National College of Ireland

Configuration Manual

MSc Research Project
MSc in Science in AI for Business (MSCAIBUS1)

Osaigbovo Daniel Adoghe
Student ID: x23139013

School of Computing
National College of Ireland

Supervisor: Dr. Devanshu Anand

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Osaigbovo Daniel Adoghe
Student ID:	x23139013
Programme:	MSc in Science in AI for Business (MSCAIBUS1)
Year:	2024
Module:	MSc Research Project
Supervisor:	Dr. Devanshu Anand
Submission Due Date:	12/08/2024
Project Title:	Configuration Manual
Word Count:	501
Page Count:	5

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:	Osaigbovo Daniel Adoghe
Date:	16th September 2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
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.	☐

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

Osaigbovo Daniel Adoghe
x23139013

1 System Requirements

1.1 Hardware:

- Google Colab for GPU acceleration.
- Local machine with internet access for initial setup (optional).

1.2 Software:

- Python 3.8 or higher.
- Google Colab account.
- **Libraries**: TensorFlow, Keras, Scikit-learn, NLTK, Matplotlib, Pandas, NumPy.

((Chollet et al.; 2015), (Pedregosa; 2011))

2 Environment Setup on Google Colab

2.1 Set Up Google Colab Environment

- Access Google Colab and create a new notebook.
- Enable GPU support in Google Colab, navigate to `Runtime > Change runtime type > Hardware accelerator` and select `GPU`.

(Google; 2021)

2.2 Upload the .ipynb File

- Click on the File menu in Colab.
- Select **Upload notebook**.
- In the dialog, click Choose File and select your .ipynb file from your computer.
- The notebook will open automatically in Colab.

2.3 Install Necessary Libraries

- Install and import libraries by running the following code in a Colab cell (Brownlee; 2017):

```
import os
import numpy as np
import csv
import pandas as pd
import matplotlib.pyplot as plt
from matplotlib.dates import DateFormatter
from sklearn.metrics import accuracy_score, precision_score, recall_score
import math
import time
import tensorflow as tf
from tensorflow.keras.layers import GRU, LSTM, Bidirectional, Dense, Flatten, Conv1D, BatchNormalization, LeakyReLU, Dropout,Permute
from tensorflow.keras.layers import Multiply, Permute, Attention, Concatenate, Dropout, LayerNormalization, Input,Flatten, Input
from tensorflow.keras import Sequential
from tensorflow.keras.utils import plot_model
from pickle import load
from sklearn.metrics import mean_squared_error
from tqdm import tqdm
import statsmodels.api as sm
from math import sqrt
from datetime import datetime, timedelta
from sklearn.preprocessing import MinMaxScaler
from pickle import dump
from nltk.sentiment.vader import SentimentIntensityAnalyzer
from nltk.sentiment.vader import SentimentIntensityAnalyzer
import unicodedata
from sklearn.model_selection import KFold
import tensorflow as tf
from tensorflow.keras.models import Model

import warnings
warnings.filterwarnings("ignore")
```

Figure 1: Enter Caption

2.4 Running the Code

- Click on each code cell in sequence, starting from the top, and press Shift + Enter to execute the code.
- Ensure that you follow the steps sequentially to avoid errors.

3 Data Preparation

3.1 Load and Preprocess Data

3.1.1 Tweet Data

Load tweet data related to the selected stock (e.g., AMZN) and perform sentiment analysis using the 'SentimentIntensityAnalyzer' from NLTK. The data is grouped by date to compute average sentiment scores.

3.1.2 Stock Data

Load historical stock data and merge it with sentiment data to create a unified dataset for analysis. Hutto and Gilbert (2014)

```
all_tweets = pd.read_csv('/content/stock_tweets 3.csv')
all_stocks = pd.read_csv('/content/stock_yfinance_data (1).csv')
```

Figure 2: Enter Caption

3.2 Add Technical Indicators

To enhance the dataset, integrate technical indicators such as Moving Averages (MA7, MA20), MACD, Bollinger Bands, and RSI.

4 Model Development and Training

4.1 LSTM Model

- Build a sequential model with two LSTM layers, each containing 50 units, followed by a Dense output layer. The model is trained using the 'adam' optimizer and mean_squared_error loss function.
- The model is trained on the preprocessed stock data.

4.2 Random Forest Model

Train a Random Forest model on selected stock features such as Open, High, Low, and Volume. The model is used as a baseline for comparison with deep learning models. (Breiman; 2001)

```
[ ]  # Create and train the Random Forest Classifier
     classifier = RandomForestClassifier(n_estimators=100, random_state=42)
     classifier.fit(X_train_scaled, y_train)
```

Figure 3: Enter Caption

4.3 GAN Model

4.3.1 Generator:

Construct a model using five LSTM blocks, followed by Dense layers and Dropout for regularization.

4.3.2 Discriminator:

Implement using Conv1D layers followed by Dense layers for binary classification of generated and real data. (Goodfellow et al.; 2014)

```
generator = make_generator_model(input_dim, output_dim, feature_size)
discriminator = make_discriminator_model(input_dim)
```

Figure 4: Enter Caption

4.4 Training and Evaluation

Train the models on the prepared dataset and evaluate their performance using metrics like RMSE, accuracy, precision, and recall.(Bishop and Nasrabadi; 2006)

5 Visualization and Results

5.1 Plotting:

Use matplotlib to generate plots for actual vs. predicted stock prices and compare model performance across different metrics.(Hunter; 2007)

5.2 Saving Results

Ensure all results are saved and documented, making them reproducible.

6 Troubleshooting and Optimization

6.1 Overfitting:

Use dropout layers and L2 regularization to prevent overfitting in the models.

6.2 Data Imbalance:

Address data imbalance issues using resampling techniques to improve model performance.

6.3 Resource Management:

Monitor GPU usage in Colab to avoid session timeouts during training.(Ng; 2018)

References

Bishop, C. M. and Nasrabadi, N. M. (2006). *Pattern recognition and machine learning*, Vol. 4, Springer.

Breiman, L. (2001). Random forests, *Machine learning* **45**: 5–32.

Brownlee, J. (2017). *Long short-term memory networks with python: develop sequence prediction models with deep learning*, Machine Learning Mastery.

Chollet, F. et al. (2015). keras.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. and Bengio, Y. (2014). Generative adversarial nets, *Advances in neural information processing systems* **27**.

Google (2021). Google colaboratory. Retrieved from `https://colab.research.google.com`.

Hunter, J. D. (2007). Matplotlib: A 2d graphics environment, *Computing in science & engineering* **9**(03): 90–95.

Hutto, C. and Gilbert, E. (2014). Vader: A parsimonious rule-based model for sentiment analysis of social media text, *Proceedings of the international AAAI conference on web and social media*, Vol. 8, pp. 216–225.

Ng, A. (2018). Machine learning yearning: Technical strategy for ai engineers, in the era of deep learning.

Pedregosa, F. (2011). Scikit-learn: Machine learning in python fabian, *Journal of machine learning research* **12**: 2825.