CÂU HỎI TRẮC NGHIỆM ÔN TẬP

Câu 1: Hạn chế lớn nhất của các thuật toán phân cụm truyền thống (như DBSCAN gốc) khi áp dụng vào dữ liệu đô thị là gì?

A. Không xử lý được dữ liệu nhiễu.

B. Sử dụng khoảng cách Euclid, bỏ qua các ràng buộc vật lý như sông ngòi, đường sá.

C. Độ phức tạp tính toán quá cao.

D. Yêu cầu quá nhiều tham số đầu vào.

Câu 2: Trong báo cáo này, "Không gian mạng" (Network Space) được mô hình hóa dưới dạng cấu trúc toán học nào?

A. Đồ thị có hướng không trọng số.

B. Đồ thị vô hướng có trọng số.

C. Cây nhị phân tìm kiếm.

D. Ma trận kề.

Câu 3: Khoảng cách giữa hai điểm p_i và p_j trong không gian mạng ($dist_N$) được định nghĩa là:

A. Khoảng cách đường chim bay (Euclid).

B. Tổng trọng số lớn nhất của các cạnh nối hai điểm.

C. Tổng trọng số nhỏ nhất của các cạnh tạo nên đường đi nối hai điểm đó.

D. Khoảng cách Manhattan.

Câu 4: Tập dữ liệu D trong mô hình mạng lưới bao gồm các điểm sự kiện nằm ở vị trí nào?

A. Chỉ nằm tại các đỉnh (Vertices) của đồ thị.

B. Nằm tại trọng tâm của đồ thị.

C. Nằm tại bất kỳ vị trí nào trên các cạnh (Edges), thay vì chỉ nằm tại đỉnh.

D. Nằm ngoài phạm vi đồ thị.

Câu 5: Vùng lân cận mạng lưới $N_{Eps}(p)$ có hình dạng đặc trưng nào?

A. Hình tròn bán kính Eps.

B. Hình vuông cạnh $2 \times Eps$.

C. Hình dạng lan tỏa dọc theo các con đường (dạng mạng nhện).

D. Hình cầu trong không gian 3 chiều.

Câu 6: Điểm lõi (Core Point) được định nghĩa là điểm có số lượng lân cận thỏa mãn điều kiện nào?

A. Nhỏ hơn $MinPts$.

B. Lớn hơn hoặc bằng $MinPts$.

C. Bằng đúng $MinPts$.

D. Lớn hơn Eps.

Câu 7: Một điểm được gọi là "Nhiễu" (Noise) khi nào?

A. Là điểm biên nhưng có mật độ cao.

B. Không phải điểm lõi và cũng không phải điểm biên.

C. Là điểm lõi nhưng nằm ở rìa đồ thị.

D. Có khoảng cách đến tâm lớn hơn Eps.

Câu 8: Thách thức lớn nhất về hiệu năng của thuật toán NS-DBSCAN gốc là gì?

A. Thiếu độ chính xác.

B. Chi phí tính toán lớn do phải xác định vùng lân cận cho hàng nghìn điểm trên đồ thị phức tạp.

C. Không xác định được nhiễu.

D. Khó cài đặt trên máy tính.

Câu 9: Tên viết tắt của thuật toán cải tiến được đề xuất trong báo cáo là gì?

A. F-DBSCAN.

B. iNS-DBSCAN.

C. Fast-DBSCAN.

D. Net-DBSCAN.

Câu 10: Chiến lược tối ưu hóa đầu tiên liên quan đến "Lược bỏ cạnh" dựa trên định lý nào?

A. Định lý Pythagore.

B. Một đường đi hợp lệ trong lân cận Eps không thể chứa cạnh đơn lẻ có trọng số lớn hơn Eps.

C. Định lý đồ thị Euler.

D. Định lý giới hạn trung tâm.

Câu 11: Trong thuật toán LSPD cải tiến, bước "Tiền xử lý" thực hiện công việc gì?

A. Sắp xếp lại các đỉnh đồ thị.

B. Xóa toàn bộ các đỉnh cô lập.

C. Rà soát và vô hiệu hóa các cạnh có trọng số $W(e) > Eps$.

D. Tính toán lại toàn bộ trọng số cạnh.

Câu 12: Điều kiện "Ngắt sớm" trong thuật toán tìm kiếm lân cận cải tiến hoạt động như thế nào?

A. Dừng khi tìm thấy 1 điểm lân cận.

B. Dừng khi hết bộ nhớ.

C. Nếu khoảng cách tích lũy $d(p) \geq Eps$, không tiếp tục xét duyệt các cạnh kề của p.

D. Dừng sau 10 bước lặp.

Câu 13: Vấn đề hiệu năng của giai đoạn "Xây dựng bảng trật tự mật độ" trong thuật toán gốc là gì?

A. Tốn quá nhiều bộ nhớ lưu trữ đồ thị.

B. Phải gọi hàm sắp xếp lại (re-sorting) hàng đợi mỗi khi cập nhật hoặc thêm điểm mới.

C. Tính toán sai mật độ điểm.

D. Không sử dụng hàng đợi ưu tiên.

Câu 14: Trong thuật toán cải tiến, để tránh việc sắp xếp lại hàng đợi Q, nhóm tác giả sử dụng kỹ thuật gì?

A. Sử dụng QuickSort liên tục.

B. Chèn điểm vào vị trí chính xác để bảo toàn thứ tự giảm dần (Cơ chế chèn bảo toàn).

C. Không sắp xếp hàng đợi nữa.

D. Sử dụng mảng băm (Hash map).

Câu 15: Ngưỡng Heuristic δ được sử dụng để lọc bỏ các điểm có mật độ quá thấp được tính xấp xỉ bằng bao nhiêu?

A. $n/2$

B. $\sqrt{n}$

C. $\ln(n)$

D. n^2

Câu 16: Tại sao việc loại bỏ các điểm "Sparse points" (mật độ rất thấp) lại giúp tăng tốc thuật toán?

A. Vì chúng không bao giờ có thể trở thành điểm lõi để tạo cụm.

B. Vì chúng gây lỗi bộ nhớ.

C. Vì chúng làm sai lệch trọng số cạnh.

D. Vì chúng luôn là điểm biên.

Câu 17: Chiến lược thứ 3 để tối ưu hóa thuật toán liên quan đến quy trình nào?

A. Tính khoảng cách.

B. Xác định nhiễu (Noise).

C. Vẽ biểu đồ.

D. Nhập dữ liệu.

Câu 18: Trong thuật toán cải tiến, Nhiễu được xác định như thế nào?

A. Kiểm tra từng điểm bằng vòng lặp If-Else.

B. Xác định ngầm định: Những điểm còn sót lại không thuộc cụm nào sau khi phân cụm xong là Nhiễu.

C. Dựa vào bảng màu sắc.

D. Người dùng tự gán nhãn thủ công.

Câu 19: Việc loại bỏ kiểm tra nhiễu tường minh giúp ích gì cho mã giả (Pseudocode)?

A. Làm mã dài hơn nhưng dễ hiểu hơn.

B. Giúp loại bỏ các phép kiểm tra dư thừa trong vòng lặp, làm mã gọn nhẹ và chạy nhanh hơn.

C. Tăng độ phức tạp thuật toán.

D. Không có tác dụng gì đáng kể.

Câu 20: Trong thuật toán hình thành cụm cải tiến (Algorithm 3), dòng lệnh nào đã được loại bỏ so với bản gốc?

A. Vòng lặp duyệt qua các điểm.

B. Kiểm tra $Density(p) \geq MinPts$.

C. Kiểm tra `if p does not belong to any cluster or noises`.

D. Khởi tạo cụm C.

Câu 21: Dữ liệu thực nghiệm được trích xuất từ nguồn nào?

A. Google Maps.

B. Dữ liệu giả lập ngẫu nhiên.

C. OpenStreetMap (OSM).

D. Bản đồ giấy được số hóa.

Câu 22: Hai chỉ số chính dùng để đánh giá hiệu năng trong bài báo cáo là gì?

A. Độ chính xác (Accuracy) và F1-Score.

B. Thời gian thực thi (Time) và Mức độ chênh lệch hiệu năng (Diff %).

C. Dung lượng bộ nhớ (RAM) và CPU Usage.

D. Số lượng cụm và Số lượng điểm nhiễu.

Câu 23: Công thức tính Diff (%) là gì?

A. $\frac{Time\_Improved - Time\_Original}{Time\_Improved} \times 100$

B. $\frac{Time\_Original - Time\_Improved}{Time\_Original} \times 100$

C. $\frac{Time\_Original}{Time\_Improved} \times 100$

D. $\frac{Time\_Improved}{Time\_Original} \times 100$

Câu 24: Kết quả thực nghiệm cho thấy thuật toán cải tiến hiệu quả nhất trong trường hợp nào?

A. Khi dữ liệu rất nhỏ.

B. Khi bán kính Eps nhỏ và $MinPts$ lớn.

C. Khi bán kính Eps lớn (không gian tìm kiếm rộng) và $MinPts$ nhỏ (xử lý nặng).

D. Khi không có nhiễu.

Câu 25: Mức cải thiện hiệu năng cao nhất (Diff %) ghi nhận được trong bảng kết quả là khoảng bao nhiêu?

A. 50%

B. 5%

C. 19.17% (xấp xỉ 20%).

D. 90%

Câu 26: Tại sao trong một số trường hợp (tác vụ nhẹ), chỉ số Diff lại mang giá trị âm (thuật toán cải tiến chậm hơn chút ít)?

A. Do thuật toán bị lỗi.

B. Do chi phí khởi tạo cấu trúc dữ liệu tối ưu hóa lấn át lợi ích khi dữ liệu thưa/ít.

C. Do máy tính bị nóng.

D. Do dữ liệu đầu vào bị sai.

Câu 27: Nhận xét về khả năng chịu tải (Scalability) của thuật toán cải tiến?

A. Kém hơn thuật toán gốc.

B. Tương đương thuật toán gốc.

C. Tốt hơn, xu hướng cải thiện rõ ràng hơn khi độ phức tạp bài toán tăng.

D. Không ổn định, lúc nhanh lúc chậm thất thường.

Câu 28: Nghiên cứu này khẳng định thuật toán iNS-DBSCAN phù hợp cho ứng dụng nào?

A. Xử lý ảnh y tế.

B. Phân tích văn bản.

C. Hệ thống thông tin địa lý (GIS) và các dịch vụ dựa trên vị trí (LBS).

D. Dự báo thời tiết toàn cầu.

Câu 29: Một trong những hướng phát triển tiếp theo được đề xuất là gì?

A. Chuyển sang sử dụng thuật toán K-Means.

B. Tối ưu hóa bộ nhớ và xử lý song song/phân tán (GPU).

C. Loại bỏ hoàn toàn tham số Eps.

D. Chỉ áp dụng cho dữ liệu 1 chiều.

Câu 30: Về mặt tham số ($Eps, MinPts$), hướng nghiên cứu tương lai đề xuất điều gì?

A. Giữ nguyên cố định.

B. Yêu cầu người dùng nhập tay chính xác hơn.

C. Nghiên cứu cơ chế tham số thích nghi (Adaptive Parameters) dùng học máy.

D. Thay thế bằng tham số ngẫu nhiên.

Câu 31: Trong thuật toán LSPD (Local Shortest-Path Distance), tại sao chi phí tính toán khoảng cách mạng lưới (Network Distance) lại cao hơn so với khoảng cách Euclid truyền thống?

A. Do phải thực hiện phép tính tích phân phức tạp.

B. Do phải duyệt qua cấu trúc đồ thị, tính tổng trọng số các cạnh và xử lý các nút giao thay vì chỉ áp dụng công thức tọa độ đơn giản.

C. Do dữ liệu mạng lưới luôn có kích thước lớn hơn dữ liệu không gian phẳng.

D. Do khoảng cách mạng lưới luôn bao gồm cả thành phần độ cao (3D).

Câu 32: Giai đoạn xây dựng trật tự mật độ trong NS-DBSCAN sử dụng chiến lược "Leo đồi" (Hill Climbing) nhằm mục đích chính là gì?

A. Để tìm ra đường đi ngắn nhất giữa hai điểm bất kỳ trong đồ thị.

B. Để loại bỏ ngay lập tức các điểm nhiễu ra khỏi tập dữ liệu.

C. Để đảm bảo các điểm có mật độ cao cục bộ được ưu tiên xử lý trước, mô phỏng cấu trúc cụm tự nhiên dạng các "ngọn đồi".

D. Để sắp xếp các điểm dữ liệu theo thứ tự ID tăng dần.

Câu 33: Sự khác biệt cốt lõi trong hình học của "Vùng lân cận Eps" (N_{Eps}) giữa DBSCAN truyền thống và NS-DBSCAN là gì?

A. DBSCAN là hình vuông, NS-DBSCAN là hình tròn.

B. DBSCAN là hình tròn bán kính Eps; NS-DBSCAN là tập hợp điểm trên đồ thị có đường đi ngắn nhất từ tâm $\leq Eps$ (dạng mạng nhện lan tỏa).

C. DBSCAN là không gian 2D, NS-DBSCAN là không gian 3D.

D. DBSCAN có bán kính cố định, NS-DBSCAN có bán kính thay đổi ngẫu nhiên.

Câu 34: Dựa trên Định lý 1, tại sao có thể khẳng định việc loại bỏ các cạnh có trọng số $W(e) > Eps$ ngay từ bước tiền xử lý là an toàn tuyệt đối?

A. Vì các cạnh dài thường là lỗi dữ liệu GPS.

B. Vì một đường đi hợp lệ trong vùng lân cận có tổng độ dài $\leq Eps$ thì không thể chứa bất kỳ cạnh thành phần nào dài hơn Eps.

C. Vì các cạnh dài làm thuật toán chạy chậm mà không mang lại thông tin gì.

D. Vì xác suất xuất hiện các cạnh dài trong đô thị là rất thấp.

Câu 35: Trong tối ưu hóa bảng mật độ, tại sao kỹ thuật "Chèn theo thứ tự giảm dần" (Descending Order Insertion) lại hiệu quả hơn phương pháp của thuật toán gốc?

A. Vì nó sử dụng ít bộ nhớ RAM hơn.

B. Vì thuật toán gốc thêm điểm vào rồi mới sắp xếp lại (Re-sorting) gây tốn kém chi phí $O(N \log N)$, trong khi cải tiến chèn trực tiếp vào vị trí đúng giúp tránh việc này.

C. Vì thư viện lập trình Python hỗ trợ chèn giảm dần tốt hơn tăng dần.

D. Vì nó giúp loại bỏ các điểm có giá trị trùng lặp.

Câu 36: Ngưỡng heuristic $\delta \approx \ln(n)$ được sử dụng trong bảng sắp xếp mật độ có vai trò gì?

A. Là công thức để tính toán bán kính Eps tối ưu.

B. Là ngưỡng để xác định số lượng cụm tối đa.

C. Là ngưỡng lọc các điểm có mật độ quá thấp (Sparse points) không có khả năng trở thành điểm lõi, giúp giảm kích thước dữ liệu đầu vào.

D. Là hằng số để cân bằng độ phức tạp thuật toán.

Câu 37: Cơ chế "Xác định nhiễu ngầm định" (Implicit Noise Identification) giúp tối ưu hóa hiệu năng như thế nào?

A. Nó sử dụng AI để dự đoán điểm nhiễu trước khi chạy thuật toán.

B. Thay vì kiểm tra điều kiện "if p is noise" trong vòng lặp, nó coi tất cả các điểm không được gán vào cụm sau khi kết thúc là nhiễu, loại bỏ khoảng 3 phép kiểm tra điều kiện cho mỗi điểm.

C. Nó tự động xóa các điểm nhiễu ra khỏi cơ sở dữ liệu gốc.

D. Nó gộp tất cả nhiễu vào một cụm riêng biệt để xử lý sau.

Câu 38: Trong kết quả thực nghiệm, tại sao có trường hợp (ví dụ $Eps = 200, MinPts = 25$) chỉ số cải thiện hiệu năng lại âm (-0.72%)?

A. Do lỗi lập trình trong thuật toán cải tiến.

B. Do máy tính thực hiện bị quá tải nhiệt.

C. Do chi phí khởi tạo (overhead) của các cấu trúc dữ liệu trong thuật toán cải tiến lớn hơn lợi ích thu được khi dữ liệu thưa hoặc khối lượng tính toán nhỏ (Trade-off).

D. Do dữ liệu đầu vào bị sai lệch.

Câu 39: Thuật toán cải tiến thể hiện ưu thế rõ rệt nhất (tốc độ nhanh hơn nhiều nhất) trong kịch bản nào?

A. Eps nhỏ và MinPts lớn (Không gian tìm kiếm hẹp).

B. Eps lớn và MinPts nhỏ (Khối lượng tính toán lớn, không gian tìm kiếm rộng).

C. Khi chạy với dữ liệu giả lập.

D. Khi chạy trên mạng lưới đường bộ đơn giản, ít ngã tư.

Câu 40: Tại sao báo cáo lại chọn dữ liệu OpenStreetMap (OSM) để thực nghiệm thay vì dữ liệu giả lập?

A. Vì dữ liệu OSM hoàn toàn miễn phí và dễ tải.

B. Để phản ánh chính xác sự phân bố phức tạp, tính ngẫu nhiên và các ràng buộc topo thực tế của mạng lưới giao thông đô thị, đảm bảo tính khách quan.

C. Vì dữ liệu giả lập không thể tạo được đồ thị vô hướng.

D. Vì thuật toán chỉ tương thích với định dạng file của OSM.

Câu 41: Vai trò quan trọng nhất của "Biểu đồ trật tự mật độ" (Density Ordering Graph) là gì?

A. Hiển thị vị trí địa lý.

B. Tính toán khoảng cách ngắn nhất.

C. Giúp trực quan hóa cấu trúc dữ liệu dưới dạng "ngọn đồi" để hỗ trợ chọn tham số MinPts và Eps phù hợp.

D. Loại bỏ cạnh thừa.

Câu 42: Tại sao thuật toán khuyến nghị ngưỡng lọc $MinPts \geq \ln(n) + 1$?

A. Vì đây là quy định của phần mềm.

B. Vì đây là ngưỡng heuristic thống kê, tự động điều chỉnh theo quy mô dữ liệu (n) để loại bỏ nhiễu ngẫu nhiên hiệu quả.

C. Vì giá trị logarit giúp làm đẹp biểu đồ.

D. Vì thuật toán cần số lẻ để hoạt động.

Câu 43: Nếu Biểu đồ trật tự mật độ có dạng "phẳng", điều đó báo hiệu gì?

A. Dữ liệu phân bố quá đồng đều (không có cụm) hoặc tham số Eps được chọn chưa hợp lý.

B. Thuật toán hoạt động hoàn hảo.

C. Mạng lưới không có đường cụt.

D. Số lượng điểm quá nhỏ.

Câu 44: Tại sao điểm sự kiện thường được ánh xạ lên cạnh (Edge) thay vì nút (Node)?

A. Để tiết kiệm bộ nhớ.

B. Vì thực tế sự kiện (tai nạn, cửa hàng) nằm dọc theo đường, gán lên cạnh giúp tính khoảng cách di chuyển chính xác hơn.

C. Vì thuật toán LSPD chỉ chạy được trên cạnh.

D. Vì các nút giao thông không được chứa điểm sự kiện.

Câu 45: Chiến lược "Cắt tỉa cạnh" kém hiệu quả nhất trong trường hợp nào?

A. Mạng lưới nhiều đường cao tốc.

B. Mạng lưới ít điểm sự kiện.

C. Mạng lưới dày đặc với nhiều con đường cụt (dead-ends) ngắn hơn Eps.

D. Mạng lưới đường một chiều.

Câu 46: Nếu không xử lý Topology kỹ càng cho cầu vượt/hầm chui, thuật toán dễ mắc lỗi gì?

A. Không hiển thị màu sắc.

B. Nhận diện nhầm các điểm trên cầu và dưới cầu là "hàng xóm" (do trùng tọa độ 2D) dù thực tế cách xa nhau.

C. Tự động xóa điểm trong hầm.

D. Tính sai mật độ do diện tích mặt đường.

Câu 47: Trong trường hợp "tệ nhất" (Worst-case) nào thuật toán cải tiến có thể chạy chậm hơn hoặc ngang bằng thuật toán gốc?

A. Dữ liệu quá lớn.

B. MinPts = 1.

C. Chọn tham số Eps quá lớn (lớn hơn mọi cạnh trong đồ thị) khiến chiến lược cắt tỉa bị vô hiệu hóa.

D. Máy tính cấu hình yếu.