Thuật toán phân cụm dựa trên tìm kiếm Tabu và triển khai song song trên Spark

Nhóm 5: Nguyễn Ngọc Anh – Lê Hằng Anh
Giảng viên: TS. Vũ Tiến Dũng

Tháng 3 năm 2024

Mục lục

Giới thiệu chung

Giới thiệu chung

- Trong số các phương pháp phân cụm, thuật toán phân cụm K–means là một trong những phương pháp phổ biến và được sử dụng rộng rãi nhất.
- Thuật toán K–means phụ thuộc vào việc khởi tạo một tập hợp các điểm trung tâm ban đầu để bắt đầu các bước tiếp theo, gán các đối tượng vào các điểm trung tâm này và sau đó xác định các điểm trung tâm mới. Sự lựa chọn của các điểm trung tâm ban đầu này ảnh hưởng đáng kể đến cấu trúc và chất lượng của các cụm cuối cùng được tạo ra.

Giới thiệu chung

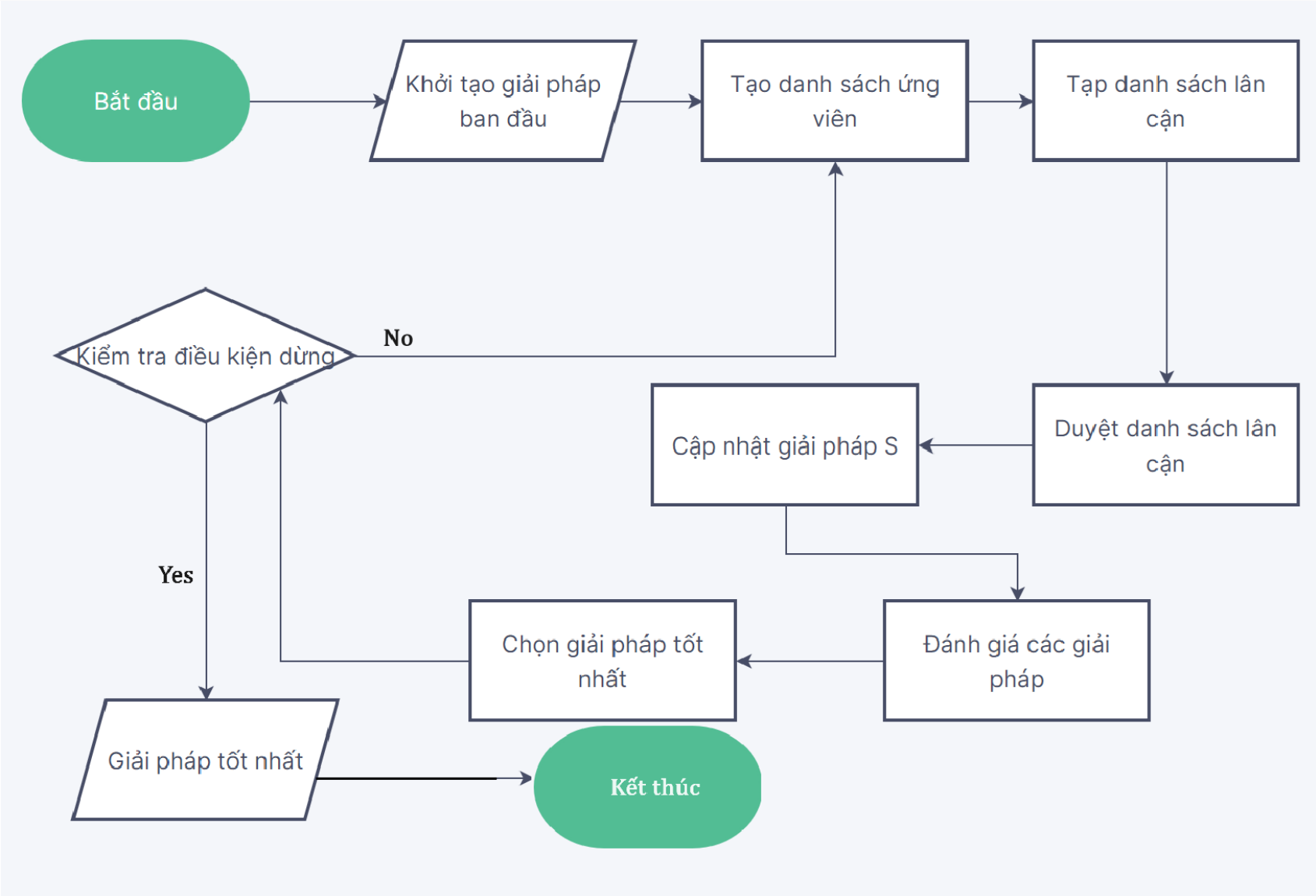
- Thuật toán Tabu Search được tạo bởi Fred Glover vào năm 1986, là một phương pháp tìm kiếm siêu mô phỏng sử dụng các phương pháp tìm kiếm cục bộ để tối ưu hóa.
- Tabu Search là một meta-heuristic hướng dẫn quy trình tìm kiếm theo phương pháp heuristic cục bộ để khám phá ra giải pháp vượt ngoài mức tối ưu cục bộ bằng cách sử dụng Tabu List

Giới thiệu chung

- Tabu Search có ba chiến lược chính:
  - Forbidding strategy: kiểm soát những gì được đưa vào Tabu List
  - Freeing strategy: kiểm soát những gì được đưa ra khỏi Tabu List
  - Short-term Stategy: quản lý tác động giữa 2 strategies trên để lựa chọn ra các giải pháp thử nghiệm

Thuật toán Tabu Search

Sơ đồ thuật toán



Xác định khu vực lân cận

- Mục đích của việc xác định khu vực lân cận trong Tabu Search là tạo ra một tập hợp các giải pháp lân cận hợp lệ để khám phá và cải thiện giải pháp hiện tại, đồng thời tránh lặp lại các bước di chuyển đã thực hiện. Điều này giúp thuật toán Tabu Search tiến gần hơn đến giải pháp tối ưu trong không gian tìm kiếm.
- Khu vực lân cận trong Tìm Kiếm Tabu là một phần của không gian giải pháp liên quan đến giải pháp hiện tại. Nó định nghĩa các bước di chuyển có thể được thực hiện để tạo ra một giải pháp mới trong lần lặp tiếp theo.

Xác định khu vực lân cận

- Khu vực lân cận được định nghĩa bằng cách tạo ra một vùng tròn xung quanh tâm cụm. Một phần tử trong vùng này được chọn làm tâm cụm mới và tất cả các phần tử còn lại được gán vào tâm cụm gần nhất để tạo ra một giải pháp lân cận.
- Tổng các khu vực lân cận của các cụm được kết hợp để tạo thành khu vực lân cận toàn cục.

Danh sách Tabu - Tabu list

- Tabu List trong Tabu Search có nhiệm vụ hướng dẫn quá trình tìm kiếm bằng cách cấm hoặc giới hạn sử dụng các bước di chuyển đã thực hiện trước đó. Nó đảm bảo rằng thuật toán không lặp lại các bước di chuyển đã được thực hiện gần đây và khám phá các giải pháp mới trong không gian tìm kiếm.

Danh sách Tabu - Tabu list

- Các yếu tố chính trong Tabu List bao gồm:
  - Bước di chuyển Tabu: Các bước di chuyển đã được thực hiện gần đây được đánh dấu là "Tabu" và không được thực hiện lại trong một khoảng thời gian nhất định.
  - Thời gian Tabu: Mỗi bước di chuyển Tabu có một thời gian Tabu tương ứng, là số lần lặp hoặc thời gian quy định mà bước di chuyển đó sẽ bị cấm.
  - Cấu trúc Tabu List: Tabu List có thể được cài đặt dưới dạng một danh sách lưu trữ các bước di chuyển Tabu hoặc dưới dạng một bảng băm để kiểm tra tính Tabu của một bước di chuyển nhanh chóng.

Danh sách ứng viên - Candidate list

- Ý chính của Candidate List là tạo ra một danh sách nhỏ hơn của khu vực lân cận để giảm công sức tính toán khi xem xét toàn bộ khu vực lân cận. Thuật toán này tập trung vào những điểm dữ liệu có khả năng cung cấp giải pháp tốt hơn.

Danh sách ứng viên - Candidate list

- Các yếu tố chính của Candidate List bao gồm:
  - Tạo khu vực lân cận cho mỗi cụm: Với mỗi cụm, thuật toán tạo khu vực lân cận dựa trên bán kính đã xác định.
  - Tính tổng khoảng cách: Thuật toán duyệt qua tất cả các cụm và tính tổng khoảng cách dựa trên công thức đã cho, sử dụng các điểm trong khu vực lân cận ngoại trừ tâm cụm.
  - Tạo danh sách ứng viên: để giảm công sức kiểm tra toàn bộ khu vực lân cận. Điều này được thực hiện bằng cách sắp xếp các điểm dữ liệu trong khu vực lân cận theo tổng khoảng cách, xác định số lượng điểm dữ liệu trong danh sách ứng viên cho mỗi cụm, lựa chọn các điểm dữ liệu đầu tiên và kết hợp các danh sách ứng viên từ các cụm.

Tạo tâm cụm đầu tiên

- Mục đích của việc tạo tâm cụm đầu tiên trong Tabu Search là khởi tạo một giải pháp ban đầu cho bài toán tối ưu. Phương pháp dựa trên việc xác định các điểm có khoảng cách lớn nhất từ các tâm cụm hiện có, sử dụng tổng khoảng cách từ các tâm cụm để đánh giá. Khác với phương pháp K-means, trong thuật toán của bài báo, các tâm cụm thực tế được chọn là các điểm dữ liệu cụ thể

Tạo tâm cụm đầu tiên

- Quá trình tạo tâm cụm đầu tiên bao gồm các bước sau:
  - Chọn ngẫu nhiên một điểm dữ liệu làm tâm cụm đầu tiên. Điều này có thể được thực hiện bằng cách chọn một điểm ngẫu nhiên từ tập hợp các điểm dữ liệu.
  - Đặt điểm dữ liệu được chọn làm tâm cụm đầu tiên vào tập hợp các tâm cụm. Điểm này sẽ đại diện cho cụm đầu tiên.
  - Tạo một tập hợp khác chứa tất cả các điểm dữ liệu trừ điểm đã được chọn làm tâm cụm đầu tiên. Điều này đảm bảo rằng các điểm dữ liệu còn lại sẽ được xem xét để tạo các tâm cụm tiếp theo.

Thuật toán Tabu Search

- Thuật toán Tabu Search dựa trên nhóm cụm có các bước chính sau:
  - Xây dựng giải pháp ban đầu.
  - Thực hiện thủ tục Tìm kiếm Tabu:
    - Duyệt qua từng cụm và tạo danh sách ứng viên cho từng cụm.
    - Duyệt qua danh sách ứng viên và thử chọn mỗi điểm dữ liệu làm tâm cụm mới.
    - Cập nhật giải pháp bằng cách gán các điểm dữ liệu không phải là tâm cụm cho tâm gần nhất.
    - Kiểm tra điều kiện Tabu và các điều kiện cải thiện giải pháp.
    - Cập nhật giải pháp tốt nhất và giải pháp tốt nhất từ trước đến nay.
    - Kiểm tra điều kiện dừng vòng lặp.

Thuật toán Tabu Search

- Chuyển đổi giữa đa dạng hóa và tập trung hóa:
  - Kiểm tra nếu cả đa dạng hóa và tập trung hóa không cải thiện giải pháp tốt nhất.
  - Chuyển đổi sang chiến lược đa dạng hóa hoặc tập trung hóa dựa trên giải pháp hiện tại.
  - Thiết lập lại các giá trị và quay lại thủ tục Tìm kiếm Tabu.

Triển khai song song

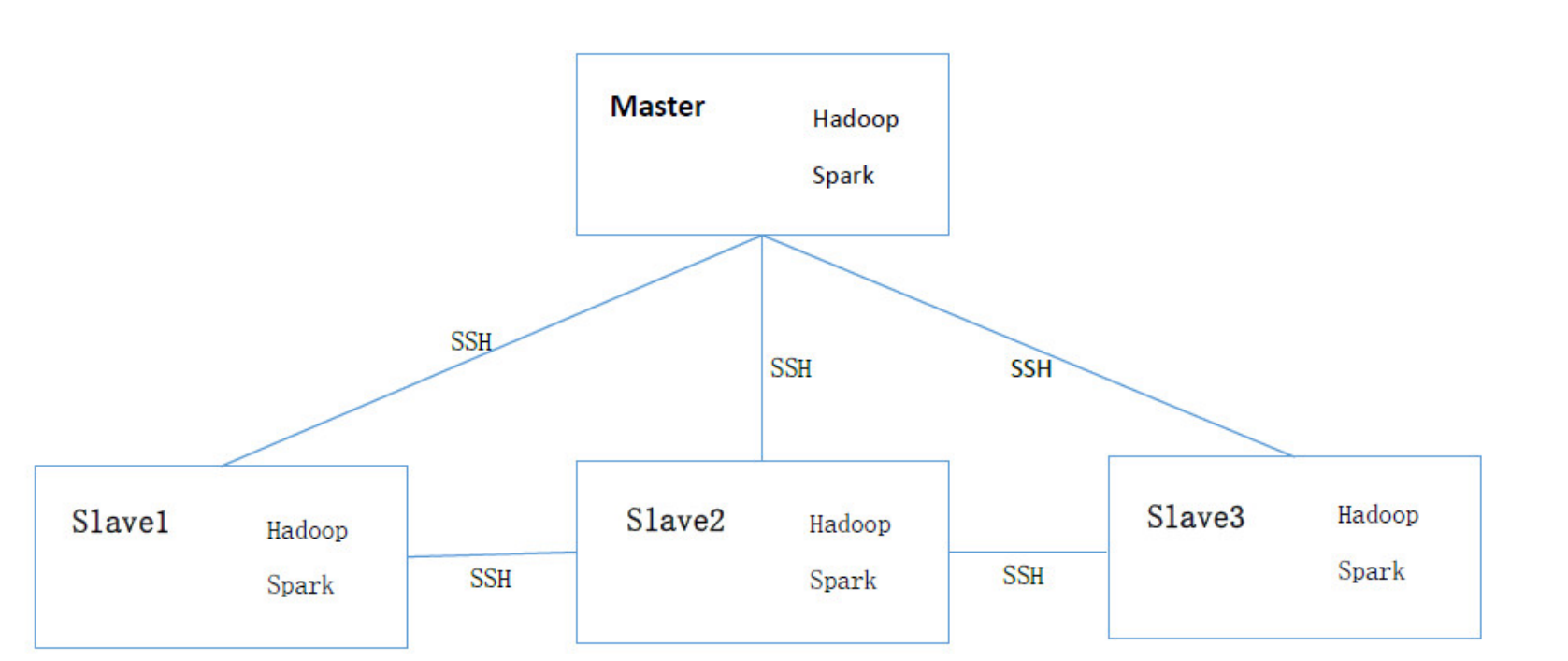
Song song hóa

- Tập trung vào việc song song hóa hai thành phần chính của thuật toán:
  - Quá trình gán lại các điểm dữ liệu cho các cụm
  - Quá trình cập nhật tâm cụm của mỗi cụm.

Song song hóa

- Sử dụng thao tác Map–Reduce trong môi trường tính toán song song, bộ dữ liệu được chia thành các khối nhỏ và xử lý đồng thời, giúp giảm thời gian tính toán.
- Quá trình reduce sử dụng kết quả đầu ra của map để thu thập điểm dữ liệu liên quan, tính toán bán kính phạm vi của cụm và thu thập danh sách ứng viên cho chiến lược tập trung hoặc đa dạng hóa.
- Trong mỗi vòng lặp, nhiều thao tác Map–Reduce được chạy đồng thời trên các bộ dữ liệu đã chia nhỏ và kết quả từ các quá trình Map–Reduce riêng lẻ được kết hợp để chọn giải pháp tốt nhất cho vòng lặp hiện tại.

Triển khai song song trên Spark



Thực nghiệm

Thông tin các tập dữ liệu

Dựa theo hướng dẫn từ bài báo, thuật toán thực hiện thử nghiệm trên hai tập dữ liệu Iris và Wine được lấy từ nguồn UCI (`https://archive.ics.uci.edu/datasets`), đã có trong thư viện **sklearn.datasets** của Python. Các trường dữ liệu được trình bày trong slide tiếp theo.

Các trường dữ liệu trong tập dữ liệu Iris

Iris là tập dữ liệu về các loại hoa lan, ta sẽ phân nhóm các giống hoa lan theo độ dài và độ rộng của cánh hoa và đài hoa.

- Số lượng quan sát: 150;
- Bộ dữ liệu có 5 trường dữ liệu như sau
  - SepalLengthCm: Độ dài của đài hoa (đơn vị cm);
  - SepalWidthCm: Độ rộng của đài hoa (đơn vị cm);
  - PetalLengthCm: Độ dài của cánh hoa (đơn vị cm);
  - PetalWidthCm: Độ rộng của cánh hoa (đơn vị cm);
  - Species: Các giống hoa.

Các trường dữ liệu trong tập dữ liệu Wine

Wine là tập dữ liệu về thành phần có trong rượu để từ đó ta phân loại chúng thành các loại rượu khác nhau.

- Số lượng quan sát: 178;
- Bộ dữ liệu có 13 trường dữ liệu sau
  - Alcohol (Nồng độ cồn);
  - Malic acid (Axít malic);
  - Ash (Tro);
  - Alcalinity of ash (Độ kiềm của tro);
  - Magnesium (Magiê);
  - Total phenols (Tổng số phenol);

Các trường dữ liệu trong tập dữ liệu Wine

- Flavanoids (Flavonoid);
- Nonflavanoid phenols (Phenol không flavonoid);
- Proanthocyanins (Proanthocyanidin);
- Color intensity (Độ đậm màu);
- Hue (Độ màu);
- $OD280/OD315$ of diluted wines (Tỷ lệ quang học);
- Proline (Proline).

Kết quả thực nghiệm

Tập dữ liệu	Số vòng lặp	Số cụm	Độ chính xác	Thời gian
Iris	500	5	0.7012	4.2296
Iris	1000	6	0.6264	4.5120
Iris	1000	8	0.6211	4.5329
Iris	1000	7	0.9052	8.0982
Wine	500	5	0.5879	5.7653
Wine	1000	6	0.7157	12.2445
Wine	1000	7	0.7288	22.4329
Wine	1500	5	0.7239	16.9110

Độ phức tạp thời gian

Như thuật toán đã trình bày, thuật toán hoạt động dựa trên các thành phần

- Số vòng lặp N_i;
- Số lượng cụm N_s;
- Số lượng điểm N_p;
- Số lượng block song song N_{mr}.

Khi đó, nếu thuật toán có N_i bước lặp thì độ phức tạp thời gian (khi khai triển trên Spark) là

$$O\left(\frac{N_s N_p}{N_{mr}}\left(N_l\left(1+log(N_s)\right)+\frac{1}{2}\left(1+log\left(\frac{N_p}{2N_{mr}}\right)\right)\right)\right)$$

Kết luận

Kết luận

- Báo cáo đã trình bày các bước của thuật toán Tabu Search phiên bản tuần tự và phiên bản song song khai triển trên Spark;
- Thuật toán K–Means truyền thống có độ phức tạp thời gian là $O(kN_lN_sN_p)$, do vậy ta thấy thuật toán Tabu khai triển trên Spark hoạt động nhanh hơn.
- Với bộ dữ liệu Iris, do số lượng quan sát ít hơn nên thời gian thực thi nhanh hơn khi thực thi trên tập Wine và độ chính xác của thuật toán trên tập Iris tốt hơn độ chính xác của thuật toán trên tập Wine.

Nhóm 5 cảm ơn thầy cô và các bạn đã lắng nghe!