

210701238

Ex No 3

Map Reduce program to process a weather dataset.

AIM:

To implement MapReduce program to process a weather dataset.

PROCEDURE:

Step 1: Prepare the Weather Dataset

Ensure your weather dataset is in a text file (weather_dataset.txt). A simple format might be:

python

Copy code

2024-01-01,23,60

2024-01-02,25,65

...

Step 2: Create Input Directory in HDFS

Upload your dataset to HDFS:

bash

Copy code

hdfs dfs -mkdir -p /weather/input

hdfs dfs -put /path/to/your/weather_dataset.txt /weather/input

Step 3: Write the MapReduce Program in Python

We'll create two Python scripts: one for the mapper and one for the reducer.

1. Mapper (mapper.py):

- Reads each line of the input file.
- Extracts the year and temperature.
- Outputs the year as the key and the temperature as the value.

2. Reducer (reducer.py):

- Receives key-value pairs from the mapper.
- Finds the maximum temperature for each year.

mapper.py:

python

Copy code

```
#!/usr/bin/env python3
```

```
import sys
```

```
for line in sys.stdin:
```

```
    line = line.strip()
```

210701238

```
if not line:
    continue
try:
    date, temperature, _ = line.split(',')
    year = date[:4]
    print(f"{year}\t{temperature}")
except ValueError:
    continue # skip lines with missing or invalid data
```

reducer.py:

python

Copy code

```
#!/usr/bin/env python3
```

```
import sys
```

```
current_year = None
```

```
max_temp = -float('inf')
```

```
for line in sys.stdin:
```

```
    line = line.strip()
```

```
    year, temperature = line.split('\t')
```

```
    temperature = int(temperature)
```

```
    if current_year == year:
```

```
        if temperature > max_temp:
```

```
            max_temp = temperature
```

```
    else:
```

```
        if current_year:
```

```
            print(f"{current_year}\t{max_temp}")
```

```
        current_year = year
```

```
        max_temp = temperature
```

```
if current_year:
```

```
    print(f"{current_year}\t{max_temp}")
```

Step 4: Make the Scripts Executable

Make sure the Python scripts are executable:

```
bash
```

Copy code

```
chmod +x mapper.py reducer.py
```

Step 5: Run the MapReduce Job Using Hadoop Streaming

Use Hadoop streaming to run the MapReduce job with Python scripts:

210701238

bash

Copy code

```
hadoop jar /usr/local/hadoop/share/hadoop/tools/lib/hadoop-streaming-*.jar \  
-input /weather/input/weather_dataset.txt \  
-output /weather/output \  
-mapper /path/to/mapper.py \  
-reducer /path/to/reducer.py
```

Replace /path/to/mapper.py and /path/to/reducer.py with the actual paths to your scripts.

Step 6: View the Results

Check the output directory for results:

bash

Copy code

```
hdfs dfs -ls /weather/output
```

```
hdfs dfs -cat /weather/output/part-00000
```

This will display the maximum temperature for each year based on your dataset.

OUTPUT:

```
senthil@senthil-2463:~$ hdfs dfs -mkdir -p /weather/input
2024-09-12 08:45:56,265 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
senthil@senthil-2463:~$ hdfs dfs -put /home/senthil/weather_dataset.txt /weather/input
2024-09-12 08:46:01,887 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
2024-09-12 08:46:02,640 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localhostTrusted = false, remoteHostTrusted = false
senthil@senthil-2463:~$ vi mapper.py
senthil@senthil-2463:~$ vi reducer.py
senthil@senthil-2463:~$ chmod +x mapper.py reducer.py
senthil@senthil-2463:~$ hadoop jar /usr/local/hadoop/share/hadoop/tools/lib/hadoop-streaming-*.jar \  
> -input /weather/input/weather_dataset.txt \  
> -output /weather/output \  
> -mapper /home/senthil/mapper.py \  
> -reducer /home/senthil/reducer.py
2024-09-12 08:48:09,561 INFO streaming.StreamJob: Output directory: /weather/output
senthil@senthil-2463:~$ hdfs dfs -ls /weather/output
hdfs dfs -cat /weather/output/part-00000
2024-09-12 08:48:16,739 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 2 items
-rw-r--r-- 3 senthil supergroup 0 2024-09-12 08:48 /weather/output/_SUCCESS
-rw-r--r-- 3 senthil supergroup 24 2024-09-12 08:48 /weather/output/part-00000
2024-09-12 08:48:18,143 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
2024-09-12 08:48:18,825 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localhostTrusted = false, remoteHostTrusted = false
2024 25
2025 32
2026 37
senthil@senthil-2463:~$
```

RESULT:

Thus, the program for weather dataset using Map Reduce has been executed successfully.