

Configuration Manual

MSc Research Project
Research Project

Mohith Srivathsa Ravikumaraiah

Student ID: 20218940

School of Computing
National College of Ireland

Supervisor: Jitendra Kumar Sharma

Configuration Manual

Mohith Srivathsa Ravikumaraiah
20218940

1 Introduction

By following the set of rules provided by the NCI research study, this document is created which contains the different tools used while doing the project, methods followed, and also the applications. To run the different deep learning models and for evaluating the results Google Colab is used as the platform and all my files and datasets are stored in my google drive for enhanced security. After the comparison of results, then LSTM model is found to be more appropriate.

After configuring the autoscaling group, as per the minimum count mentioned while creating, the auto scaling group launches the EC2 instance. in my case it will be 1, so it will launch 1 EC2 instance. after that I have established the connection through SSH for that created instance. And all my files and dependencies are stored in the EBS volume. For data set forecasting the lstm.py Python file will increase the CPU load when performing total predictions/input sample. Finally, depending upon the CPU utilization the target tracking policy of the autoscaling group will modify the desired count and this will in turn adds or terminates resources(EC2 instances)

2 Deep learning models setup

Import all mentioned libraries with
pip3 install command
before running the program.

The figures are as follows:

Figure 1: Google colab Setup1

Figure 2: Run all for running all code blocks in order2

Figure 3: All the libraries imported3

Figure 4: Google drive mount code to google colab 4

3 AWS Configuration

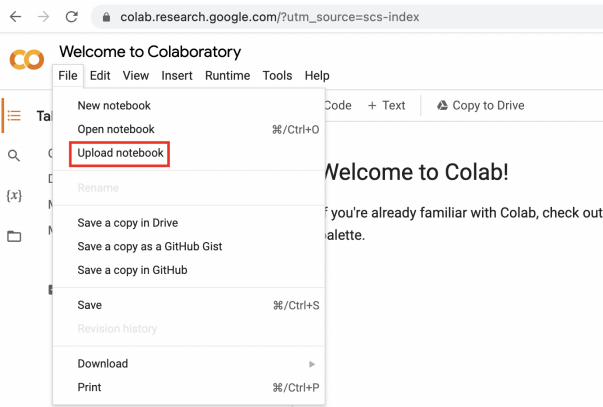
The Detailed screenshots of AWS Console is shown in this section. The figures are as follows:

Figure 5: Creating Launch Template – Dashboard5

Figure 6: Creating Launch Template – Summary6

Figure 7: Creating AutoScaling Group – Dashboard7

Figure 8: Creating AutoScaling Group – Summary8



What is Colab?

Figure 1: Google colab Setup

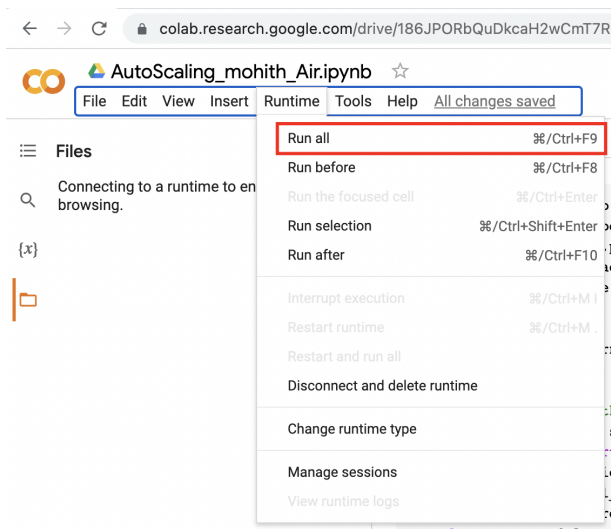


Figure 2: Run all for running all code blocks in order

```

import numpy as np #for arrays
import pandas as pd #for data analysis and machine learning tasks
import matplotlib.pyplot as plt #for visualization like plots, graphs and charts
from matplotlib.backends.backend_pdf import PdfPages
matplotlib inline

import warnings
warnings.filterwarnings("ignore") # for ignoring warnings

import math

import pickle #Python is primarily used in serialising and deserialising a Python object structure
import seaborn as sns #seaborn is a Python data visualization library based on matplotlib
from seaborn import heatmap #step 1 - Import the required Python packages, Load the dataset, Create a Python Numpy array, Create a Pivot in Python, Create a
from sklearn.metrics import f1_score, mean_absolute_error, mean_squared_error #The sklearn.metrics module implements several loss, score, and utility func
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler,MinMaxScaler
from statsmodels.graphics.tsaplots import plot_acf
from statsmodels.graphics.tsaplots import plot_pacf

from tensorflow.keras.models import Sequential #Sequential API is used to create models layer-by-layer
import tensorflow as tf #tensorflow is an open-source end-to-end platform - a library for multiple machine learning tasks
from tensorflow.keras.layers import Dense, LSTM, Input, Dropout, Bidirectional, Layer #keras is a high-level neural network library that runs on top of Tensor
from tensorflow.keras.models import Model, fit_generator
import tensorflow.keras.backend as K # import 'k'
from tensorflow.keras.layers import Attention # import attention layer

```

Figure 3: All the libraries imported

```

[2] from google.colab import drive
drive.mount('/content/drive')

Mounted at /content/drive

```

Figure 4: Google drive mount code to google colab

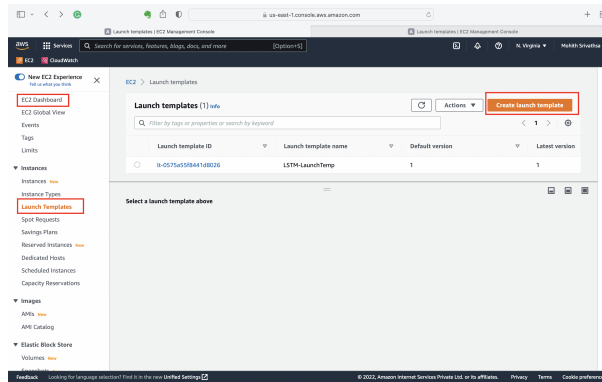


Figure 5: Creating Launch Template – Dashboard

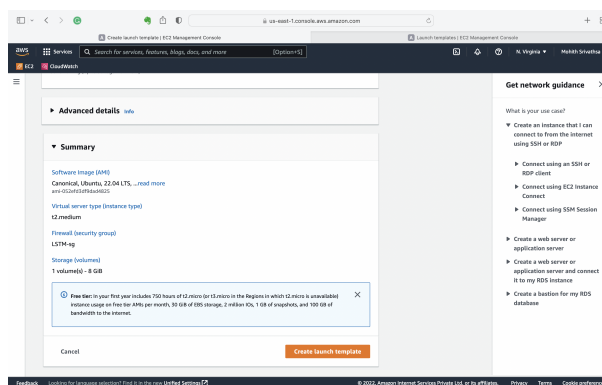


Figure 6: Creating Launch Template – Summary

Figure 9: Creating Dynamic Scaling Policy⁹

Figure 10: Configuring Target tracking policy by setting cpu utilization threshold of 80%¹⁰

Figure 11: LSTM AutoScaling Group- Group Details¹¹

Figure 12: LSTM AutoScaling Group- Launch Templates¹²

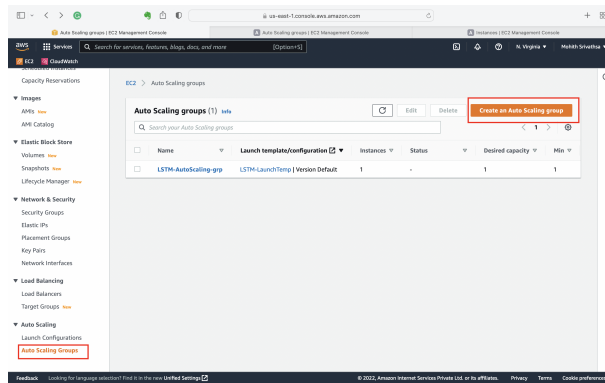


Figure 7: Creating AutoScaling Group – Dashboard

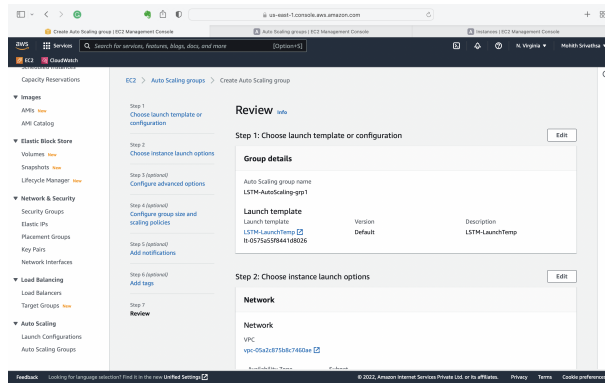


Figure 8: Creating AutoScaling Group – Summary

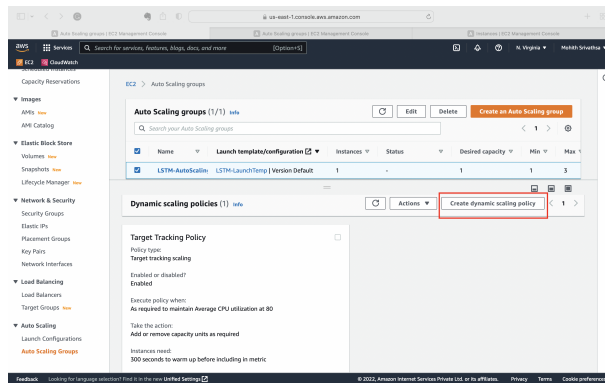


Figure 9: Creating Dynamic Scaling Policy

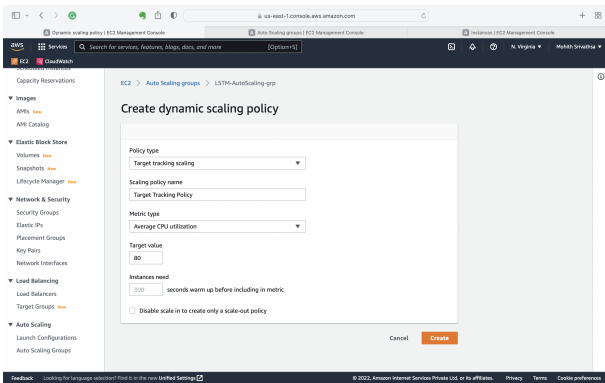


Figure 10: Configuring Target tracking policy by setting cpu utilization threshold of 80%

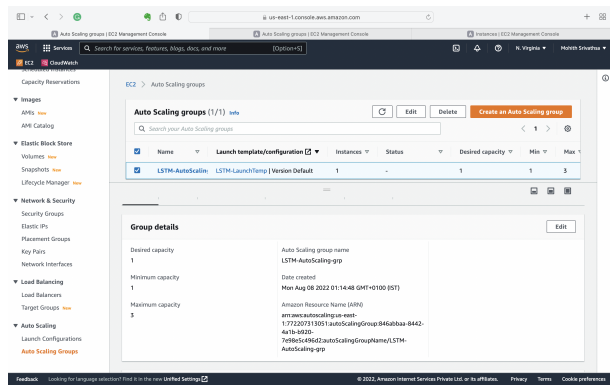


Figure 11: LSTM AutoScaling Group- Group Details

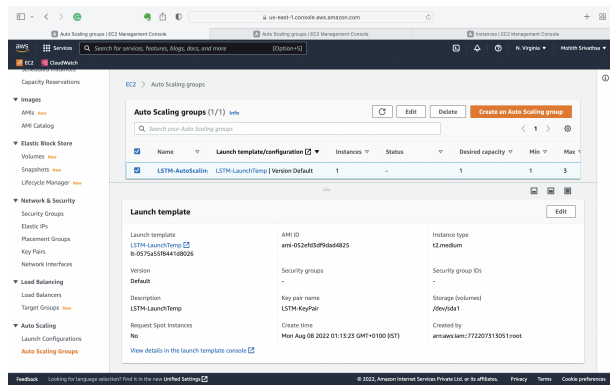


Figure 12: LSTM AutoScaling Group- Launch Templates