National College of Ireland

Configuration Manual

MSc Research Project
MSc in Cloud Computing

Abhishek Goud Chathurpally
Student ID: 22236783

School of Computing
National College of Ireland

Supervisor: Shreyas Setlur Arun

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Abhishek Goud Chathurpally		
Student ID:	22236783		
Programme:	MSc in Cloud Computing	**Year:**	2014-2025
Module:	MSc Research Project		
Lecturer:	Shreyas Setlur Arun		
Submission Due Date:	29/01/2025		
Project Title:	Enhancing Predictive Analysis through Machine Learning Models in Cloud Computing Environments		
Word Count:	**1190**	**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: Abhishek Goud Chathurpally

Date: 28/01/2025

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

Abhishek Goud Chathurpally
Student ID: 22236783

1 Environment Setup

Configuring the environment for preprocessing, training of the models, and deployment on cloud infrastructure is the first step in implementation of this thesis project. On this part, it gives an overview of what you'll need to prepare before installing to your local or cloud environment

1.1 Hardware Requirements :

- **Local Machine**: Minimum system requirements are so that initial data preprocessing and model experimentation can be preformed using system with at least 8 GB RAM, 50 GB of free disk space, and a modern processor (e.g., Intel i5 or AMD Ryzen 5).

- **AWS Cloud :** Run on scalable infrastructure using Amazon Web Services. EC2, S3, Lambda, and API Gateway are what they need. Make it so that you can access your AWS Management console.

1.2 Software Requirements :

- **Python:** For performing any of the programming task, install Python 3.9 or later. To manage dependencies, it's best to setup a virtual environment

- **Libraries:** Install the following Python libraries using pip install
    - **pandas:** Used for (data) manipulation and preprocessing**.**
    - **numPy:** For numerical computations.
    - **scikit-learn:** For developing evaluation of machine learning models.
    - **jilbab:** For model serialization.
    - **boto3:** To use AWS services inside of Python scripts.
    - **matplotlib (optional):** For exploratory data analysis if you have any data visualization tasks

1.3 Setting Up AWS :

- **AWS Account:** For the initial experimentation, always use the AWS Free Tier.

- **AWS CLI**: AWS Command Line Interface (CLI) CLI command line tool is installed on your local machine to automate cloud operations. You can use aws configure to configure your credentials. You will need
    - AWS Access Key ID
    - AWS Secret Access Key

- Default region (e.g., us-east-1)

List available S3 buckets with the command and test the connection.
"aws s3 ls"

- **IAM Roles :** Permissions for IAM roles
  - An EC2 instance role with permissions to read/write from S3.
  - An S3 and CloudWatch permission helper lambda execution role

2 Data Preprocessing and Model Development

2.1 Dataset: In the dataset we have stock price data from Yahoo finance data, which include opening price, closing price, high price, low price, adjusted close, volume etc. Download the dataset, put it on an S3 bucket for cloud storage

2.2 Data Preprocessing: Do some of the local pre-processing using python and Pandas library.

- **Handle Missing Values:** Find out how to deal with unknown values. You can interpolate or if the row is incomplete, remove it.

- **Feature Engineering:** Derive additional features such as:
  - **Daily Return:** Closing price of the day versus previous day in percentage change.
  - **Moving Average**: 5 Days window rolling average.

- **Normalization:** If your numerical features (e.g., closing prices) are not normalized to be on the same scale, they will likely spoil your machine learning model.

- **Train-Test Split:** Use **"train_test_split"** from Scikit-learn to split the dataset into 80% training and 20% testing subsets.

2.3 Model Training: Use a Random Forest Regressor to predict stock prices:

- **Initialization**: Import Scikit-learn and setup a random forest regressor with default parameters.

- **Hyperparameter Tuning**: To optimize parameters such as: use GridSearchCV
  - **n_estimators:** Number of trees in the forest
  - **max_depth:** Maximum depth of each tree.
  - **min_samples_split:** Minimum samples required to split a node.

- **Training:** Now, on the processed training dataset train the model.

- **Evaluation:** The model is evaluated using MSE and R^2 Score on the test dataset.

2.4 Model Serialization: Save the trained model using Joblib

```
import joblib
joblib.dump(model, 'random_forest_model.pkl')
```

For deployment you can upload the serialized model to your S3 bucket.

3 Cloud Deployment

3.1 AWS S3:

- Use AES 256 bits encryption for data security and enabling bucket encryption.
- We set up S3 bucket to store the dataset, the model, and the final results.
- Use the AWS CLI to upload files

  "**aws s3 cp random_forest_model.pkl s3://<bucket-name>/"**

3.2 AWS EC2 for Training:

- Use EC2 instance (for example t2.medium) in the case of a scalable cloud environment and train the model.
- Install Python, required libraries on the instance.
- Put the dataset on the instance or access it directly on S3.
- Back on S3, use the same script that train the model locally by loading serialized model back again to train the model.

3.3 AWS Lambda for Inference:

- **Create a Lambda Function:**
  - Run with python3.9.
  - We write a function which will take a load the model from S3 and process incoming requests.
  - Example code snippet

```
import boto3
import joblib
import json

def lambda_handler(event, context):
    s3 = boto3.client('s3')
    bucket = '<bucket-name>'
    key = 'random_forest_model.pkl'
    s3.download_file(bucket, key, '/tmp/random_forest_model.pkl')
    model = joblib.load('/tmp/random_forest_model.pkl')

    input_data = json.loads(event['body'])
    prediction = model.predict([input_data['features']])
    return {
        'statusCode': 200,
        'body': json.dumps({'prediction': prediction.tolist()})
    }
```

3.4 API Gateway

- We expose the Lambda function as a REST Api using API gateway.
- Make the API accept POST request with payload JSON having input features.
- Facilitate HTTPS for a secure communication.

3.5 Monitoring and Logging

- Use AWS CloudWatch to monitor:
    - Errors and lambdas invocations.
    - API Gateway traffic.
    - S3 bucket access.

- Alerts for anomalies or unauthorized access setup.

References

Abbas, Z., & Myeong, S. (2023). Enhancing industrial cybersecurity, focusing on formulating a practical strategy for making predictions through machine learning tools in cloud computing environments. Electronics, 12(12), 2650.

Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5–32.

Lopez Garcia, A., et al. (2020). A cloud-based framework for machine learning workloads and applications. IEEE Access, 8, 18681-18692.

Zhang, S., Li, Y., & Liu, H. (2021). Cloud security and privacy issues in machine learning. Journal of Cloud Computing, 10(1), 65-79.