**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

Logo, company name

Description automatically generated

**BÁO CÁO CUỐI KỲ**Môn học : Mạng Xã Hội

Học kỳ I (2022-2023)

**ĐỀ TÀI:**

**PHÂN TÍCH DỮ LIỆU NHỮNG CUỘC KHỦNG BỐ Ở THỔ NHĨ KỲ**

Sinh viên thực hiện:

Trần Nhật Tân MSSV: 19522177

GVHD: TS. Nguyễn Thị Kim Phụng

Lớp: IS353.N12.HTCL

**Thành phố Hồ Chí Minh, tháng 11 năm 2022**

**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

Logo, company name

Description automatically generated

**BÁO CÁO CUỐI KỲ**Môn học : Mạng Xã Hội

Học kỳ I (2022-2023)

**ĐỀ TÀI:**

**PHÂN TÍCH DỮ LIỆU NHỮNG CUỘC KHỦNG BỐ Ở THỔ NHĨ KỲ**

Sinh viên thực hiện:

Trần Nhật Tân MSSV: 19522177

GVHD: TS. Nguyễn Thị Kim Phụng

Lớp: IS353.N12.HTCL

**Thành phố Hồ Chí Minh, tháng 11 năm 2022**

Mục lục

[**LỜI CẢM ƠN** 3](#_Toc123329073)

[**NHẬN XÉT CỦA GIẢNG VIÊN** 4](#_Toc123329074)

[**I. TỔNG QUAN** 5](#_Toc123329075)

[**1.Giới thiệu dataset** 5](#_Toc123329076)

[**2. Xác định bài toán** 5](#_Toc123329077)

[**3. Mô tả dữ liệu** 5](#_Toc123329078)

[**II. XỬ LÝ VÀ PHÂN TÍCH DỮ LIỆU** 7](#_Toc123329079)

[**1. Các thư viện được sử dụng** 7](#_Toc123329080)

[**2. Làm sạch dữ liệu** 8](#_Toc123329081)

[**3. Chuyển đổi dataframe thành đồ thị** 11](#_Toc123329082)

[**III. CÁC ĐỘ ĐO VÀ KẾT QUẢ** 15](#_Toc123329083)

[**1. Degree Centrality** 16](#_Toc123329084)

[**2. Betweenness Centrality** 18](#_Toc123329085)

[**3. Closeness Centrality** 21](#_Toc123329086)

[**4. Page Rank** 23](#_Toc123329087)

[**5. EigenVector Centrality** 26](#_Toc123329088)

[**IV.THUẬT TOÁN PHÁT HIỆN CỘNG ĐỒNG** 29](#_Toc123329089)

[**1. Thuật toán Girvan Newman** 29](#_Toc123329090)

[**2. Thuật toán Louvain** 32](#_Toc123329091)

[**V.** **ĐỒ THỊ 10 NOTE** 40](#_Toc123329092)

[**THAM KHẢO** 44](#_Toc123329093)

[**Hết.** 44](#_Toc123329094)

# **LỜI CẢM ƠN**

Trên thực tế không có sự thành công nào mà không gắn liền với những sự hỗ trợ, giúp đỡ dù ít hay nhiều, dù trực tiếp hay gián tiếp của người khác. Với lòng biết ơn sâu sắc nhất, đầu tiên em xin gửi lời cảm ơn chân thành đến tập thể quý Thầy Cô Trường Đại học Công nghệ thông tin – Đại học Quốc gia TP.HCM và quý Thầy Cô khoa Hệ thống thông tin đã giúp cho nhóm có những kiến thức cơ bản làm nền tảng để thực hiện đề tài này.

Đặc biệt nhóm chúng em xin gửi lời cảm ơn chân thành tới cô Nguyễn Thị Kim Phụng – giảng viên môn Mạng Xã Hội đã tận tình giúp đỡ, trực tiếp chỉ bảo, hướng dẫn em trong suốt quá trình làm đề tài. Nhờ đó, em đã tiếp thu được nhiều kiến thức bổ ích trong việc vận dụng cũng như kỹ năng làm đề tài. Nếu không có những lời hướng dẫn, dạy bảo của cô thì em nghĩ đề tài này của em rất khó có thể hoàn thiện được. Một lần nữa, em xin chân thành cảm ơn cô.

Cuối cùng, bản thân em đã làm việc hết công suất để hoàn thành tốt đề tài của mình. Xin chân thành cảm ơn!

# **NHẬN XÉT CỦA GIẢNG VIÊN**

# **I. TỔNG QUAN**

## **1.Giới thiệu dataset**

Khủng bố là hoạt động phá hoại, đe dọa bằng lời nói, hình ảnh hoặc video giết người do cá nhân hoặc tổ chức thực hiện làm thiệt mạng người, đặc biệt là thường dân, hoặc gây tổn thất cho xã hội và cộng đồng để tác động vào tâm lý đối phương gây hoang mang khiếp sợ, nhằm mục đích chính trị hoặc tôn giáo. Một đặc tính thống nhất của khủng bố là việc sử dụng bừa bãi bạo lực đối với những người không có khả năng chống cự với mục đích là sự nổi tiếng cho một nhóm, một phong trào, một cá nhân hoặc gây áp lực lên đối thủ chính trị buộc họ phải chấp nhận một giải pháp chính trị có lợi cho mình. Các tổ chức khủng bố có thể khai thác nỗi sợ hãi của con người để hỗ trợ đạt được những mục tiêu này. Đối tượng bị khủng bố gây thiệt hại có thể là tính mạng, sức khỏe, danh dự nhân phẩm, tài sản (của cá nhân, tổ chức hay của nhà nước) hoặc sự vững mạnh của một chính quyền nhà nước.

Khủng bố là mối quan tâm lớn đối với các quốc gia trên toàn thế giới, vì nó thường báo hiệu mở đầu cho chiến tranh và dân thường bị mắc kẹt trong cuộc chiến. Chúng ta không thể thay đổi thực tế mà chúng ta không hiểu đầy đủ, vì vậy đây là một số dữ liệu liên quan về chủ đề này. Trong đồ án này, em sẽ tập trung phân tích về độ phổ biến của những mục tiêu tấn công ở các thành phố. Môn học mạng xã hội có thể phân tích mạng phân bố các cuộc tấn công ở các thành phố trong quốc gia Thổ Nhĩ Kỳ và khai thác dữ liệu trên mạng này.

## **2. Xác định bài toán**

- Input: Tập dữ liệu ban đầu trên nguồn dữ liệu Kaggle đã qua tiền xử lý dữ liệu

- Ouput: Đưa ra độ đo, đưa ra cộng đồng phục vụ cho việc phân tích mạng xã hội “Turkey Terrorism”

## **3. Mô tả dữ liệu**

**a. Nguồn dữ liệu**

- Link dữ liệu: [terrorism-in-turkey | Kaggle](https://www.kaggle.com/datasets/senakaradeniz/terrorisminturkey)

**b. Mô tả dữ liệu**

**-** Mỗi dòng trong tập dữ liệu là một vụ khủng bố tại Thổ Nhĩ Kỳ.

- Dữ liệu gồm 547 dòng và 14 thuộc tính.

|  |  |  |
| --- | --- | --- |
| Tên cột | Kiểu dữ liệu | Ý nghĩa |
| Id | Int | Id của dữ liệu |
| Year | Int | Năm xảy ra vụ khủng bố |
| Country | String | Quốc gia (Thổ Nhĩ Kỳ) |
| Region | String | Vùng của quốc gia (Trung Đông) |
| city | String | Thành phố xảy ra vụ khủng bố |
| AttackType | String | Kiểu tấn công |
| Killed | Int | Số người tử vong |
| Wounded | Int | Số người bị thương |
| Target | String | Mục tiêu của cuộc tấn công khủng bố |
| Summary | String | Diễn biến chi tiết của cuộc khủng bố |
| Group | String | Tổ chức khủng bố |
| Target\_type | String | Loại mục tiêu của cuộc tấn công khủng bố |
| Weapon\_type | String | Loại vũ khí dùng để khủng bố |
| Motive | String | Động cơ khủng bố |

- Bảng dữ liệu:

Graphical user interface, table

Description automatically generated

**II. XỬ LÝ VÀ PHÂN TÍCH DỮ LIỆU**

## **1. Các thư viện được sử dụng**

Text

Description automatically generated

## **2. Làm sạch dữ liệu**

Dữ liệu sẽ được đọc từ file csv đưa vào dataframe. Sau đó xoá bỏ dữ liệu trùng lặp. Kết quả ta nhận được bộ dữ liệu 547 dòng và 2 cột.

A picture containing graphical user interface

Description automatically generated

Table

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Table

Description automatically generated

Graphical user interface, text, application

Description automatically generated

- Khi kiểm tra số lượng dữ liệu ta phân tích 162 thành phố và 18 mục tiêu tấn công với 511 cạnh.

## **3. Chuyển đổi dataframe thành đồ thị**

**a. Đồ thị 2 phía**

- Node: là tên của loại mục tiêu tấn công và tên các thành phố của nó.

- Edge: cho ta biết loại mục tiêu tấn công đó đã được thực hiện ở thành phố nào.

*Code hiển thị đồ thị 2 phía:*

A screenshot of a computer

Description automatically generated with medium confidence

*Đồ thị 2 phía:*

Diagram

Description automatically generated

- Khi nhìn vào đồ thị 2 phía thì ta thấy 1 thành phố có thể có nhiều loại mục tiêu tấn công, và 1 loại mục tiêu tấn công có thể xảy ra ở nhiều thành phố.

**b. Đồ thị 1 phía**

- Node: là các thành phố (city).

- Edge: Hai thành phố có cùng mục tiêu tấn công sẽ hợp thành 1 cạnh.

Điều này nói lên rằng các vụ khủng bố tại các thành phố khác nhau tại Thổ Nhĩ Kỳ có nhiều mục tiêu tấn công khác nhau, các thành phố khác nhau có thể có cùng mục tiêu tấn công của bọn khủng bố.

*Code hiển thị đồ thị 1 phía:*

*Graphical user interface, text, application

Description automatically generated*

*Đồ thị 1 phía:*

Chart, scatter chart

Description automatically generated

*- Xuất dữ liệu đồ thị 1 phía ra file csv để thực hiện trên Gephi:*

Graphical user interface, application

Description automatically generated

Table

Description automatically generated

# **III. CÁC ĐỘ ĐO VÀ KẾT QUẢ**

- Sau khi import file csv dữ liệu ở trên vào Data Laboratory trên Gephi ta thu được các độ đo cần thiết để so sánh kết quả giữa Gephi và Python

Graphical user interface, application, table

Description automatically generated

## **1. Degree Centrality**

- Hệ số này sẽ giúp chúng ta đo lường được số lượng của các mối quan hệ trực tiếp của một actor nào đó với các thành viên khác trong mạng lưới. Giá trị của hệ số này chạy từ 0.00 đến 1.00 và khi giá trị càng gần tới 1.00 thì tính trung tâm trực tiếp của actor càng lớn, tức là càng nằm ở vị trí trung tâm của mạng lưới. Công thức tính như sau:

A picture containing clock

Description automatically generated

Trong đó,

k = Tổng số các mối quan hệ trực tiếp của actor

n = Tổng số actor trong mạng lưới

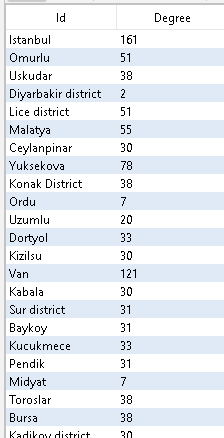
**\*So sánh kết quả Python và Gephi theo giá trị Degree giảm dần:**

*Python:*

Graphical user interface, text

Description automatically generated

*Gephi:*



## **2. Betweenness Centrality**

- Theo quan điểm của Freeman, một actor nào đó trong mạng lưới có thể ít gắn kết với các thành viên khác trong mạng lưới (tức hệ số trung tâm trực tiếp thấp), cũng không "gần gũi" lắm với mọi thành viên trong mạng lưới (tức hệ số trung tâm lân cận thấp), nhưng lại là "cầu nối" (bridge), là "nhà trung gian" cần thiết trong mọi cuộc trao đổi trong mạng lưới.

- Hệ số này cũng đi từ 0.00 đến 1.00. Khi một actor nào đó có hệ số trung tâm trung gian càng gần đến 1.00 thì số lượng quan hệ giữa các actor khác phải "thông qua" actor này càng nhiều và do đó ảnh hưởng của actor cũng càng lớn.

Cách tính hệ số trung tâm trung gian như sau:

Text

Description automatically generated with low confidence

Trong đó

n(j,z;x) = Tổng số lần làm "trung gian" của actori

n = Tổng số actor trong mạng

*Code Python:*

Graphical user interface, text, application, email

Description automatically generated

Text

Description automatically generated

*Gephi:*

*Table

Description automatically generated*

=> Độ đo betweenness là vai trò trung gian của các mối quan hệ giữa các thành phố trong mạng, độ đo càng cao thì sự giao thoa của các thành phố càng lớn

## **3. Closeness Centrality**

- Điểm yếu của hệ số trung tâm trực tiếp là nó chỉ tính các mối quan hệ trực tiếp của actor mà thôi nên chưa chắc actor có hệ số trung tâm trực tiếp cao là người "gần gũi" với mọi thành viên khác trong mạng. Tính gần gũi hay lân cận cũng là một trong những tiêu chí quan trọng thể hiện vị thế của actor trong mạng, bởi một actor càng gần gũi với các thành viên trong mạng lưới bao nhiêu thì actor đó càng dễ có nhiều thông tin, càng có nhiều uy thế và do đó càng dễ gây ảnh hưởng lên toàn bộ mạng lưới. Để đo lường hệ số này, chúng ta sẽ tính tổng số "bước" (step) của "đoạn đường" ngắn nhất (geodesic path) mà actor phải "đi" để đến được với tất cả các thành viên khác trong mạng lưới.

- Hệ số này cũng có giá trị đi từ 0.00 đến 1.00, càng gần đến 1.00 thì actor càng gần với mọi thành viên khác trong mạng lưới, tức đoạn đường phải đi để đến với mọi actor khác càng ngắn và ngược lại. Công thức tính hệ số này như sau:

Diagram

Description automatically generated with low confidence

Trong đó:

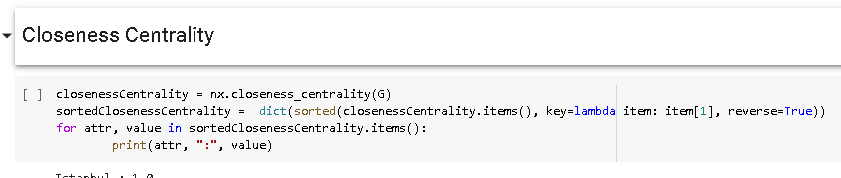
n = Tổng số actor trong mạng lưới

∑d (x,y) = Tổng số "bước" (step) của đoạn đường ngắn nhất mà actori phải đi để đến với mọi actor trong mạng

*Code Python:*

**\*So sánh kết quả Python và Gephi theo giá trị Degree giảm dần:**

*Python*

****

**Text

Description automatically generated with medium confidence**

*Gephi*

**Table

Description automatically generated**

=> Độ do closeness là độ gần khoảng cách của các node với các node khác trong mạng, độ đo càng cao thì thành phố đó đó xuất hiện mục tiêu tấn công có chung cạnh nhiều nhất với các thành phố khác.

## **4. Page Rank**

- PageRank là thứ hạng trang. PageRank được phát triển tại đại học Stanford bởi Lary Page (cũng bởi vậy mà có tên PageRank) và sau đó bởi Sergey Brin như một phần dự án công cụ tìm kiếm mới.Theo Google một cách tóm lược thì PageRank chỉ được đánh giá từ hệ thống liên kết đường dẫn. Trang của bạn càng nhận nhiều liên kết ( phải là dofollow ) trỏ đến thì mức độ quan trọng trang của bạn càng tăng.

Công thức PageRank có dạng như sau:

A picture containing text, watch, clock, gauge

Description automatically generated

Trong đó:

d: hằng số Google quy định. Thông thường, d = 0.85.

PR(pj): PageRank của các đỉnh đi vào đỉnh i.

L(j): số link out của các đỉnh đi vào đỉnh i.

*Code Python:*

Graphical user interface, text, application

Description automatically generated

**\*So sánh kết quả Python và Gephi theo giá trị Degree giảm dần:**

*Python*

Text

Description automatically generated

*Gephi*

Table

Description automatically generated

## **5. EigenVector Centrality**

- Egenvector là các vectơ (khác 0) không thay đổi hướng khi bất kỳ phép biến đổi tuyến tính nào được áp dụng. Nó chỉ thay đổi bởi một hệ số vô hướng. Tóm lại, chúng ta có thể nói, nếu A là một phép biến đổi tuyến tính từ một không gian vectơ V và **x** là một vectơ trong V, không phải là một vectơ 0, thì v là một ký hiệu riêng của A nếu A (X) là một bội số vô hướng của **x** .

Một  **Eigenspace** của vector **x** bao gồm một tập hợp tất cả các eigenvector có giá trị riêng tương đương gọi chung với vector zero. Mặc dù vậy, vectơ 0 không phải là một ký hiệu riêng.

Giả sử A là ma trận “n × n” và λ là giá trị riêng của ma trận A, khi đó **x** , một vector khác 0, được gọi là eigenvector nếu nó thỏa mãn biểu thức dưới đây;

A **x**  = λ **x**

**x** là một ký tự riêng của A tương ứng với giá trị riêng, λ.

*Code Python:*

Text

Description automatically generated

Text

Description automatically generated with medium confidence

**\*So sánh kết quả Python và Gephi theo giá trị Degree giảm dần:**

*Python*

Text

Description automatically generated

*Gephi*

Table

Description automatically generated

# **IV.THUẬT TOÁN PHÁT HIỆN CỘNG ĐỒNG**

## **1. Thuật toán Girvan Newman**

- Thuật toán lần đầu tiên được đề xuất bởi Freeman. Theo Freeman, các cạnh được coi là cạnh có số lượng con đường ngắn nhất giữa các cặp đỉnh khác nhau chạy qua nó. Cạnh nối có ảnh hưởng rất lớn đến dòng chảy của thông tin giữa các nút khác, đặc biệt là trong trường hợp thông tin lưu truyền trong mạng chủ yếu theo con đường ngắn nhất. Thuật toán điển hình nhất trong các thuật toán chia này là thuật toán Girvan-Newman. Để tìm các cạnh trong mạng nối hai đỉnh thuộc hai cộng đồng khác nhau, khái quát đây là cạnh có độ trung gian cao, và xác định độ do trung gian này bằng cách tính số đường đi ngắn nhất giữa các cặp đỉnh mà có qua nó. Với một đồ thị m cạnh và n 13 đỉnh thì thời gian tính toán cho giai đoạn này là O(mn) .Với đồ thị có trọng số, độ đo trung gian của cạnh có trọng số đơn giản được tính bằng độ đo trung gian của cạnh không có trọng số chia cho trọng số của cạnh đó. Nếu một mạng lưới bao gồm các cộng đồng hoặc nhóm chúng chỉ được liên kết nối yếu bằng một nhóm cạnh, thì tất cả các đường đi ngắn nhất giữa các cộng đồng khác nhau sẽ phải đi dọc theo một trong số ít các cạnh thuộc nhóm cạnh đó. Vì vậy, các cạnh kết nối các cộng đồng sẽ là cạnh có độ đô trung gian cao. Bằng cách loại bỏ các cạnh, thuật toán Girvan-Newman tách được thành các nhóm riêng biệt. Thuật toán được thực hiện theo các bước sau:

1. Tính độ đo trung gian cho tất cả các cạnh trong mạng.

2. Hủy bỏ các cạnh có độ trung gian cao nhất.

3. Tính lại độ trung gian cho tất cả các cạnh bị ảnh hưởng theo các cạnh đã loại bỏ.

4. Lặp lại từ bước 2 cho đến khi không còn các cạnh trung gian.

*Code cài đặt và sử dụng thuật toán:*

Graphical user interface, text, application, email

Description automatically generated

Diagram

Description automatically generated

Số cụm 2

**Danh sách các cụm:**

**Cụm 1:**

['Dortyol', 'Malatya', 'Caglayan', 'Bitlis', 'Konak District', 'Akdana', 'Mardinkapi', 'Izmit', 'Ceylanpinar', 'Mesepinar', 'Islahiye district', 'Iskenderun', 'Saricam District', 'Derik district', 'Taskoy', 'Silvan', 'Besikduzu', 'Semdinli', 'Sultangazi district', 'Cizre district', 'Siran', 'Adana', 'Kozluk', 'Cayirli', 'Sirnak', 'Nusaybin', 'Pulumur', 'Catak', 'Magara', 'Samsun', 'Pozanti', 'Yesilkoy', 'Erzincan', 'Dogansar District', 'Uzumlu', 'Kadikoy district', 'Baykoy', 'Gecitli', 'Beytussebap district', 'Dogubeyazit district', 'Kabala', 'Hizan district', 'Guclukonak District', 'Silopi', 'Saray', 'Celtiksuyu', 'Alacakaya', 'Karliova', 'Hani district', 'Nazimiye district', 'Belenoluk', 'Caldiran district', 'Isparta', 'Sur district', 'Geyiksuyu', 'Kucukmece', 'Aydin', 'Diyadin District', 'Gaziantep', 'Mezra', 'Aritis', 'Bulanik', 'Lice', 'Genc', 'Dicle', 'Sason', 'Istanbul', 'Diyadin', 'Tatvan', 'Konya', 'Kizilsu', 'Arabaslik', 'Bakacak', 'Toroslar', 'Cankiri', 'Resadiye District', 'Mersin', 'Seyhan District', 'Cat district', 'Cizre', 'Cukurca', 'Beytussebap', 'Suruc', 'Hatay', 'Gecitalan', 'Semdinli district', 'Ordu', 'Pendik', 'Payas', 'Daglica', 'Buca district', 'Servi', 'Antalya', 'Cigdemli', 'Ziyaret', 'Dazya', 'Ogulveren', 'Gebze district', 'Erzin', 'Yuksekova', 'Omurlu', 'Cukurca district', 'Ercis', 'Kulp district', 'Yuksekova District', 'Diyarbakir district', 'Cizre District', 'Yenikoy', 'Ercis district', 'Diyarbakir', 'Antakya', 'Yenimahalle District', 'Yuksekova district', 'Duzici district', 'Kigi', 'Akcay', 'Tokat', 'Elazig', 'Kiziltepe District', 'Bukardi', 'Eskisehir', 'Nusaybin district', 'Van', 'Beldeler', 'Reyhanli', 'Sanliurfa', 'Torul', 'Malazgirt', 'Genc district', 'Bademli', 'Of', 'Bursa', 'Batman', 'Kemer', 'Karakocan district', 'Ankara', 'Egirmec', 'Hani', 'Izmir', 'Hozat district', 'Kasimli', 'Ozalp', 'Dikbiyik', 'Yenisehir', 'Hakkari', 'Pervari', 'Midyat', 'Solhan', 'Kocakoy', 'Bingol', 'Mus', 'Refahiye', 'Cinar', 'Marmaris', 'Tunceli', 'Omerli', 'Kirikkale', 'Lice district', 'Pinarbasi', 'Uskudar']

**Cụm 2:**

['Kusadasi district', 'Cesme']

## **2. Thuật toán Louvain**

Phương pháp Louvain để phát hiện cộng đồng là một thuật toán để phát hiện các cộng đồng trong mạng. Nó tối đa hóa điểm mô-đun cho mỗi cộng đồng, trong đó mô-đun định lượng chất lượng của việc gán các nút cho cộng đồng. Điều này có nghĩa là đánh giá mức độ kết nối của các nút trong cộng đồng với mật độ cao hơn như thế nào so với mức độ kết nối của chúng trong một mạng ngẫu nhiên. Thuật toán Louvain là một thuật toán phân cụm phân cấp, hợp nhất một cách đệ quy các cộng đồng thành một nút duy nhất và thực hiện phân cụm mô-đun trên các đồ thị cô đọng

*Code cài đặt và sử dụng thuật toán LouvainGraphical user interface, text, application

Description automatically generated*

*Đồ thị phân cụm sử dụng Louvain*

Diagram

Description automatically generated

Graphical user interface, text, application

Description automatically generated

*Danh sách các cụm*

*A picture containing text

Description automatically generated*

Số cụm: 5

**Cụm 1:**

*Table

Description automatically generated*

**Cụm 2:**

Table

Description automatically generated

Cụm 3:

Table

Description automatically generated

Cụm 4:

Text

Description automatically generated

Cụm 5:

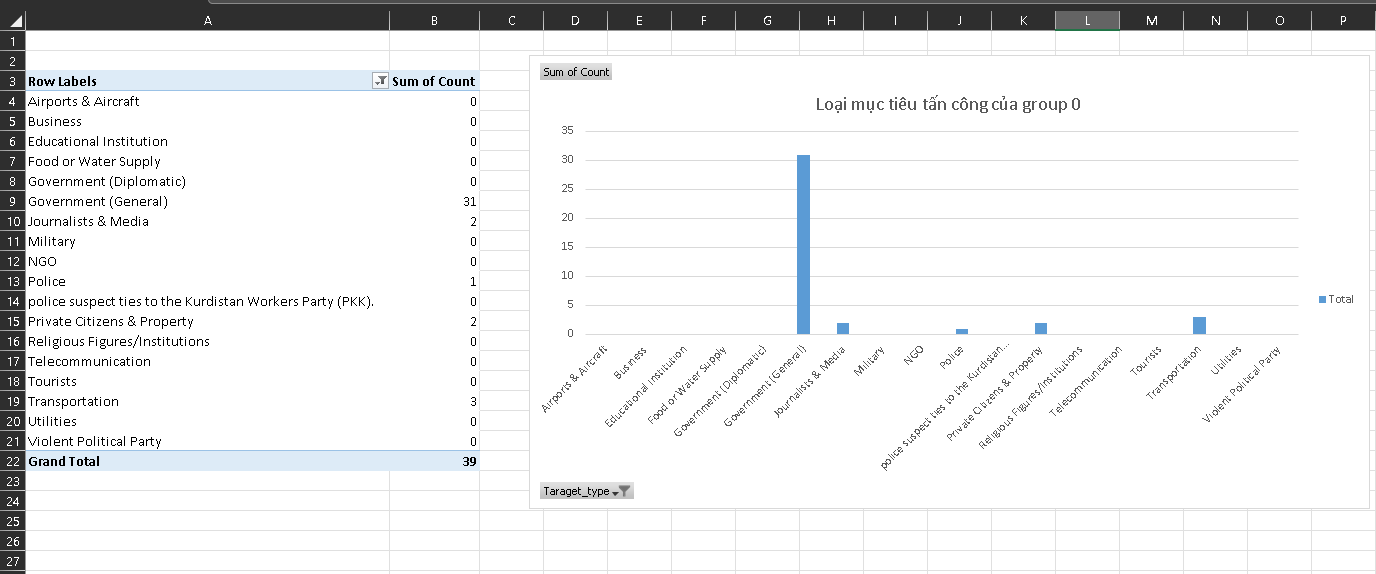
Table

Description automatically generated

Code xuất file excel cụm:

A picture containing text

Description automatically generated



Ta có thể thấy cụm đầu tiên là cụm gồm những thành phố có mục tiêu tấn công là Government.

Graphical user interface, chart

Description automatically generated

Cụm thứ 2 có loại mục tiêu tấn công là Military.

Graphical user interface, chart, application

Description automatically generated

Cụm 3 loại mục tiêu tấn công là Transportation.

Graphical user interface, chart

Description automatically generated

Mục tiêu tấn công chủ yếu của cụm 4 là Business và Private Citizens and property.

Graphical user interface, chart

Description automatically generated

Cụm cuối gồm nhiều loại mục tiêu tấn công.

# **V.** **ĐỒ THỊ 10 NOTE**

Gephi:

Diagram

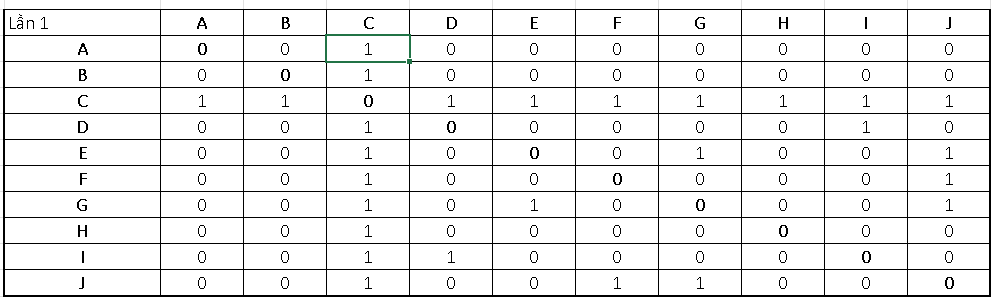
Description automatically generated

Các độ đo:

Table

Description automatically generated

Ma trận kề:



Bảng mapping:

Graphical user interface, chart

Description automatically generated

Betweeness:

Graphical user interface

Description automatically generated with medium confidence

Closeness:

Chart

Description automatically generated

Page rank:

Graphical user interface, application, table, Excel

Description automatically generated

EigenVector:

Calendar

Description automatically generated

File excel tính tay các độ đo sẽ được em đính kèm trong project ạ. Tên file “tinhtay.xlsx”.

# **THAM KHẢO**

1. Website môn học Mạng Xã Hội
2. "Page Rank Algorithm and Implementation," [Online]. Available: https://www.geeksforgeeks.org/page-rank-algorithm-implementation/.
3. "Girvan–Newman algorithm," [Online]. Available: https://en.wikipedia.org/wiki/Girvan%E2%80%93Newman\_algorithm.
4. "PageRank," [Online]. Available: https://vi.wikipedia.org/wiki/PageRank.
5. "Betweenness centrality," [Online]. Available: https://en.wikipedia.org/wiki/Betweenness\_centrality.
6. M. Telatnik, "How To Get Started with Social Network Analysis," 27 05 2020. [Online]. Available: https://towardsdatascience.com/how-to-get-started-with-social-network-analysis-6d527685d374.
7. "Is it possible to find closeness centrality using Gephi?," [Online]. Available: https://stackoverflow.com/questions/28727120/is-it-possible-to-find-closeness-centrality-using-gephi.
8. D. Liyan , L. Yongli , Y. Han , L. Huang and R. Mao , "The Algorithm of Link Prediction on Social Network," 17 09 2013. [Online]. Available: https://www.hindawi.com/journals/mpe/2013/125123/.

# **Hết.**