

**BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC NHA TRANG  
KHOA CÔNG NGHỆ THÔNG TIN**

-----



**ĐỒ ÁN MÔN HỌC  
XỬ LÝ DỮ LIỆU LỚN (INS358)**

**XÂY DỰNG CHƯƠNG TRÌNH TỔNG HỢP CÁC  
TWEETS CỦA ELON MUSK TRÊN  
HADOOP/SPARK**

**Sinh viên thực hiện: Trần Duy Phát**

**MSSV: 63135054**

**Lớp: 63.CNT-3**

**Giảng viên: TS. Nguyễn Đình Hưng**

**Khánh Hòa - 2024**

## KẾT QUẢ ĐÁNH GIÁ ĐỒ ÁN MÔN HỌC

*Họ và tên sinh viên:*

*MSSV:*

*Lớp:*

Nội dung	Trọng số	Điểm
<b>1. Giải quyết vấn đề</b>		
1.1. Phân tích bài toán; thu thập, khảo sát và chuẩn bị dữ liệu; thiết kế giải thuật	20%	
1.2. Cài đặt, triển khai ứng dụng trên Hadoop	20%	
1.3. Cài đặt, triển khai ứng dụng trên Spark	20%	
<b>2. Báo cáo bài tập lớn</b>		
2.1. Nội dung báo cáo	20%	
2.2. Vấn đáp	20%	
<b>Điểm trung bình</b>		

*Giảng viên*

### **Lời cam đoan**

Tôi cam đoan đây là công trình do tôi tự thực hiện. Các nội dung nghiên cứu, số liệu và kết quả thực nghiệm là trung thực. Các số liệu, công trình sử dụng của tác giả khác đều được trích dẫn nguồn gốc rõ ràng.

Tất cả phần mềm sử dụng trong đồ án này đều là mã nguồn mở.

Nếu phát hiện có bất kì sự gian lận nào, tôi xin chịu hoàn toàn trách nhiệm.

Trần Duy Phát

# Mục lục

<b>Mục lục .....</b>	<b>4</b>
<b>Danh sách hình .....</b>	<b>6</b>
<b>Chương 1. Giới thiệu.....</b>	<b>7</b>
1.1 Tổng quát về dữ liệu lớn .....	7
1.2 Mục tiêu của đề tài .....	7
1.3 Cấu trúc của đồ án .....	7
<b>Chương 2. Nội dung và phương pháp thực hiện.....</b>	<b>8</b>
2.1 Phân tích bài toán.....	8
2.2 Thu thập dữ liệu .....	8
2.3 Cài đặt và triển khai ứng dụng trên Hadoop.....	8
2.3.1 Cài đặt Hadoop.....	8
2.3.2 Xây dựng giải thuật.....	15
2.3.3 Lập trình ứng dụng .....	15
2.3.4 Thực thi ứng dụng.....	17
2.4 Cài đặt và triển khai ứng dụng trên Spark.....	19
2.4.1 Cài đặt Spark.....	19
2.4.2 Lập trình ứng dụng .....	20
2.4.3 Thực thi ứng dụng.....	22
<b>Chương 3. Kết luận .....</b>	<b>24</b>
3.1 Đánh giá chung .....	24

3.1.1 Những kết quả đạt được .....	24
3.1.2 Một số hạn chế .....	24
3.2 Hướng phát triển .....	24
Tài liệu tham khảo.....	25

# Danh sách hình

Hình 2.1 Cài đặt SSH.....	9
Hình 2.2 Minh hoạ khi kiểm tra SSH .....	9
Hình 2.3 Ảnh minh hoạ Hadoop đã chạy.....	14
Hình 2.4 Hình ảnh khi truy cập Hadoop .....	14
Hình 2.5 Mô tả thuật toán hoạt động .....	15
Hình 2.6 Ảnh minh hoạ khi chạy hoàn thành MapReduce .....	18
Hình 2.7 Kết quả trả về. ....	18
Hình 2.8 Kết quả của thực thi .....	19
Hình 2.9 Ảnh minh hoạ Pyspark đã chạy .....	20
Hình 2.10 Ảnh minh hoạ code tạo python của ứng dụng spark.....	20
Hình 2.11 Ảnh minh hoạ dòng lệnh chạy ứng dụng cho Spark. ....	22
Hình 2.12 Ảnh minh hoạ kết quả về đếm số tweet theo ngày.....	23
Hình 2.13 Ảnh minh hoạ kết quả về đếm số tweet theo giờ. ....	23
Hình 2.14 Kết quả khung giờ Elon Musk hay đăng tweet nhất. ....	23

# Chương 1.

## GIỚI THIỆU

### 1.1 Tổng quát về dữ liệu lớn

Dữ liệu lớn (Big Data) đề cập đến tập hợp các dữ liệu với khối lượng khổng lồ, đa dạng và được tạo ra với tốc độ nhanh, vượt xa khả năng xử lý của các hệ thống truyền thống. Việc khai thác và phân tích dữ liệu lớn không chỉ giúp các doanh nghiệp và tổ chức hiểu rõ hơn về xu hướng thị trường, hành vi người tiêu dùng mà còn hỗ trợ tối ưu hóa quy trình sản xuất, nâng cao hiệu quả hoạt động và ra quyết định dựa trên các thông tin chính xác. Công nghệ tiên tiến như trí tuệ nhân tạo và học máy đang được ứng dụng rộng rãi nhằm chuyển đổi dữ liệu thành giá trị thực tiễn, đồng thời tạo ra cơ hội đổi mới sáng tạo trong kỷ nguyên số. Tuy nhiên, sự bùng nổ của dữ liệu lớn cũng đặt ra những thách thức về cơ sở hạ tầng, bảo mật thông tin và quản lý dữ liệu hiệu quả.

### 1.2 Mục tiêu của đề tài

Đề tài nhằm đạt được các mục tiêu chính sau:

- Nghiên cứu toàn diện khái niệm và những ứng dụng thực tiễn của dữ liệu lớn.
- Khám phá và phân tích các phương pháp, công nghệ cùng các công cụ tiêu biểu được ứng dụng trong lĩnh vực xử lý dữ liệu lớn.
- Áp dụng kiến thức đã học để thiết kế và xây dựng một ứng dụng cơ bản xử lý dữ liệu lớn.

### 1.3 Cấu trúc của đồ án

Đồ án gồm các phần sau:

- Chương 1: Giới thiệu.
- Chương 1: Nội dung và phương pháp thực hiện.
- Chương 1: Kết luận

# Chương 2.

## NỘI DUNG VÀ PHƯƠNG PHÁP THỰC HIỆN

### 2.1 Phân tích bài toán

Cho bộ dữ liệu chứa các tweets của Elon Musk chứa: id, thời gian tạo (create\_at) và nội dung (text). Yêu cầu bài toán: Đếm số tweet của từng ngày, đếm số tweet theo từng khung giờ để xác định khung giờ mà Elon Musk thường đăng tweet.

### 2.2 Thu thập dữ liệu

Trong dự án này tôi sử dụng nguồn dữ liệu:

[https://github.com/nd-hung/Big-Data/blob/main/datasets/ElonMusk\\_tweets.csv](https://github.com/nd-hung/Big-Data/blob/main/datasets/ElonMusk_tweets.csv)

### 2.3 Cài đặt và triển khai ứng dụng trên Hadoop

#### 2.3.1 Cài đặt Hadoop

##### 1. Tạo tài khoản quản trị Hadoop

Để đảm bảo bảo mật, tạo một tài khoản riêng để quản lý Hadoop:

```
sudo adduser hdoop
```

Nhập mật khẩu khi được yêu cầu.

Cấp quyền sudo cho tài khoản:

```
sudo usermod -aG sudo hdoop
```

Đăng nhập tài khoản Hadoop:

```
su - hdoop
```

##### 2. Cài đặt SSH

SSH giúp truy cập từ xa. Cài đặt SSH bằng lệnh:

```
sudo apt install openssh-server openssh-client -y
```

Tạo cặp khóa SSH để đăng nhập không cần mật khẩu:



```
ssh-keygen -t rsa -P "" -f ~/.ssh/id_rsa
```

```
hdoop@phat-VirtualBox:~$ ssh-keygen -t rsa -P "" -f ~/.ssh/id_rsa
Generating public/private rsa key pair.
/home/hdoop/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Your identification has been saved in /home/hdoop/.ssh/id_rsa.
Your public key has been saved in /home/hdoop/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:J/WJSzrbSmEkCpkGBuAWBDqXtMwqDzbYBURS3P9mSk0 hdoop@phat-VirtualBox
The key's randomart image is:
+---[RSA 2048]---+
|@0*.  
|+*o*.  
|oo@ ... ..  
|o* o ..oE. o .  
|=o. . +S + o  
|oo. ..=* .  
| . . ++ .  
| . . +  
| o..  
+---[SHA256]---+
```

Hình 2.1 Cài đặt SSH

Thêm khóa vào danh sách được ủy quyền:

```
cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
```

```
chmod 0600 ~/.ssh/authorized_keys
```

Kiểm tra SSH:

```
ssh localhost
```

```
phat@phat-VirtualBox:~$ ssh localhost
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:vWRSjvQOT+YPIbgKuVHYfrSi9aGy2EdPcrXg69+HrHg.
Are you sure you want to continue connecting (yes/no)? yes
```

Hình 2.2 Minh họa khi kiểm tra SSH

Nhập **yes** khi được hỏi.

Lưu và khởi động lại hệ thống. Kiểm tra trạng thái:

```
cat /proc/sys/net/ipv6/conf/all/disable_ipv6
```

Nếu kết quả là 1, IPv6 đã bị tắt.

#### 4. Cài đặt Java

Hadoop 3.2.x yêu cầu Java 8:

```
sudo apt install openjdk-8-jdk -y
```

#### 5. Cài đặt Hadoop

Tải Hadoop 3.2.2:

```
wget https://archive.apache.org/dist/hadoop/common/hadoop-3.2.2/hadoop-3.2.2.tar.gz
```

Giải nén:

```
tar xzf hadoop-3.2.2.tar.gz
```

#### 6. Thiết lập biến môi trường

Mở tệp `~/.bashrc`:

```
sudo nano ~/.bashrc
```

Thêm vào cuối:

```
# Hadoop Environment Variables
export HADOOP_HOME=/home/hadoop/hadoop-3.2.2
export PATH=$PATH:$HADOOP_HOME/sbin:$HADOOP_HOME/bin
export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib/native"
Djava.library.path=$HADOOP_HOME/lib/native"
```

Lưu và áp dụng:

```
source ~/.bashrc
```

#### 7. Cấu hình Hadoop

Mở `hadoop-env.sh`:

```
sudo nano $HADOOP_HOME/etc/hadoop/hadoop-env.sh
```

Thêm:

```
export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
```

Thiết lập thư mục tạm:

```
sudo mkdir -p /app/hadoop/tmp  
sudo chown hdoop:hdoop /app/hadoop/tmp  
sudo chmod 750 /app/hadoop/tmp
```

Mở *core-site.xml*:

```
sudo nano $HADOOP_HOME/etc/hadoop/core-site.xml
```

Thêm:

```
<configuration>  
  <property>  
    <name>hadoop.tmp.dir</name>  
    <value>/app/hadoop/tmp</value>  
  </property>  
  <property>  
    <name>fs.defaultFS</name>  
    <value>hdfs://localhost:9000</value>  
  </property>  
</configuration>
```

Mở *hdfs-site.xml*:

```
sudo nano $HADOOP_HOME/etc/hadoop/hdfs-site.xml
```

Thêm:

```
<configuration>
```

```
<property>

  <name>dfs.replication</name>

  <value>1</value>

</property>

</configuration>
```

Mở *mapred-site.xml*:

```
sudo nano $HADOOP_HOME/etc/hadoop/mapred-site.xml
```

Thêm:

```
<configuration>

  <property>

    <name>mapreduce.framework.name</name>

    <value>yarn</value>

  </property>

</configuration>
```

Mở *yarn-site.xml*:

```
sudo nano $HADOOP_HOME/etc/hadoop/yarn-site.xml
```

Thêm:

```
<configuration>

  <property>

    <name>yarn.nodemanager.aux-services</name>

    <value>mapreduce_shuffle</value>

  </property>

</configuration>
```

## 8. Định dạng HDFS

```
hdfs namenode -format
```

## 9. Khởi động và dừng Hadoop

Khởi động:

```
start-dfs.sh  
start-yarn.sh
```

Hoặc:

```
start-all.sh
```

Dừng Hadoop:

```
stop-dfs.sh  
stop-yarn.sh
```

Hoặc:

```
stop-all.sh
```

## 10. Truy cập Hadoop qua trình duyệt

Mở trình duyệt và truy cập:

```
http://localhost:9870
```

Hoàn tất quá trình cài đặt và cấu hình Hadoop trên Ubuntu.

```
phat@phat-VirtualBox:~$ su - hdoop  
Password:  
hdoop@phat-VirtualBox:~$ start-all.sh  
WARNING: Attempting to start all Apache Hadoop daemons as hdoop in 10 seconds.  
WARNING: This is not a recommended production deployment configuration.  
WARNING: Use CTRL-C to abort.  
Starting namenodes on [localhost]  
localhost: hdoop@localhost: Permission denied (publickey,password).  
Starting datanodes  
localhost: hdoop@localhost: Permission denied (publickey,password).  
Starting secondary namenodes [phat-VirtualBox]  
phat-VirtualBox: hdoop@phat-virtualbox: Permission denied (publickey,password).  
Starting resourcemanager  
Starting nodemanagers  
localhost: hdoop@localhost: Permission denied (publickey,password).  
hdoop@phat-VirtualBox:~$ S
```

Hình 2.3 Ảnh minh họa Hadoop đã chạy

HadoopOverviewDatanodesDatanode Volume FailuresSnapshotStartup ProgressUtilities

Overview 'localhost:9000' (active)

Started:

Wed Mar 17 09:34:35 +0700 2021

Version:

3.2.2, r7a3bc90b05f257c8ace2f76d74264906f0f7a932

Compiled:

Sun Jan 03 16:26:00 +0700 2021 by hexiaoqiao from branch-3.2.2

Cluster ID:

CID-9c0ec093-b121-4a41-90f2-5dc95f7dd815

Block Pool ID:

BP-3652805-127.0.1.1-1615948455243

Summary

Security is off.

Safemode is off.

27 files and directories, 13 blocks (13 replicated blocks, 0 erasure coded block groups) = 40 total filesystem object(s).

Heap Memory used 65.31 MB of 121.88 MB Heap Memory. Max Heap Memory is 1.88 GB.

Non Heap Memory used 56.27 MB of 57.52 MB Committed Non Heap Memory. Max Non Heap Memory is <unbounded>.

Configured Capacity:	39.79 GB
Configured Remote Capacity:	0 B
DFS Used:	432 KB (0%)
Non DFS Used:	8.38 GB
DFS Remaining:	29.36 GB (73.79%)
Block Pool Used:	432 KB (0%)
DataNodes usages% (Min/Median/Max/stdDev):	0.00% / 0.00% / 0.00% / 0.00%
Live Nodes	1 (Decommissioned: 0, In Maintenance: 0)
Dead Nodes	0 (Decommissioned: 0, In Maintenance: 0)
Decommissioning Nodes	0
Entering Maintenance Nodes	0
Total Datanode Volume Failures	0 (0 B)
Number of Under-Replicated Blocks	0
Number of Blocks Pending Deletion (including replicas)	0
Block Deletion Start Time	Wed Mar 17 09:34:35 +0700 2021
Last Checkpoint Time	Wed Mar 17 09:34:15 +0700 2021
Enabled Erasure Coding Policies	RS-6-3-1024k

NameNode Journal Status

Current transaction ID: 180

Journal Manager	State
FileJournalManager(root=/tmp/hadoop-hdoop/dfs/name)	EditLogOutputStream(/tmp/hadoop-hdoop/dfs/name/current/edits_inprogress_0000000000000000180)

NameNode Storage

Storage Directory	Type	State
/tmp/hadoop-hdoop/dfs/name	IMAGE_AND_EDITS	Active

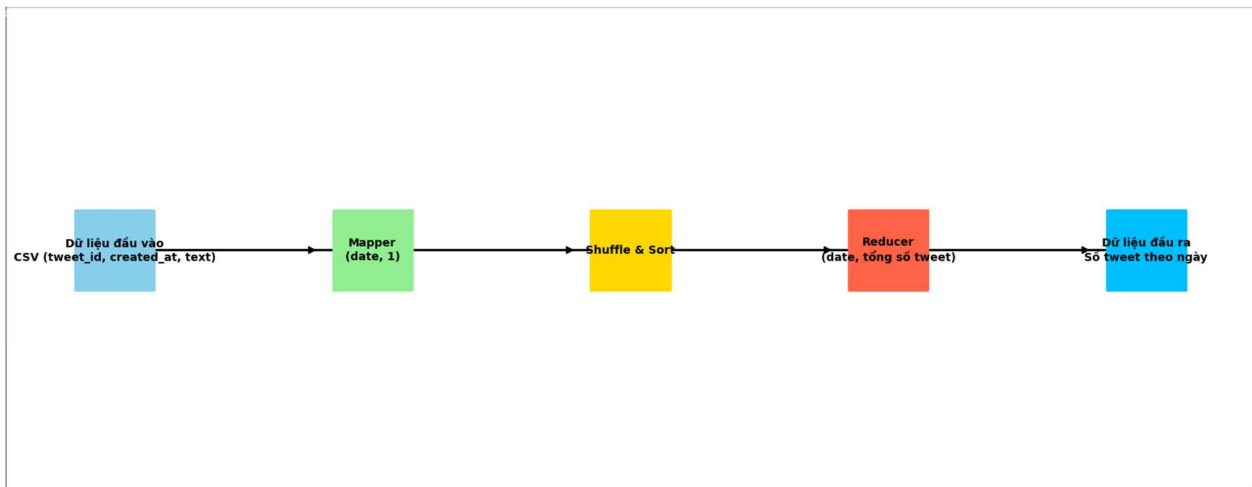
DFS Storage Types

Storage Type	Configured Capacity	Capacity Used	Capacity Remaining	Block Pool Used	Nodes In Service
DISK	39.79 GB	432 KB (0%)	29.36 GB (73.79%)	432 KB	1

Hadoop, 2021.

Hình 2.4 Hình ảnh khi truy cập Hadoop

### 2.3.2 Xây dựng giải thuật



Hình 2.5 Mô tả thuật toán hoạt động

### 2.3.3 Lập trình ứng dụng

#### 1. Đăng nhập tài khoản hdoop

Mở terminal và chạy lệnh:

```
su - hdoop
```

#### 2. Tạo các file Python cho MapReduce

##### 2.1. Tạo file mapper\_date.py

Mở terminal và nhập lệnh:

```
nano mapper_date.py
```

Dán nội dung sau:

```
#!/usr/bin/python3  
  
import sys  
  
import csv
```

```
for line in sys.stdin:

    try:

        row = next(csv.reader([line])) # Đọc dòng CSV

        tweet_id, created_at, text = row

        date = created_at[:10] # Lấy ngày từ created_at (YYYY-MM-DD HH:MM:SS)

        print(f'{date}\t1') # Output: (date, 1)

    except Exception:

        continue # Bỏ qua dòng lỗi
```

Lưu file (Ctrl+X, nhấn Y, Enter).

## 2.2. Tạo file reducer\_date.py

Mở terminal và nhập lệnh:

```
nano reducer_date.py
```

Dán nội dung sau:

```
#!/usr/bin/python3

import sys

from collections import defaultdict

tweet_count = defaultdict(int)

for line in sys.stdin:

    day, count = line.strip().split("\t")

    tweet_count[day] += int(count)

for day in sorted(tweet_count):
```



```
print(f'{day}\t{tweet_count[day]}")
```

Lưu file.

### 2.3. Cấp quyền thực thi

Chạy lệnh sau:

```
chmod +x mapper_date.py reducer_date.py
```

### 3. Upload dữ liệu lên HDFS

Chạy Hadoop:

```
start-all.sh
```

Giả sử file dữ liệu ElonMusk\_tweets.csv đã được lưu tại /home/phan/Downloads/tweet. Tải file lên HDFS:

```
hdfs dfs -mkdir -p /user/hadoop/data
```

```
hdfs dfs -copyFromLocal /home/phan/Downloads/tweet /user/hadoop/data
```

#### 2.3.4 Thực thi ứng dụng

1. Kiểm tra file trong HDFS:

```
hdfs dfs -ls /user/hadoop/data
```

Xóa output cũ (nếu có):

```
hdfs dfs -rm -r /user/hadoop/data/tweet_count_by_date
```

2. Chạy job MapReduce:

```
hadoop jar $HADOOP_HOME/share/hadoop/tools/lib/hadoop-streaming-3.2.2.jar \  
-file mapper_date.py -mapper mapper_date.py \  
-file reducer_date.py -reducer reducer_date.py \  
-input /user/hadoop/data/tweet \
```

```
-output /user/hadoop/data/tweet_count_by_date
```

### 3. Xem kết quả:

```
hdfs dfs -cat /user/hadoop/data/tweet_count_by_date/part-00000
```

### 4. Dừng Hadoop

Sau khi hoàn tất công việc, dừng Hadoop:

```
stop-all.sh
```

```
hadoop@phat-VirtualBox:~/tweet$ hadoop jar $HADOOP_HOME/share/hadoop/tools/lib/hadoop-streaming-3.2.2.jar -file mapper_date.py -mapper mapper_date.py -file reducer_date.py -reducer reducer_date.py -i
input /user/hadoop/data/tweet -output /user/hadoop/data/tweet_count_by_date
2025-03-29 15:21:11,072 WARN streaming.StreamJob: -file option is deprecated, please use generic option -files instead.
packageJobJar: [mapper_date.py, reducer_date.py, /tmp/hadoop-unjar2042567983735921260/] [] /tmp/streamjob8613303830259746956.jar tmpDir=null
2025-03-29 15:21:11,641 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
2025-03-29 15:21:11,834 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
2025-03-29 15:21:11,991 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/hadoop/.staging/job_1743236090982_0002
2025-03-29 15:21:12,223 INFO mapred.FileInputFormat: Total input files to process : 1
2025-03-29 15:21:12,299 INFO mapreduce.JobSubmitter: number of splits:2
2025-03-29 15:21:12,394 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1743236090982_0002
2025-03-29 15:21:12,395 INFO mapreduce.JobSubmitter: Executing with tokens: []
2025-03-29 15:21:12,541 INFO conf.Configuration: resource-types.xml not found
2025-03-29 15:21:12,541 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2025-03-29 15:21:12,611 INFO impl.YarnClientImpl: Submitted application application_1743236090982_0002
2025-03-29 15:21:12,636 INFO mapreduce.Job: The url to track the job: http://phat-VirtualBox:8088/proxy/application_1743236090982_0002/
2025-03-29 15:21:12,637 INFO mapreduce.Job: Running job: job_1743236090982_0002
2025-03-29 15:21:17,727 INFO mapreduce.Job: Job job_1743236090982_0002 running in uber mode : false
2025-03-29 15:21:17,728 INFO mapreduce.Job: map 0% reduce 0%
2025-03-29 15:21:21,810 INFO mapreduce.Job: map 50% reduce 0%
2025-03-29 15:21:22,820 INFO mapreduce.Job: map 100% reduce 0%
2025-03-29 15:21:27,858 INFO mapreduce.Job: map 100% reduce 100%
```

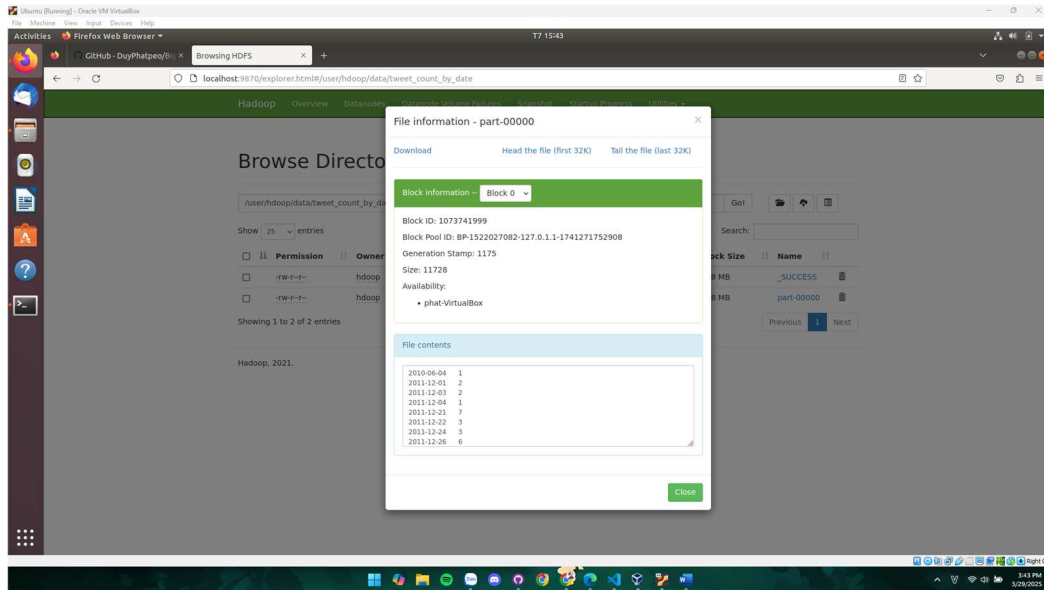
Hình 2.6 Ảnh mình hoạ khi chạy hoàn thành MapReduce

```
hadoop@phat-VirtualBox:~/tweet$ hdfs dfs -cat /user/hadoop/data/tweet_count_by_date/part-00000
2010-06-04      1
2011-12-01      2
2011-12-03      2
2011-12-04      1
2011-12-21      7
2011-12-22      3
2011-12-24      3
2011-12-26      6
2011-12-27      2
2011-12-28      2
2011-12-29      5
2011-12-30      8
2011-12-31      3
2012-01-01      7
2012-01-03      3
2012-01-06      3
2012-01-11      1
2012-01-12      3
2012-01-13      1
2012-01-14      2
2012-01-17      3
```

Hình 2.7 Kết quả trả về.

Khi truy cập vào Hadoop:

```
localhost:9870/explorer.html#/user/hadoop/date/tweet_count_by_date
```



Hình 2.8 Kết quả của thực thi

## 2.4 Cài đặt và triển khai ứng dụng trên Spark

### 2.4.1 Cài đặt Spark

#### 1. Tải và giải nén Spark

Tải phiên bản Spark 3.5.5:

```
wget https://dlcdn.apache.org/spark/spark-3.5.5/spark-3.5.5-bin-hadoop3.tgz
```

Giải nén:

```
tar xvf spark-3.5.5-bin-hadoop3.tgz
```

Di chuyển thư mục Spark:

```
sudo mv spark-3.5.5-bin-hadoop3 /opt/spark
```

#### 2. Thiết lập biến môi trường

Mở file `~/.bashrc` và thêm các dòng sau:

```
echo "export SPARK_HOME=/opt/spark" >> ~/.bashrc
```

```
echo "export PATH=$SPARK_HOME/bin:$SPARK_HOME/sbin:$PATH" >> ~/.bashrc
```

```
echo "export PYSPARK_PYTHON=python3" >> ~/.bashrc
```

Nạp lại cấu hình:

```
source ~/.bashrc
```

### 3. Khởi động PySpark

pyspark

```
phat@phat-VirtualBox:~$ pyspark
Python 3.6.9 (default, Mar 10 2023, 16:46:00)
[GCC 8.4.0] on linux
Type "help", "copyright", "credits" or "license()" for more information.
25/03/25 09:58:04 WARN Utils: Your hostname, phat-VirtualBox resolves to a loopback address: 127.0.1.1
1: using 10.0.2.15 instead (on interface enp0s3)
25/03/25 09:58:04 WARN Utils: Set SPARK_LOCAL_IP if you need to bind to another address
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
25/03/25 09:58:06 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Welcome to

  ____      __
 / ___ |__ / /_  __
/ /___/ __// __/ / 
/____/_/____/_/____/
version 3.5.5

Using Python version 3.6.9 (default, Mar 10 2023 16:46:00)
Spark context Web UI available at http://10.0.2.15:4040
Spark context available as 'sc' (master = local[*], app id = local-1742871487152).
SparkSession available as 'spark'.
>>> □
```

### Hình 2.9 Ảnh minh hoạ Pyspark đã chạy

### 2.4.2 Lập trình ứng dụng

## Mở terminal và tạo file Python

Nhập lệnh sau để mở trình chỉnh sửa nano:

nano tweet\_analysis.py

Sau đó, dán nội dung code vào file và lưu lại (**Ctrl + X → Y → Enter**).

```
phat@phat-VirtualBox:~$ nano tweet_analysis.py
```

### Hình 2.10 Ảnh minh họa code tạo python của ứng dụng spark

Sau đó dán vào file python: `:///mnt/data/ElonMusk_tweets.csv` là nơi lưu file dữ liệu

```
from pyspark.sql import SparkSession
```

```

# Khởi tạo SparkSession

spark = SparkSession.builder.appName("TweetAnalysis").getOrCreate()

sc = spark.sparkContext

# Đọc file dữ liệu

rdd = sc.textFile("file:///mnt/data/ElonMusk_tweets.csv")

# Bỏ dòng tiêu đề

header = rdd.first()

rdd = rdd.filter(lambda line: line != header)

# (a) Đếm số tweet theo ngày

def extract_date(line):

    fields = line.split(",")

    if len(fields) < 2:

        return None

    date = fields[1].split(" ")[0] # Lấy phần YYYY-MM-DD

    return (date, 1)

tweet_by_date = rdd.map(extract_date).filter(lambda x: x is not
None).reduceByKey(lambda a, b: a + b)

tweet_by_date_sorted = tweet_by_date.sortByKey()

tweet_by_date_sorted.coalesce(1).saveAsTextFile("tweet_count_by_date")

# In ra 10 dòng đầu tiên

tweet_by_date_sorted.take(5)

# (b) Đếm số tweet theo khung giờ

def extract_hour(line):

```

```

fields = line.split(",")

if len(fields) < 2:

    return None

hour = fields[1].split(" ")[1].split(":")[0] # Lấy giờ (HH)

return (hour, 1)

tweet_by_hour = rdd.map(extract_hour).filter(lambda x: x is not
None).reduceByKey(lambda a, b: a + b)

tweet_by_hour_sorted = tweet_by_hour.sortByKey()

tweet_by_hour_sorted.coalesce(1).saveAsTextFile("tweet_count_by_hour")

# In ra 10 dòng đầu tiên

tweet_by_hour_sorted.take(5)

# (c) Tìm khung giờ Elon Musk hay đăng tweet nhất

most_active_hour = tweet_by_hour.max(lambda x: x[1])

print(f"Khung giờ Elon Musk hay đăng tweet nhất: {most_active_hour[0]}h với
{most_active_hour[1]} tweet")

# Dừng SparkSession

spark.stop()

```

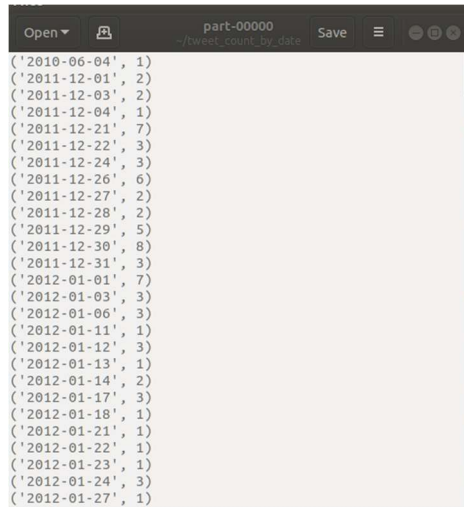
### 2.4.3 Thực thi ứng dụng

Trong terminal, nhập lệnh sau để chạy:

```
spark-submit tweet_analysis.py
```

```
phat@phat-VirtualBox:~$ spark-submit tweet_analysis.py
```

Hình 2.11 Ảnh minh hoạ dòng lệnh chạy ứng dụng cho Spark.

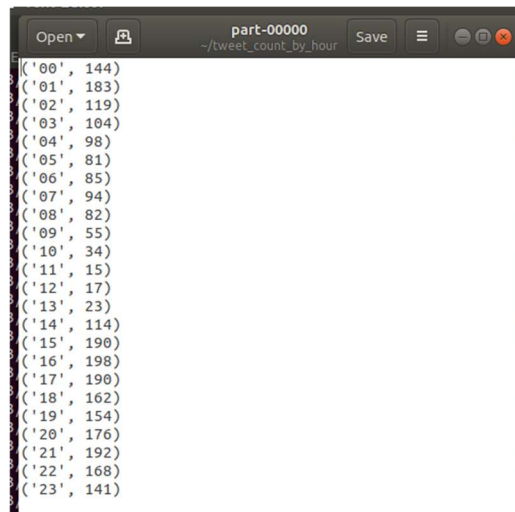


```

('2010-06-04', 1)
('2011-12-01', 2)
('2011-12-03', 2)
('2011-12-04', 1)
('2011-12-21', 7)
('2011-12-22', 3)
('2011-12-24', 3)
('2011-12-26', 6)
('2011-12-27', 2)
('2011-12-28', 2)
('2011-12-29', 5)
('2011-12-30', 8)
('2011-12-31', 3)
('2012-01-01', 7)
('2012-01-03', 3)
('2012-01-06', 3)
('2012-01-11', 1)
('2012-01-12', 3)
('2012-01-13', 1)
('2012-01-14', 2)
('2012-01-17', 3)
('2012-01-18', 1)
('2012-01-21', 1)
('2012-01-22', 1)
('2012-01-23', 1)
('2012-01-24', 3)
('2012-01-27', 1)

```

**Hình 2.12 Ảnh minh họa kết quả về đếm số tweet theo ngày.**

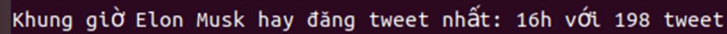


```

('00', 144)
('01', 183)
('02', 119)
('03', 104)
('04', 98)
('05', 81)
('06', 85)
('07', 94)
('08', 82)
('09', 55)
('10', 34)
('11', 15)
('12', 17)
('13', 23)
('14', 114)
('15', 190)
('16', 198)
('17', 190)
('18', 162)
('19', 154)
('20', 176)
('21', 192)
('22', 168)
('23', 141)

```

**Hình 2.13 Ảnh minh họa kết quả về đếm số tweet theo giờ.**



Khung giờ Elon Musk hay đăng tweet nhất: 16h với 198 tweet

**Hình 2.14 Kết quả khung giờ Elon Musk hay đăng tweet nhất.**

# Chương 3.

## KẾT LUẬN

### 3.1 Đánh giá chung

#### 3.1.1 Những kết quả đạt được

- Hệ thống hoạt động ổn định trên Hadoop, hỗ trợ phân tích dữ liệu lớn.
- Ứng dụng Apache Spark giúp cải thiện tốc độ xử lý.
- Kết quả hỗ trợ phân tích xu hướng trên mạng xã hội.

#### 3.1.2 Một số hạn chế

- Hiệu suất xử lý dữ liệu lớn chưa được tối ưu.
- Một số lỗi trong dữ liệu chưa được xử lý hoàn toàn.
- Giao diện trực quan còn hạn chế, cần cải thiện.
- Việc triển khai Spark vẫn đang thử nghiệm, cần tối ưu thêm.

### 3.2 Hướng phát triển

- Mở rộng phân tích theo chủ đề, cảm xúc của tweet.
- Xây dựng giao diện trực quan với biểu đồ, dashboard.
- Ứng dụng AI, Machine Learning để phân tích sâu hơn.
- So sánh hiệu suất giữa Hadoop MapReduce và Spark.
- Thử nghiệm Spark Streaming để xử lý dữ liệu thời gian thực.
- Nghiên cứu thêm các công nghệ như Apache Flink, Druid, Data Lake.



# Tài liệu tham khảo

[1] Nguyễn Đình Hưng – Tài liệu môn xử lý dữ liệu lớn - 2025