

# Big Data Wrangling With Google Books Ngrams

## Table of Contents

<b>Introduction</b>	<b>1</b>
<b>Goal</b>	<b>2</b>
<b>Create a Cluster</b>	<b>2</b>
<b>[Side Note] Create EC2 Roles</b>	<b>7</b>
Identity and Access Management	7
<b>SSH into head node</b>	<b>9</b>
<b>Copy data to HDFS</b>	<b>11</b>
<b>Transfer file HDFS -&gt; Laptop</b>	<b>12</b>

## Introduction

The scope of this data processing and analysis report is to document the workflow involved in filtering and reducing big data of Google Ngrams down to a manageable size, and then doing some analysis locally on our machine after extracting data.

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, and span 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](#). This data has been converted to CSV and hosted on a public S3 bucket.

1. Spin up a new EMR cluster on AWS for using Spark and EMR notebooks.
2. Copy data from S3 to HDFS
3. Analyze and filter data using Spark.
4. Read filtered data on local machine
5. Plot the number of occurrences of the token (the frequency column) of data over the years.

## Goal

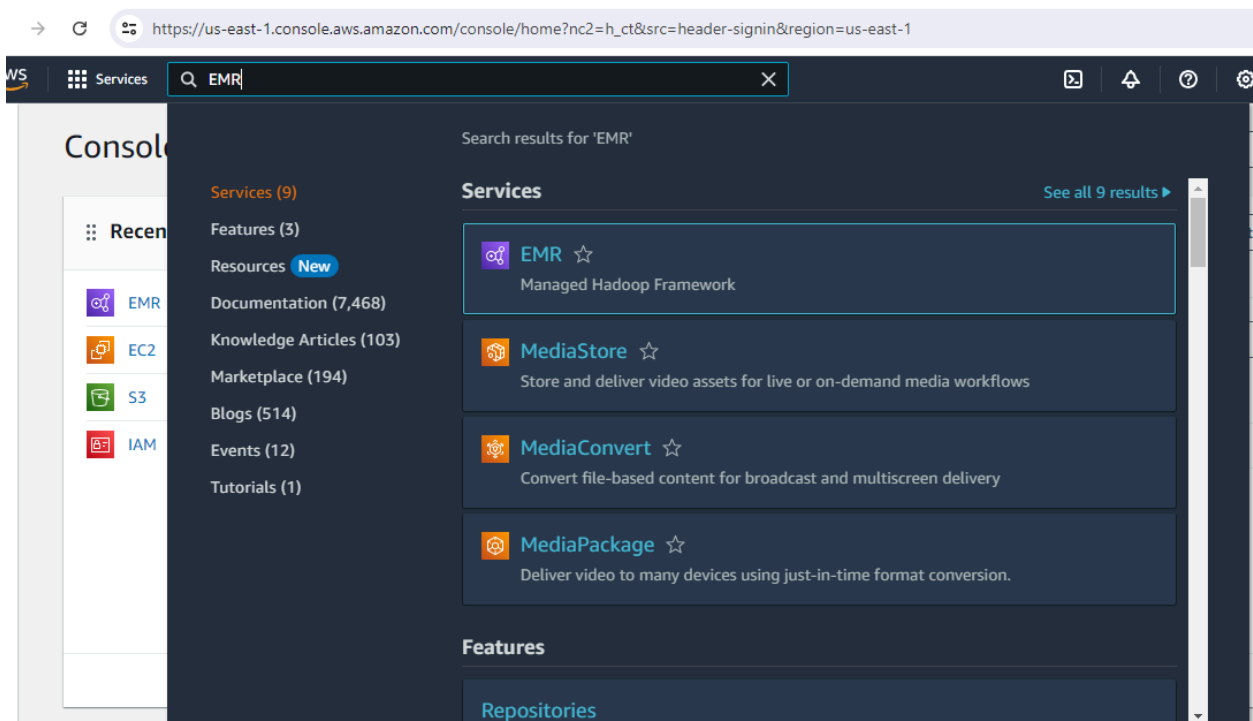
We will set up an EMR cluster, SSH into it, and copy the relevant file directly onto the HDFS.

We will also investigate the pros and cons of Hadoop vs Spark.

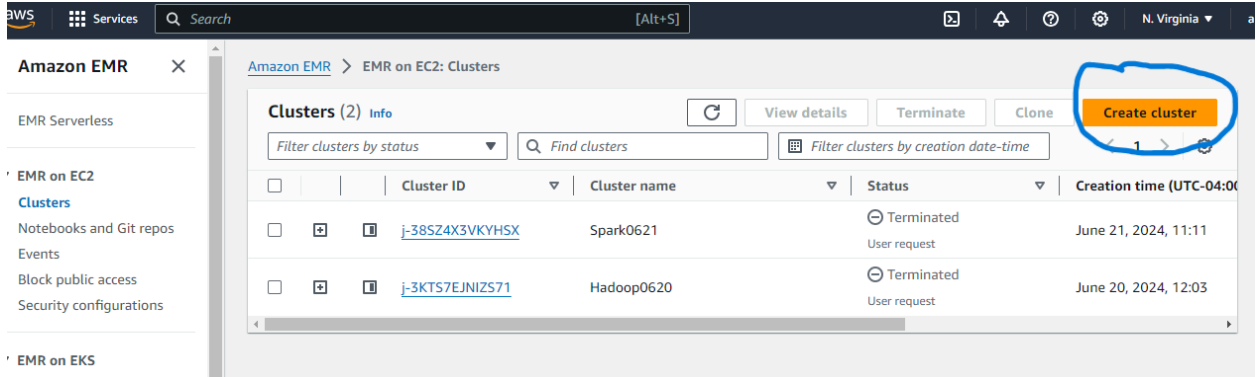
## Create a Cluster

**Create a new EMR cluster on AWS for using Spark and EMR notebooks.**

1. Create an AWS account if you don't already have one.
2. Sign in to the AWS console
3. On the Console Home, type 'EMR' in the Search bar and select EMR from the results



4. On the EMR home page, click on Create Cluster, or if you have an old cluster with the settings you want, you can select that cluster and click on Clone.



## 5. Cluster Settings

- Give your cluster a name.
- In the 'Release' dropdown, select **emr-6.10.0**.
- Select the **Custom** application bundle, and tick the boxes for Hadoop, Hue, JupyterHub, Livy, Hive, and Spark.


Ngrams0625


**Amazon EMR release** [Info](#)


A release contains a set of applications which can be installed on your cluster.


emr-6.10.0


**Application bundle**


Spark  
  


Core  
Hadoop  
  


HBase  
  


Presto  
  


Trino  
  


Custom  
  


☐ Flink 1.16.0  
☐ HCatalog 3.1.3  
☒ Hue 4.10.0  
☒ Livy 0.7.1  
☐ Phoenix 5.1.2  
☒ Spark 3.3.1  
☐ Tez 0.10.2  
☐ ZooKeeper 3.5.10

☐ Ganglia 3.7.2  
☒ Hadoop 3.3.3  
☐ JupyterEnterpriseGateway 2.6.0  
☐ MXNet 1.9.1  
☐ Pig 0.17.0  
☐ Sqoop 1.4.7  
☐ Trino 403

☐ HBase 2.4.15  
☒ Hive 3.1.3  
☒ JupyterHub 1.5.0  
☐ Oozie 5.2.1  
☐ Presto 0.278  
☐ TensorFlow 2.11.0  
☐ Zeppelin 0.10.1

**AWS Glue Data Catalog settings**

## Instance Groups

6. Remove the **task** instance group.
7. Allocate 2 nodes to the **core** instance group.

▼

Cluster configuration - *required*

Info

Choose a configuration method for the primary, core, and task node groups for your cluster.

☒ Uniform instance groups

Choose the same EC2 instance type and purchasing option (On-Demand or Spot) for all nodes in your node group. [Learn more](#)

☐ Flexible instance fleets

Choose from the widest variety of provisioning options for the EC2 instances in your cluster. Diversify instance types and purchasing options, and use an allocation strategy. [Learn more](#)

Uniform instance groups

Primary

Choose EC2 instance type

m5.xlarge

4 vCore 16 GiB memory EBS only storage

On-Demand price: \$0.192 per instance/hour

Lowest Spot price: \$0.078 (us-east-1f)

▼

Actions ▼

☐ Use high availability

Launch highly available, more resilient cluster with three primary nodes on On-Demand Instances. This configuration applies for the lifetime of your cluster. [Learn more](#)

► Node configuration - *optional*

## Cluster Termination

- Set cluster termination to 4h idle time.
- Turn termination protection off.

## ▼ Cluster termination and node replacement [Info](#)

Choose termination settings and protect your cluster from accidental shutdown.

### Termination option

- ☐ Manually terminate cluster
- ☐ Automatically terminate cluster after last step ends
- ☒ Automatically terminate cluster after idle time (Recommended)

### Idle time

Enter the time until your cluster terminates.

0 days ▼ 04:00:00

Choose a time that is greater than 1 minute (00:01:00) and less than 7 days. The time is in hh:mm:ss (24-hour) format.

### ☐ Use termination protection

Protects your cluster from accidental termination. If on, you must first turn off protection to terminate the cluster. We recommend turning on termination protection for your long running clusters.

### Unhealthy node replacement - *new* [Info](#)

- ☒ Turn on  
Amazon EMR gracefully stops processes on unhealthy nodes to minimize data loss and job interruptions. It quickly replaces unhealthy nodes with new EC2 instances to keep your jobs running smoothly.
- ☐ Turn off  
Amazon EMR adds unhealthy nodes to a denylist while keeping them in the cluster, allowing you continued access for troubleshooting.

## Security and Access Management

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

## ▼ Security configuration and EC2 key pair [Info](#)

Choose a security configuration or create a new one that you can reuse with other clusters.

### Security configuration

Select your cluster encryption, authentication, authorization, and instance metadata service settings.

### Amazon EC2 key pair for SSH to the cluster [Info](#)

## ▼ Identity and Access Management (IAM) roles - *required* [Info](#)

Choose or create a service role and instance profile for the EC2 instances in your cluster.

### Amazon EMR service role [Info](#)

The service role is an IAM role that Amazon EMR assumes to provision resources and perform service-level actions with other AWS services.

- ☒ **Choose an existing service role**  
Select a default service role or a custom role with IAM policies attached so that your cluster can interact with other AWS services.

- ☐ **Create a service role**  
Let Amazon EMR create a new service role so that you can grant and restrict access to resources in other AWS services.

Service role

EMR\_DefaultRole



### EC2 instance profile for Amazon EMR

The instance profile assigns a role to every EC2 instance in a cluster. The instance profile must specify a role that can access the resources for your steps and bootstrap actions.

- ☒ **Choose an existing instance profile**  
Select a default role or a custom instance profile with IAM policies attached so that your cluster can interact with your resources in Amazon S3.

- ☐ **Create an instance profile**  
Let Amazon EMR create a new instance profile so that you can specify a custom set of resources for it to access in Amazon S3.

Instance profile

EMR\_EC2\_DefaultRole



## Create Cluster

- Click the 'Create Cluster' button.

The cluster creation process will start. It may take some time for the EC2 instances to spin up and all of the cluster software to be configured.

Once the cluster has been created it will enter a 'Waiting' state.

## [Side Note] Create EC2 Roles

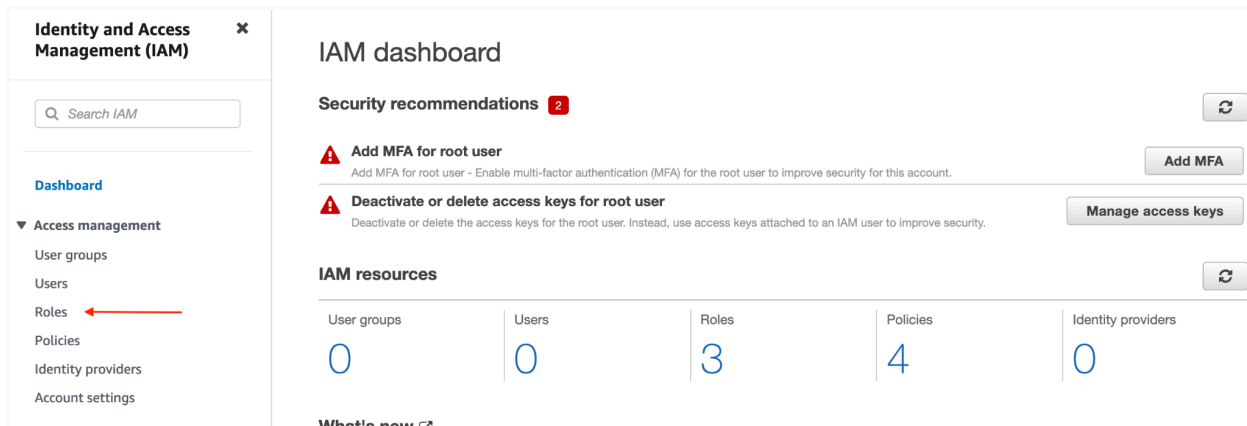
Creating the EC2 roles in case they are not found during the 'Identity and Access Management' block above.

### Identity and Access Management

These settings determine how our cluster is allowed to interact with other AWS services. The permissions come in pre-defined 'roles'. As this is our first time set-up, we will need double check (or create) the correct roles.

In the future, you can skip these steps and go straight to selecting a 'Service Role' and 'Instance Profile' from the drop-down menus.

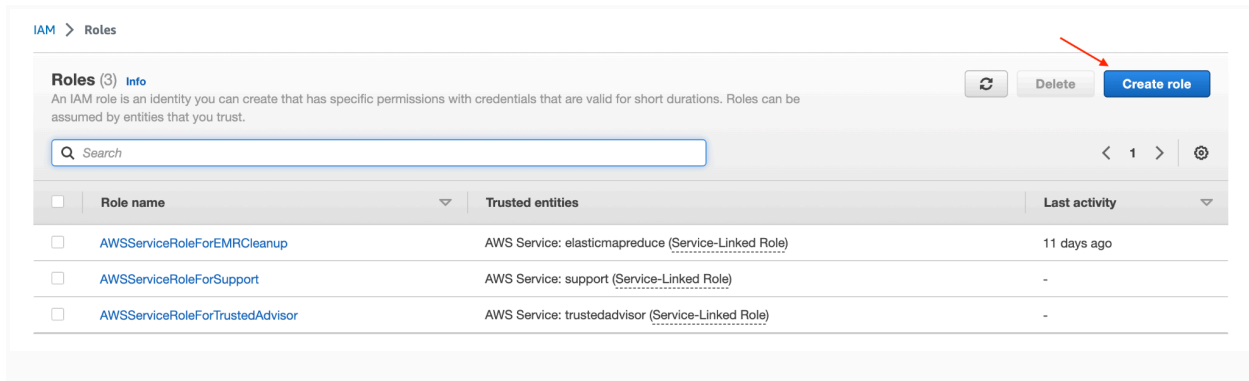
1. Leave the EMR cluster set-up window on hold for now.
2. In a new browser tab, open the AWS IAM page, by searching 'IAM' in the search bar.
3. Go to the 'Roles' tab through the left-side menu.



The screenshot shows the AWS IAM dashboard. On the left, the 'Identity and Access Management (IAM)' header is visible, along with a search bar and a navigation menu. The 'Roles' option in the 'Access management' section is highlighted with a red arrow. The main content area shows the 'IAM dashboard' with 'Security recommendations' (2) and 'IAM resources'. The 'IAM resources' section displays a table with the following data:

User groups	Users	Roles	Policies	Identity providers
0	0	3	4	0

4. If 'EMR\_DefaultRole' and 'EMR\_EC2\_DefaultRole' already show in the list of roles, you have the relevant roles. Otherwise, click 'Create Role'.



5. Select 'AWS service' as trusted entity type, and search and select 'EMR' as a use case, before clicking 'next'.

The screenshot shows the 'Select trusted entity' page in the AWS IAM console. Under 'Trusted entity type', the 'AWS service' option is selected. Under 'Use case', the 'EMR' option is selected. The 'Next' button is highlighted with a red arrow.

**Select trusted entity** [Info](#)

**Trusted entity type**

☒ **AWS service**  
Allow AWS services like EC2, Lambda, or others to perform actions in this account.

☐ **AWS account**  
Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.

☐ **Web identity**  
Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.

☐ **SAML 2.0 federation**  
Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.

☐ **Custom trust policy**  
Create a custom trust policy to enable others to perform actions in this account.

**Use case**  
Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

**Common use cases**

☐ **EC2**  
Allows EC2 instances to call AWS services on your behalf.

☐ **Lambda**  
Allows Lambda functions to call AWS services on your behalf.

Use cases for other AWS services:

☒ **EMR**  
Allows Elastic MapReduce to call AWS services such as EC2 on your behalf.

☐ **EMR Role for EC2**  
Allows EC2 instances in an Elastic MapReduce cluster to call AWS services such as S3 on your behalf.

☐ **EMR - Cleanup**  
Allows EMR to terminate instances and delete resources from EC2 on your behalf.

[Cancel](#) [Next](#)



6. The 'Permissions Policies' should be pre-populated with "AmazonElasticMapReduceRole". Click 'Next'.



## Add permissions [Info](#)

### Permissions policies (1) [Info](#)

The type of role that you selected requires the following policy.

Policy name <a href="#">↗</a>	Type	Attached entities
  <a href="#">AmazonElasticMapReduceRole</a>	AWS managed	1

- On the final page, fill in the 'Role Name' as 'EMR\_DefaultRole', and at the bottom, click 'Create Role'. You should be brought back to the IAM page.
- Repeat steps 4-7 to create another role, this time with 'EMR Role for EC2' as use case, which should pre-populate the "AmazonElasticMapReduceforEC2Role". Call it 'EMR\_EC2\_DefaultRole'.

## SSH into head node



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

- You must know the location of the private key (.ppk or .pem) file on your local machine.
- Once the cluster is created, give it a few minutes for the status to change from Starting to Waiting.
- Click on Connect to Primary Node using SSH

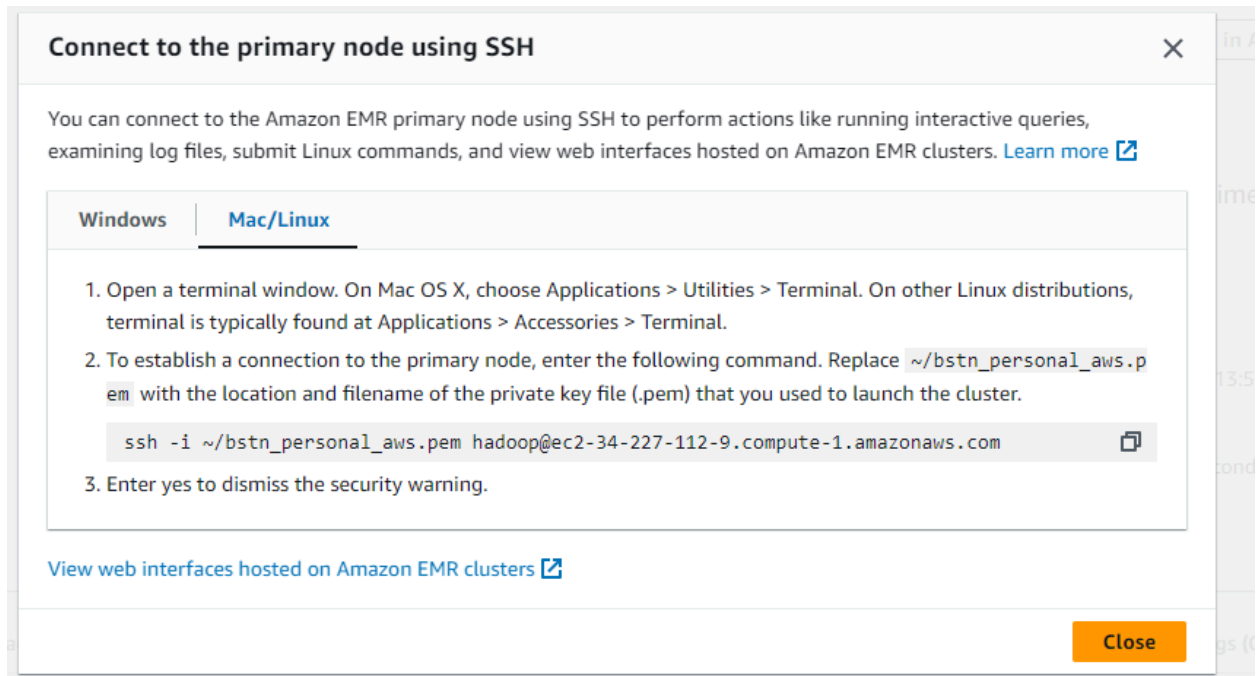
Amazon EMR > EMR on EC2: Clusters > Ngrams0625

**Ngrams0625** Updated less than a minute ago [↻](#) [Terminate](#) [Clone in AWS CLI](#) [Clone](#)

▼ Summary

Cluster info	Applications	Cluster management	Status and time
Cluster ID j-28CF6407RNE5O	Amazon EMR version emr-6.10.0	Log destination in Amazon S3 <a href="#">aws-logs-339712953454-us-east-1/elasticmapreduce</a>	Status  Starting
Cluster configuration Instance groups	Installed applications Hadoop 3.3.3, Hive 3.1.3, Hue 4.10.0, JupyterHub 1.5.0, Livy 0.7.1, Spark 3.3.1	Primary node public DNS  <a href="#">ec2-34-227-112-9.compute-1.amazonaws.com</a>	Creation time June 25, 2024, 13:53 (UTC-04:00)
Capacity 1 Primary   2 Core   0 Task		<a href="#">Connect to the Primary node using SSH</a> Connect to the Primary node using SSM <a href="#">↗</a>	Elapsed time 3 minutes, 17 seconds

- This will give you the address of the machine to ssh into.



5. On Windows, open Putty
6. Under Hostname, put the name of the remote machine. e.g. - [hadoop@ec2-34-227-112-9.compute-1.amazonaws.com](https://hadoop@ec2-34-227-112-9.compute-1.amazonaws.com)
7. In Putty, in the Menu on the left side, go to Connection -> SSH -> Auth -> Credentials. Put the path to your private key file there.
8. Under Connection -> SSH-> Tunnels, do the following setting so that your jupyter on the remote machine can be seen on your localhost.
  - a. Source Port - set to 9995
  - b. Destination port - set to the remote address followed by 9443. E.g. `ec2-34-227-112-9.compute-1.amazonaws.com:9443`
  - c. Click on 'Add'
  - d. Make sure the two boxes "Local ports accept connections from other hosts" and "Remote ports do the same (SSH-2 only)" are checked.
9. Go to Session, give it a new name under Saved Sessions and hit Save.
10. Connect to the session. If you see a ssh session that says EMR, the connection is successful.

```

hadoop@ip-172-31-90-118:~
Using username "hadoop".
Authenticating with public key "bstn_personal_aws"
Last login: Tue Jun 25 18:06:20 2024

      #_
    ~\  ####_      Amazon Linux 2
    ~~ \_#####\
    ~~   \###|      AL2 End of Life is 2025-06-30.
    ~~    \#/
    ~~     V~' '->
    ~~~
    ~.._./_/_/_/_/
    _/m/'      A newer version of Amazon Linux is available!
                Amazon Linux 2023, GA and supported until 2028-03-15.
                https://aws.amazon.com/linux/amazon-linux-2023/

3 package(s) needed for security, out of 4 available
Run "sudo yum update" to apply all updates.

EEEEEEEEEEEEEEEEEEEE MMMMMMMM          MMMMMMMM RRRRRRRRRRRRRRRR
E::::::::::::::::::::E M::::::::M          M::::::::M R:::::::::R
EE:::::EEEEEEEEEE::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
EEEEEEEEEEEEEEEEEEEE MMMMMMMM          MMMMMMMM RRRRRRR      RRRRRR

[hadoop@ip-172-31-90-118 ~]$

```

## Copy data to HDFS

Copy the data folder from the S3 bucket *directly* into a directory on the Hadoop File System (HDFS)

Source path - [s3://brainstation-dsft/eng\\_1M\\_1gram.csv](s3://brainstation-dsft/eng_1M_1gram.csv)

Destination folder- /user/hadoop/eng\_1M\_1gram

Command to run:

hadoop distcp [s3://brainstation-dsft/eng\\_1M\\_1gram.csv](s3://brainstation-dsft/eng_1M_1gram.csv) /user/hadoop/eng\_1M\_1gram

If everything went right, you will see the file when you do `hadoop fs -ls`

## Transfer file HDFS -> Laptop

**Steps to get the resulting file from HDFS to your local machine.**

1. Stored csv is in user/livy
2. `hadoop fs -ls /user/livy`
3. `hadoop fs -getmerge HADOOP_FILE LOCAL_FILE`` to get back to regular data

`hadoop fs -getmerge /user/livy/ngram_data.csv ngramLocal.csv`

4. `sudo cp MY_FILE /mnt/var/lib/jupyter/home/jovyan/`

`sudo cp ngramLocal.csv /mnt/var/lib/jupyter/home/jovyan/`

5. Download it onto your machine. Also make sure to Download the spark notebook before you terminate the cluster

The screenshot shows the JupyterHub web interface. At the top, there's a browser address bar with the URL `https://localhost:9995/user/jovyan/tree?`. Below the header, there are tabs for 'Files', 'Running', and 'Clusters'. The 'Files' tab is active, showing a file browser interface. At the top of the file browser, there are buttons for 'Duplicate', 'Rename', 'Move', 'Download', 'View', 'Edit', and a trash icon. On the right, there are buttons for 'Upload', 'New', and a refresh icon. The main area displays a list of files and folders. The files are:

Name	Last Modified	File size
0621_spark_june2024_complete.ipynb	Running an hour ago	1.01 MB
Google Ngrams.ipynb	Running 8 minutes ago	31.9 kB
jupyterhub-proxy.pid	2 hours ago	2 B
jupyterhub.sqlite	in a few seconds	102 kB
jupyterhub_cookie_secret	2 hours ago	65 B
ngramLocal.csv	seconds ago	7.34 kB

The file 'ngramLocal.csv' is selected, indicated by a blue checkmark in the first column and a blue circle around the row.