National College of Ireland

Configuration Manual

MSc Research Project
Cloud Computing

Shreya Dhumal
Student ID: 21195773

School of Computing
National College of Ireland

Supervisor: Shreyas Setlur Arun

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Shreya Dhumal
Student ID:	21195773
Programme:	MSc Cloud Computing
Year:	2023
Module:	MSc Research Project
Supervisor:	Shreya Setlur Arun
Submission Due Date:	14/12/2023
Project Title:	Configuration Manual
Word Count:	1123
Page Count:	7

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:	Shreya Dhumal
Date:	13th 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):	

Improving Dynamic Cloud Workload Prediction and Resource Management with AdaptiveCloudEnsemble (ACE): A Concept Drift-Aware Approach

Shreya Dhumal
21195773

1 Introduction to System Architecture

The AdaptiveCloudEnsemble (ACE) project design is divided into three key steps that ensure a continuous flow and integration of data for predictive analytics in cloud environments. Figure 1 shows the system architecture of the project. The upward Pipeline starts with data collection via AWS Cloud9 and AWS Kinesis Datastreams, then temporary storage in AWS Glue. The processed data is then copied to an output Glue table in the downward pipeline and supplied to AWS S3 in parquet format via Kinesis Firehose. To know more details about setting up the pipeline follow the steps given in the AWS blog Amazon Web Services (2021). Finally, in the model creation process using AWS SageMaker Studio Notebook, the ACE model is built with ensemble algorithms such as ARF, SRP, and XGBoost, each of which is capable of drift detection and retraining.

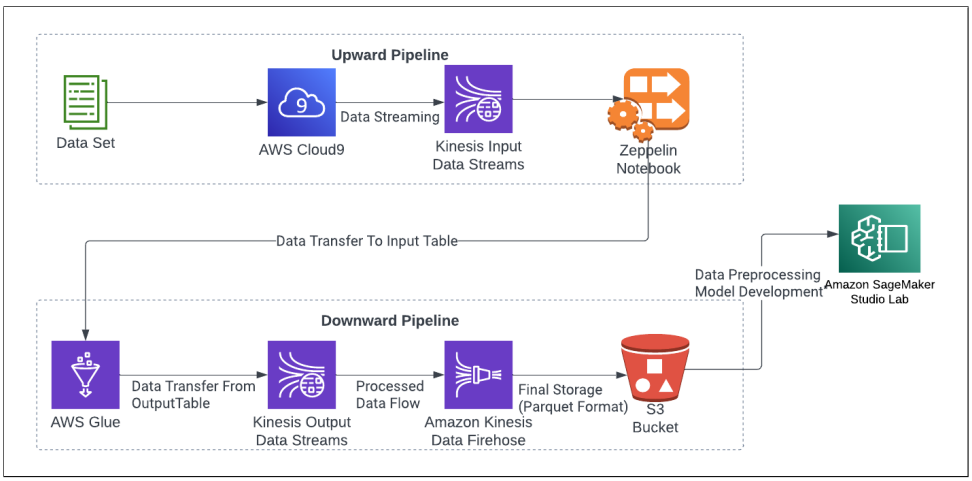

Figure 1: System Architecture

Prerequisites:

- AWS account with access to the required services
- Basic understanding of AWS services
- Understanding of Python programming and machine learning principles
- Understanding of ensemble learning and concept drift

2 Upward Pipeline

The upward pipeline describes the first step in the ACE project's system architecture where data is ingested and first stored:

1. Setting Up AWS Cloud9: Create an AWS Cloud9 environment that will act as an integrated development environment (IDE) for creating, running, and debugging data-streaming programs. Download the 'IoT_2020_b_0.01_fs' dataset from the Github Repository `https://github.com/Shre02/FinalProject` and upload it to Cloud9. Run the Python code with the command python 3 iotproducer.py to start sending the records from the dataset to Kinesis datastreams.

2. Configuring AWS Kinesis Datastreams: In the AWS Management Console, create a Kinesis Datastream. This stream will receive data from the IoT2020 dataset in real-time. In the datastream creation step give the name as 'iot-input-stream' and select the On-Demand option for Data stream capacity and create the stream.

3. Setting up Apache Flink Zeppelin Notebook: Select the Managed Apache Flink application from the Kinesis services menu in the management console. Choose the 'Create Studio Notebook' option on the application console, and then under the Creation Method, choose the Quick Create with sample option to create the notebook using the default IAM role.

To grant necessary access to services such as Kinesis streams, the Glue database, and S3, after creating the Notebook, open the IAM role console by clicking on the name of the role in the Studio Notebook Details section. Attach suitable policies to the role to grant the necessary permissions.

The next step is to specify the IAM policies for the notebook's source and destination, as well as the name of the Glue database that will be attached to the Notebook. Select the Edit IAM Permissions option under the Studio Notebook Details section choose the 'Create' option under the AWS Glue database section and create the Glue database with the desired name. Once the database is created go back to the previous console and select the newly created Glue database. Under the Included sources and destination in IAM policy options enter 'iot-input-stream' as the source and 'iot-output-stream' as the destination.

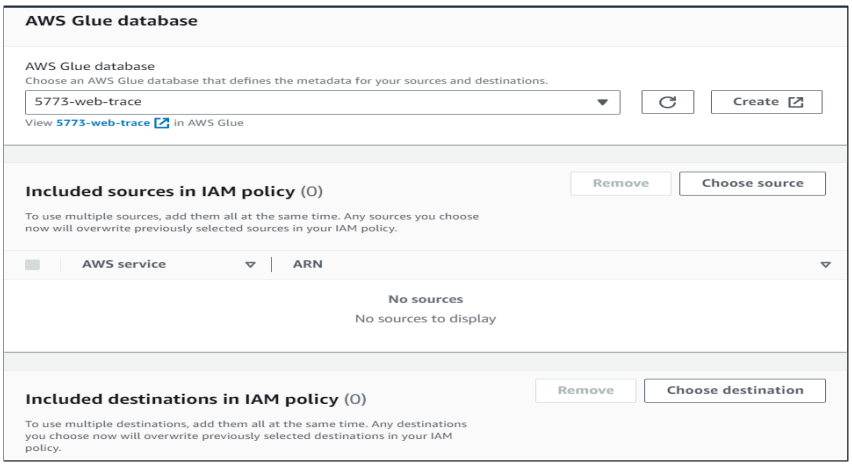

Figure 2: Studio IAM Permissions

4. Creation of Glue Tables: The next step is to link AWS Glue to the Kinesis Datastream. Kinesis data is transmitted to an AWS Glue table, where it is temporarily stored. Search for AWS Glue in the AWS Management Console. Open the Data Catalog option in the AWS Glue console and click Add Table under the newly created database. Add a new table in the database with Kinesis as the type of source in the Data Store section and add the 'iot-input-stream' as the source of the Glue table. In the next step add the table schema similar to the schema of the dataset and create the table.

With this, the Upward Pipeline will be set up and ready for the datastream ingestion from the dataset into the Kinesis input Stream and store the records in a temporary Glue Database.

3 Downward Pipeline

1. Transferring records to Output Glue Table: Create another Glue table by following the steps given above with a similar schema as the input Glue table. Now with the help of the Zeppelin Notebook write a query to transfer records from the input table to the output table.

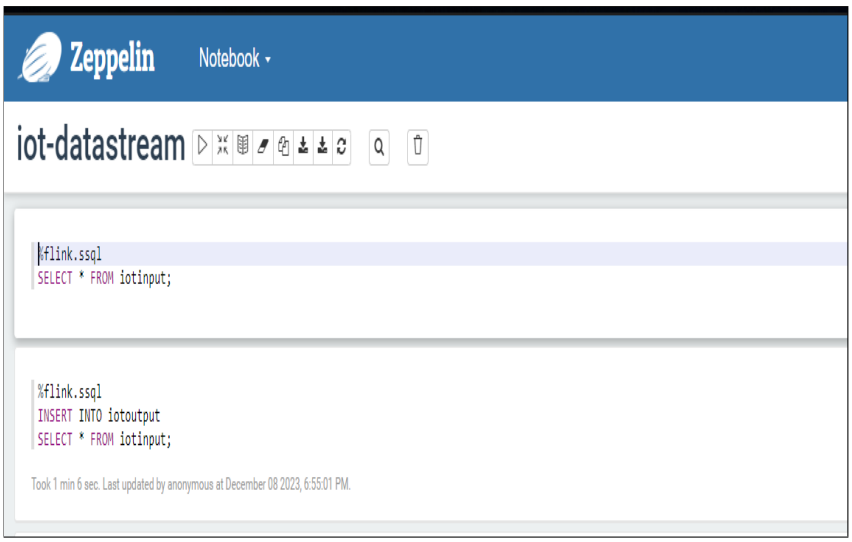

Figure 3: Zeppeli Notebook

2. Setting Up Kinesis Stream Data Firehose Delivery Stream: To set up the Kinesis Firehose Delivery Stream first the source and destination of the stream should be created. To do that first create output Kinesis Datastream by following the steps given in Section 2.

Create a S3 bucket by navigating to the S3 console from the AWS management console. Make sure to uncheck the Block all public access option in the bucket creation form as shown in Figure 4 as the records need to be accessed by AWS Sagemaker Studio Notebook in the next step. After the bucket is created under the properties section edit the Static website hosting and enable the static website hosting option to make the bucket public 5.

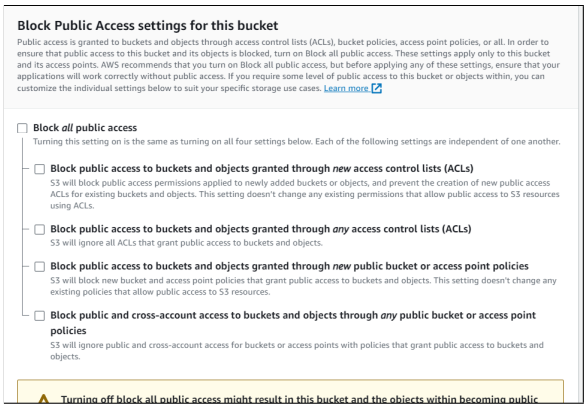

Figure 4: S3 Bucket Access

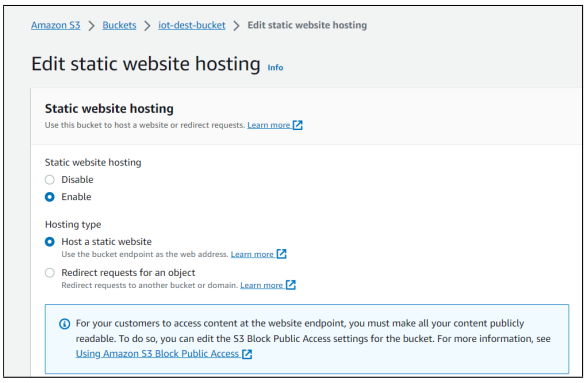

Figure 5: Static Website Hosting

Once the source and destination are set, proceed to the Kinesis Firehose Delivery stream creation by selecting Kinesis datastreams as the source and the S3 bucket as the destination. Under the Transform and Convert records option check the Enable record format conversion to convert the records into parquet format with the options selected as shown in Figure 6 and create the Delivery stream.

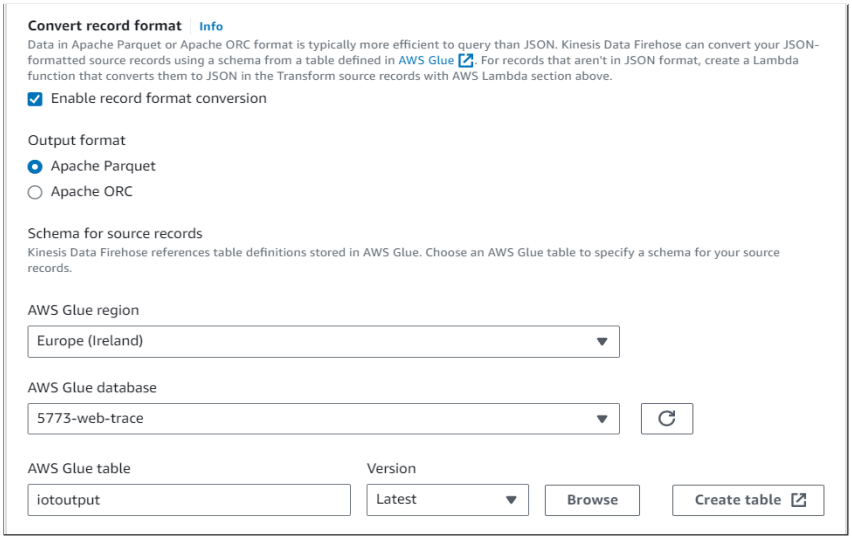

Figure 6: Kinesis Firehose Record Format

The downward pipeline will be successfully constructed with the data records from the input Glue table securely saved in the S3 bucket, assisted by the AWS Managed Apache Flink Zeppelin notebook and Firehose delivery streams.

4 Data Preprocessing and Model Development

4.1 Setting Up AWS Sagemaker Studio Notebook

Open the Amazon Sagemaker Studio Notebook Amazon Web Services (2023) from the AWS account and upload the Final(2).ipynb notebook from your device. Select the Data Science, Python 3 Kernel for the notebook. Next, you have to install the required libraries using Pip which is a package manager for Python packagesPython Packaging Authority

(2023). The required packages are given below:
1. Pandas
2. Dask
3. Lightgbm
4. River

Now execute each cell of the notebook to implement the model and get results.

4.2 Results

By following all the steps and executing all the cells of the given notebook, the below results are produced,

Figure 7: ACE Model Result

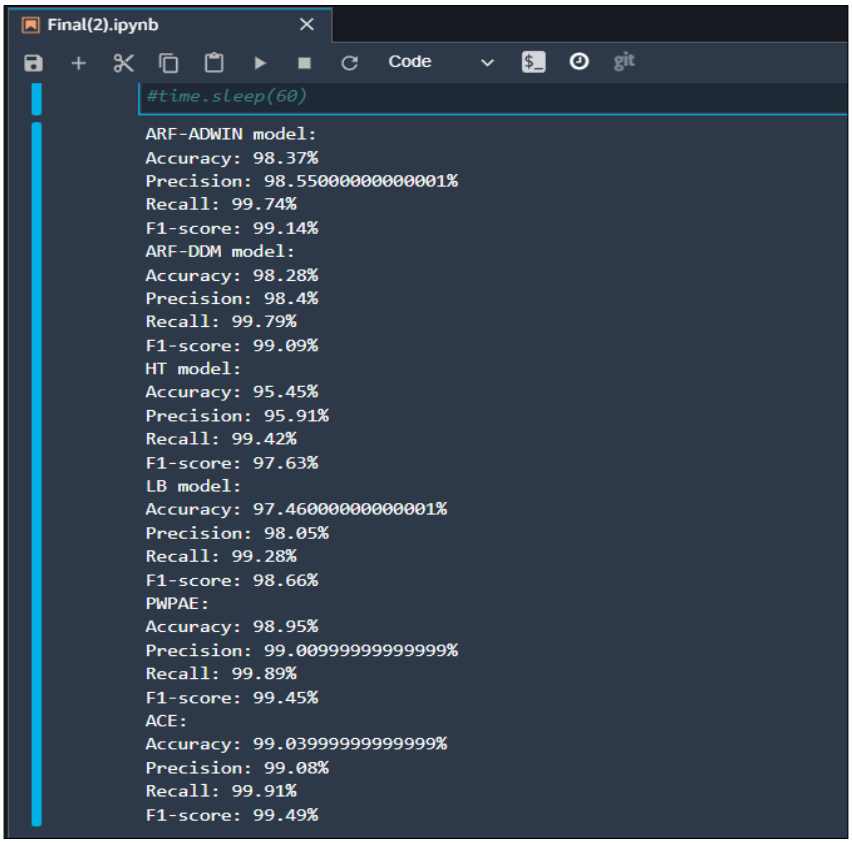

Figure 8: Comparison Results

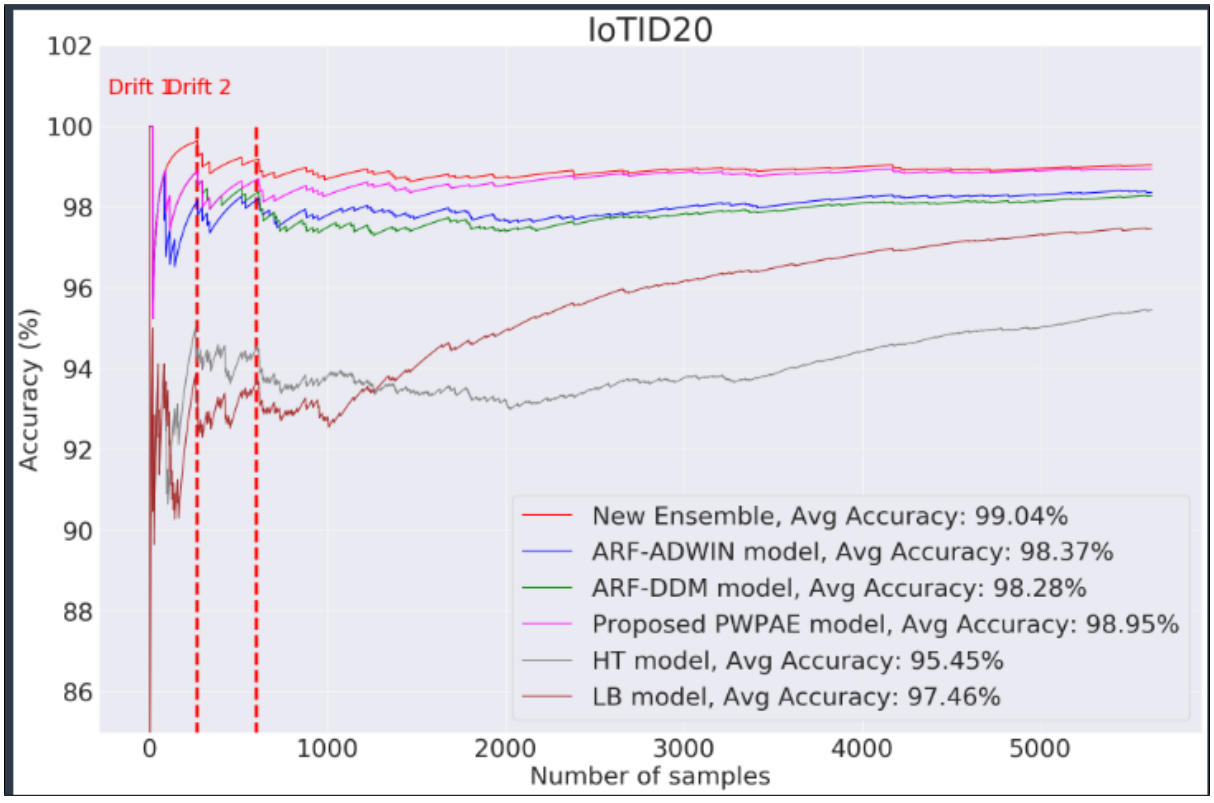

Figure 9: Graphical Representation

References

Amazon Web Services (2021). Introducing amazon kinesis data analytics studio: Quickly interact with streaming data using sql, python, or scala, https://aws.amazon.com/blogs/aws/introducing-amazon-kinesis-data-analytics-studio-quickly-interact-with-streaming-da Accessed: [11/12/2023].

Amazon Web Services (2023). Amazon sagemaker notebooks, https://docs.aws.amazon.com/sagemaker/latest/dg/notebooks.html. Accessed: [11/12/2023].

Python Packaging Authority (2023). pip: The python package installer, https://pypi.org/project/pip/. Accessed: [11/12/2023].