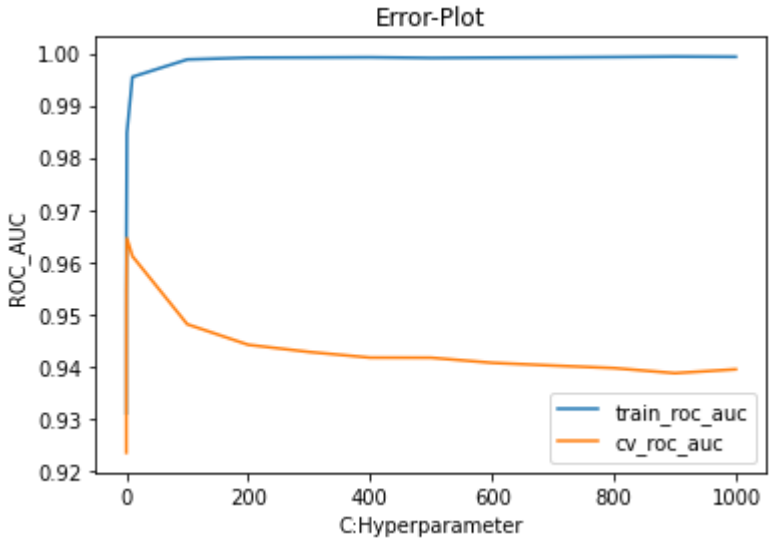
```
Normalizer(copy=True, norm='l2')
```

```
tfidf_train=nor.transform(tf_x_train)
tfidf_test=nor.transform(tf_x_test)
```

```
%%time
cv_timeseries=TimeSeriesSplit(n_splits=5).split(tfidf_train)
tf_gcv=GridSearchCV(lg,param_grid=parameter,scoring="roc_auc",cv=cv_timeseries,return_train_
tf_gcv.fit(tfidf_train,y_train)
print(tf_gcv.best_params_)
print(tf_gcv.best_score_)
```

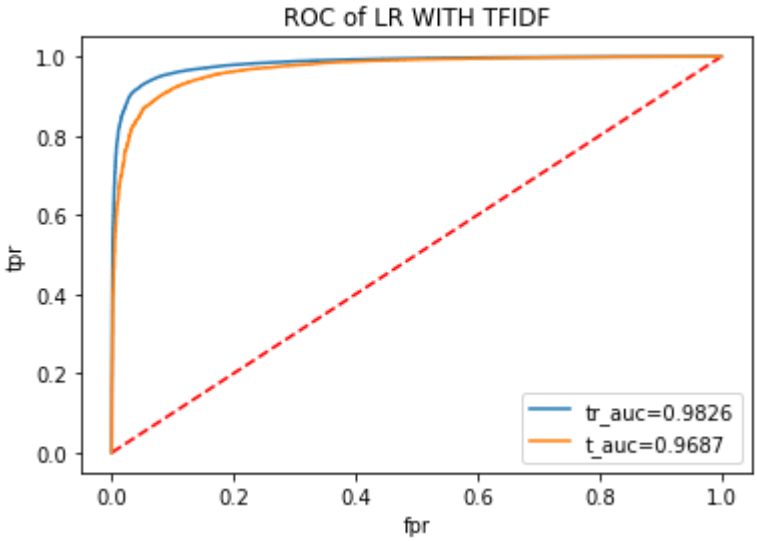
```
{'C': 1}
0.9645459820122072
CPU times: user 4.78 s, sys: 1.33 s, total: 6.11 s
Wall time: 1min 16s
```

```
plot_cv_results(tf_gcv,C)
plot_model("LR WITH TFIDF",tf_gcv,tfidf_train,tfidf_test,y_train,y_test)
```



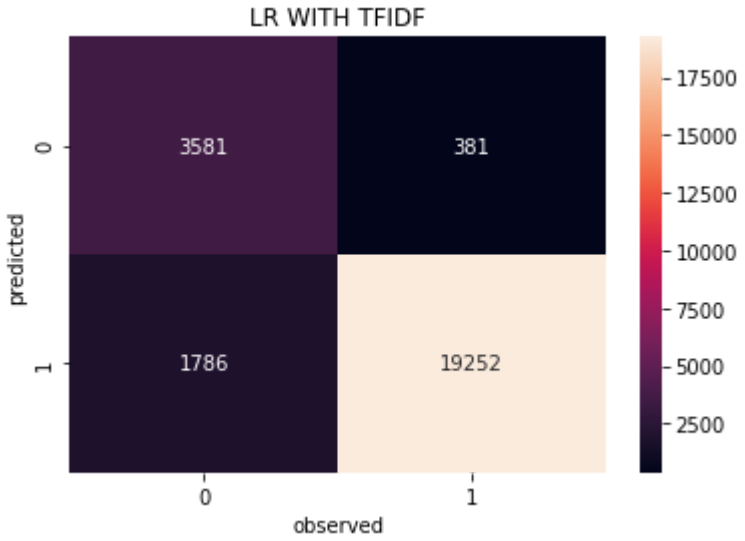
LR WITH TFIDF

==========



the classification report:

	precision	recall	f1-score	support
0	0.67	0.90	0.77	3962
1	0.98	0.92	0.95	21038
accuracy			0.91	25000
macro avg	0.82	0.91	0.86	25000
weighted avg	0.93	0.91	0.92	25000




```
from gensim.models import Word2Vec
```

```
def avg_words(x_train):
  i=0
  list_of_sentance=[]
  for sentance in x_train:
    list_of_sentance.append(sentance.split())

  w2v_models=Word2Vec(list_of_sentance,size=50,min_count=10,workers=8)
  w2v_words=list(w2v_models.wv.vocab)

  sent_vectors = []

  for sent in list_of_sentance:
      sent_vec = np.zeros(50)
      cnt_words =0;
      for word in sent: #
          if word in w2v_words:
              vec = w2v_models.wv[word]
              sent_vec += vec
              cnt_words += 1
      if cnt_words != 0:
          sent_vec /= cnt_words
      sent_vectors.append(sent_vec)

  return np.array(sent_vectors)



def tfidf(x_train):
    # S = ["abc def pqr", "def def def abc", "pqr pqr def"]
  model = TfidfVectorizer()
  list_of_sentance=[]
  for sentance in x_train:
    list_of_sentance.append(sentance.split())

  w2v_models=Word2Vec(list_of_sentance,size=50,min_count=10,workers=8)
  w2v_words=list(w2v_models.wv.vocab)

  tf_idf_matrix = model.fit_transform(x_train)
  # we are converting a dictionary with word as a key, and the idf as a value
  dictionary = dict(zip(model.get_feature_names(), list(model.idf_)))
  # TF-IDF weighted Word2Vec
  tfidf_feat = model.get_feature_names() # tfidf words/col-names
  # final_tf_idf is the sparse matrix with row= sentence, col=word and cell_val = tfidf

  tfidf_sent_vectors = []; # the tfidf-w2v for each sentence/review is stored in this list
  row=0;
  for sent in list_of_sentance: # for each review/sentence
      sent_vec = np.zeros(50) # as word vectors are of zero length
      weight_sum =0; # num of words with a valid vector in the sentence/review
      for word in sent: # for each word in a review/sentence
          if word in w2v_words and word in tfidf_feat:
              vec = w2v_models.wv[word]
              #tf_idf = tf_idf_matrix[row, tfidf_feat.index(word)]
              # to reduce the computation we are
              # dictionary[word] = idf value of word in whole courpus
              # sent.count(word) = tf valeus of word in this review
              tf_idf = dictionary[word]*(sent.count(word)/len(sent))
              sent_vec += (vec * tf_idf)
              weight_sum += tf_idf
```

```
      if weight_sum != 0:
          sent_vec /= weight_sum
      tfidf_sent_vectors.append(sent_vec)
      row += 1
  return np.array(tfidf_sent_vectors)


avg_x_train=avg_words(x_train)
avg_x_test=avg_words(x_test)
```

```
Buffered data was truncated after reaching the output size limit.
```

```
%time
w_x_train=tfidf(x_train)
w_x_test=tfidf(x_test)
```

```
CPU times: user 3 µs, sys: 0 ns, total: 3 µs
Wall time: 6.68 µs
Buffered data was truncated after reaching the output size limit.Buffered data was trun
```

```
nor.fit(avg_x_train)
```

```
Normalizer(copy=True, norm='l2')
```

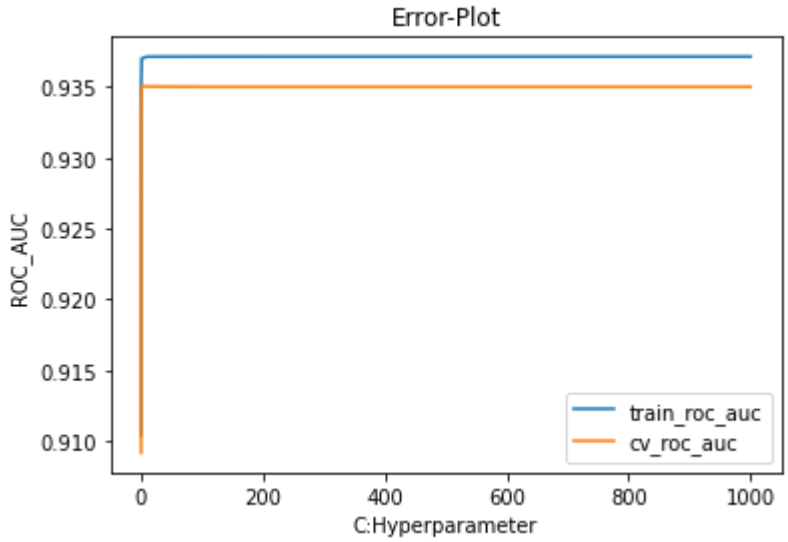
```
avg_train=nor.transform(avg_x_train)
avg_test=nor.transform(avg_x_test)


nor.fit(w_x_train)
```

```
Normalizer(copy=True, norm='l2')
```

```
w_train=nor.transform(w_x_train)
w_test=nor.transform(w_x_test)


%%time
cv_timeseries=TimeSeriesSplit(n_splits=5).split(avg_train)
avg_gcv=GridSearchCV(lg,param_grid=parameter,scoring="roc_auc",cv=cv_timeseries,return_trai
avg_gcv.fit(avg_train,y_train)
print(avg_gcv.best_params_)
print(avg_gcv.best_score_)
```

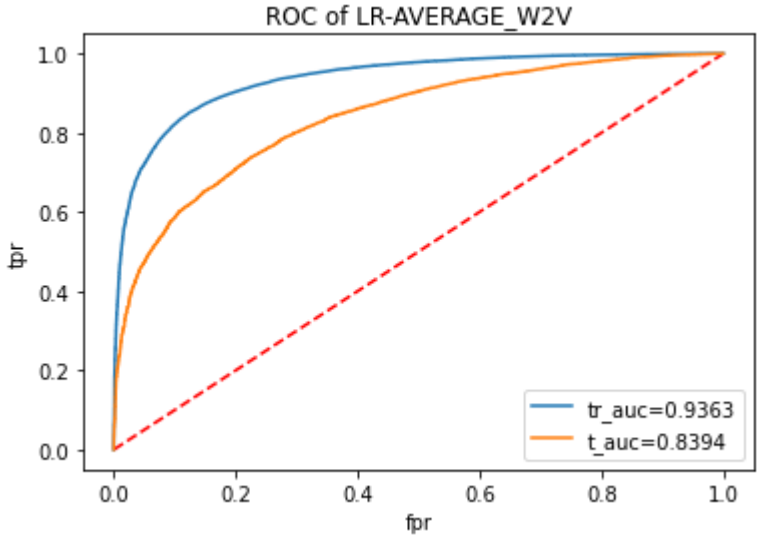
```
{'C': 1}
0.9350397069136711
CPU times: user 8.75 s, sys: 27.8 s, total: 36.6 s
Wall time: 2min 15s
```

```
plot_cv_results(avg_gcv,C)
plot_model("LR-AVERAGE_W2V",avg_gcv,avg_train,avg_test,y_train,y_test)
```



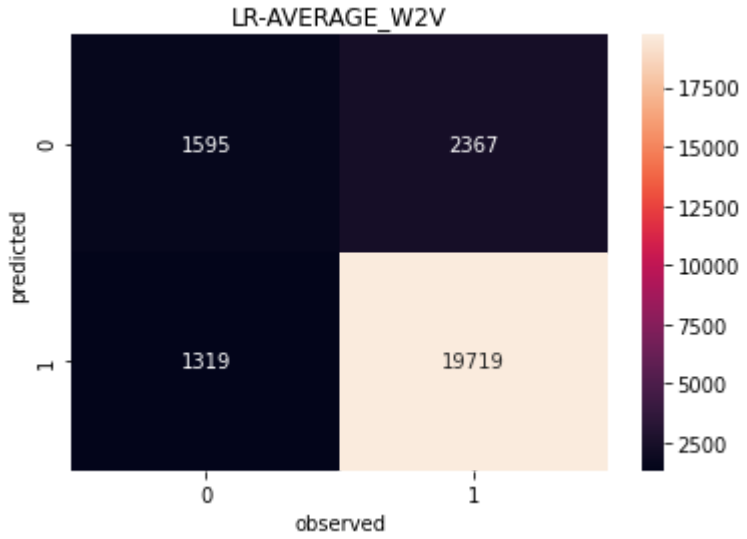
LR-AVERAGE_W2V

==========



the classification report:

	precision	recall	f1-score	support
0	0.55	0.40	0.46	3962
1	0.89	0.94	0.91	21038
accuracy			0.85	25000
macro avg	0.72	0.67	0.69	25000
weighted avg	0.84	0.85	0.84	25000

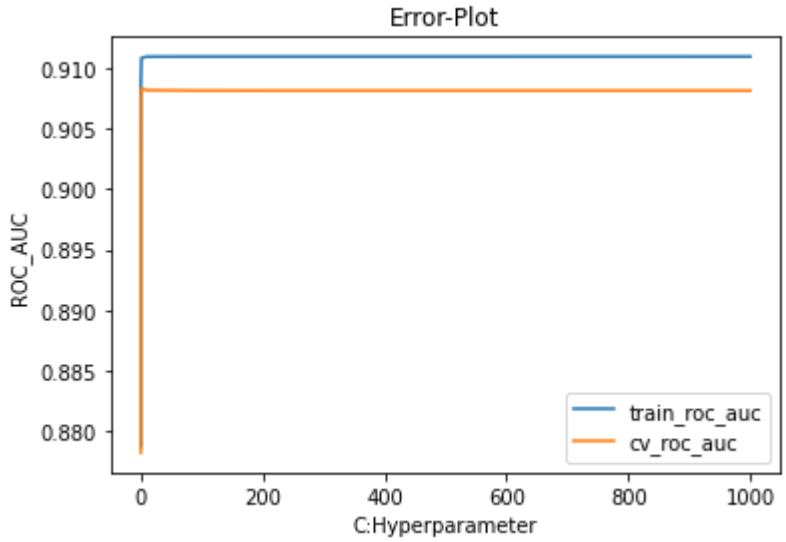



```
%%time
cv_timeseries=TimeSeriesSplit(n_splits=5).split(w_train)
```

```
w_gcv=GridSearchCV(lg,param_grid=parameter,scoring="roc_auc",cv=cv_timeseries,return_train_s
w_gcv.fit(w_train,y_train)
print(w_gcv.best_params_)
print(w_gcv.best_score_)
```

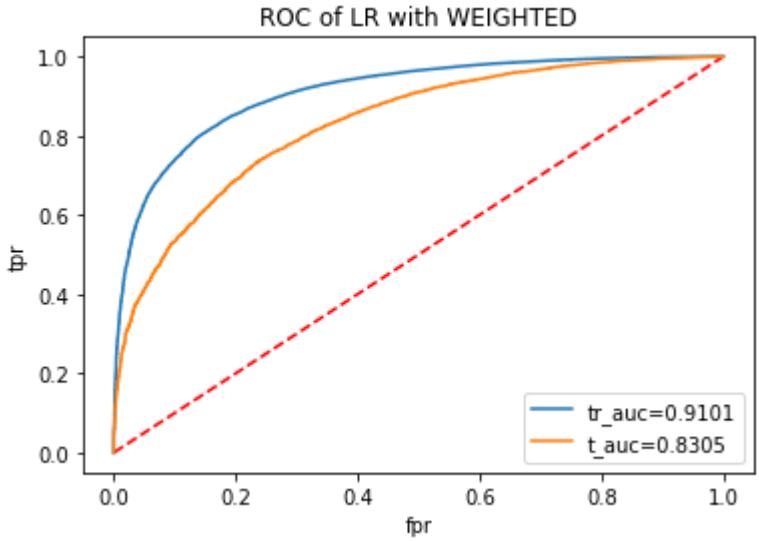
```
{'C': 1}
0.9082819288744375
CPU times: user 8.82 s, sys: 27.7 s, total: 36.5 s
Wall time: 1min 55s
```

```
plot_cv_results(w_gcv,C)
plot_model("LR with WEIGHTED",w_gcv,w_train,w_test,y_train,y_test)
```



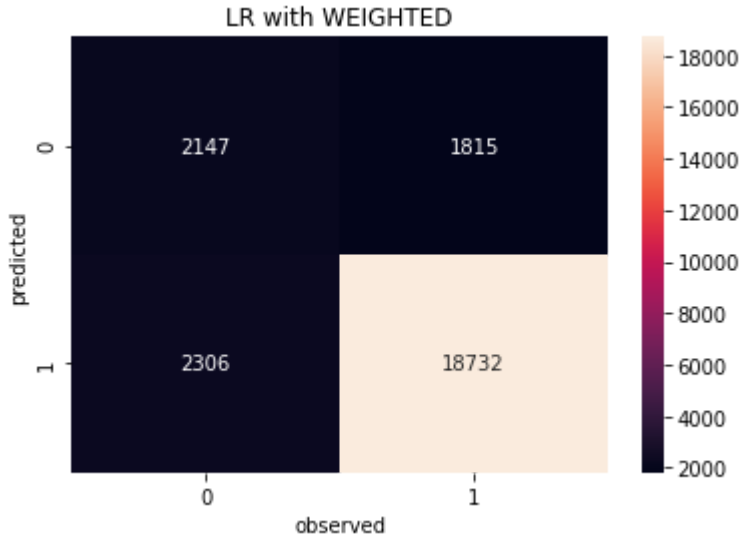
LR with WEIGHTED

==========



the classification report:

	precision	recall	f1-score	support
0	0.48	0.54	0.51	3962
1	0.91	0.89	0.90	21038
accuracy			0.84	25000
macro avg	0.70	0.72	0.71	25000
weighted avg	0.84	0.84	0.84	25000




```
from prettytable import PrettyTable
```

```
pt=PrettyTable()


pt.add_column("LOGISTIC REGRESSION",["BOW","TFIDF","AVG-W2V","TFIDF-W-W2V"])
pt.add_column("HYPERPARAMETER:C",[10,1,1,1])
pt.add_column("ROC_AUC",[0.96,0.98,0.93,0.90])
pt.add_column("TRAIN_AUC",[0.98,0.98,0.93,0.91])
pt.add_column("TEST_AUC",[0.97,0.96,0.83,0.83])
pt.add_column("TIME(min:sec)",["1:18","1:16","2:15","1:55"])
print(pt)
```

```
+---------------------+------------------+---------+-----------+----------+------------
| LOGISTIC REGRESSION | HYPERPARAMETER:C | ROC_AUC | TRAIN_AUC | TEST_AUC | TIME(min:se
+---------------------+------------------+---------+-----------+----------+------------
|         BOW         |        10        |   0.96  |    0.98   |   0.97   |      1:18
|        TFIDF        |        1         |   0.98  |    0.98   |   0.96   |      1:16
|       AVG-W2V       |        1         |   0.93  |    0.93   |   0.83   |      2:15
|     TFIDF-W-W2V     |        1         |   0.9   |    0.91   |   0.83   |      1:55
+---------------------+------------------+---------+-----------+----------+------------
```

CONCLUSION:

- As you can see our C=1/lambda is "1" in many cases hence our model neither overfitted nor underfitted
- Training Time using L-BFGS-B Solver is less as compared to LIBLINEAR Solver and it has low latency in used in Internet Companies.
- LIBLINEAR SOLVER without preprpcessing/standardization/normalization has High Accuracy.
- L1- Lasso Regularization has less dense matrix as compared to L2-Ridge Regularization as we can see coefficient in Lasso is ZERO compared to Ridge.
- Overall Score of Logistic Regression with TFIDF is best. From precision score to roc_auc_score to auc_