Configuration Manual

MSc Research Project
AI for Business

Aida Metenova
Student ID: 22247700

School of Computing
National College of Ireland

Supervisor: Dr. Muslim Jameel Syed

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: ....Aida Metenova ....

Student ID: ......x22247700..

Programme: ......MSCAIBUS.. **Year:** ...2023-24..

Module: ......MSc Research Project...

Lecturer: ......Dr. Muslim Jameel Syed...

Submission Due Date: ......12 Aug 2024...

Project Title: ...... Configuration Manual ....

Word Count: ......303.................................. **Page Count:** ....6......................................

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: ......Aida Metenova..

Date: ......11 Aug 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

Aida Metenova
x22247700

1 System Configuration

The project was executed in Google Colab which is a cloud-based environment that provides access to hardware like CPUs and GPUs. The configuration of the Colab included access to Python 3 and a minimum of 12GB of RAM.

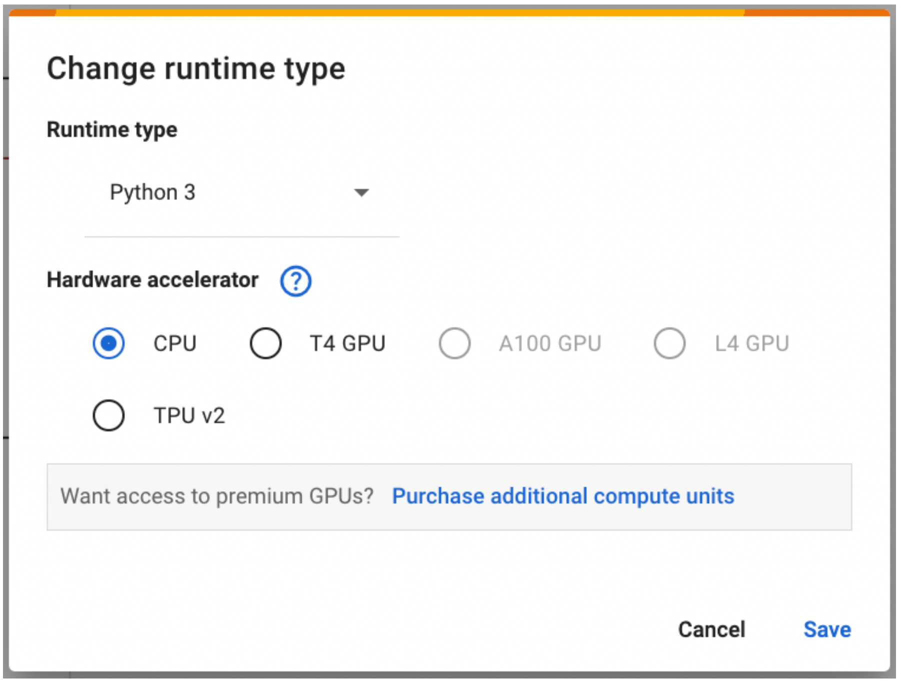

Figure 1. Colab runtime configuration

2 Software Requirement

The project used the following software provided by Google Colab:
- Python 3
- Jupyter Notebook integrated within Colab.

3 Python Libraries

The following Python libraries were used in this project:
- pandas
- scikit-learn
- TensorFlow
- Keras

- matplotlib
- seaborn
- numpy

The libraries were installed using the following command:

```
# Install Necessary Libraries
!pip install pandas scikit-learn tensorflow matplotlib seaborn
```

Figure 2. Import of the libraries

4 Dataset

The dataset used was taken from open-sources repository and uploaded directly into the Colab environment (Khakzad & Khakzad, 2021). It contained features like temperature, pressure, flow velocity, and pH level.

5 Data Preprocessing

Data preprocessing steps included:

- Dropping unnecessary columns and rows
- Handling missing values
- Scaling features for standardization.

```
# Step 2: Data Cleaning and Preprocessing
# Drop columns with irrelevant metadata
data_cleaned = data.drop(columns=['Unnamed: 8', 'Unnamed: 9', 'Unnamed: 10'])

# Drop any rows that are not part of the main data (e.g., rows containing headers again)
data_cleaned = data_cleaned.dropna()

# Define features and target variable
X = data_cleaned.drop(columns=['Corrosion rate'])
y = data_cleaned['Corrosion rate']

# Standardize the features
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
```

Figure 3. Preprocessing steps

6 Model Training and Testing

The following models were trained and tested with the preprocessed dataset:

- Random forest regression
- Support vector regression (SVR)
- Convolutional neural network (CNN)
- Long short-term memory (LSTM).

Each model training was carried out with the following code structures.

1. Random forest

```
# Random Forest
rf = RandomForestRegressor()
rf.fit(X_train, y_train)
rf_pred = rf.predict(X_test)
```

Figure 4. Random forest

2. Support vector regression

```
# Support Vector Machine
svm = SVR()
svm.fit(X_train, y_train)
svm_pred = svm.predict(X_test)
```

Figure 5. SVR

3. Convolution neural network

```
# CNN Model
cnn = Sequential()
cnn.add(Conv1D(64, kernel_size=2, activation='relu', input_shape=(X_train.shape[1], 1)))
cnn.add(Flatten())
cnn.add(Dense(1))
cnn.compile(optimizer='adam', loss='mse')
cnn.fit(X_train.reshape(-1, X_train.shape[1], 1), y_train, epochs=10, validation_split=0.2)
cnn_pred = cnn.predict(X_test.reshape(-1, X_test.shape[1], 1))
```

Figure 6. CNN

4. Long short-term memory

```
# LSTM Model
lstm = Sequential()
lstm.add(LSTM(50, return_sequences=True, input_shape=(X_train.shape[1], 1)))
lstm.add(LSTM(50))
lstm.add(Dense(1))
lstm.compile(optimizer='adam', loss='mse')
lstm_history = lstm.fit(X_train.reshape(-1, X_train.shape[1], 1), y_train, epochs=10, validation_split=0.2)
```

Figure 7. LSTM

7 Model Evaluation

The model evaluation involved plotting and comparing the following metrics:

- Training and validation loss
- Actual vs. Predicted values
- Residual plots

The plots were generated using Matplotlib and Seaborn libraries.

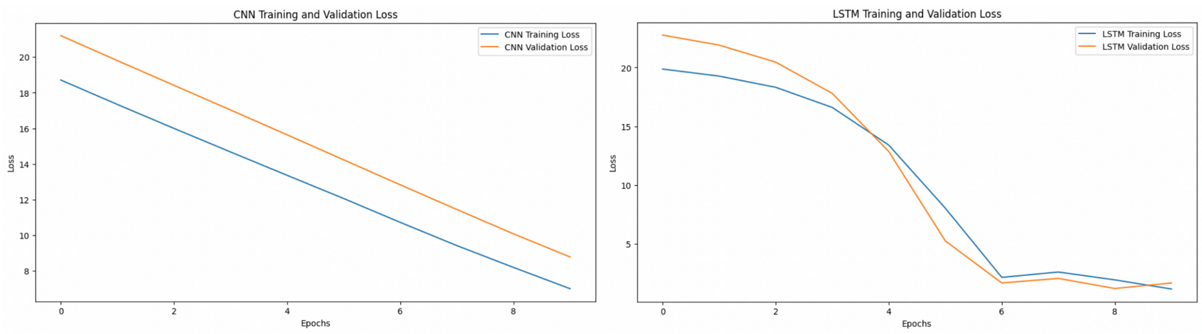

Figure 8. Training and validation loss

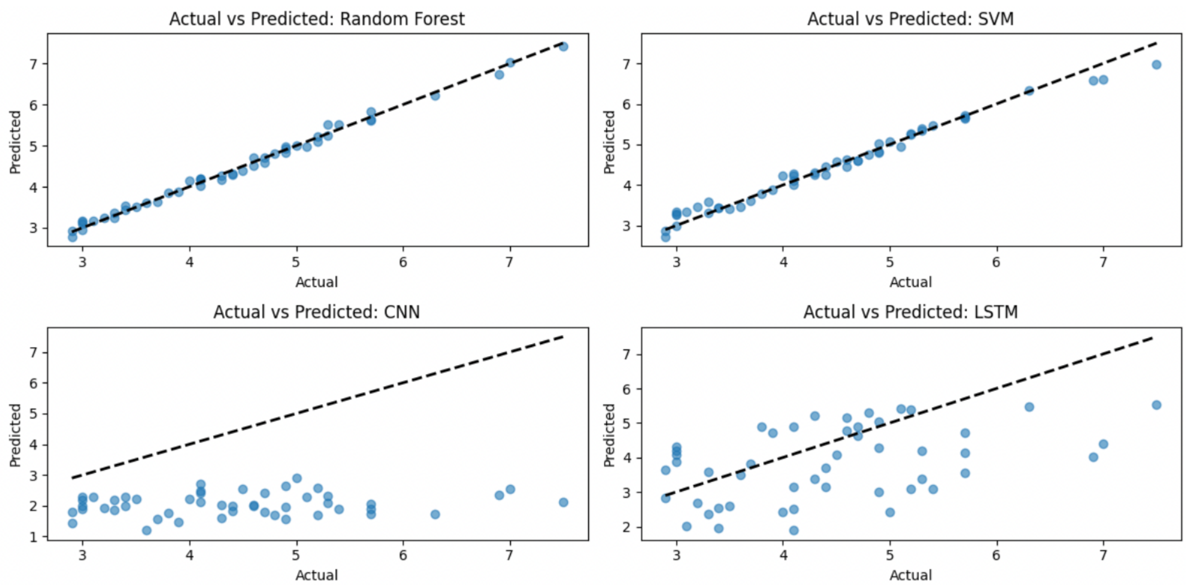

Figure 9. Actual vs. predicted values

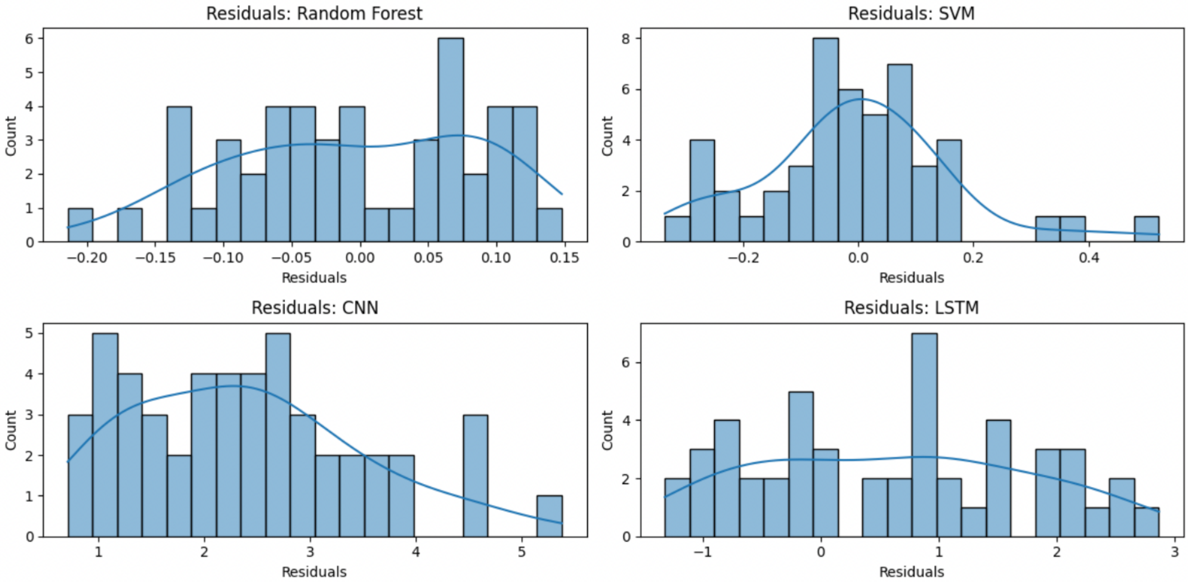

Figure 10. Residuals

Model	Train MAE	Validation MAE	Train RMSE	Validation RMSE	Train R2 Score	Validation R2 Score
Random Forest	0.230	0.313	0.326	0.451	0.891	0.819
SVM	0.294	0.354	0.395	0.484	0.841	0.802
CNN	0.099	0.318	0.141	0.419	0.977	0.835
LSTM	0.168	0.286	0.236	0.384	0.938	0.854

Figure 11. Comparison of the results

References

Khakzad, S., & Khakzad, N. (2021). *Simulation data for CO2 corrosion rate of oil pipeline*. Retrieved from Mendeley Data: https://data.mendeley.com/datasets/4nydhxjymw/1