National College of Ireland

Configuration Manual

MSc Research Project
Cloud Computing

Bhuvan Prashanth
Student ID: 22163654

School of Computing
National College of Ireland

Supervisor: Dr. Rashid Mijumbi

National College of Ireland
Project Submission Sheet
School of Computing

National College of Ireland

Student Name:	Bhuvan Prashanth
Student ID:	22163654
Programme:	Cloud Computing
Year:	2023
Module:	MSc Research Project
Supervisor:	Dr. Rashid Mijumbi
Submission Due Date:	14/12/2023
Project Title:	Efficient Real-Time Data Deduplication Techniques for Improving Data Quality in Urban Taxi Trip Streams
Word Count:	425
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:	Bhuvan Prashanth
Date:	14th December 2023

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

Bhuvan Prashanth
22163654

1 Introduction

Outlined here are the steps for configuring the Notebook file on AWS SageMaker.

1.1 Requirement

Before commencing the setup process, verify that the following requirements have been fulfilled:

- Possession of data in CSV format.
- An active AWS account.
- Access to SageMaker with Compute-Optimized instances.
- Proficiency in Boto 3 and Python.
- Install Pycharm , NymPy, Dask, Matlib libraries
- Optionally, contemplate the use of VScode for future tasks.

1.2 Implementation

1. Log in to your AWS Account and Go to AWS S3 buckets

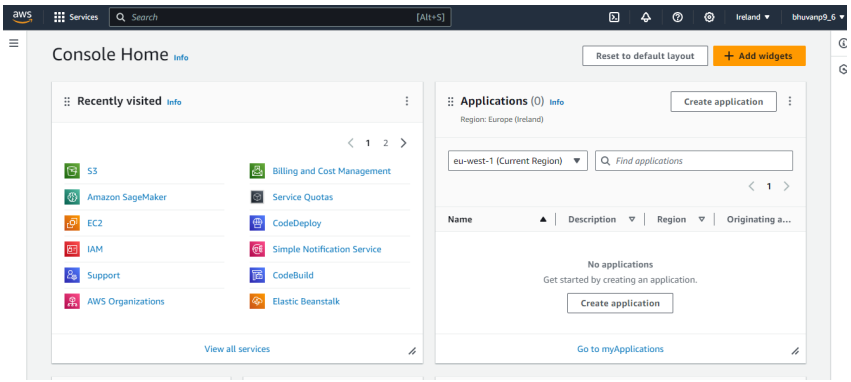

Figure 1: AWS Console Page

2. Next step Create S3 Bucket for storing csv files.

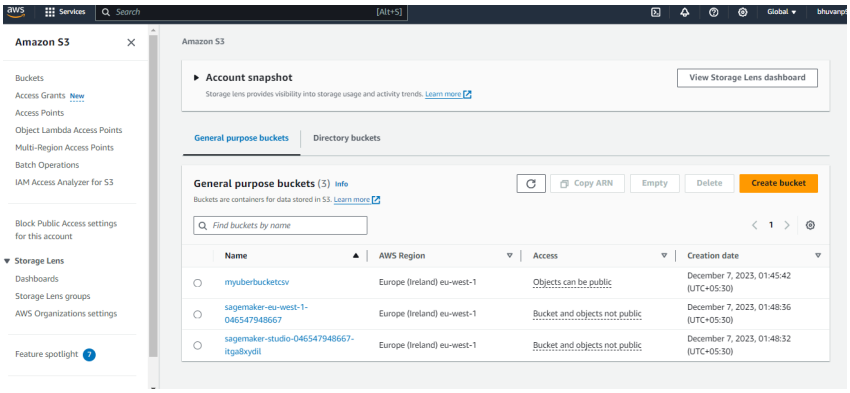

Figure 2: AWS S3 Bucket

3. Upload the CSV files to the s3 bucket.

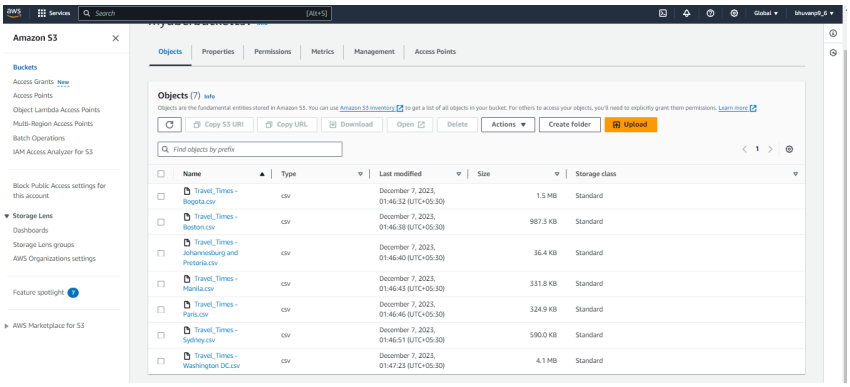

Figure 3: Adding CSV files to S3 Bucket

4. Open IAM in new tab and configure user and its roles with s3 Full AccessPermision

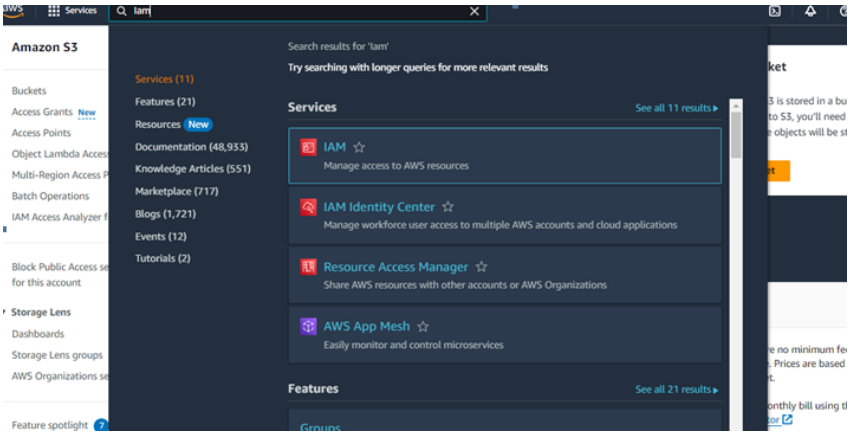

Figure 4: IAM Role

5. Setup Policies for User

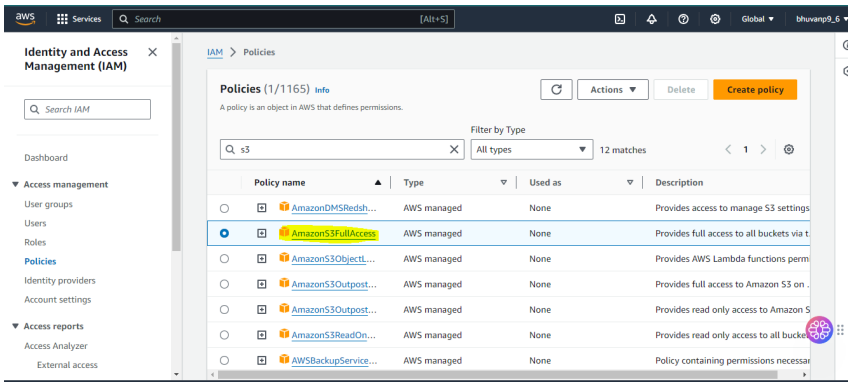

Figure 5: Set Policies to S3 bucket

6. Create notebook instance to upload the .ipynb and dataset files as shown

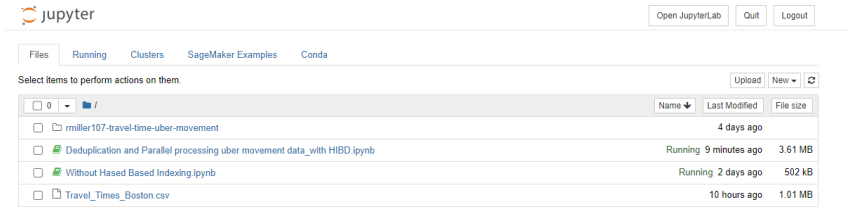

Figure 6: Jpyter Notebook

2 Python Libraries

1. Pip Install all the dependencies

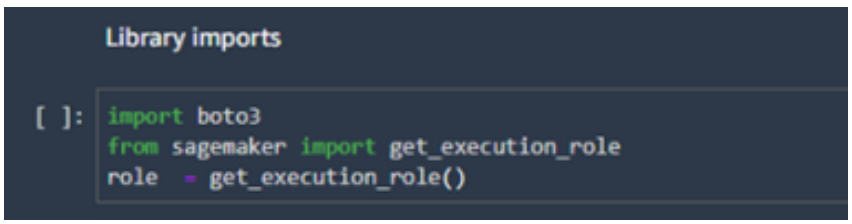

Figure 7: Importing Boto3 library

2. Let us import the necessary libraries.

```python
import pandas as pd
import matplotlib.pyplot as plt
from shapely.geometry import Polygon
from geopandas import GeoDataFrame
```

Figure 8: Import the python library

3. Install running the command "Pip install boto3 pandas shapely geopandas"
4. Run the HIBD notebook after configuring your bucket

```python
import pandas as pd
import boto3


# Set your AWS credentials (if not configured through the AWS CLI)
aws_access_key_id = 'AKIAQVVTZKR5WET6QXXW'
aws_secret_access_key = 'U/KNnekBzq3OJ3KHgydHJpdJeY8AMQGF5ZAkJBdi'

# Create an S3 client
s3 = boto3.client('s3', aws_access_key_id=aws_access_key_id, aws_secret_access_key=aws_secret_access_key)

# Specify the bucket and object key
bucket_name = 'myuberbucketcsv'
object_key = 'Travel_Times - Boston.csv'

# Specify the local file name to save
local_file_name = 'Travel_Times_Boston.csv'

# Download the file
s3.download_file(bucket_name, object_key, local_file_name)

# Read the CSV file into a Pandas DataFrame
uber_data = pd.read_csv(local_file_name)

# Now 'df' is your Pandas DataFrame
# You can use df.head() to check the first few rows of the DataFrame
print(uber_data.head())
```

Figure 9: Configuring S3 credentails in the HIBD code

5. Output related to Performance Metrics.

```
Column Names: Index(['Origin Movement ID', 'Origin Display Name', 'Origin Geometry',
       'Destination Movement ID', 'Destination Display Name',
       'Destination Geometry', 'Date Range', 'Mean Travel Time (Seconds)',
       'Range - Lower Bound Travel Time (Seconds)',
       'Range - Upper Bound Travel Time (Seconds)', 'start_date'],
      dtype='object')
Detected date column: Date Range
Deduplication Accuracy: 100.00%
Processing Time: 0.07 seconds
Resource Consumption: 536.95 MB
```

Figure 10: Output for HIBD Performace Metrics

6. Repeat the same procedure for the second notebook "Dask Parallel Process".

```
In [7]: import os
import pandas as pd
import dask.dataframe as dd
import matplotlib.pyplot as plt
import seaborn as sns

# Specify the folder paths
data_folder_path = r'./rmiller107-travel-time-uber-movement'
original_folder_path = os.path.join(data_folder_path, 'original')
data_folder_path = os.path.join(data_folder_path, 'data')

# Function to read all CSV files in a folder
def read_csv_files(folder_path, prefix=''):
    data_frames = {}
    for file_name in os.listdir(folder_path):
        if file_name.endswith('.csv'):
            full_path = os.path.join(folder_path, file_name)
            # Use the file name as the DataFrame key
            df_key = f"{prefix}{file_name.split('.')[0]}"
            data_frames[df_key] = pd.read_csv(full_path)
    return data_frames

# Read CSV files in the 'original' folder
original_data_frames = read_csv_files(original_folder_path, prefix='original_')

# Read CSV files in the 'data' folder
data_data_frames = read_csv_files(data_folder_path, prefix='data_')
```

Figure 11: Dask Parallel Process Code

7. Output for Mean travel time for Dask

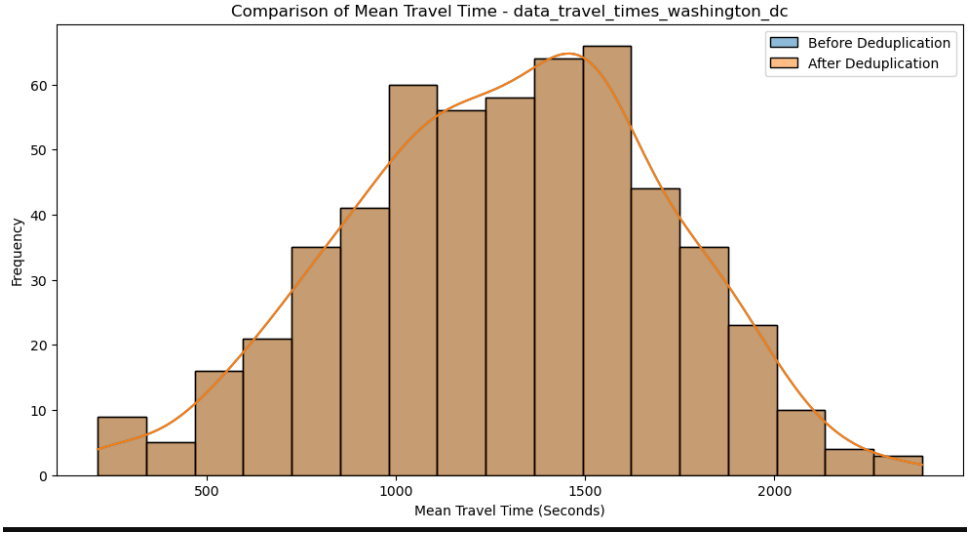

Figure 12: Dask Parallel Process Code

8. Performance Metrics Output Graphs for HIBD and Dask

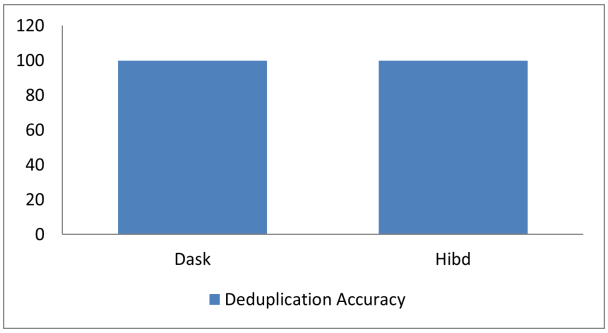

Figure 13: Deduplication Accuracy for HIBD & Dask

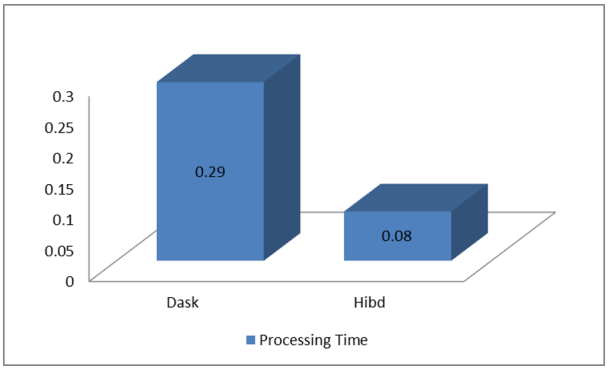

Figure 14: Performance Speed for HIBD & Dask

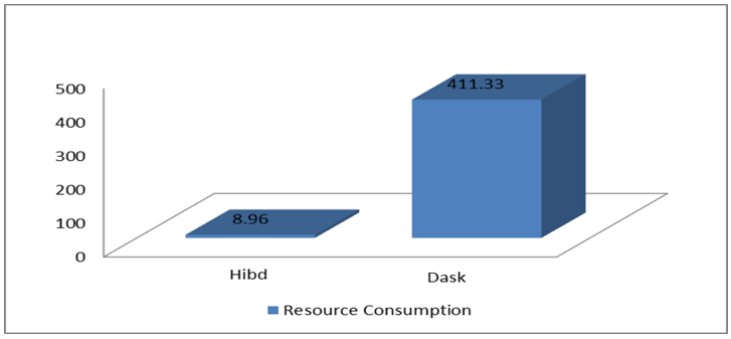

Figure 15: Resource consumption for HIBD & Dask

3 Dependency Installations

1. Install "pip install package-name" command
2. Install "Pip Install pandas"
3. Install "pip install matplotlib"
4. Install "Pip install geopandas"