Unit 4 Deliverable 1
Jashkirat Virdi

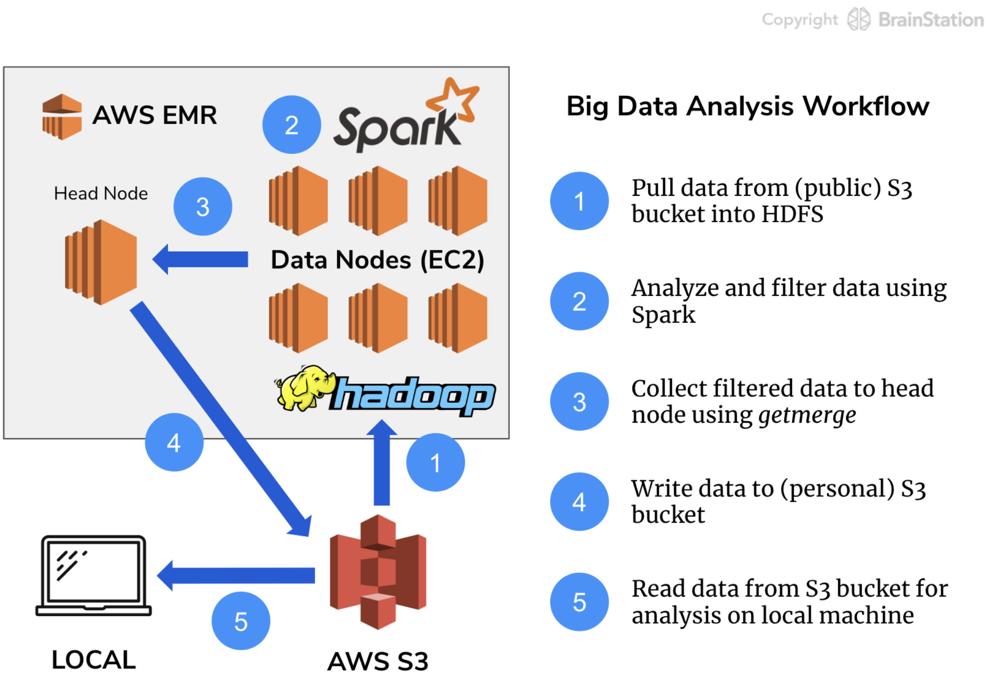
Big Data Wrangling With Google Books Ngrams

In this assignment, I will apply the skills I've learned in the Big Data Fundamentals unit to load, filter, and visualize a large real-world dataset in a cloud-based distributed computing environment using Hadoop, Spark, Hive, and the S3 filesystem. I will prepare a professional report to summarize my findings and include an appendix with screenshots of the steps completed to setup EMR Cluster

The Google Ngrams dataset was created by Google's research team by analyzing all of the content in Google Books. These digitized texts represent approximately 4% of all books ever printed, spanning a time period from the 1800s into the 2000s.

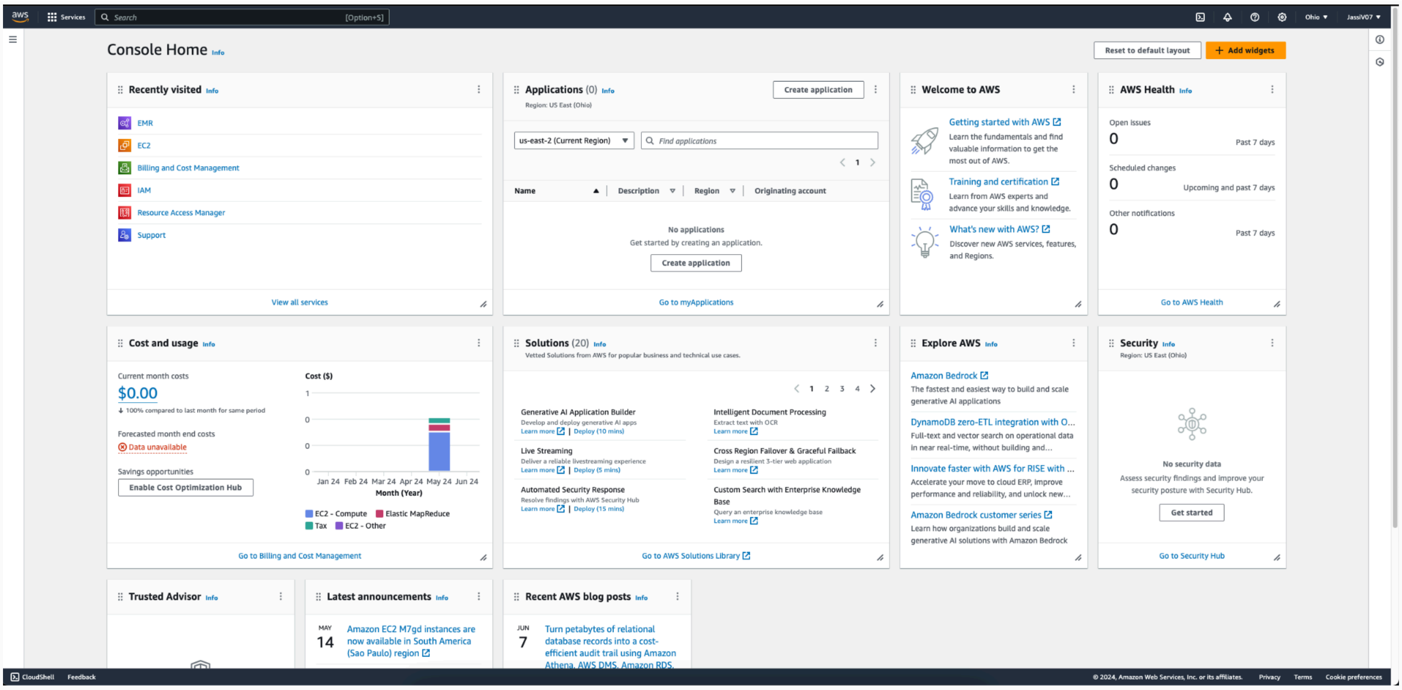
The dataset is hosted in a public S3 bucket as part of the Amazon S3 Open Data Registry. For this assignment, the data has been converted to CSV and hosted on a public S3 bucket, which can be accessed here: s3://brainstation-dsft/eng_1M_1gram.csv.

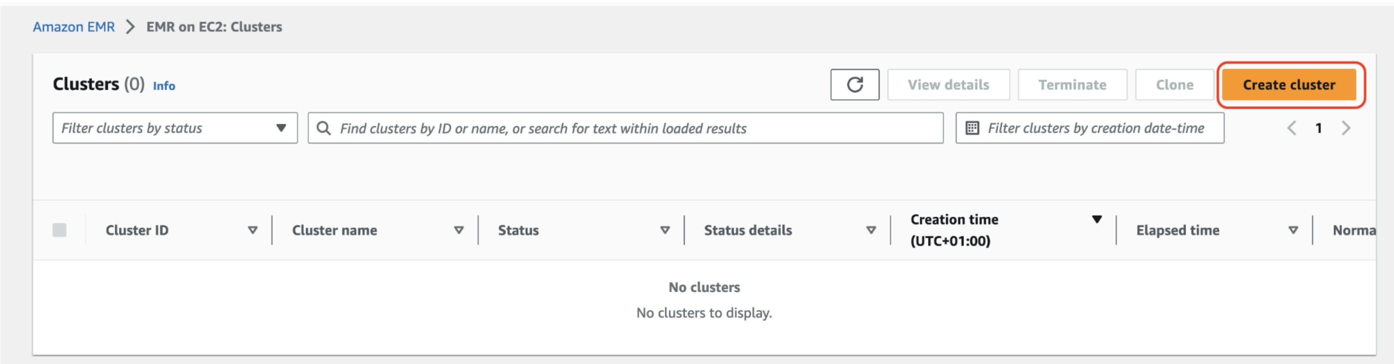
As part of this workflow, I will filter and reduce the data down to a manageable size, and then perform some analysis locally on my machine after extracting data from the cloud and processing it using Big Data tools. The workflow and steps in the process are illustrated below:



Unit 4 Deliverable 1
Jashkirat Virdi

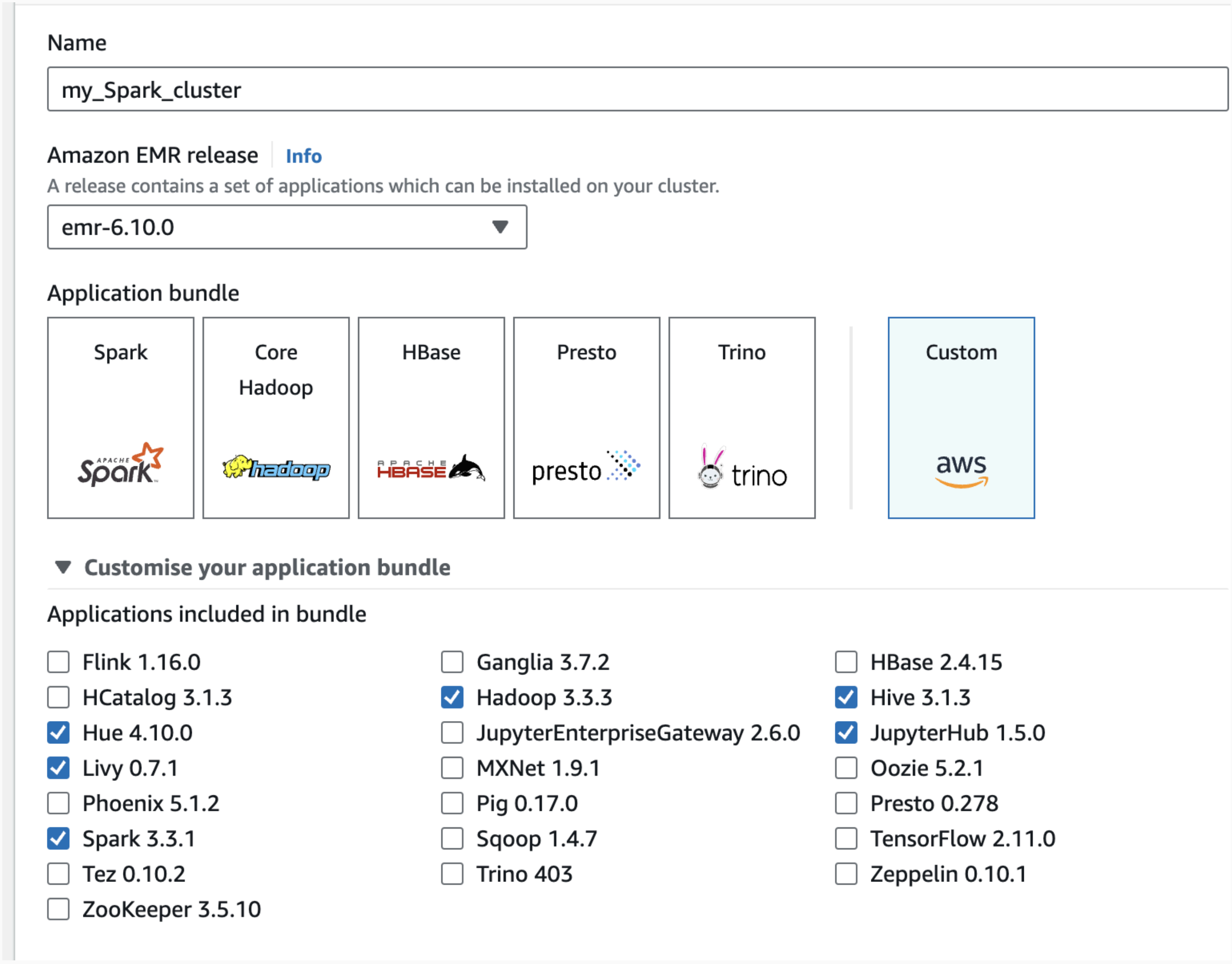
01. Spin up a new EMR cluster on AWS for using Spark and EMR notebooks
   a. Go to https://aws.amazon.com and sign in to your account.
   b. Navigate to the EMR panel in AWS and click 'Create Cluster':





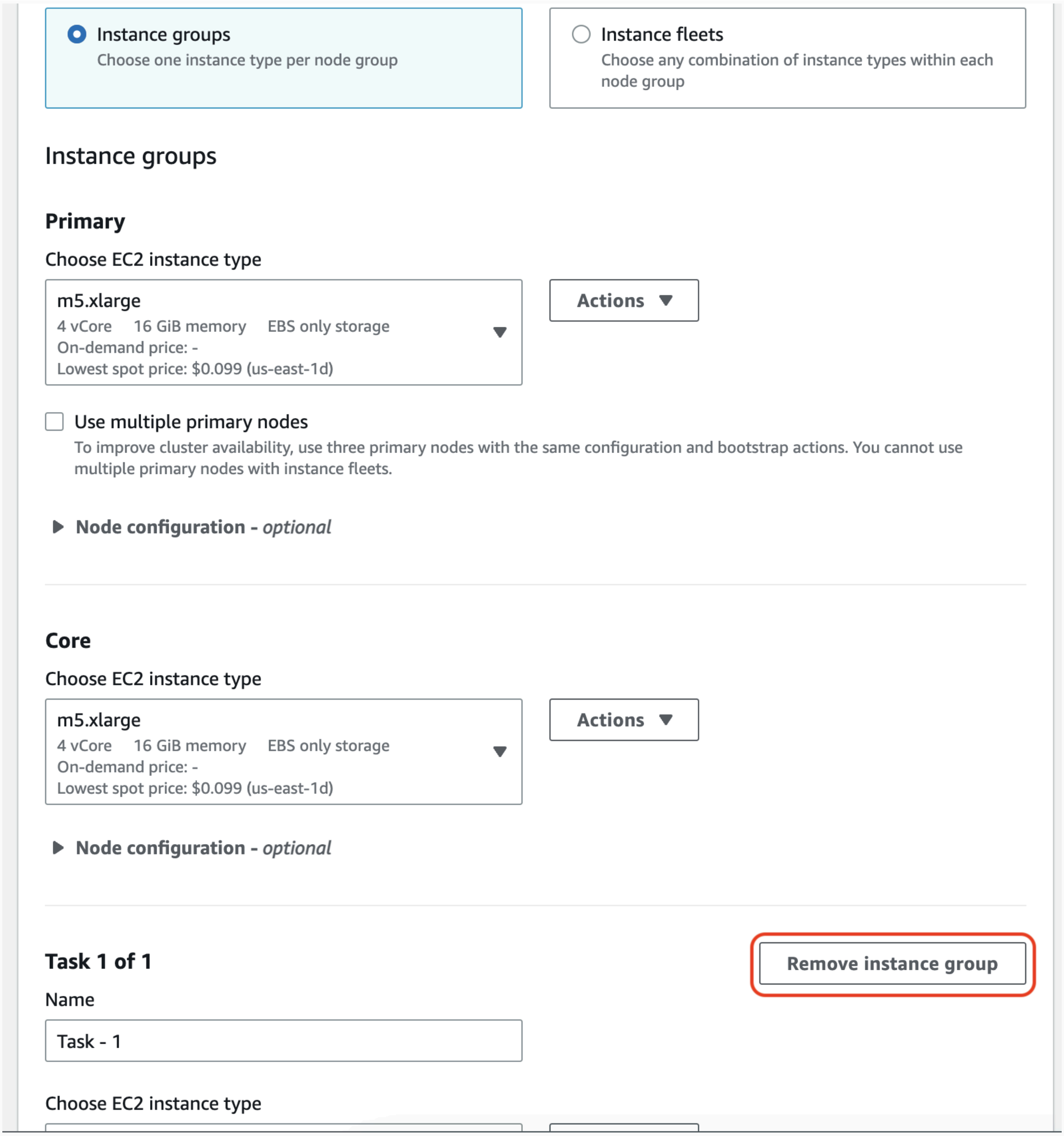
Unit 4 Deliverable 1
Jashkirat Virdi

c. Name the Cluster and Select Applications.
   i. Give your cluster a name.
   ii. In the 'Release' dropdown, select emr-6.10.0.
   iii. Select the Custom application bundle, and tick the boxes for Hadoop, Hue, JupyterHub, Livy, Hive, and Spark.



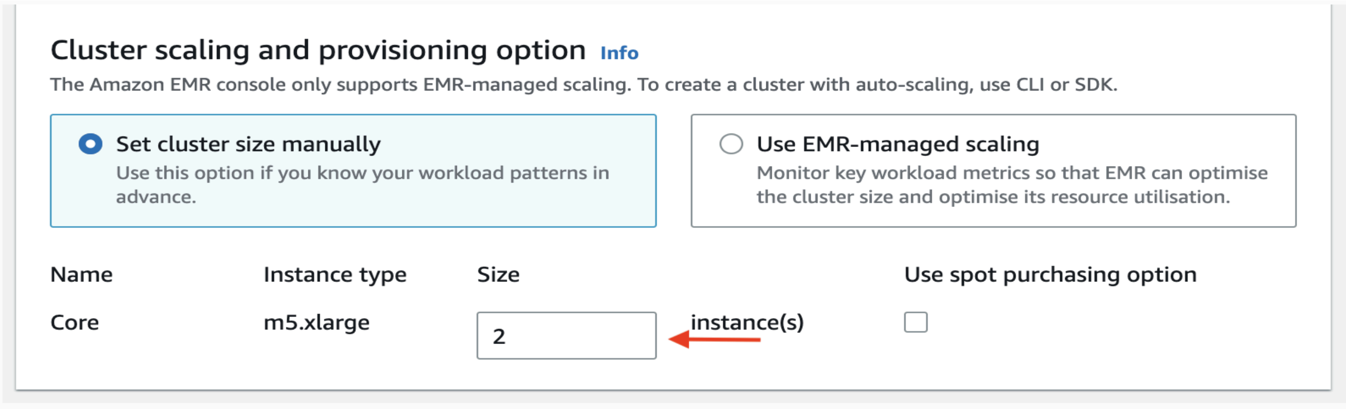
d. Instance Group

i. Remove the Task instance group



Unit 4 Deliverable 1
Jashkirat Virdi

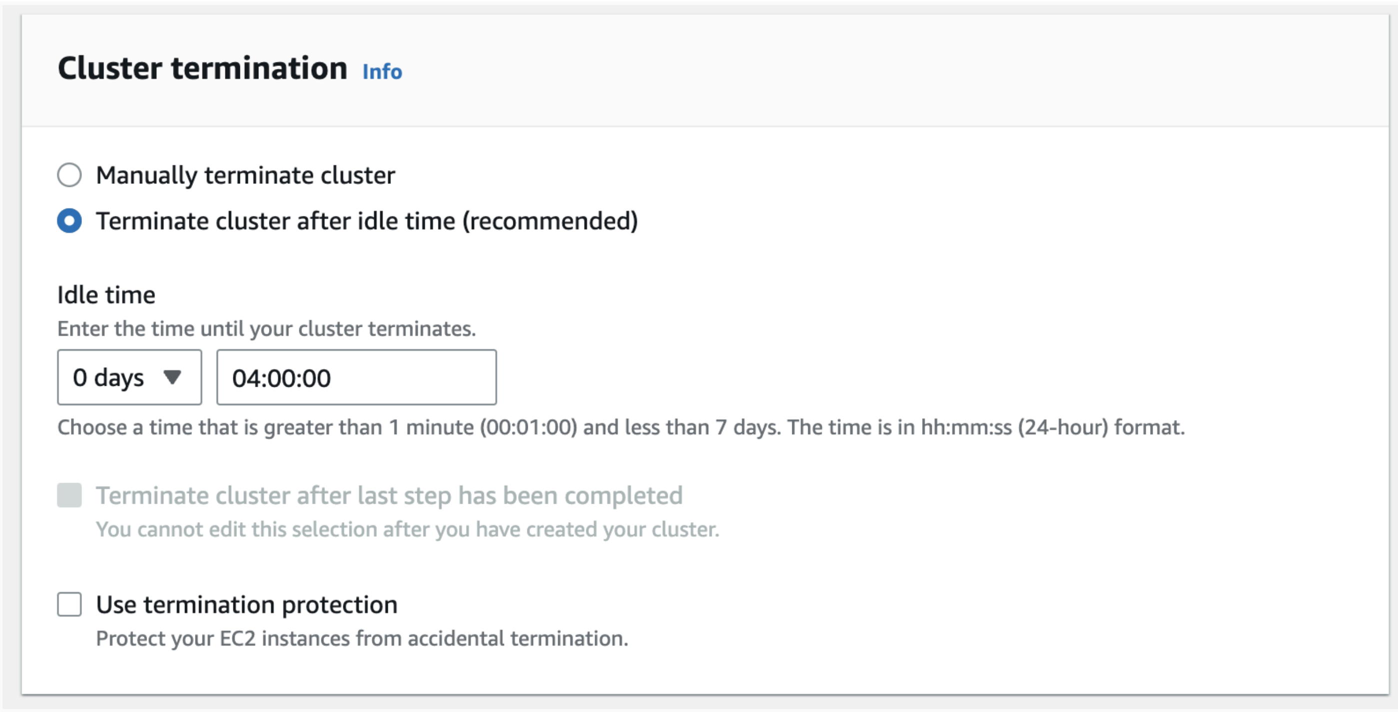
ii. Allocate 2 Nodes to the core instance group



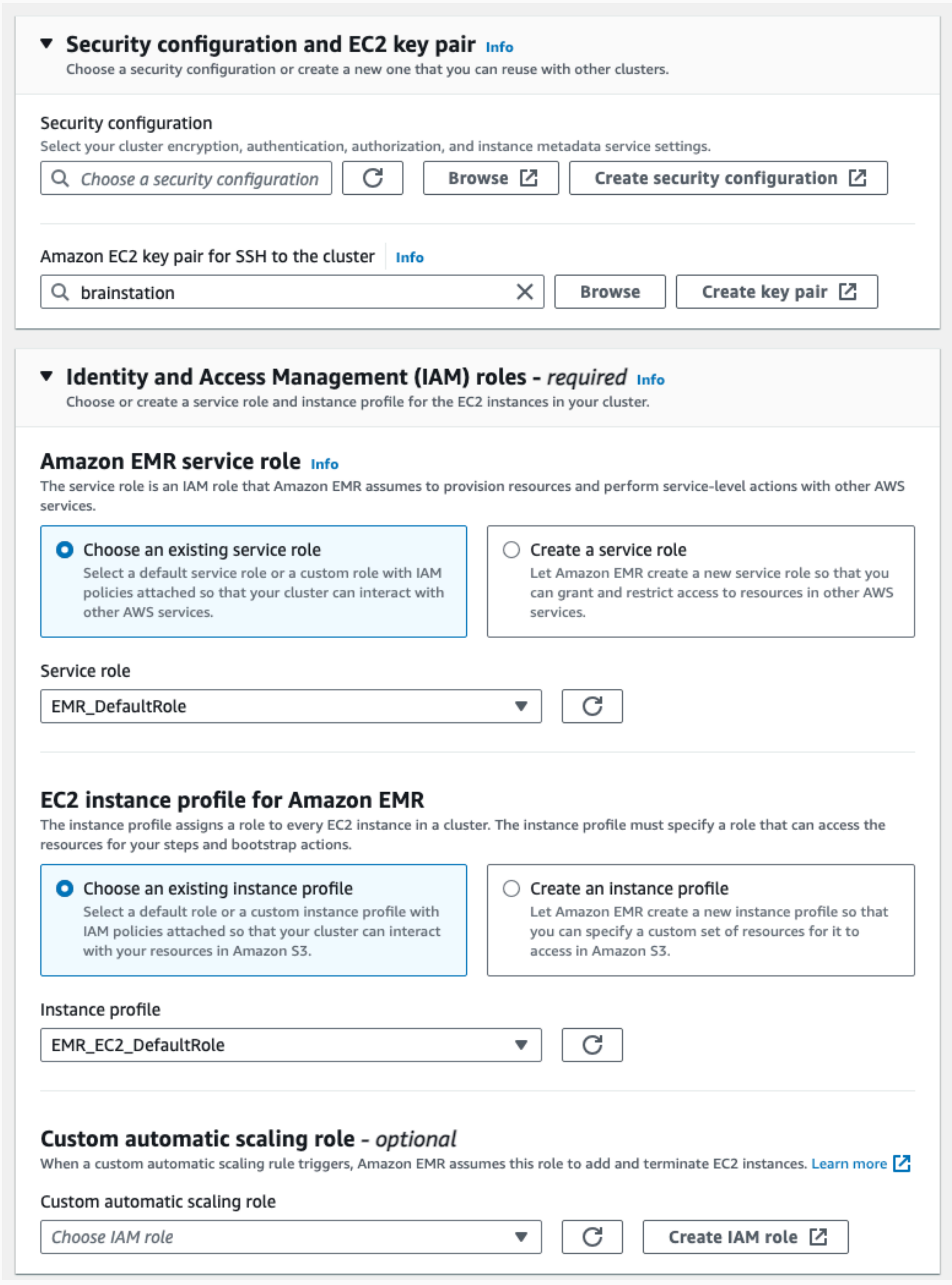
e. Cluster Termination

i. Set cluster termination to 4h idle time.

ii. Turn termination protection off.



f. Security and Access Management
   i. Select your key pair (keys are associated to geographies so if you switched recently, you might need to create a new key pair).
   ii. In Identity and Access Management, choose the EMR_DefaultRole and EMR_EC2_DefaultRole



Unit 4 Deliverable 1
Jashkirat Virdi

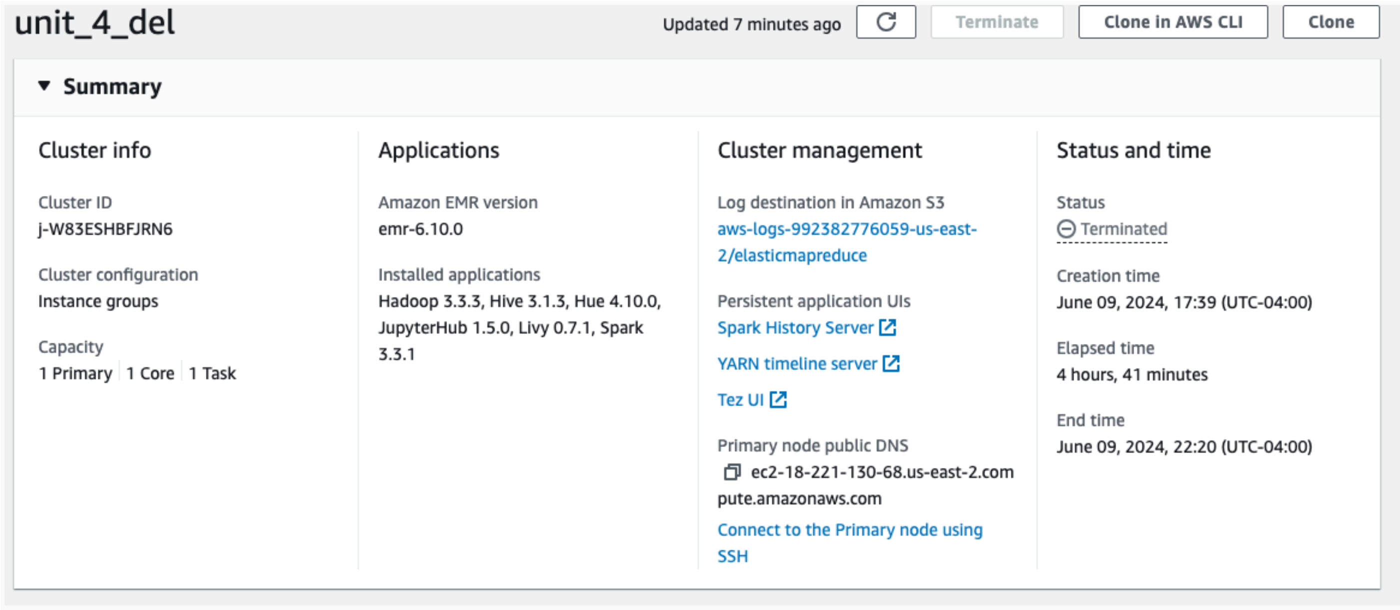
Note Once you create the cluster the process will start but it will take some time for it to get fully loaded and all of the things to get configured, the state will change from Starting to Waiting.

02.Connect to the head node of the cluster using SSH

a. Connect to the Primary Node of the cluster, and access JupyterHub in a browser window, edit the following bash command:

b. Replace 'xxxxxxxxx' with your 'Primary node public DNS' found on the overview page of your cluster.

```
ssh -i mykey.pem -L 9995:localhost:9443
hadoop@xxxxxxxxxxxxxx.compute.amazonaws.com
```



Unit 4 Deliverable 1
Jashkirat Virdi

c. Paste the command in the terminal

```
sh -i brainstation.pem -L 9995:localhost:9443 hadoop@ec2-18-221-130-68.us-east-2.compute.amazonaws.com
t login: Sun Jun  9 22:51:49 2024
,     #_
~\_  ####_        Amazon Linux 2
~  \_#####\
~     \###|       AL2 End of Life is 2025-06-30.
~       \#/ ___
~~       V~' '->
 ~~~         /    A newer version of Amazon Linux is available!
   ~~._.   _/
      _/ _/       Amazon Linux 2023, GA and supported until 2028-03-15.
    _/m/'           https://aws.amazon.com/linux/amazon-linux-2023/


EEEEEEEEEEEEEEEEEE MMMMMMMM           MMMMMMMM RRRRRRRRRRRRRRR
E::::::::::::::::E M:::::::M         M:::::::M R::::::::::::::R
EE:::::EEEEEEEEE:::E M::::::::M       M::::::::M R:::::RRRRRR:::::R
  E::::E       EEEEE M:::::::::M     M:::::::::M RR::::R      R::::R
  E::::E             M::::::M:::M   M:::M::::::M   R:::R      R::::R
  E:::::EEEEEEEEEE   M:::::M M:::M M:::M M:::::M   R:::RRRRRR:::::R
  E::::::::::::::E   M:::::M  M:::M:::M  M:::::M   R:::::::::::RR
  E:::::EEEEEEEEEE   M:::::M   M:::::M   M:::::M   R:::RRRRRR::::R
  E::::E             M:::::M    M:::M    M:::::M   R:::R      R::::R
  E::::E       EEEEE M:::::M     MMM     M:::::M   R:::R      R::::R
EE:::::EEEEEEEE::::E M:::::M             M:::::M   R:::R      R::::R
E::::::::::::::::E M:::::M             M:::::M RR::::R      R::::R
EEEEEEEEEEEEEEEEEE MMMMMMM             MMMMMMM RRRRRRR      RRRRRR

doop@ip-172-31-42-62 ~]$
```

03.Copy the data folder from the S3 bucket directly into a directory on the Hadoop File System (HDFS) named /user/hadoop/eng_1M_1gram.

a. Paste the following command in the terminal

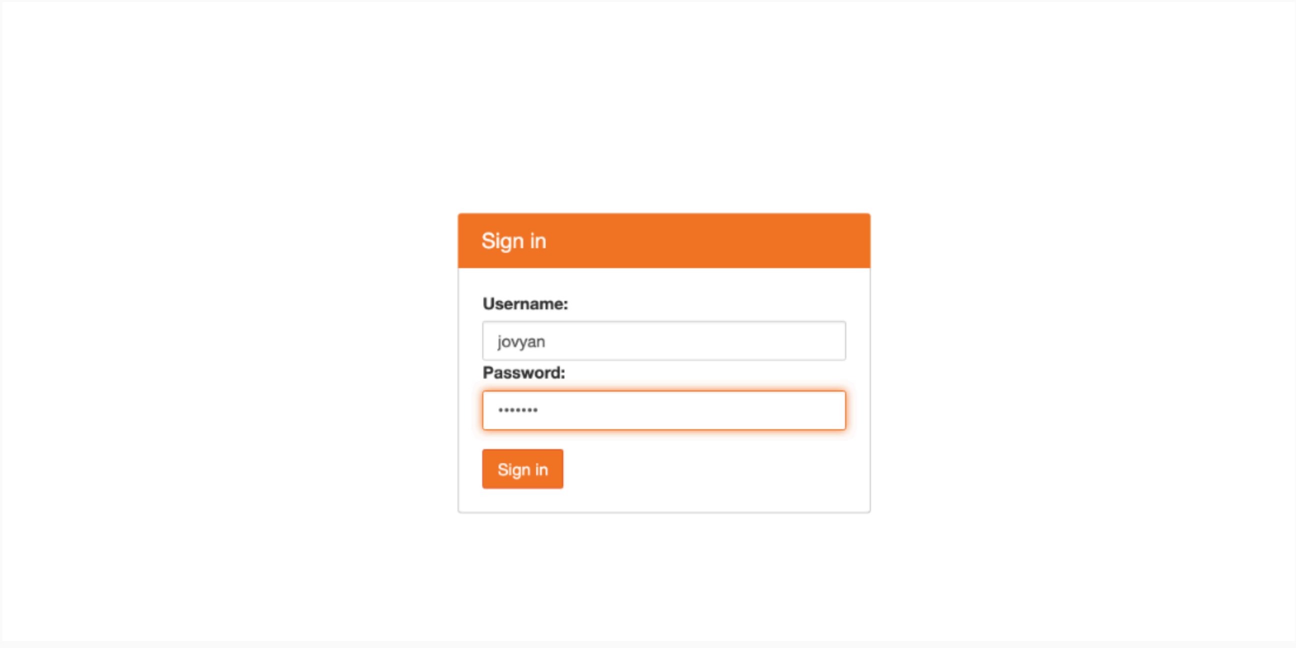
```
hadoop distcp s3://brainstation-dsft/eng_1M_1gram.csv /user/hadoop/eng_1M_1gram
```

b. Check /user/hadoop/eng_1M_1gram

```
[hadoop@ip-172-31-42-62 ~]$ hadoop fs -ls
Found 3 items
drwxr-xr-x   - hadoop hdfsadmingroup          0 2024-06-09 23:32 .sparkStaging
-rw-r--r--   1 hadoop hdfsadmingroup 5292105197 2024-06-09 22:24 eng_1M_1gram
```

Unit 4 Deliverable 1
Jashkirat Virdi

c. Access JupyterHub in a browser window at `https://localhost:9995`
d. Login using username `jovyan` and password `jupyter`



e. Open a New PySpark Notebook

Unit 4 Deliverable 1
Jashkirat Virdi

04. Check Step_4_Big_Data.ipynb Notebook for the Below.

a. Describe the dataset (examples include size, shape, schema) in pyspark
b. Create a new DataFrame from a query using Spark SQL, filtering to include only the rows where the token is "data" and describe the new dataset
c. Write the filtered data back to a directory in the HDFS from Spark using df.write.csv().

05.Collect the contents of the directory into a single file on the local drive of the head node using getmerge and move this file into a S3 bucket in your account.

a. Check hadoop directory using `hadoop fs -ls`

```
[hadoop@ip-172-31-42-62 ~]$ hadoop fs -ls /user/hadoop/
Found 3 items
drwxr-xr-x   - hadoop hdfsadmingroup          0 2024-06-09 23:32 /user/hadoop/.sparkStaging
-rw-r--r--   1 hadoop hdfsadmingroup 5292105197 2024-06-09 22:24 /user/hadoop/eng_1M_1gram
drwxr-xr-x   - livy   hdfsadmingroup          0 2024-06-09 23:25 /user/hadoop/filtered_data
```

b. Use `hadoop fs -getmerge` to merge the contents into a single file on the local drive of the head node.
c. Move the file into an S3 bucket in your account using `aws s3 cp`.

```
[hadoop@ip-172-31-42-62 ~]$ aws s3 cp filtered_data.csv  s3://aws-emr-studio-992382776059-us-east-2/171796
5722053/e-9BBQW3CFDNRYZC7JG8NQ3YHG9/
upload: ./filtered_data.csv to s3://aws-emr-studio-992382776059-us-east-2/1717965722053/e-9BBQW3CFDNRYZC7J
G8NQ3YHG9/filtered_data.csv
```

Unit 4 Deliverable 1
Jashkirat Virdi

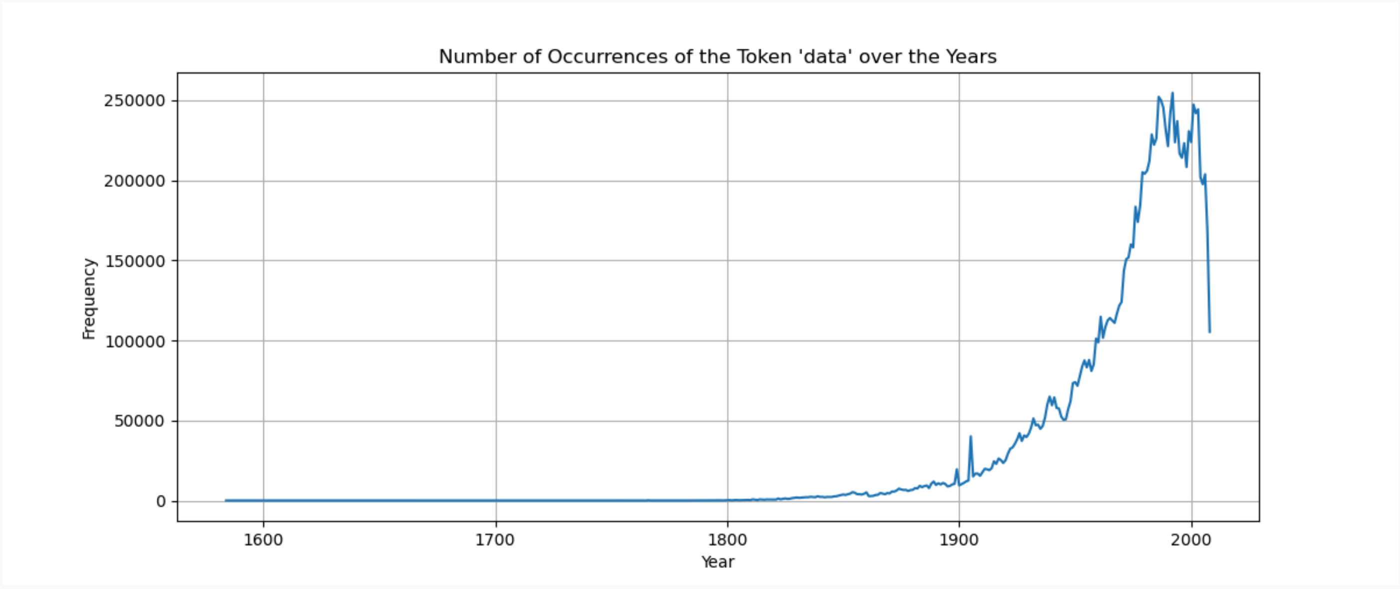
Check Step_6_Big_Data.ipynb Notebook for the Below.

06. On your local machine in python, read the CSV data from the S3 folder into a pandas DataFrame Note you must have first authenticated on your machine using aws configure on the command line to complete this step.

    a. Open Jupyter Notebook and import necessary libraries
    b. Read Data into dataframe

07. Plot the number of occurrences of the token (the frequency column) of data over the years using matplotlib.

    a. Data Wrangling
    b. Plot number of occurrences of the token **`data`** over the years



The heavy usage of the word "data" in books began in the mid-1800s and increased exponentially, reaching a peak between the 1990s and 2000s, likely due to the absence of more recent data.

Unit 4 Deliverable 1
Jashkirat Virdi

08.Compare Hadoop and Spark as distributed file systems.

a. What are the advantages/ differences between Hadoop and Spark? List two advantages for each.
    i. **Advantages of Hadoop:**

        1. **Storage Capability (HDFS):** Hadoop Distributed File System (HDFS) allows for the storage and processing of large data sets across distributed clusters. It ensures high fault tolerance and reliability, making it suitable for storing vast amounts of unstructured data.

        2. **Cost-Effective:** Hadoop is often more cost-effective for large-scale batch processing due to its open-source nature and compatibility with commodity hardware, reducing the need for expensive proprietary solutions

    ii. **Advantages of Spark:**

        1. **Speed:** Spark processes data much faster than Hadoop MapReduce due to its in-memory computation capabilities. It can cache data in memory, which significantly speeds up iterative algorithms and interactive data analysis.

        2. **Ease of Use:** Spark provides high-level APIs in Java, Scala, Python, and R, making it more accessible for developers. It also includes built-in libraries for streaming, machine learning, and graph processing, simplifying complex data processing tasks.

b. Explain how the HDFS stores the data.

HDFS (Hadoop Distributed File System) stores data by splitting large files into smaller blocks, typically 128MB or 256MB each. These blocks are distributed across multiple nodes in a cluster for parallel processing. Each block is replicated across multiple nodes (usually three) to ensure fault tolerance and reliability. The NameNode manages the metadata and directory structure, while DataNodes store the actual data blocks. This design allows HDFS to handle large-scale data storage with high availability and resilience.