ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

🙤🙧🟍🙥🙦

Logo, company name

Description automatically generated

ĐỒ ÁN MÔN HỌC

MẠNG XÃ HỘI

PHÂN TÍCH DỮ LIỆU CHUỖI CUNG ỨNG CHO

CÁC LOẠI SẢN PHẨM THÔNG QUA CÁC THƯỚC ĐO VỀ

TRỌNG TÂM VÀ KẾT QUẢ PHÂN CỤM

SINH VIÊN THỰC HIỆN

ĐẶNG HOÀNG QUÂN – 18520339

GIẢNG VIÊN HƯỚNG DẪN

NGUYỄN THỊ KIM PHỤNG

TP. HỒ CHÍ MINH – 11/2021

**MỤC LỤC**

[**DANH MỤC HÌNH ẢNH** 1](#_Toc89898509)

[**I.** **GIỚI THIỆU** 1](#_Toc89898510)

[**II.** **XÁC ĐỊNH BÀI TOÁN** 1](#_Toc89898511)

[**III.** **DỮ LIỆU** 1](#_Toc89898512)

[1. Giới thiệu nguồn dữ liệu 1](#_Toc89898513)

[2. Xử lý và phân tích dữ liệu 7](#_Toc89898514)

[2.1. Đọc dữ liệu từ file vào DataFrame 7](#_Toc89898515)

[2.2. Làm sạch dữ liệu 7](#_Toc89898516)

[2.3. Chuyển đổi DataFrame thành đồ thị 8](#_Toc89898517)

[**IV.** **XẾP HẠNG CÁC LOẠI SẢN PHẨM** 10](#_Toc89898518)

[1. Định nghĩa các hàm hiển thị kết quả độ đo 10](#_Toc89898519)

[2. Degree Centrality 11](#_Toc89898520)

[2.1. Code chạy độ đo 11](#_Toc89898521)

[2.2. Kết quả độ đo 11](#_Toc89898522)

[2.3. Nhận xét 12](#_Toc89898523)

[3. Betweenness Centrality 13](#_Toc89898524)

[3.1. Code chạy độ đo 13](#_Toc89898525)

[3.2. Kết quả độ đo 13](#_Toc89898526)

[3.3. Nhận xét 13](#_Toc89898527)

[4. Closeness Centrality 14](#_Toc89898528)

[4.1. Code chạy độ đo 14](#_Toc89898529)

[4.2. Kết quả độ đo 15](#_Toc89898530)

[4.3. Nhận xét 15](#_Toc89898531)

[5. Eigenvector Centrality 16](#_Toc89898532)

[5.1. Code chạy độ đo 16](#_Toc89898533)

[5.2. Kết quả độ đo 16](#_Toc89898534)

[5.3. Nhận xét 17](#_Toc89898535)

[6. PageRank 18](#_Toc89898536)

[6.1. Code chạy độ đo 18](#_Toc89898537)

[6.2. Kết quả độ đo 18](#_Toc89898538)

[6.3. Nhận xét 19](#_Toc89898539)

[**V.** **THUẬT TOÁN PHÁT HIỆN CỘNG ĐỒNG** 20](#_Toc89898540)

[1. Định nghĩa các hàm hiển thị kết quả phân cụm 20](#_Toc89898541)

[2. Thuật toán Louvain 21](#_Toc89898542)

[2.1. Code chạy thuật toán 21](#_Toc89898543)

[2.2. Đồ thị phân cụm 21](#_Toc89898544)

[2.3. Ý nghĩa các cụm 22](#_Toc89898545)

[2.4. Nhận xét 23](#_Toc89898546)

[3. Thuật toán K-Means 23](#_Toc89898547)

[3.1. Code chạy thuật toán 23](#_Toc89898548)

[3.2. Đồ thị phân cụm 25](#_Toc89898549)

[3.3. Ý nghĩa các cụm 25](#_Toc89898550)

[3.4. Nhận xét 26](#_Toc89898551)

[4. Gaussian Mixture Model 27](#_Toc89898552)

[4.1. Code chạy thuật toán 27](#_Toc89898553)

[4.2. Đồ thị phân cụm 27](#_Toc89898554)

[4.3. Ý nghĩa các cụm 27](#_Toc89898555)

[4.4. Nhận xét 28](#_Toc89898556)

[5. Trực quan hóa các điểm chung trong các cụm 29](#_Toc89898557)

[**VI.** **TÀI LIỆU THAM KHẢO** 30](#_Toc89898558)

**DANH MỤC HÌNH ẢNH**

[Hình 1. Đọc dữ liệu từ file vào DataFrame 7](#_Toc89898343)

[Hình 2. Kiểm tra các giá trị bị khuyết 7](#_Toc89898344)

[Hình 3. Loại bỏ các giá trị trùng lặp 7](#_Toc89898345)

[Hình 4. Chuyển đổi DataFrame thành đồ thị 8](#_Toc89898346)

[Hình 5. Đồ thị 2 phía 8](#_Toc89898347)

[Hình 6. Code hiển thị đồ thị 2 phía 9](#_Toc89898348)

[Hình 7. Đồ thị 1 phía 9](#_Toc89898349)

[Hình 8. Code hiển thị đồ thị 1 phía 10](#_Toc89898350)

[Hình 9. Code hiển thị Cliques 10](#_Toc89898351)

[Hình 10. Hàm in kết quả độ đo 10](#_Toc89898352)

[Hình 11. Hàm trực quan hóa kết quả độ đo 11](#_Toc89898353)

[Hình 12. Code chạy Degree Centrality 11](#_Toc89898354)

[Hình 13. Kết quả Degree Centrality 11](#_Toc89898355)

[Hình 14. Kết quả Degree Centrality trên Gephi 12](#_Toc89898356)

[Hình 15. Code chạy Betweenness Centrality 13](#_Toc89898357)

[Hình 16. Kết quả Betweenness Centrality 13](#_Toc89898358)

[Hình 17. Kết quả Betweenness Centrality trên Gephi 14](#_Toc89898359)

[Hình 18. Code chạy Closeness Centrality 14](#_Toc89898360)

[Hình 19. Kết quả Closeness Centrality 15](#_Toc89898361)

[Hình 20. Kết quả Closeness Centrality trên Gephi 16](#_Toc89898362)

[Hình 21. Code chạy Eigenvector Centrality 16](#_Toc89898363)

[Hình 22. Kết quả Eigenvector Centrality 17](#_Toc89898364)

[Hình 23. Kết quả Eigenvector Centrality trên Gephi 18](#_Toc89898365)

[Hình 24. Code chạy PageRank 18](#_Toc89898366)

[Hình 25. Kết quả PageRank 19](#_Toc89898367)

[Hình 26. Kết quả PageRank trên Gephi 20](#_Toc89898368)

[Hình 27. Hàm lấy đặc điểm chung của 1 cụm 20](#_Toc89898369)

[Hình 28. Hàm in ra các cộng đồng cùng điểm chung của các node trong chúng 21](#_Toc89898370)

[Hình 29. Hàm trực quan hóa các cộng đồng 21](#_Toc89898371)

[Hình 30. Code chạy thuật toán Louvain 21](#_Toc89898372)

[Hình 31. Đồ thị phân cụm sử dụng Louvain 22](#_Toc89898373)

[Hình 32. Kết quả phân cụm sử dụng Louvain 22](#_Toc89898374)

[Hình 33. Chuyển đổi đồ thị thành ma trận kề làm đầu vào cho K-Means 23](#_Toc89898375)

[Hình 34. Code triển khai Elbow Method cho thuật toán K-Means 24](#_Toc89898376)

[Hình 35. Chọn ra số cụm k cho thuật toán K-Means bằng Elbow Method 24](#_Toc89898377)

[Hình 36. Chạy K-Means với đầu vào là ma trận kề cùng số cụm k đã chọn 25](#_Toc89898378)

[Hình 37. Đồ thị phân cụm sử dụng K-Means 25](#_Toc89898379)

[Hình 38. Kết quả phân cụm sử dụng K-Means 25](#_Toc89898380)

[Hình 39. Code chạy mô hình Gaussian Mixture 27](#_Toc89898381)

[Hình 40. Đồ thị phân cụm sử dụng GMM 27](#_Toc89898382)

[Hình 41. Kết quả phân cụm sử dụng GMM 27](#_Toc89898383)

[Hình 42. Các khu vực đã cung cấp các sản phẩm của cụm 0 khi dùng Louvain 29](#_Toc89898384)

[Hình 43. Các khu vực đã cung cấp các sản phẩm của cụm 1 khi dùng Louvain 29](#_Toc89898385)

[Hình 44. Các khu vực đã cung cấp các sản phẩm của cụm 2 khi dùng Louvain 30](#_Toc89898386)

1. **GIỚI THIỆU**

Ngày nay, nhiều công ty phải đối mặt với một thách thức dường như đầy mâu thuẫn: làm thế nào để giảm chi phí vận hành đồng thời tăng mức độ dịch vụ khách hàng. Thiết kế mạng lưới chuỗi cung ứng phù hợp cung cấp giải pháp cho cả hai vấn đề trên. Mặc dù có nhiều yếu tố cần xem xét khi thiết kế mạng lưới chuỗi cung ứng, nhưng quá trình này không quá phức tạp với các đối tác phù hợp.

Môn học mạng xã hội sẽ giúp phân tích và xem xét chuỗi cung ứng của công ty bằng cách sử dụng các thước đo về trọng tâm và phân cụm liên quan đến các khía cạnh nổi bật (dựa trên đường đi ngắn nhất, phổ, khoảng cách, ...) để từ đó đưa ra các chiến lược vận hành phù hợp nhất.

1. **XÁC ĐỊNH BÀI TOÁN**

* Input: Tập dữ liệu ban đầu trên nguồn dữ liệu Kaggle được qua tiền xử lý dữ liệu.
* Output: Đưa ra độ đo, cộng đồng phục vụ cho việc phân tích mạng xã hội **DataCo SMART SUPPLY CHAIN FOR BIG DATA ANALYSIS.**

1. **DỮ LIỆU**
2. Giới thiệu nguồn dữ liệu

* Link dataset: <https://www.kaggle.com/shashwatwork/dataco-smart-supply-chain-for-big-data-analysis>.
* Dữ liệu **DataCo SMART SUPPLY CHAIN FOR BIG DATA ANALYSIS** được cung cấp trên nền tảng Kaggle chứa dữ liệu của chuỗi cung ứng được sử dụng bởi công ty DataCo Global.
* Dataset bao gồm tập hợp các sản phẩm đã bán của công ty, chi tiết tài chính (lãi, lỗ, tổng doanh thu, v.v.), chi tiết giao hàng và chi tiết khách hàng như doanh số, nhân khẩu học và chi tiết giao dịch.
* Dữ liệu có kích thước 91 MB bao gồm 180519 dòng với 54 thuộc tính:

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Type** | **Mô tả** | **Các giá trị** |
| Type | char | Loại giao dịch thực hiện | * CASH * DEBIT * PAYMENT * TRANSFER |
| Days for shipping (real) | int | Số ngày giao hàng thực tế |  |
| Days for shipment (scheduled) | int | Số ngày giao hàng theo dự kiến |  |
| Benefit per order | float | Thu nhập cho mỗi đơn hàng được đặt | = Order Profit Per Order |
| Sales per customer | float | Tổng doanh số theo khách hàng | = Order Item Total |
| Delivery Status | char | Trạng thái giao hàng của đơn hàng | * Advance shipping * Late delivery * Shipping canceled * Shipping on time |
| Late\_delivery\_risk | int | Biến phân loại cho biết gửi muộn hay không | * 1 – Gửi muộn * 0 – Không gửi muộn |
| Category Id | int | Mã danh mục sản phẩm | = Product Category Id |
| Category Name | char | Tên danh mục sản phẩm |  |
| Customer City | char | Thành phố nơi khách hàng thực hiện mua hàng |  |
| Customer Country | char | Đất nước nơi khách hàng thực hiện mua hàng |  |
| Customer Email | char | Email của khách hàng | XXXXXXXXX |
| Customer Fname | char | Tên khách hàng |  |
| Customer Id | int | ID khách hàng |  |
| Customer Lname | char | Họ khách hàng |  |
| Customer Password | char | Mật khẩu khách hàng | XXXXXXXXX |
| Customer Segment | char | Phân khúc khách hàng | * Consumer * Corporate * Home Office |
| Customer State | char | Tiểu bang của cửa hàng đã đăng ký giao dịch mua |  |
| Customer Street | char | Đường của cửa hàng đã đăng ký giao dịch mua |  |
| Customer Zipcode | float | Mã bưu điện khách hàng |  |
| Department Id | int | Mã bộ phận của cửa hàng |  |
| Department Name | char | Tên bộ phận của cửa hàng |  |
| Latitude | float | Vĩ độ tương ứng với vị trí của cửa hàng |  |
| Longitude | float | Kinh độ tương ứng với vị trí của cửa hàng |  |
| Market | char | Thị trường của nơi được giao hàng | * Africa * Europe * LATAM * Pacific Asia * USCA |
| Order City | char | Thành phố của đơn hàng được đặt |  |
| Order Country | char | Quốc gia của đơn hàng được đặt |  |
| Order Customer Id | int | Mã đặt hàng của khách | = Customer Id |
| order date (DateOrders) | datetime | Ngày đặt hàng |  |
| Order Id | int | Mã đơn hàng |  |
| Order Item Cardprod Id | int | Mã sản phẩm được tạo thông qua đầu đọc RFID | = Product Card Id |
| Order Item Discount | float | Giá trị chiết khấu của mặt hàng trong đơn hàng | = Sales - Order Item Total |
| Order Item Discount Rate | float | Phần trăm chiết khấu của mặt hàng trong đơn hàng |  |
| Order Item Id | int | Mã mặt hàng được đặt trong đơn hàng |  |
| Order Item Product Price | float | Giá của sản phẩm khi không giảm giá | = Product Price |
| Order Item Profit Ratio | float | Tỷ lệ lợi nhuận của mặt hàng trong đơn hàng |  |
| Order Item Quantity | int | Số lượng sản phẩm c mỗi mặt hàng trong đơn hàng |  |
| Sales | float | Doanh số bán hàng | = Order Item Product Price \* Order Item Quantity |
| Order Item Total | float | Tổng số tiền cho mặt hàng trong đơn hàng | = Sales per customer |
| Order Profit Per Order | float | Lợi nhuận cho mỗi mặt hàng trong đơn hàng | = Benefit per order |
| Order Region | char | Khu vực nơi đơn đặt hàng được tiến hành giao hay khu vực sẽ cung cấp loại sản phẩm | * Southeast Asia * South Asia * Oceania * Eastern Asia * West Asia * West of USA * US Center * West Africa * Central Africa * North Africa * Western Europe * Northern * Caribbean * South America * East Africa * Southern Europe * East of USA * Canada * Southern Africa * Central Asia * Europe * Central America * Eastern Europe * South of USA |
| Order State | char | Tiểu bang nơi đơn hàng được tiến hành giao |  |
| Order Status | char | Trạng thái đơn hàng | * CANCELED * CLOSED * COMPLETE * ON\_HOLD * PAYMENT\_REVIEW * PENDING * PENDING\_PAYMENT * PROCESSING * SUSPECTED\_FRAUD |
| Order Zipcode | float | Mã bưu điện đơn hàng |  |
| Product Card Id | int | Mã sản phẩm | = Order Item Cardprod Id |
| Product Category Id | int | Mã danh mục sản phẩm | = Category Id |
| Product Description | float | Mô tả sản phẩm |  |
| Product Image | char | Liên kết đến hình ảnh của sản phẩm |  |
| Product Name | char | Tên sản phẩm |  |
| Product Price | float | Giá sản phẩm | = Order Item Product Price |
| Product Status | int | Trạng thái sản phẩm | * 1 – Không có sẵn * 0 – Có sẵn |
| shipping date (DateOrders) | datetime | Ngày và thời gian chính xác của lô hàng |  |
| Shipping Mode | char | Chế độ vận chuyển | * First Class * Same Day * Second Class * Standard Class |

1. Xử lý và phân tích dữ liệu
   1. Đọc dữ liệu từ file vào DataFrame

Text

Description automatically generated

Hình 1. Đọc dữ liệu từ file vào DataFrame

* 1. Làm sạch dữ liệu
* Kiểm tra các giá trị bị khuyết 🡺 không phát hiện giá trị nào nên không cần loại bỏ:

Graphical user interface, application, Word

Description automatically generated

Hình 2. Kiểm tra các giá trị bị khuyết

* Loại bỏ các giá trị trùng lặp 🡺 Kết quả cuối cùng nhận được là 1 bộ dữ liệu gồm 691 dòng và 2 cột:

Text

Description automatically generated

Hình 3. Loại bỏ các giá trị trùng lặp

* 1. Chuyển đổi DataFrame thành đồ thị

Text

Description automatically generated

Hình 4. Chuyển đổi DataFrame thành đồ thị

* + 1. Đồ thị 2 phía
* Node: danh mục các loại sản phầm (Category Name) và khu vực nơi mà đơn đặt hàng được tiến hành giao hay có thể nói cách khác là khu vực sẽ cung cấp loại sản phẩm (Order Region).
* Edge: mối quan hệ hình thành khi 1 loại sản phẩm được cung cấp từ 1 khu vực.

Text

Description automatically generated

Hình 5. Đồ thị 2 phía

* Nhìn vào đồ thị có thể thấy cùng 1 loại sản phẩm có thể được cung cấp từ nhiều khu vực và 1 khu vực cũng có thể cung cấp nhiều loại sản phẩm.
* Code hiển thị đồ thị 2 phía:

Text

Description automatically generated

Hình 6. Code hiển thị đồ thị 2 phía

* + 1. Đồ thị 1 phía
* Node: danh mục các loại sản phầm (Category Name).
* Edge: 2 loại sản phẩm cùng được cung cấp từ 1 khu vực sẽ được nối với nhau tạo thành cạnh, ý nghĩa nói lên với cùng 1 khu vực có thể cung cấp nhiều loại sản phẩm.
* Weight: trọng số là số khu vực giống nhau đã cung cấp 2 loại sản phẩm.

Diagram

Description automatically generated

Hình 7. Đồ thị 1 phía

* Code hiển thị đồ thị 1 phía:

Graphical user interface, text

Description automatically generated

Hình 8. Code hiển thị đồ thị 1 phía

* + 1. Cliques
* Clique được định nghĩa là 1 đồ thị con hoàn chỉnh **cực đại** của 1 đồ thị trong đó mỗi node được kết nối với tất cả các node khác. Từ **cực đại** có nghĩa là nếu ta thêm một node khác vào Clique, nó sẽ không còn là 1 Clique.
* Code hiển thị Cliques:

A picture containing application

Description automatically generated

Hình 9. Code hiển thị Cliques

* Dễ thấy 2 Clique rút ra được chỉ khác nhau ở các node ngoài rìa (màu đỏ) có thể nhìn thấy rõ ràng ở đồ thị 1 phía.

1. **XẾP HẠNG CÁC LOẠI SẢN PHẨM**
2. Định nghĩa các hàm hiển thị kết quả độ đo

* Hàm in kết quả độ đo đã sắp xếp giảm dần:

Text

Description automatically generated

Hình 10. Hàm in kết quả độ đo

* Hàm trực quan hóa kết quả độ đo:

Text

Description automatically generated

Hình 11. Hàm trực quan hóa kết quả độ đo

1. Degree Centrality
   1. Code chạy độ đo

Text

Description automatically generated

Hình 12. Code chạy Degree Centrality

* 1. Kết quả độ đo

A picture containing timeline

Description automatically generated

Hình 13. Kết quả Degree Centrality

* 1. Nhận xét
* Degree của 1 node thể hiện số lượng node mà 1 node nhất định kết nối. Loại sản phẩm có Degree càng cao thì càng có nhiều liên kết, nghĩa là các khu vực mà cung cấp loại sản phẩm này cũng sẽ cung cấp các loại sản phẩm khác.
* Những loại sản phẩm có Degree Centrality cao thể hiện các loại sản phẩm này được cung cấp bởi nhiều khu vực khác nhau, vì vậy nó mới kết nối với nhiều loại sản phẩm khác nhau.
* Như trong hình 13, ta có thể thấy các node ở phía ngoài rìa đồ thị sẽ có Degree Centrality thấp hơn. Bên cạnh đó, kết quả độ đo thực hiện trên Python và Gephi không có sự khác biệt.

Table

Description automatically generatedTable

Description automatically generated

Hình 14. Kết quả Degree Centrality trên Gephi

1. Betweenness Centrality
   1. Code chạy độ đo

Text

Description automatically generated

Hình 15. Code chạy Betweenness Centrality

* 1. Kết quả độ đo

A picture containing timeline

Description automatically generated

Hình 16. Kết quả Betweenness Centrality

* 1. Nhận xét
* Betweenness Centrality là thước đo độ “ở giữa” trong đồ thị dựa trên ý tưởng về đường đi ngắn nhất. Betweenness Centrality của 1 node là 1 phần nhỏ của các đường đi ngắn nhất đi qua node đó.
* Các node có độ “ở giữa” cao hoạt động như 1 cầu nối giữa các phần biệt lập khác nhau của mạng. Các loại sản phẩm có Betweenness Centrality cao thể hiện nó có liên kết với nhiều loại sản phẩm khác, vì vậy các loại sản phẩm này được các khu vực cung cấp nhiều hơn các loại sản phẩm khác.
* Như trong hình 16, ta có thế thấy các node màu cam có Betweenness Centrality cao nhất, nghĩa là nó quan trọng trong việc kết nối giữa các node, các node muốn kết nối với nhau phải thông qua node đó. Bên cạnh đó, kết quả độ đo thực hiện trên Python và Gephi không có sự khác biệt.

Table

Description automatically generatedTable

Description automatically generated

Hình 17. Kết quả Betweenness Centrality trên Gephi

1. Closeness Centrality
   1. Code chạy độ đo

Text

Description automatically generated

Hình 18. Code chạy Closeness Centrality

* 1. Kết quả độ đo

A picture containing timeline

Description automatically generated

Hình 19. Kết quả Closeness Centrality

* 1. Nhận xét
* Closeness Centrality là 1 cách phát hiện các node có thể phân phối luồng đi 1 cách hiệu quả qua mạng, node quan trọng là node ở gần với các node khác.
* Kết quả của Closeness Centrality thể hiển top “độ gần” của các node đến tất cả các node trong mạng, đồng nghĩa với việc top các loại sản phẩm sẽ cần được cung cấp nhiều nhất. Bên cạnh đó, kết quả độ đo thực hiện trên Python và Gephi không có sự khác biệt.

Table

Description automatically generatedTable

Description automatically generated

Hình 20. Kết quả Closeness Centrality trên Gephi

1. Eigenvector Centrality
   1. Code chạy độ đo

Text

Description automatically generated

Hình 21. Code chạy Eigenvector Centrality

* 1. Kết quả độ đo

A picture containing timeline

Description automatically generated

Hình 22. Kết quả Eigenvector Centrality

* 1. Nhận xét
* Eigenvector Centrality đo lường tầm quan trọng của 1 node trong mạng đồng thời xem xét tầm quan trọng của các node lân cận. Các kết nối đến các node trung tâm của Eigenvector có điểm số cao đóng góp nhiều hơn vào điểm số chung so với các kết nối ngang bằng với các nút có điểm số thấp. Nói cách khác, 1 node có nhiều kết nối có thể có điểm Eigenvector thấp nếu tất cả các kết nối của nó đều có các node có điểm thấp.
* Loại sản phẩm có điểm Eigenvector cao khi nó được kết nối với nhiều node có điểm cao khác, đồng nghĩa với việc các loại sản phẩm có điểm Eigenvector càng cao sẽ càng được cung cấp bởi nhiều khu vực, vì các loại sản phẩm kết nối với loại sản phẩm đó cũng được cung cấp bởi nhiều khu vực.
* Như trong hình 22, ta có thể thấy các node màu cam có ảnh hưởng nhất mạng. Bên cạnh đó, kết quả độ đo thực hiện trên Python cũng có sự khác biệt so với Gephi.

Table

Description automatically generatedTable

Description automatically generated

Hình 23. Kết quả Eigenvector Centrality trên Gephi

1. PageRank
   1. Code chạy độ đo

Text

Description automatically generated

Hình 24. Code chạy PageRank

* 1. Kết quả độ đo

A picture containing graphical user interface

Description automatically generated

Hình 25. Kết quả PageRank

* 1. Nhận xét
* PageRank của Google là 1 biến thể phổ biến của Eigenvector Centrality được chuẩn hóa kết hợp với các bước nhảy ngẫu nhiên. Do đó, không có gì ngạc nhiên khi Pagerank mang lại kết quả tương đương hoặc có thể tốt hơn cả Eigenvector.
* Loại sản phẩm có PageRank càng cao càng được cung cấp bởi nhiều khu vực vì các loại sản phẩm kết nối với loại sản phẩm có PageRank cao cũng được cung cấp bởi nhiều khu vực khác nhau.
* Như trong hình 25, ta có thể thấy PageRank đã có sự phân hóa rõ rệt hơn các loại sản phẩm quan trọng trong mạng so với các độ đo trước. Bên cạnh đó, kết quả độ đo thực hiện trên Python và Gephi không có sự khác biệt.

Table

Description automatically generated Table

Description automatically generated

Hình 26. Kết quả PageRank trên Gephi

1. **THUẬT TOÁN PHÁT HIỆN CỘNG ĐỒNG**
2. Định nghĩa các hàm hiển thị kết quả phân cụm

* Hàm lấy đặc điểm chung của các node trong 1 cụm:

Graphical user interface, text, application, email

Description automatically generated

Hình 27. Hàm lấy đặc điểm chung của 1 cụm

* Hàm in ra các cộng đồng đã phát hiện được cùng điểm chung của các node trong chúng:

Text

Description automatically generated

Hình 28. Hàm in ra các cộng đồng cùng điểm chung của các node trong chúng

* Hàm trực quan hóa các cộng đồng:

Text

Description automatically generated

Hình 29. Hàm trực quan hóa các cộng đồng

1. Thuật toán Louvain
   1. Code chạy thuật toán

Text

Description automatically generated

Hình 30. Code chạy thuật toán Louvain

* 1. Đồ thị phân cụm

A picture containing text, indoor, decorated

Description automatically generated

Hình 31. Đồ thị phân cụm sử dụng Louvain

* Dễ thấy thuật toán rút ra được 3 cụm
  1. Ý nghĩa các cụm

Text

Description automatically generated

Hình 32. Kết quả phân cụm sử dụng Louvain

* Cụm thứ 0:
* Gồm **24 Node**: Accessories, Baseball & Softball, Boxing & MMA, Camping & Hiking, Cardio Equipment, Cleats, Electronics, Fishing, Fitness Accessories, Girls' Apparel, Golf Apparel, Golf Balls, Golf Gloves, Golf Shoes, Hockey, Hunting & Shooting, Indoor/Outdoor Games, Lacrosse, Men's Footwear, Shop By Sport, Tennis & Racquet, Trade-In, Water Sports, Women's Apparel.
* Ý nghĩa: những loại sản phẩm được cung cấp ở cả **22 khu vực** gồm Canada, Caribbean, Central Africa, Central America, East Africa, East of USA, Eastern Asia, Eastern Europe, North Africa, Northern Europe, Oceania, South America, South Asia, South of USA, Southeast Asia, Southern Africa, Southern Europe, US Center, West Africa, West Asia, West of USA, Western Europe.
* Cụm thứ 1:
* Gồm **18 Node**: Baby, Books, CDs, Cameras, Children's Clothing, Computers, Consumer Electronics, Crafts, DVDs, Garden, Health and Beauty, Men's Clothing, Music, Pet Supplies, Sporting Goods, Toys, Video Games, Women's Clothing.
* Ý nghĩa: những loại sản phẩm được cung cấp ở cả **3 khu vực** gồm Oceania, South Asia, Southeast Asia.
* Cụm thứ 2:
* Gồm **8 Node**: As Seen on TV!, Basketball, Golf Bags & Carts, Kids' Golf Clubs, Men's Golf Clubs, Soccer, Strength Training, Women's Golf Clubs.
* Ý nghĩa: những loại sản phẩm được cung cấp ở cả **3 khu vực** gồm Northern Europe, Southern Europe, Western Europe.
  1. Nhận xét
  + Nếu nhìn bằng mắt thường ta cũng dễ dàng nhận ra đồ thị có thể có 3 cụm. Louvain đã phát hiện được 3 cụm đó khá rõ rệt với nhau. Như vậy chứng tỏ thuật toán Louvain đã phân cụm khá tốt.

1. Thuật toán K-Means
   1. Code chạy thuật toán
      1. Chuyển đổi đồ thị thành ma trận kề

A screenshot of a computer

Description automatically generated with low confidence

Hình 33. Chuyển đổi đồ thị thành ma trận kề làm đầu vào cho K-Means

* + 1. Chọn ra số cụm bằng Elbow Method

Text

Description automatically generated

Hình 34. Code triển khai Elbow Method cho thuật toán K-Means

* Tính khoảng cách từng phần tử trong ma trận kề (50 x 50) với các tâm cụm (k x 50).
* Ma trận (50 x k) tương ứng với 50 loại sản phẩm, với hàng là các khoảng cách giữa mỗi phần tử trong ma trận kề với các tâm cụm.
* Lấy tổng các khoảng cách của mỗi phần tử trong ma trận kề với tâm cụm mà tại đó có khoảng cách giữa 2 bên là nhỏ nhất / số lượng phần tử trong ma trận kề.
* Cuối cùng, tính tổng biến thiên khoảng cách nhỏ nhất của trong cụm.

Chart, line chart

Description automatically generated

Hình 35. Chọn ra số cụm k cho thuật toán K-Means bằng Elbow Method

* + 1. Huấn luyện với số cụm đã chọn
* Chọn số cụm k là 2 để huấn luyện cũng như sẽ là số lượng centroid sẽ tạo ra.
* Tiến hành huấn luyện và hiển thị kết quả phân cụm với đầu vào là ma trận kề đã có.

Text

Description automatically generated

Hình 36. Chạy K-Means với đầu vào là ma trận kề cùng số cụm k đã chọn

* 1. Đồ thị phân cụm

A picture containing text, indoor, day

Description automatically generated

Hình 37. Đồ thị phân cụm sử dụng K-Means

* 1. Ý nghĩa các cụm

Text

Description automatically generated

Hình 38. Kết quả phân cụm sử dụng K-Means

* Cụm thứ 0:
* Gồm **26 Node**: As Seen on TV!, Baby, Basketball, Books, CDs, Cameras, Children's Clothing, Computers, Consumer Electronics, Crafts, DVDs, Garden, Golf Bags & Carts, Health and Beauty, Kids' Golf Clubs, Men's Clothing, Men's Golf Clubs, Music, Pet Supplies, Soccer, Sporting Goods, Strength Training, Toys, Video Games, Women's Clothing, Women's Golf Clubs.
* Ý nghĩa: các node trong cụm này **không** có điểm chung gì đồng thời với tất cả các node khác. Đây có thể đơn giản chỉ là phần còn lại sau khi phát hiện được cụm 1.
* Cụm thứ 1:
* Gồm **24 Node**: Accessories, Baseball & Softball, Boxing & MMA, Camping & Hiking, Cardio Equipment, Cleats, Electronics, Fishing, Fitness Accessories, Girls' Apparel, Golf Apparel, Golf Balls, Golf Gloves, Golf Shoes, Hockey, Hunting & Shooting, Indoor/Outdoor Games, Lacrosse, Men's Footwear, Shop By Sport, Tennis & Racquet, Trade-In, Water Sports, Women's Apparel.
* Ý nghĩa: những loại sản phẩm được cung cấp ở cả **22 khu vực** gồm Canada, Caribbean, Central Africa, Central America, East Africa, East of USA, Eastern Asia, Eastern Europe, North Africa, Northern Europe, Oceania, South America, South Asia, South of USA, Southeast Asia, Southern Africa, Southern Europe, US Center, West Africa, West Asia, West of USA, Western Europe.
  1. Nhận xét
  + Từ kết quả có thể thấy K-Means chỉ phân biệt được các node có phân bố dạng chụm lại với nhau tạo thành 1 cụm (màu vàng) nhưng không phân biệt được hay tìm được các đặc điểm chung của các node có phân bố dạng đường vòng (màu đỏ).
  + Nguyên nhân có thể do khi sử dụng thuật toán K-Means thì mỗi điểm dữ liệu được gán 1 cách dứt khoát thuộc 1 trung tâm cụm. Ngoài ra, K-Means không tính đến covariance (hiệp phương sai) để thể hiện tính liên quan của dữ liệu.
  + Nếu ta có 2 điểm cách đều nhau từ trung tâm cụm nhưng 1 điểm theo xu hướng này và điểm kia thì không, K-Means sẽ coi chúng là như nhau, vì K-Means sử dụng khoảng cách Euclide.

1. Gaussian Mixture Model
   1. Code chạy thuật toán

Text

Description automatically generated

Hình 39. Code chạy mô hình Gaussian Mixture

* 1. Đồ thị phân cụm

A picture containing indoor, decorated

Description automatically generated

Hình 40. Đồ thị phân cụm sử dụng GMM

* 1. Ý nghĩa các cụm

Text

Description automatically generated with medium confidence

Hình 41. Kết quả phân cụm sử dụng GMM

* Cụm thứ 0:
* Gồm **24 Node**: Accessories, Baseball & Softball, Boxing & MMA, Camping & Hiking, Cardio Equipment, Cleats, Electronics, Fishing, Fitness Accessories, Girls' Apparel, Golf Apparel, Golf Balls, Golf Gloves, Golf Shoes, Hockey, Hunting & Shooting, Indoor/Outdoor Games, Lacrosse, Men's Footwear, Shop By Sport, Tennis & Racquet, Trade-In, Water Sports, Women's Apparel.
* Ý nghĩa: những loại sản phẩm được cung cấp ở cả **22 khu vực** gồm Canada, Caribbean, Central Africa, Central America, East Africa, East of USA, Eastern Asia, Eastern Europe, North Africa, Northern Europe, Oceania, South America, South Asia, South of USA, Southeast Asia, Southern Africa, Southern Europe, US Center, West Africa, West Asia, West of USA, Western Europe.
* Cụm thứ 1:
* Gồm **18 Node**: Baby, Books, CDs, Cameras, Children's Clothing, Computers, Consumer Electronics, Crafts, DVDs, Garden, Health and Beauty, Men's Clothing, Music, Pet Supplies, Sporting Goods, Toys, Video Games, Women's Clothing.
* Ý nghĩa: những loại sản phẩm được cung cấp ở cả **3 khu vực** gồm Oceania, South Asia, Southeast Asia.
* Cụm thứ 2:
* Gồm **8 Node**: As Seen on TV!, Basketball, Golf Bags & Carts, Kids' Golf Clubs, Men's Golf Clubs, Soccer, Strength Training, Women's Golf Clubs.
* Ý nghĩa: những loại sản phẩm được cung cấp ở cả **3 khu vực** gồm Northern Europe, Southern Europe, Western Europe.
  1. Nhận xét
  + Gaussian Mixture Model là một mô hình xác suất giả định tất cả các điểm dữ liệu được tạo ra từ một hỗn hợp của 1 số hữu hạn của các phân phối Gaussian với các tham số không xác định và là phần mở rộng các ý tưởng đằng sau K-Means.
  + Vì vậy Gaussian Mixture Model (GMM) đã khắc phục được các nhược điểm khi sử dụng thuật toán K-Means và đã cho ra kết quả phân cụm tương tự khi sử dụng thuật toán Louvain.

1. Trực quan hóa các điểm chung trong các cụm

* Do K-Means có vẻ không detect cụm tốt với dataset này, còn Louvain thì có kết quả phân cụm tương tự như khi sử dụng GMM nên ở đây ta sẽ trực quan hóa các khu vực đã cung cấp các loại sản phẩm của các cụm được detect khi sử dụng thuật toán Louvain.
* Cụm 0: các loại sản phẩm thường được cung cấp trên toàn cầu.

Chart, bar chart, histogram

Description automatically generated

Hình 42. Các khu vực đã cung cấp các sản phẩm của cụm 0 khi dùng Louvain

* Cụm 1: các loại sản phẩm thưởng được cung cấp ở các khu vực Châu Á.

Chart, bar chart

Description automatically generated

Hình 43. Các khu vực đã cung cấp các sản phẩm của cụm 1 khi dùng Louvain

* Cụm 2: các loại sản phẩm thưởng được cung cấp ở các khu vực Châu Âu.

Chart, bar chart

Description automatically generated

Hình 44. Các khu vực đã cung cấp các sản phẩm của cụm 2 khi dùng Louvain

1. **TÀI LIỆU THAM KHẢO**
2. <https://youtube.com/playlist?list=PLoROMvodv4rPLKxIpqhjhPgdQy7imNkDn>
3. <https://www.coursera.org/learn/python-social-network-analysis>
4. <https://github.com/Geometrein/helsinki-city-bikes>