National College of Ireland

Configuration Manual

MSc Research Project
MSc. Data Analytics

Ojas Shivadatta Acharekar
Student ID: x23268492

School of Computing
National College of Ireland

Supervisor: Prof. Qurrat Ul Ain

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	OJAS SHIVADATTA ACHAREKAR
Student ID:	x23268492
Programme:	MSc. Data Analytics **Year:** 2024 – 2025
Module:	MSc. Research Project
Lecturer:	Prof. Qurrat Ul Ain
Submission Due Date:	15th September 2025
Project Title:	Sentiment-Driven Credit Risk Analysis: Hybrid Framework with Country-Level Fusion and Explainable AI
Word Count:	951 **Page Count:** 9

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: OJAS SHIVADATTA ACHAREKAR

Date: 15th September 2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	yes
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	**yes**

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual for my Research Project

Ojas Shivadatta Acharekar
Student ID: x23268492

1 Introduction

1.1 Background

Credit risk analysis plays an important role in the financial industry by assessing the likelihood of a borrower defaulting on their financial obligations. With the rise of social media, valuable insights can be drawn from online financial discussions. This project leverages sentiment analysis from financial tweets to enhance traditional credit risk models.

My study integrates sentiment scores derived from financial tweets into credit risk prediction models using machine learning techniques. It is divided into two phases: **Phase 1** focuses on sentiment extraction through natural language processing (NLP), while **Phase 2** combines these sentiments with credit risk data to improve risk classification models.

1.2 Aim of the Study

To enhance credit risk classification by incorporating public sentiment extracted from financial tweets using advanced NLP and deep learning models.

1.3 Research Objectives

1. To clean and preprocess unlabelled financial tweets.
2. To label tweet sentiments using VADER and visualize them.
3. To train deep learning models (LSTM, BiLSTM) for tweet sentiment classification.
4. To merge sentiment scores with a structured credit risk dataset.
5. To apply preprocessing techniques and balance class distributions using SMOTE.
6. To train and evaluate multiple machine learning models for credit risk prediction.
7. To use Explainable AI (LIME) to interpret and visualize model decisions.

1.4 Research Question

To what extent does integrating the sentiment scores extracted from financial tweets using deep learning models affect the accuracy, precision, recall, and F1 score of ML models in credit risk classification compared to models that only used structured financial data?

2 Environmental Setup

2.1 Hardware Requirements (My Local machine configurations but takes lot of time on training)

- 16GB RAM
- 500GB SSD Storage
- Intel Core i5 minimum

2.2 Software Requirements

- Operating System: Windows 10 or 11, both will work
- Python 3.10 because various machine learning models are not supported with the latest version of Python 3.10 is found to be stable
- Kaggle or Jupyter Notebook

2.3 Programming Tools

- IDE: Kaggle
- Language: Python
- GPU: Tesla T4 If you are running on Google Colab
- Visualization: Matplotlib, Seaborn
- Libraries: Pandas, Numpy, NLTK, TensorFlow, Keras, Scikit-learn, Gensim, Imbalanced-learn, LIME, CatBoost, XGBoost, WordCloud

3 Libraries Required

Library	Purpose
pandas	Data loading and manipulation
numpy	Numerical operations
nltk	Text preprocessing (stopwords, punctuation removal, etc.)
gensim	Word embeddings (Word2Vec)
tensorflow/keras	Deep learning models (LSTM, BiLSTM)
sklearn	ML models, evaluation metrics, and preprocessing
imblearn	SMOTE for class balancing
seaborn, matplotlib	Data visualization
lime	Explainable AI
xgboost, catboost	Gradient boosting models
wordcloud	Visualization of word frequencies
Vader Sentiment	Sentiment labelling of tweets

Install all the libraries that are required to perform the analysis and Implementation of all the modules, libraries, and packages used.

4 Dataset Details

4.1 Tweet Dataset

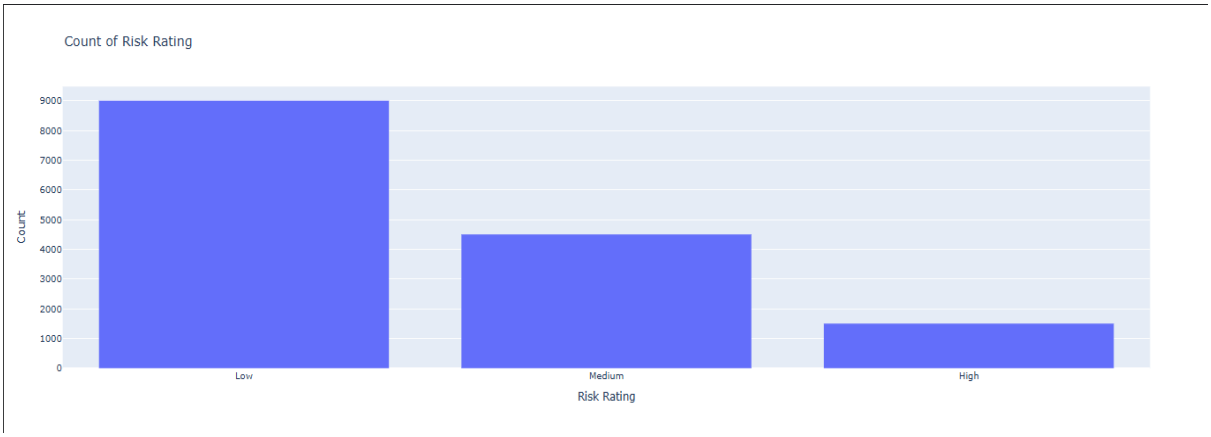
- Source: https://www.kaggle.com/datasets/davidwallach/financial-tweets
- Contains unlabelled financial tweets.
- Used for sentiment analysis and model training.

4.2 Credit Risk Dataset

- Source: https://www.kaggle.com/datasets/preethamgouda/financial-risk
- Contains structured credit risk data with country-level financial features.
- Target: Credit risk classification.

4.3 Merging Strategy

- Datasets are merged on the **country column**.
- The sentiment score derived from the tweet dataset is added as a new feature to the credit risk dataset.



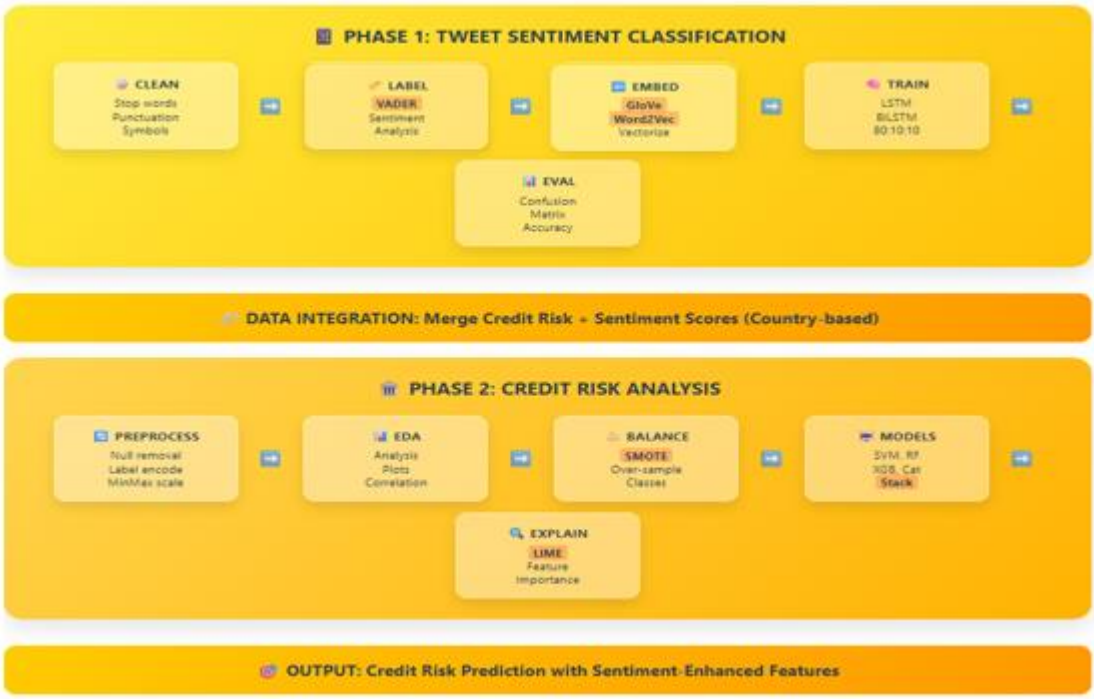
5 Project Flow

Phase 1: Sentiment Extraction from Tweets

1. **Text Cleaning:**
   - Handle null values which are there in the dataset for text column
   - Stop words, punctuation, symbols, and whitespace was removed
2. **Sentiment Labelling:**
   - Use TextBlob (VADER) Sentiment Analyzer for sentiment generation
3. **Text Preprocessing:**
   - Word embeddings using GloVe and TF-IDF
4. **Visualization:**
   - WordCloud and CountPlot for sentiment distribution
5. **Model Training:**
   - Deep Learning Models: LSTM, BiLSTM
6. **Evaluation:**
   - Confusion matrix, classification report, accuracy/loss graphs

Phase 2: Credit Risk Prediction

1. **Load and Merge Datasets (on Country level)**
2. **Data Cleaning and Preprocessing:**
   - Handle null values, encode labels, scale features using MinMaxScaler
3. **Class Balancing:**
   - Apply SMOTE for oversampling and imbalanced data
4. **Split Data:**
   - 90% Train / 10% Test
5. **Model Training:**
   - SVM, Random Forest, XGBoost, CatBoost, Stacking Classifier
6. **Evaluation Metrics:**
   - Classification report
   - Confusion matrix
7. **Explainability:**
   - Using LIME explainable AI for enhancing the interpretability, which features that contributed most to a model's decision



6 Implementation

Phase 1: Sentiment classification Using LSTM & BiLSTM using different Embedding techniques like (TF-IDF Vectorization & GLoVe Embedding)

```
# Libraries for NLP, Machine Learning, Visualization, and Deep Learning

import re # For regular expressions (text cleaning)
import json
import string
import joblib # For saving/loading models and preprocessors
import pickle
import wordninja # For splitting concatenated words

import math, nltk
import numpy as np
import pandas as pd
import seaborn as sns
from tqdm import tqdm
import tensorflow as tf # For deep learning models
import plotly.express as px # For interactive plotting

from textblob import TextBlob # For text sentiment analysis
from wordcloud import WordCloud # For generating word clouds
import matplotlib.pyplot as plt
from nltk.corpus import stopwords
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer # For TF-IDF vectorization
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report # For model evaluation metrics

from tensorflow.keras.models import Sequential, load_model # For sequential neural networks and model loading
from tensorflow.keras.layers import Embedding, LSTM, Dense, Dropout, Bidirectional # For layers like LSTM, Embedding, Dropout, Bidirectional
from tensorflow.keras.layers import Dropout, Embedding, GlobalMaxPooling1D, MaxPooling1D, Add, Flatten # Additional layers for CNNs and pooling
from tensorflow.keras.preprocessing.text import Tokenizer # For converting text to sequences
from tensorflow.keras.preprocessing.sequence import pad_sequences # For padding sequences to a uniform length
```

Above are the libraries required for sentiment analysis, and they are already installed using the pip command

[5]:
```
# Load cleaned sentiment dataset
dataframe=pd.read_csv('/kaggle/input/country-sentiments/cleaned_sentiment.csv') # Load CSV file into DataFrame
dataframe.head()
```

[5]:

	Unnamed: 0	id	text	timestamp	source	symbols	company_names	url	verified	country
0	0	1019696670777503700	According to Gran , the company has no plans t...	Wed Jul 18 21:33:26 +0000 2018	GoldmanSachs	GS	The Goldman Sachs	https://twitter.com/i/web/status/1019696670777...	True	United States of America
1	1	1019709091038548000	Technopolis plans to develop in stages an area...	Wed Jul 18 22:22:47 +0000 2018	StockTwits	M	Macy's	https://twitter.com/i/web/status/1019709091038...	True	United States of America
2	2	1019711413798035500	The international electronic industry company ...	Wed Jul 18 22:32:01 +0000 2018	TheStreet	AIG	American	https://buff.ly/2L3kmc4	True	Other
3	3	1019716662587740200	With the new production plant the company woul...	Wed Jul 18 22:52:52 +0000 2018	MarketWatch	BTC	Bitcoin	https://twitter.com/i/web/status/1019716662587...	True	Other
4	4	1019718460287389700	According to the company 's updated strategy f...	Wed Jul 18 23:00:01 +0000 2018	Forbes	ORCL	Oracle	http://on.forbes.com/6013DqDDU	True	United States of America

+ Code + Markdown

Data for Sentiment analysis, important ones are the text and country columns

[12]:
```
# Text Cleaning Pipeline: Apply all cleaning functions step-by-step on the 'text' column

dataframe['clean_tweet']=dataframe['text'].apply(lambda x: html_references(x)) # Remove URLs, HTML entities, lowercase
dataframe['clean_tweet']=dataframe['clean_tweet'].apply(lambda x: decontraction(x)) # Expand contractions
dataframe['clean_tweet']=dataframe['clean_tweet'].apply(lambda x: filter_punctuations_etc(x)) # Remove punctuation and non-printable chars
dataframe['clean_tweet']=dataframe['clean_tweet'].apply(lambda x: separate_alphanumeric(x)) # Separate alphanumeric tokens
dataframe['clean_tweet']=dataframe['clean_tweet'].apply(lambda x: unique_char(cont_rep_char, x)) # Reduce repeated characters
dataframe['clean_tweet']=dataframe['clean_tweet'].apply(lambda x: split_attached_words(x)) # Split attached words
dataframe['clean_tweet']=dataframe['clean_tweet'].apply(lambda x: stopwords_shortwords(x)) # Remove stopwords and short words
```

Cleaning steps carried out on the text column

[14]:
```
dataframe.drop('timestamp', axis=1, inplace=True) # Remove the 'timestamp' column from the dataframe
dataframe.head(5)
```

[14]:

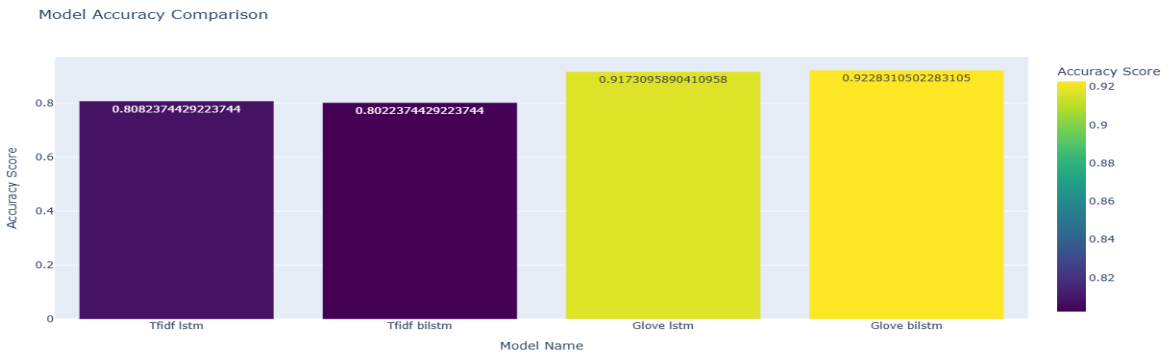
	company_names	country	text	clean_tweet
0	The Goldman Sachs	United States of America	According to Gran , the company has no plans t...	according gran company plans move production r...
1	Macy's	United States of America	Technopolis plans to develop in stages an area...	techno polis plans develop stages area less 10...
2	American	Other	The international electronic industry company ...	international electronic industry company el c...
3	Bitcoin	Other	With the new production plant the company woul...	new production plant company would increase ca...
4	Oracle	United States of America	According to the company 's updated strategy f...	according company updated strategy years 20092...

This is the clear view of clean and normal text.

[20]:

	company_names	country	text	clean_tweet	sentiment	sentiment_score	country_sen_score	country_sentiment
0	The Goldman Sachs	United States of America	According to Gran , the company has no plans t...	according gran company plans move production r...	neutral	0.000000	0.056285	positive
1	Macy's	United States of America	Technopolis plans to develop in stages an area...	techno polis plans develop stages area less 10...	negative	-0.166667	0.056285	positive
2	American	Other	The international electronic industry company ...	international electronic industry company el c...	neutral	0.000000	0.081773	positive
3	Bitcoin	Other	With the new production plant the company woul...	new production plant company would increase ca...	negative	-0.064802	0.081773	positive
4	Oracle	United States of America	According to the company 's updated strategy f...	according company updated strategy years 20092...	negative	-0.016667	0.056285	positive

Data after using the TextBlob Library for sentiment generation.



Deep Learning models for sentiment analysis

Accuracy of trained models with different embedding techniques .

Phase 2 Merging dataset Based on the country level

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import plotly.express as px
import seaborn as sns
import warnings  # Warning control

from sklearn.preprocessing import LabelEncoder
from sklearn.preprocessing import MinMaxScaler  # Feature scaling to range [0,1]
from sklearn.model_selection import train_test_split, GridSearchCV  # Data splitting and hyperparameter tuning
from cuml.svm import SVC  # GPU-accelerated Support Vector Classifier from cuML
from sklearn import metrics  # Performance metrics
from sklearn.metrics import classification_report, accuracy_score, confusion_matrix  # Specific metrics
from sklearn.ensemble import RandomForestClassifier  # CPU Random Forest
from catboost import CatBoostClassifier  # Gradient boosting with categorical features support
from xgboost import XGBClassifier  # Extreme Gradient Boosting classifier
from cuml.ensemble import RandomForestClassifier  # GPU-accelerated Random Forest from cuML
from mlxtend.classifier import StackingClassifier  # Ensemble stacking classifier
import joblib
import os

warnings.filterwarnings('ignore')  # Suppress warnings for cleaner output
```

+ Code + Markdown

[8]:
```
# First, create a dictionary from sentiments for faster lookup
sentiment_dict = sentiments.set_index('Country')['country_sentiment'].to_dict()

# Get the 'Other' sentiment score as the default fallback
default_score = sentiment_dict.get('Other', 0)  # You can set default=0 if 'Other' doesn't exist

# Map the sentiment scores to df['Country'] using the dictionary, fallback to 'Other' score
df['country_sentiment'] = df['Country'].apply(lambda x: sentiment_dict.get(x, default_score))
```

[9]:
```
df.head(5)
```

[9]:

	Age	Gender	Education Level	Marital Status	Income	Credit Score	Loan Amount	Loan Purpose	Employment Status	Years at Current Job	...	Debt-to-Income Ratio	Assets Value	Number of Dependents	City	State	Country	Previous Defaults	Marital Status Change	Risk Rating	country_sentiment
0	49	Male	PhD	Divorced	72799.0	688.0	45713.0	Business	Unemployed	19	...	0.154313	120228.0	0.0	Port Elizabeth	AS	Cyprus	2.0	2	Low	positive
1	57	Female	Bachelor's	Widowed	NaN	690.0	33835.0	Auto	Employed	6	...	0.148920	55849.0	0.0	North Catherine	OH	Turkmenistan	3.0	2	Medium	positive
2	21	Non-binary	Master's	Single	55687.0	600.0	36623.0	Home	Employed	8	...	0.362398	180700.0	3.0	South Scott	OK	Luxembourg	3.0	2	Medium	positive
3	59	Male	Bachelor's	Single	26508.0	622.0	26541.0	Personal	Unemployed	2	...	0.454964	157319.0	3.0	Robinhaven	PR	Uganda	4.0	2	Medium	positive
4	25	Non-binary	Bachelor's	Widowed	49427.0	766.0	36528.0	Personal	Unemployed	10	...	0.143242	287140.0	NaN	New Heather	IL	Namibia	3.0	1	Low	positive

5 rows × 21 columns

Merging the data based on country

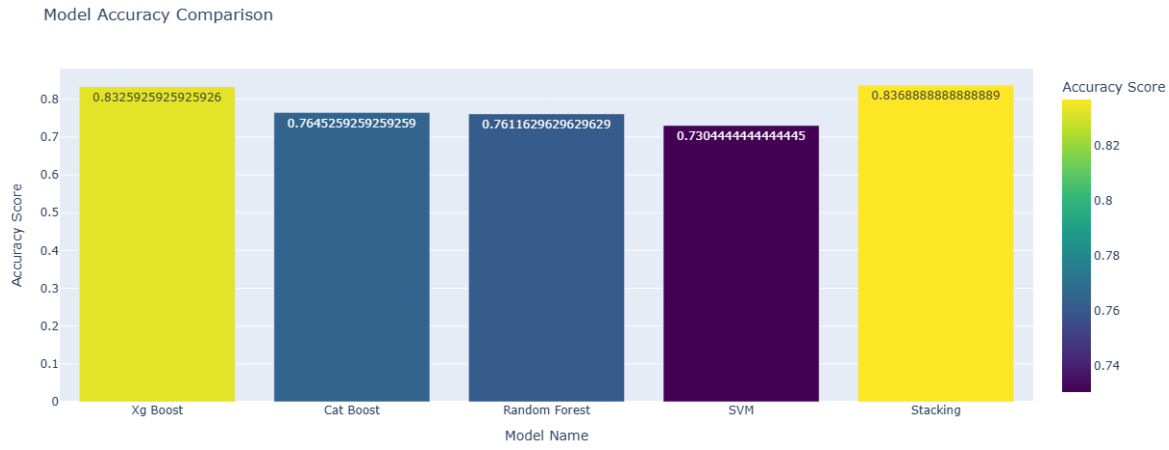
[27]:
```
from imblearn.over_sampling import SMOTE  # Import SMOTE for handling class imbalance
smote =SMOTE()  # Initialize SMOTE object
sm_x,sm_y=smote.fit_resample(X,y)  # Apply SMOTE to resample the feature and target data
joblib.dump(smote,'/kaggle/working/models/smote.sav')  # Save the SMOTE object to a file
```

Data Balance (Using Smote)

The data was imbalanced, and because of this problem, it was a need to have some algorithm so that data can be balanced and the results or training of the model should not be hampered.

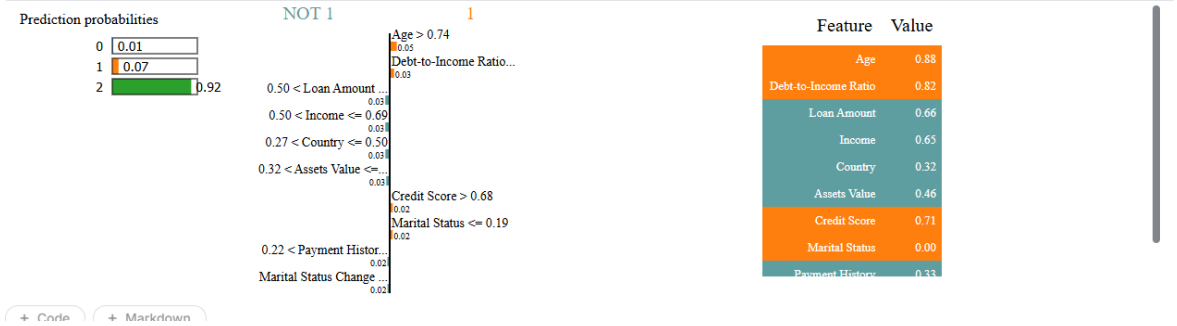
For this reason, I used a simple approach of duplicating the minority class using SMOTE. SMOTE is abbreviated to Synthetic Minority Oversampling Technique. The problem of data imbalance is resolved in the dataset by oversampling the minority classes of the dataset.

Scaling and Label encoding were also performed on the data, then models were trained.
Training models
SVM
Random Forest
CatBoost
XGBoost
Stacking(Having the best parameters from all the models and combining them to have a improved accuracy)



Machine learning models for credit risk classification with sentiment scores

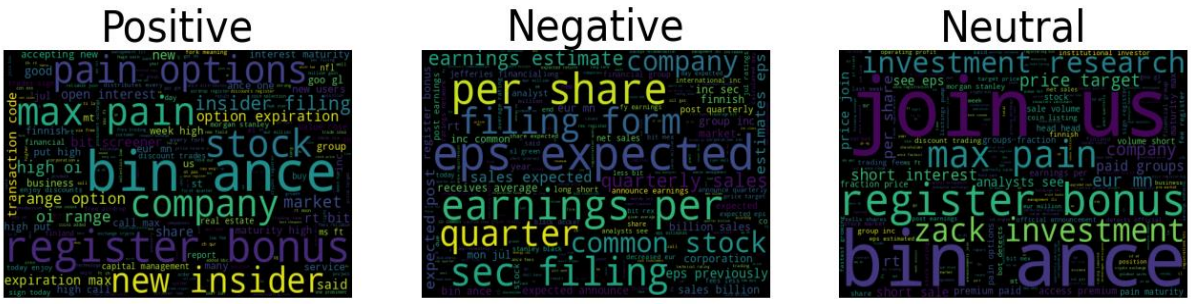
I have integrated the stacking model with LIME of few credit profiles and this interpretation on the data was being observed.



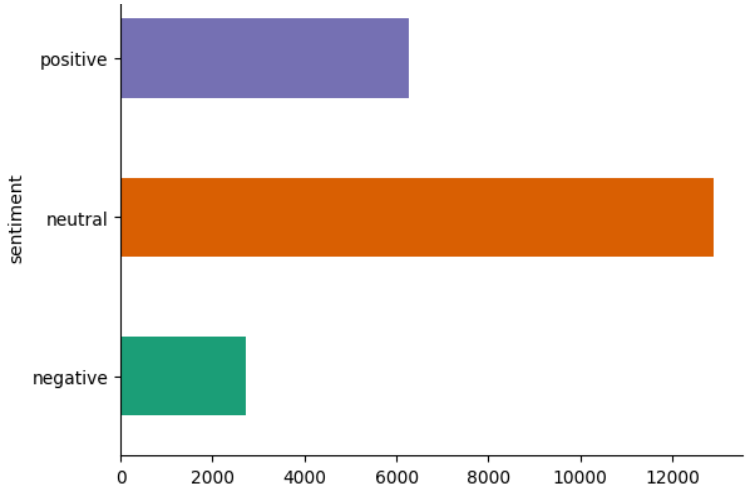
Lime Explainable AI

7 Output

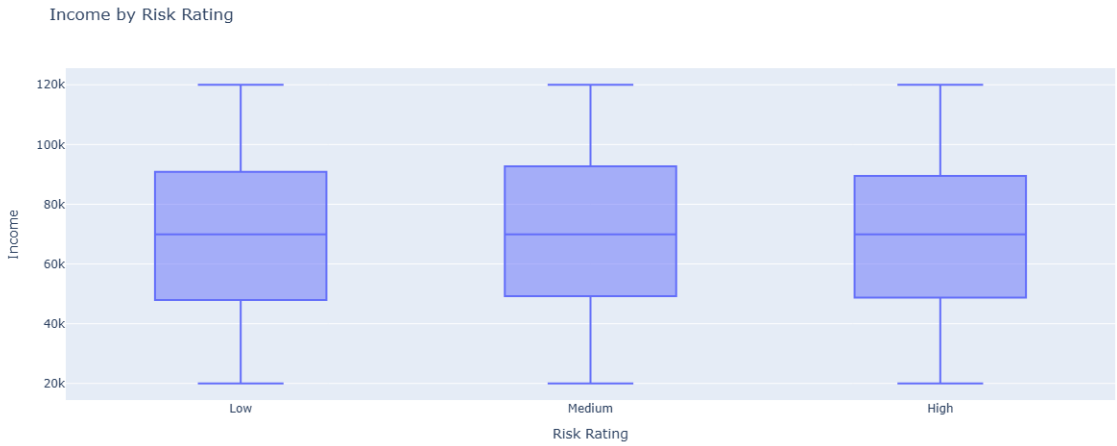
EDA for sentiment Classification

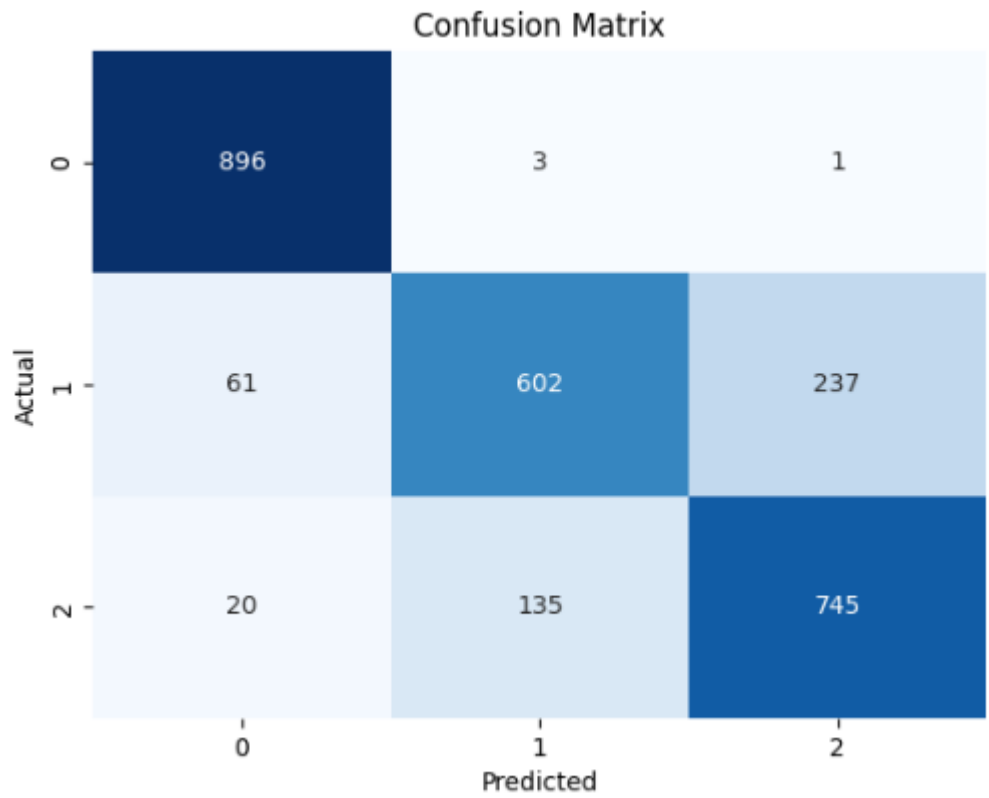


Word Cloud



EDA for Credit Risk Classification Problem





For Stacking Algorithm, the above is the confusion matrix