

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ
VIỆN TRÍ TUỆ NHÂN TẠO

BÁO CÁO MÔN HỌC KỸ THUẬT VÀ CÔNG NGHỆ DỮ
LIỆU LỚN

ĐỀ TÀI
K-MEANS VÀ LẬP TRÌNH MAPREDUCE HÓA
TRONG PHÂN CỤM ẢNH

Nhóm sinh viên thực hiện:

1. Vũ Việt Hùng - 22022585
2. Hồ Minh Hoàng - 22022567
3. Nguyễn Trọng Huy - 22022545
4. Nguyễn Đức Anh - 22022661
5. Hà Kim Dương - 22022621

Giảng viên hướng dẫn:

TS. Trần Hồng Việt

ThS. Ngô Minh Hương

HÀ NỘI, 11/2024

MỞ ĐẦU

Trong bài tập lần này chúng em tìm hiểu về dữ liệu lớn và công nghệ lưu trữ và xử lý dữ liệu lớn Hadoop HDFS. Từ đó áp dụng sử dụng thuật toán KMeans Mapreduce để xử lý phân cụm ảnh với dữ liệu được lưu trên HDFS. Với mục đích tìm hiểu học hỏi về cách hoạt động của hệ thống HDFS, cách thực hiện bài toán MapReduce và ứng dụng trong thực tế với bài toán phân cụm ảnh.

Báo cáo gồm 4 chương:

Chương 1: Tổng quan về dữ liệu lớn.

Chương 2: Thuật toán phân cụm K Means.

Chương 3: Ứng dụng của Kmeans và MapReduce trong phân cụm ảnh.

Chương 4: Kết luận và hướng phát triển.

MỤC LỤC

CHƯƠNG 1 TỔNG QUAN VỀ DỮ LIỆU LỚN.....	3
1. Thế nào là dữ liệu lớn.....	3
2. Đặc trưng cơ bản của bigdata.....	3
3. Tổng quan về hadoop.....	3
4. Tổng quan về MapReduce.....	4
CHƯƠNG 2 THUẬT TOÁN PHÂN CỤM K-MEANS.....	4
1. Giới thiệu thuật toán K-means clustering.....	4
2. Triển khai thuật toán K-means clustering.....	5
3. Ví dụ về một vòng lặp của thuật toán KMeans:.....	6
CHƯƠNG 3 ỨNG DỤNG CỦA MAPREDUCE VÀ K-MEANS TRONG PHÂN CỤM ẢNH.....	7
1. Giới thiệu bài toán.....	7
2. Cách tiếp cận.....	8
2.1 Xử lý dữ liệu đầu vào.....	8
2.2 Thuật toán K Means.....	8
3. Triển khai.....	10
3.1: Thuật toán KMean Mapreduce.....	10
3.2 Chuẩn bị dữ liệu:.....	11
4. Thực nghiệm.....	12
CHƯƠNG 4 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....	13
1. Ý nghĩa của phân cụm ảnh.....	13
2. Hướng phát triển thêm của phân cụm ảnh trong Big Data:.....	14

NHIỆM VỤ CỦA CÁC THÀNH VIÊN

VŨ VIỆT HÙNG	HÀ KIM DƯƠNG	NGUYỄN ĐỨC ANH	NGUYỄN TRỌNG HUY	HỒ MINH HOÀNG
<ul style="list-style-type: none"> - Triển khai thuật toán KMean Mapreduce cho nhiều ảnh - Làm ppt (phần 3 + 4) 	<ul style="list-style-type: none"> - Viết report (phần 1 + 2) - Tham gia tìm hiểu về dữ liệu lớn Hadoop - Tìm hiểu về thuật toán Kmean Mapreduce 	<ul style="list-style-type: none"> - Triển khai thuật toán KMean Mapreduce cho nhiều ảnh - Viết report (phần 3) - Thuyết trình 	<ul style="list-style-type: none"> - Làm ppt (phần 1 + 2) - Tham gia tìm hiểu về dữ liệu lớn Hadoop - Tìm hiểu về thuật toán Kmean-Mapreduce 	<ul style="list-style-type: none"> - Viết report (phần 4) - Tham gia tìm hiểu về dữ liệu lớn - Tìm hiểu về ý nghĩa ứng dụng của thuật toán KMean Mapreduce trong phân cụm ảnh

CHƯƠNG 1 TỔNG QUAN VỀ DỮ LIỆU LỚN

1. Thế nào là dữ liệu lớn.

- Big data là tập dữ liệu lớn mà chúng ta không thể xử lý bằng phương pháp truyền thống.

2. Đặc trưng cơ bản của bigdata

- (1) Khối lượng lớn (Volume): Khối lượng dữ liệu rất lớn và đang ngày càng tăng lên, tính đến 2014 thì có thể trong khoảng vài trăm terabyte.
- (2) Tốc độ (Velocity): Khối lượng dữ liệu gia tăng rất nhanh.
- (3) Đa dạng (Variety): Ngày nay hơn 80% dữ liệu được sinh ra là phi cấu trúc (tài liệu, blog, hình ảnh,...)
- (4) Độ tin cậy/chính xác (Veracity): Bài toán phân tích và loại bỏ dữ liệu thiếu chính xác và nhiễu đang là tính chất quan trọng của bigdata.
- (5) Giá trị (Value): Giá trị thông tin mang lại.

3. Tổng quan về hadoop

- Apache Hadoop là một framework hỗ trợ cho việc lưu trữ và xử lý dữ liệu phân tán trên nhiều máy.
- Thành phần :



- + Hadoop Common
- + HDFS:Hệ thống file phân tán
- + Hadoop YARN Framework dùng cho lập lịch job và quản lý tài nguyên của hệ thống
- + MapReduce : Hệ thống xử lý dữ liệu của Hadoop

4. Tổng quan về MapReduce

- MapReduce là một mô hình được Google phát triển độc quyền, nhằm mục đích xử lý các dữ liệu lớn theo hướng phân tán và song song thuật toán trong một cụm máy MapReduce gồm 2 thủ tục chính:

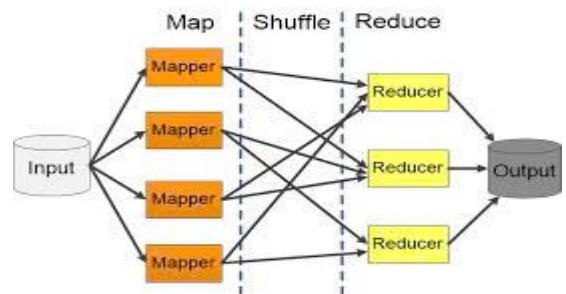
 - + Thủ tục Map: Có vai trò lọc và phân loại dữ liệu
 - + Thủ tục Reduce: Tổng hợp dữ liệu

B1: Phân rã từ nghiệp vụ chính (do người dùng muốn thể hiện) thành các công việc con để chia từng công việc con này về các máy tính trong hệ thống thực hiện xử lý một cách song song.

B2: Thu thập lại kết quả

Mô hình MapReduce:

- + Hàm Map : Hàm Map tiếp nhận mảnh dữ liệu input, rút trích thông tin cần thiết các từ trong phần tử (ví dụ: lọc dữ liệu, hoặc trích dữ liệu) tạo kết quả trung gian
- + Shuffle: Dữ liệu đầu ra từ bước Map được sắp xếp và gom nhóm để chuẩn bị cho bước Reduce
- + Hàm Reduce: tổng hợp kết quả trung gian, tính toán để cho kết quả cuối cùng

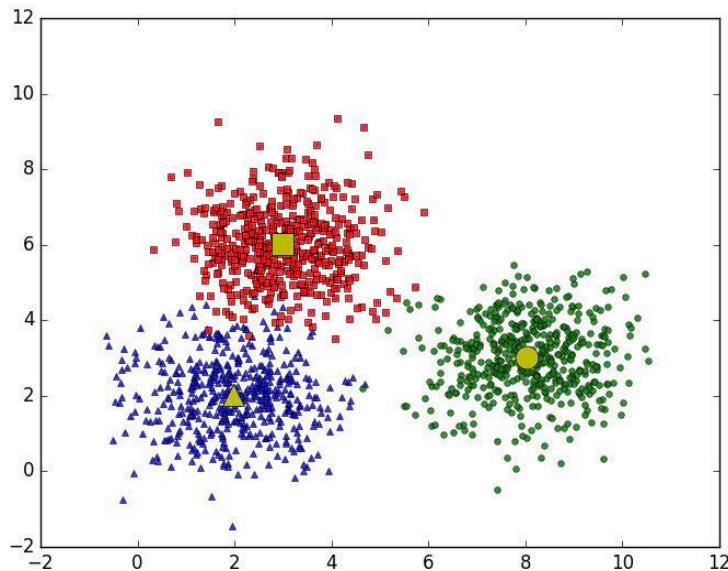


CHƯƠNG 2 THUẬT TOÁN PHÂN CỤM K-MEANS

1. Giới thiệu thuật toán K-means clustering

- Thuật toán K-means được giới thiệu năm 1957 bởi Stuart Lloyd và là phương pháp phổ biến nhất cho việc phân cụm ,dựa trên việc phân vùng dữ liệu
- Một trong những thuật toán cơ bản của Unsupervised Learning
- Chúng ta không biết nhãn (label) của từng điểm dữ liệu. Mục đích là làm thế nào để phân dữ liệu thành các cụm (cluster) khác nhau sao cho *dữ liệu trong cùng một cụm có tính chất giống nhau*.
- Biểu diễn dữ liệu: $D = \{x_1, x_2, x_3, \dots, x_r\}$ với x_i là vector n chiều trong không gian Euclidean. K-means phân cụm D thành K cụm dữ liệu:
 - + Mỗi cụm có một điểm trung tâm gọi là centroid

- + K là một hằng số cho trước
- Ý tưởng: Chia 1 bộ dữ liệu thành các cụm khác nhau
- Ví dụ về 3 cụm dữ liệu:



2. Triển khai thuật toán K-means clustering

Input: Dữ liệu $X = \{x_1, x_2, x_3, \dots, x_N\} \in \mathbb{R}^{d \times N}$ (N điểm dữ liệu, mỗi điểm có d chiều)

Số lượng cluster K: Số cụm cần tìm ($K < N$).

Output: Centers M: Tập hợp các center cuối cùng của K cụm, được cập nhật đến khi hội tụ

Label vector Y = (y_1, y_2, \dots, y_N) với mỗi y_i là một vector one-hot chỉ định cluster của x_i

Các bước thực hiện:

Bước 1: Chọn ngẫu nhiên K điểm trong dữ liệu X làm các center ban đầu $M^{(0)}$

Bước 2: Với mỗi điểm dữ liệu x_i , tính khoảng cách đến tất cả các center hiện tại. Gán điểm x_i vào cụm có center gần nhất:

$$j = \arg \min_j \|x_i - m_j\|_2^2$$

Bước 3: Cập nhật center Với mỗi cụm k, tính center mới m_k bằng trung bình cộng của tất cả các điểm dữ liệu thuộc về cụm đó:

$$m_k = \frac{1}{|C_k|} \sum_{x_i \in C_k} x_i$$

Trong đó, C_k là tập hợp các điểm dữ liệu thuộc cụm k, và $|C_k|$ là số điểm trong cụm.

Bước 4: Kiểm tra tính hội tụ.Nếu việc gán dữ liệu Y không thay đổi so với vòng lặp trước, hoặc centers M không thay đổi đáng kể, dừng thuật toán

Bước 5: Nếu chưa hội tụ quay lại bước 2

3. Ví dụ về một vòng lặp của thuật toán KMeans:

Giả sử ta có 6 điểm dữ liệu tương ứng với 6 điểm ảnh:

Point 1: (255, 0, 0)

Point 2: (254, 1, 2)

Point 3: (0, 255, 0)

Point 4: (1, 254, 1)

Point 5: (0, 0, 255)

Point 6: (2, 3, 250)

Với K = 3 tương ứng với 3 tâm được khởi tạo:

Centroid 1: (200, 0, 0)

Centroid 2: (0, 200, 0)

Centroid 3: (0, 0, 200)

Sử dụng khoảng cách Euclidean để tính khoảng cách giữa các điểm ảnh và tâm .

	Centroid 1: (200, 0, 0)	Centroid 2: (0, 200, 0)	Centroid 3: (0, 0, 200)
Point 1: (255, 0, 0)	55.0	320.2	320.2
Point 2: (254, 1, 2)	54.2	319.1	318.1
Point 3: (0, 255, 0)	320.2	55	320.2
Point 4: (1, 254, 1)	318.4	54.2	318.1
Point 5: (0, 0, 255)	320.2	320.2	55
Point 6: (2, 3, 250)	318.3	319.1	54.2

- Phân các điểm dữ liệu về các cụm:
 - Centroid 1: (Point 1, Point 2)

- Centroid 2: (Point 3, Point 4)
- Centroid 3: (Point 5, Point 6)

Tính toán Centroid mới:

$$\text{Centroid 1 mới : } \left(\frac{255+254}{2}, \frac{0+1}{2}, \frac{0+2}{2} \right) = (254.5, 0.5, 1.0)$$

$$\text{Centroid 2 mới : } \left(\frac{0+1}{2}, \frac{255+254}{2}, \frac{0+1}{2} \right) = (0.5, 254.5, 0.5)$$

$$\text{Centroid 3 mới : } \left(\frac{0+2}{2}, \frac{0+3}{2}, \frac{255+250}{2} \right) = (1.0, 1.5, 252.5)$$

CHƯƠNG 3 ỨNG DỤNG CỦA MAPREDUCE VÀ K-MEANS TRONG PHÂN CỤM ẢNH

1. Giới thiệu bài toán.

- Trong bài tập lần này chúng em áp dụng thuật toán K Means cho phân cụm một hình ảnh bất kỳ. Mục tiêu của bài toán này là giảm một bức ảnh có hàng triệu màu xuống chỉ còn K màu sắc tương ứng với K cụm sao cho giữ được đặc trưng của ảnh.

Input: Một hình ảnh hoặc một tập hình ảnh

Output: Một hình ảnh hoặc một tập các hình ảnh đã được phân cụm thành K cụm màu sắc.

❖ Ứng dụng của việc phân cụm ảnh:

- Giảm số lượng màu sắc (Color Quantization): Giảm số lượng màu trong ảnh từ hàng triệu màu xuống một số lượng giới hạn K, giúp giảm kích thước ảnh hoặc tối ưu hóa hiệu suất hiển thị.
- Chuẩn bị dữ liệu cho các thuật toán máy học cần ít đặc trưng hơn.
- Nhận diện đối tượng (Object detection).
- Tách nền trong ảnh (Background removal).
- Hỗ trợ phân tích y tế (ví dụ: phân vùng tế bào trong ảnh X-quang).
- Nhóm các vùng trong ảnh có cùng đặc điểm để nhận diện các vùng có ý nghĩa, chẳng hạn như bầu trời, cây cối, hoặc đường phố.
- Phân loại cảnh quan (Scene classification).
- Phân tích vệ tinh (Satellite image analysis).
- Tiền xử lý cho các thuật toán thị giác máy tính.
- Giảm dữ liệu đầu vào và trích xuất các đặc trưng quan trọng giúp giảm tài nguyên tính toán và giảm bộ nhớ cần thiết.

....

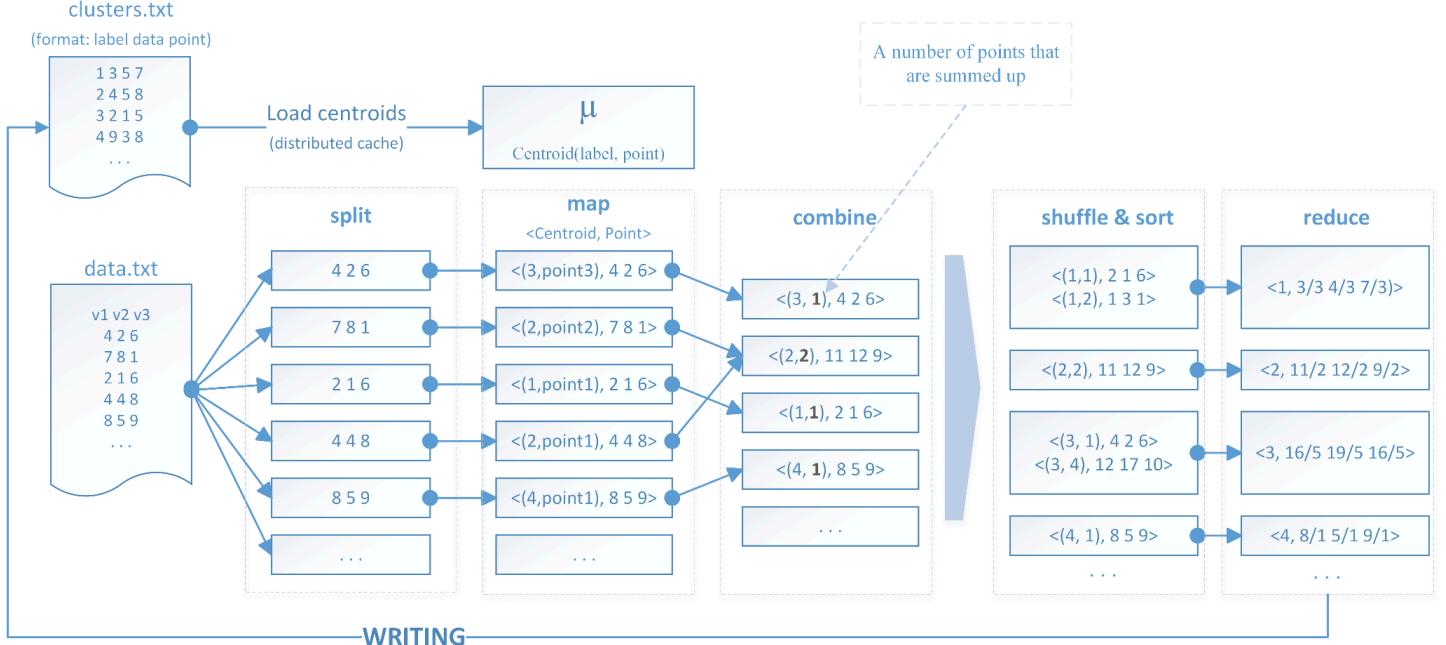
2. Cách tiếp cận.

2.1 Xử lý dữ liệu đầu vào.

- Từ 1 ảnh ban đầu được trích xuất ra các điểm ảnh thành một mảng 2 chiều với các hàng là 3 giá trị màu RGB lưu lại dưới dạng file txt : *points.txt*
- Khởi tạo K tâm cụm ngẫu nhiên với K là số màu muốn giảm xuống. Lưu các tâm cụm khởi tạo ngẫu nhiên dưới dạng file txt: *clusters.txt*
- Làm tương tự với tập các ảnh mỗi ảnh sẽ có một file *points.txt* và *clusters.txt* tương ứng. 2 file này sẽ làm dữ liệu đầu vào cho thuật toán K-means MapReduce.

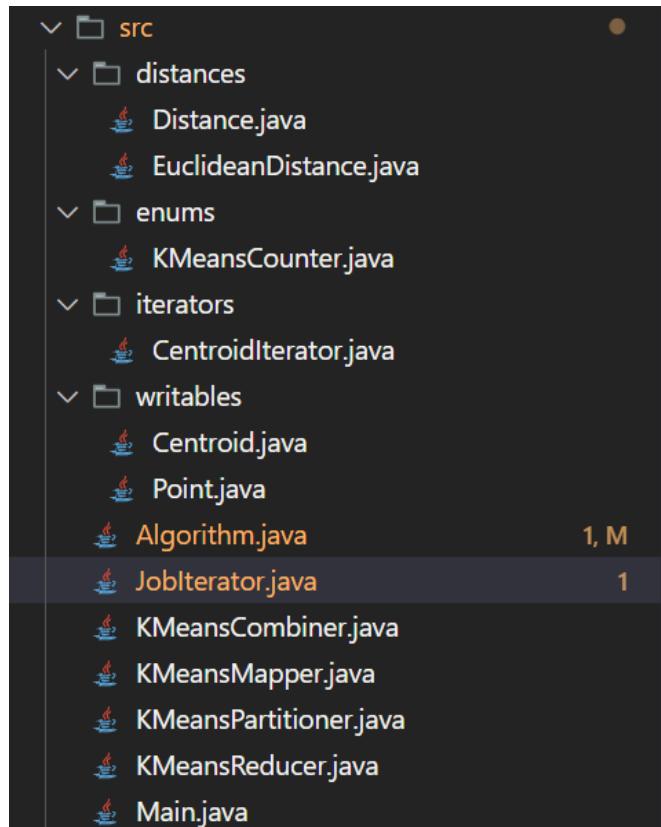
2.2 Thuật toán K Means.

- Dữ liệu đầu vào sẽ được tải lên hệ thống phân tán HDFS.
- Pha Map ban đầu sẽ chia các điểm dữ liệu trong *clusters.txt* và *points.txt* thành các cặp <Key, Value> với Key là các tâm cụm (Centroid) được khởi tạo ban đầu trong *clusters.txt* mỗi points sẽ được khởi tạo có dạng (label, point) và Value là giá trị Point gần nhất với Centroid đó được tính bằng khoảng cách Euclidean. Đầu ra của file pha Map là chia dữ liệu thành các cặp <Centroid, Point>.
- Tiếp theo là Pha Combiner: Để giảm điểm dữ liệu từ Map tới Reduce bằng cách tính toán tổng giá trị các Points có cùng Centroid và số lượng Points nằm trong Centroid đó. VD: <(Centroid, Point.number), (sum_x, sum_y, sum_z)>
- Shuffle and Sort được Hadoop thực hiện tự động sắp xếp các điểm dữ liệu theo Centroid.
- Tiếp đến là pha Reduce tính toán Centroid mới bằng cách tính trung bình các điểm dữ liệu hiện tại và so sánh xem đã hội tụ chưa bằng cách tính hiệu giữa Centroid cũ và Centroid mới so sánh với tham số *delta*. Mỗi Reducer sẽ xử lý 1 phân cụm
- Lặp lại các bước trên cho tới khi hội tụ hoặc vượt quá lần lặp (max_inter).



3. Triển khai

3.1: Thuật toán KMean Mapreduce



Cấu trúc thư mục.

- *Algorithm.java*: Kiểm tra thuật toán đã hội tụ hoặc vượt qua số vòng lặp khởi tạo chưa nếu chưa khởi tạo 1 JobIterator và chạy tiếp.
- *JobIterator.java*: Xử lý một vòng lặp cập nhật Centroid dựa trên output của vòng lặp trước, xóa output cũ trước khi thực hiện. Nếu là vòng lặp đầu tiên thì sẽ lấy dữ liệu từ file đầu vào (*clusters.txt* và *points.txt*)
- *KMeansMapper*: Thực hiện pha Map như mô tả trên.
- *KMeansCombiner*: Thực hiện pha Combiner.
- *KMeansPartitioner*: Thực hiện phân tách Key
- *KMeansReducer*: Thực hiện pha Reduce như mô tả.
- *Distance*: Tính toán khoảng cách giữa Centroid và Point
- *CentroidIteration*: Tính toán các điểm gần nhất với từng Centroid
- *Centroid*: Lớp chứa thông tin về Centroid như tọa độ nhãn
- *Point*: Lớp chứa thông tin về Point như tọa độ, số lượng Point nằm trong 1 cụm

3.2 Chuẩn bị dữ liệu:

- Với mỗi hình ảnh các điểm dữ liệu sẽ được lưu lại dưới file *points.txt* và các cụm sẽ được khởi tạo ngẫu nhiên. mỗi point sẽ là một điểm ảnh gồm 3 giá trị màu RGB

```

points_path = join(dst_folder_points, f'points_{i}.txt')
clusters_path = join(dst_folder_clusters, f'clusters_{i}.txt')

# load and write points
img = cv2.imread(src_img_path).reshape((-1, 3)).astype(np.float32)
with open(points_path, 'w') as f:
    f.write(nparray_to_str(img))
print(f'Points saved in: {points_path}')

```

Ví dụ về một số point:

```

226.0 226.0 226.0
228.0 228.0 228.0
230.0 230.0 230.0
230.0 230.0 230.0
231.0 231.0 231.0
232.0 232.0 232.0

```

- Sinh ngẫu nhiên K phân cụm mỗi cụm có dạng giống như một point và gán nhãn là 1 số nguyên với low và high là số pixel nhỏ nhất và cao nhất có trong ảnh.

```

s = np.random.uniform(low=img.min(), high=img.max(), size=(k, 3))
tmp_labels = np.arange(1, k + 1).reshape((k, 1))
clusters = np.hstack((tmp_labels, s))

with open(clusters_path, 'w') as f:
    f.write(nparray_to_str(clusters))
    print(f'Centroids saved in: {clusters_path}')

```

- Ví dụ về 10 cụm được sinh ngẫu nhiên

```

1.0 123.43619335929367 14.544450204433163 45.41094346453308
2.0 37.947022164854125 66.80143896455897 144.05882094141325
3.0 169.98683945923315 101.81618156691214 97.99323251316115
4.0 79.58871135641203 121.36745539716324 47.41490332492972
5.0 191.9175692008476 85.78895243032021 206.30861738296528
6.0 102.9980211447009 187.4671552673256 83.31649835119846
7.0 164.9333189146586 156.55082121944497 21.2137413004920250
8.0 233.55506214545616 8.271447864458288 136.60715169926635
9.0 49.91910366389993 143.44638003661234 134.85143983766545
10.0 54.36750089455553 217.0868340166224 42.85651392353406

```

4. Thực nghiệm

- Dưới đây là kết quả bọn em đã chạy thực nghiệm:
- link github: https://github.com/anhduc1234567/bigdata_final_project

Job ID	Application Priority	StartTime	LaunchTime	FinishTime	State	FinalStatus	Running Containers	Allocated CPU Vcores	Allocated Memory MB	Allocated GPUs	Reserved CPU Vcores	Reserved Memory MB	Reserved GPUs	% of Queue	% of Cluster	Progress
sfault	0	Thu Dec 5 14:46:08 +0700 2024	Thu Dec 5 14:46:09 +0700 2024	Thu Dec 5 14:46:26 +0700 2024	FINISHED	SUCCEEDED	N/A	N/A	N/A	N/A	N/A	N/A	N/A	0.0	0.0	<div style="width: 100%; background-color: #ccc;"></div>
sfault	0	Thu Dec 5 14:45:37 +0700 2024	Thu Dec 5 14:45:38 +0700 2024	Thu Dec 5 14:45:56 +0700 2024	FINISHED	SUCCEEDED	N/A	N/A	N/A	N/A	N/A	N/A	N/A	0.0	0.0	<div style="width: 100%; background-color: #ccc;"></div>
sfault	0	Thu Dec 5 14:45:05 +0700 2024	Thu Dec 5 14:45:06 +0700 2024	Thu Dec 5 14:45:24 +0700 2024	FINISHED	SUCCEEDED	N/A	N/A	N/A	N/A	N/A	N/A	N/A	0.0	0.0	<div style="width: 100%; background-color: #ccc;"></div>
sfault	0	Thu Dec 5 14:44:20 +0700 2024	Thu Dec 5 14:44:21 +0700 2024	Thu Dec 5 14:44:52 +0700 2024	FINISHED	SUCCEEDED	N/A	N/A	N/A	N/A	N/A	N/A	N/A	0.0	0.0	<div style="width: 100%; background-color: #ccc;"></div>
sfault	0	Thu Dec 5 14:43:20 +0700 2024	Thu Dec 5 14:43:21 +0700 2024	Thu Dec 5 14:43:55 +0700 2024	FINISHED	SUCCEEDED	N/A	N/A	N/A	N/A	N/A	N/A	N/A	0.0	0.0	<div style="width: 100%; background-color: #ccc;"></div>
sfault	0	Thu Dec 5 14:42:14 +0700 2024	Thu Dec 5 14:42:14 +0700 2024	Thu Dec 5 14:42:49 +0700 2024	FINISHED	SUCCEEDED	N/A	N/A	N/A	N/A	N/A	N/A	N/A	0.0	0.0	<div style="width: 100%; background-color: #ccc;"></div>
sfault	0	Thu Dec 5 14:40:54 +0700 2024	Thu Dec 5 14:40:59 +0700 2024	Thu Dec 5 14:41:55 +0700 2024	FINISHED	SUCCEEDED	N/A	N/A	N/A	N/A	N/A	N/A	N/A	0.0	0.0	<div style="width: 100%; background-color: #ccc;"></div>
sfault	0	Wed Dec 4 20:58:16 +0700 2024	Wed Dec 4 20:58:17 +0700 2024	Wed Dec 4 20:58:56 +0700 2024	FINISHED	SUCCEEDED	N/A	N/A	N/A	N/A	N/A	N/A	N/A	0.0	0.0	<div style="width: 100%; background-color: #ccc;"></div>
sfault	0	Wed Dec 4 20:57:31 +0700 2024	Wed Dec 4 20:57:32 +0700 2024	Wed Dec 4 20:58:02 +0700 2024	FINISHED	SUCCEEDED	N/A	N/A	N/A	N/A	N/A	N/A	N/A	0.0	0.0	<div style="width: 100%; background-color: #ccc;"></div>
sfault	0	Wed Dec 4 20:56:47 +0700 2024	Wed Dec 4 20:56:47 +0700 2024	Wed Dec 4 20:57:13 +0700 2024	FINISHED	SUCCEEDED	N/A	N/A	N/A	N/A	N/A	N/A	N/A	0.0	0.0	<div style="width: 100%; background-color: #ccc;"></div>
sfault	0	Wed Dec 4 -----	Wed Dec 4 -----	Wed Dec 4 -----	FINISHED	SUCCEEDED	N/A	N/A	N/A	N/A	N/A	N/A	N/A	0.0	0.0	<div style="width: 100%; background-color: #ccc;"></div>

Thông tin một số job được thực hiện qua giao diện của Hadoop

- Hình ảnh ban đầu:



- Khi khởi tạo 10 cụm ngẫu nhiên:

Hình ảnh trước khi chạy MapReduce phân cụm:



Hình ảnh sau khi chạy MapReduce với số lần lặp 30:



- Quan mặc dù chỉ với 10 cụm tuy nhiên hình ảnh được hiển thị với màu sắc rất tốt gần giống với ảnh gốc.

CHƯƠNG 4 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

1. Ý nghĩa của phân cụm ảnh

Phân cụm ảnh (image clustering) trong Big Data là một phương pháp tổ chức và phân nhóm các hình ảnh dựa trên đặc điểm hoặc tính chất tương đồng của chúng mà không cần thông tin nhãn trước. Đây là một nhiệm vụ quan trọng trong xử lý dữ liệu lớn, đặc biệt là khi dữ liệu không có cấu trúc như hình ảnh.

- Tổ chức và cấu trúc hóa dữ liệu phi cấu trúc:

- Hình ảnh là dạng dữ liệu phi cấu trúc, khó xử lý trực tiếp. Phân cụm biến tập dữ liệu không có tổ chức thành các nhóm hợp lý, dựa trên các tiêu chí tương đồng về nội dung hoặc đặc điểm (màu sắc, hình dạng, hoa văn,...).
 - Ví dụ: Trong hệ thống lưu trữ hình ảnh của mạng xã hội, phân cụm có thể nhóm các hình ảnh thành các cụm như “chân dung”, “phong cảnh”, “đồ ăn”.
- Hỗ trợ phân tích xu hướng và mẫu hình (patterns):
 - Tìm ra các xu hướng tiềm ẩn, chẳng hạn như các loại hình ảnh thường xuất hiện cùng nhau trong một bối cảnh nào đó.
 - Ví dụ: Trong thương mại điện tử, các hình ảnh sản phẩm giống nhau về mẫu mã có thể thuộc cùng một dòng sản phẩm.
- Hỗ trợ học sâu (Deep Learning):
 - Phân cụm cung cấp dữ liệu có tổ chức để huấn luyện các mô hình học sâu hiệu quả hơn, đặc biệt là trong các bài toán semi-supervised hoặc unsupervised (học bán giám sát hoặc không giám sát).

- Ứng dụng đặc biệt trong lĩnh vực chuyên biệt:
 - Y tế: Nhóm các hình ảnh chụp X-quang, MRI để hỗ trợ bác sĩ nhận biết bệnh.
 - An ninh: Phân nhóm ảnh từ camera giám sát để phát hiện hành vi bất thường.

2. Hướng phát triển thêm của phân cụm ảnh trong Big Data:

- Tích hợp học sâu (Deep Learning): Kết hợp các mô hình học sâu như Autoencoder, Convolutional Neural Networks (CNNs) với phân cụm để cải thiện chất lượng.
- Học tự giám sát (Self-supervised Learning): Sử dụng các phương pháp như Contrastive Learning để cải thiện phân cụm khi dữ liệu không có nhãn.
- Phân cụm theo ngữ cảnh: Kết hợp thông tin từ nhiều nguồn (như văn bản, metadata) để phân cụm ảnh chính xác hơn.
- Ứng dụng thời gian thực: Nghiên cứu các giải thuật phân cụm nhanh để xử lý luồng dữ liệu hình ảnh theo thời gian thực.
- Tăng cường hiệu năng với phân tán: Áp dụng các công cụ như Apache Spark hoặc Hadoop để xử lý tập dữ liệu cực lớn.
- Xử lý dữ liệu đa phương thức: Kết hợp ảnh với các loại dữ liệu khác như âm thanh, video để tăng tính liên kết và hiệu quả.
- Phân cụm theo ý nghĩa sâu hơn (Semantic Clustering): Sử dụng các mô hình ngôn ngữ lớn (như CLIP của OpenAI) để phân cụm dựa trên ý nghĩa hơn là đặc điểm vật lý.