BỘ GIÁO DỤC VÀ ĐÀO TẠO

ĐẠI HỌC CÔNG NGHỆ TP.HCM

MẠNG XÃ HỘI

Biên soạn:

ThS. Hứa Thị Phượng Vân

www.hutech.edu.vn

MẠNG XÃ HỘI

Ấn bản 2024

Các ý kiến đóng góp về tài liệu học tập này,xin gửivề e-mail của ban biên tập : tailieuhoctap@hutech.edu.vn

MỤC LỤC

DANH MỤC HÌNH ẢNH

DANH MỤC BẢNG

HƯỚNG DẪN

MÔ TẢ MÔN HỌC

Môn học "Mạng xã hội" là học phần thú vị và hấp dẫn, mang đến cho sinh viên sự tiếp cận các khái niệm và ứng dụng cơ bản về mạng xã hội và phân tích mạng xã hội trong phân tích học tập. Sinh viên sẽ tìm hiểu về cấu trúc và sự phát triển của mạng xã hội và cách phân tích thực tế dữ liệu mạng quy mô lớn và cách suy luận về nó. Sinh viên cũng sẽ biết cách sử dụng Rstudio để thao tác, phân tích và trực quan hóa mạng dữ liệu và các mạng xã hội hiện nay.

Môn học này giúp sinh viên trang bị kỹ năng phân tích, kỹ năng giao tiếp hiệu quả thông qua viết, thuyết trình, thảo luận, đàm phán, làm chủ tình huống, sử dụng hiệu quả các công cụ và phương tiện hiện đại. Sinh viên cũng cần có năng lực lập kế hoạch, điều phối, quản lý, khai thác, bảo trì các hệ thống, phần mềm và các ứng dụng, giải pháp thuộc lĩnh vực công nghệ phần mềm, hệ thống thông tin và mạng máy tính..

NỘI DUNG MÔN HỌC

BÀI 1. TỔNG QUAN VỀ MẠNG XÃ HỘI

BÀI 2. ĐỒ THỊ MẠNG XÃ HỘI

BÀI 3. CỘNG ĐỒNG MẠNG XÃ HỘI

BÀI 4. DỰ ĐOÁN LIÊN KẾT MẠNG XÃ HỘI

BÀI 5. PHÂN TÍCH MẠNG XÃ HỘI

BÀI 6. MỘT SỐ CHƯƠNG TRÌNH PHÂN TÍCH MẠNG XÃ HỘI

YÊU CẦU MÔN HỌC

Sinh viên cần đảm bảo một số yêu cầu sau:

- *Có kiến thức cơ bản về thống kê và toán học.*

- *Hiểu biết về cơ bản của ngôn ngữ lập trình, hoặc có khả năng nhanh chóng nắm bắt kiến thức lập trình.*

- *Có khả năng tìm hiểu thêm và đề xuất các dự án phân tích dữ liệu tự chọn để áp dụng kiến thức học được vào thực tế.*

CÁCH TIẾP CẬN NỘI DUNG MÔN HỌC

Môn học "Phân tích và trực quan dữ liệu" mở đầu với việc khám phá sâu rộng về phân tích dữ liệu. Sinh viên bắt đầu từ cơ bản với tổng quan về phân tích và trực quan hóa dữ liệu, sau đó nắm vững ngôn ngữ lập trình R để triển khai các phân tích.

Việc thực hành cài đặt R và xử lý dữ liệu giúp sinh viên làm quen với các thao tác cơ bản như lọc, sắp xếp, và nối bảng dữ liệu. Khám phá thống kê dữ liệu và trực quan hóa thông qua biểu đồ và đồ thị đóng vai trò quan trọng trong việc mô tả và hiểu rõ dữ liệu.

Môn học mở rộng kiến thức vào lĩnh vực học máy cơ bản, hướng dẫn cách áp dụng mô hình để dự đoán và phân loại dữ liệu. Hồi quy và phân lớp được giảng dạy kỹ lưỡng để sinh viên có thể sử dụng chúng hiệu quả.

Cuối cùng, môn học đưa vào các kỹ thuật phân tích dữ liệu nâng cao như giảm số chiều và phân cụm, giúp sinh viên sáng tỏ cấu trúc ẩn trong dữ liệu và mở rộng tầm nhìn về thế giới của phân tích dữ liệu.

PHƯƠNG PHÁP ĐÁNH GIÁ MÔN HỌC

- Để học hiệu quả môn này, người học cần tiếp cận mỗi buổi học với chiến lược ôn tập cẩn thận. Người học nên bắt đầu bằng việc ôn tập kiến thức đã học, trả lời câu hỏi ôn tập và hoàn thành bài tập để đảm bảo sự củng cố kiến thức.
- Trước khi bắt đầu bài học mới, người học cần đọc mục tiêu để nắm rõ hướng đi. Việc đọc nội dung bài học cẩn thận giúp nắm bắt thông tin chính. Khi hoàn thành mỗi phần của bài học, người học nên tự kiểm tra bằng cách trả lời câu hỏi ôn tập, giúp củng cố và ghi nhớ kiến thức. Cuối cùng, việc thực hiện bài tập sau mỗi bài học giúp áp dụng kiến thức vào thực tế, đồng thời kiểm tra sự hiểu biết của bản thân người học.
- Đánh giá của môn học này chia thành hai phần chính:
    - Điểm quá trình (trọng số 50% điểm môn học) đánh giá sự tham gia chuyên cần cùng hiệu quả trong việc hoàn thành bài tập.

- Điểm cuối kỳ (trọng số 50% điểm môn học) đánh giá kiến thức thông qua báo cáo đề tài đồ án cuối kì và ứng dụng kiến thức đã học trong thời gian học tập suốt học kỳ.

BÀI 1. TỔNG QUAN VỀ MẠNG XÃ HỘI

Học xong bài này, sinh viên sẽ đạt được những mục tiêu sau:

- *Hiểu biết cơ bản về lịch sử phát triển của Mạng xã hội.*

- *Hiểu biết cơ bản về ý nghĩa của việc phân tích Mạng xã hội và một số Mạng xã hội hiện nay.*

- *Nắm được một số dạng toán cơ bản về phân tích mạng xã hội.*

- *Thể hiện thái độ tích cực, kiên nhẫn, hợp tác và sẵn sàng học hỏi trong quá trình học tập và ứng dụng kiến thức về Mạng xã hội.*

1.1 MẠNG XÃ HỘI LÀ GÌ?

Mạng xã hội (social network) là dịch vụ nối kết các thành viên cùng sở thích trên internet lại với nhau với nhiều mục đích khác nhau không phân biệt không gian và thời gian. Những người tham gia vào mạng xã hội còn được gọi là cư dân mạng.

Mạng xã hội có những tính năng như chat, email, phim ảnh, voice chat, chia sẻ file, blog và bình luận. Mạng xã hội đổi mới hoàn toàn cách liên lạc và trở thành một phần tất yếu trong đời sống hàng ngày cho hàng trăm triệu thành viên trên khắp thế giới. Có nhiều phương thức để các thành viên tìm kiếm bạn bè, đối tác như dựa theo nhóm (ví dụ: tên trường, tên thành phố, quê quán, nơi công tác,...), dựa trên thông tin cá nhân (email, số điện thoại), hoặc dưa trên sở thích cá nhân (thể thao, phim ảnh, sách báo, ca nhạc, hoặc ẩm thực,…), lĩnh vực quan tâm (kinh doanh, mua bán,…).

Hiện nay trên thế giới có hàng trăm mạng xã hội khác nhau như MySpace, Facebook, Orkut, Bebo, CyWorld, Mixi, YuMe, Twitter,…

Mạng xã hội là một cấu trúc mang tính xã hội tạo thành từ các node, mỗi node là một actor (cá nhân hay tổ chức). Mạng xã hội kết nối các thành viên, người dùng trên Internet lại với nhau thông qua các liên kết và theo các tiêu chí nào đó, với nhiều mục đích khác nhau, không phân biệt không gian và thời gian thông qua mạng máy tính. Nói cách khác, mạng xã hội là mạng máy tính lớn, nhiều thành viên và đơn giản là hệ thống của những mối quan hệ con người với con người, do đó, bản thân Facebook hay Twitter không phải là mạng xã hội mà là những dịch vụ trực tuyến được tạo lập để xây dựng và phản ánh mạng xã hội.

Hình 1.1 biểu diễn một mạng xã hội. Có thể biểu diễn mạng xã hội bằng đồ thị. Các đỉnh trong đồ thị đại diện cho các cá nhân, thực thể, cơ quan. Các cung đại diện cho tương tác giữa các đỉnh.

Hình 1.1. Mạng xã hội

Các đỉnh của đồ thị được dùng để biểu diễn các node của mạng xã hội. Các cung dùng để biểu diễn liên kết giữa các đỉnh, các cung có thể có hướng hoặc vô hướng và có thể được đánh trọng số.

1.1.1 Lịch sử hình thành mạng xã hội

Năm 1995 với sự ra đời của trang Classmate với mục đích kết nối bạn học, tiếp theo là sự xuất hiện của SixDegrees vào năm 1997 với mục đích giao lưu kết bạn dựa theo sở thích.

Năm 2004, MySpace ra đời với các tính năng như nhúng video và nhanh chóng thu hút hàng chục ngàn thành viên mới mỗi ngày, các thành viên cũ của Friendster cũng lũ lượt chuyển qua MySpace và trong vòng một năm, MySpace trở thành mạng xã hội đầu tiên có nhiều lượt xem hơn cả Google và được tập đoàn News Corporation mua lại với giá 580 triệu USD.

Năm 2006, sự ra đời của Facebook đánh dấu bước ngoặt mới cho hệ thống mạng xã hội trực tuyến với nền tảng lập trình "Facebook Platform" cho phép thành viên tạo ra những công cụ (apps) mới cho cá nhân mình cũng như các thành viên khác dùng. Facebook Platform nhanh chóng gặt hái được thành công vượt bậc, mang lại hàng trăm tính năng mới cho Facebook và đóng góp không nhỏ cho con số trung bình 19 phút mà các thành viên bỏ ra trên trang này mỗi ngày. Cũng tại thời điểm này, mạng xã hội Twitter ra đời, Twitter đã trở thành một hiện tượng phổ biến toàn cầu. Những tweet có thể chỉ là dòng tin vặt cá nhân cho đến những cập nhật thời sự tại chỗ kịp thời và nhanh chóng hơn cả truyền thông chính thống.

1.1.2 Các khả năng của mạng xã hội

Mạng xã hội cho phép:

- Tạo ra một hệ thống trên nền Internet cho phép người dùng giao lưu và chia sẻ thông tin một cách hiệu quả.
- Xây dựng một mẫu định danh trực tuyến nhằm phục vụ những yêu cầu của cộng đồng.
- Nâng cao vai trò tạo lập và quản lý các quan hệ dựa trên những mối quan tâm chung trong những cộng đồng, thúc đẩy sự liên kết các tổ chức xã hội.
- Cung cấp tính năng: chat, email, phim ảnh, voice chat, chia sẻ file, hình ảnh và bình luận,…

1.1.3 Đặc điểm của mạng xã hội

Mạng xã hội có những đặc điểm sau đây:

- Kích thước lớn: mạng có hàng triệu, hàng tỷ node;
- Trọng số của cạnh: những kết nối giữa các node thường có trọng số nhằm mô tả mức độ tương tác giữa các node. Ví dụ những trọng số phản ánh số lượng truy cập email giữa các cá nhân,…;
- Tích chất động của mạng xã hội: tức là số lượng những node và liên kết thay đổi theo thời gian;
- Thuộc tính của node: mỗi node có thể có một tập các thuộc tính hoặc tính năng. Ví dụ: tuổi, giới tính, vị trí, chất lượng,… Các node có thể thuộc vào những kiểu khác nhau như loại nhà cung cấp, người mua hàng và những cung có thể chỉ tồn tại giữa những node của các kiểu khác nhau.

1.1.4 Những khó khăn trong xử lý dữ liệu trên mạng xã hội

- Nguồn dữ liệu: Các mạng xã hội không phải là nguồn dữ liệu tốt. Do có rất nhiều mạng xã hội, mỗi mạng lại có những quy tắc, những ràng buộc riêng và thu hút những kiểu người dùng là khác nhau;
- Thu thập số lượng lớn dữ liệu cho việc phân tích dữ liệu;
- Lọc tin nhắn và thư rác thực sự là vấn đề khó khăn trong nhiều mạng xã hội;
- Mạng xã hội thực tế thường là những mạng động. Với số lượng node và liên kết thay đổi theo thời gian.

1.2 PHÂN TÍCH MẠNG XÃ HỘI

Phân tích mạng xã hội là lĩnh vực nghiên cứu nhằm nhận thức sự phức tạp xã hội bằng cách diễn tả và phân tích các mạng xã hội với mô hình toán học dựa trên phân tích đồ thị. Một đồ thị được phân thành 2 nhóm: đồ thị có hướng và vô hướng. Node đại diện cho tác nhân (người, nhóm, tổ chức,…), các liên kết đại diện mối liên hệ giữa các tác nhân.

1.2.1 Bài toán tìm kiếm thông tin trên mạng xã hội

Các công nghệ tìm kiếm hiện nay nói chung vẫn chỉ dừng lại ở mức tìm kiếm nội dung trong các bài viết, tin nhắn được đăng tải trên các mạng xã hội. Trong khi nhu cầu tìm hiểu và phân tích thông tin không chỉ ở khả năng tìm kiếm nội dung thông thường, mà còn nhận dạng và phân tích các đối tượng và các mối quan hệ giữa chúng. Hiện có một số phần mềm cho phép phân tích, xử lý các thông tin dựa trên quan hệ kiểu. Tuy nhiên có thể nhận thấy rằng, hầu hết những công cụ đó cần phải có một cơ sở tri thức ban đầu để tạo ra đồ thị quan hệ, thông thường là từ một kiểu file như csv, xml hay các bảng CSDL…, từ đó mới bắt đầu thực thi các phân tích liên quan tới đồ thị thể hiện các mối quan hệ. Việc thu thập các node trong mạng xã hội, quan hệ giữa các node hay các thuộc tính khác thường không được định nghĩa mà có thể nhờ một phần mềm khác, hay cũng có thể do người dùng trực tiếp đưa vào. Cách thức này hạn chế ở chỗ khó nắm bắt được các thay đổi trên mạng xã hội vốn dĩ có tính chất động và thay đổi theo thời gian thực. Điểm hay nhất là tạo ra các tri thức tự động từ phân tích dữ liệu được thu thập trên mạng xã hội.

1.2.2 Bài toán phân tích mạng xã hội dựa trên cấu trúc mạng xã hội

Việc khai thác thông tin từ mạng xã hội phục vụ các mục tiêu kinh doanh, quảng bá… đang thu hút nhiều người. Phân tích mạng xã hội là nghiên cứu các mô hình tương tác giữa các thực thể xã hội. Lĩnh vực này đang thu hút sự chú ý rất nhiều của các ngành như xã hội học, sinh học,… Trong đó một số bài toán quan trọng như:

Bài toán dự đoán liên kết trên mạng xã hội

Phần lớn những nghiên cứu trong khai phá mạng xã hội tập trung vào bài toán dự đoán liên kết để lấy được những thông tin thú vị trên mạng xã hội như: bài toán gán nhãn các node. Tuy nhiên do mạng xã hội có tính chất động nên những liên kết có thể thay đổi theo thời gian. Ví dụ, liên kết tình bạn liên tục được tạo ra hoặc ngắt kết nối theo thời gian. Câu hỏi đặt ra là làm thể nào có thể dự đoán liên kết giữa hai nút trong

mạng. Quá trình dự đoán có thể sử dụng cấu trúc của mạng, thông tin thuộc tính của những node khác hoặc dựa trên quan sát những liên kết hiện có trong mạng.

Bài toán khám phá cộng đồng trên mạng xã hội

Cộng đồng là một tập những cá thể được tổ chức với nhau bằng lợi ích chung. Họ có thể là cùng sở thích, cùng mục tiêu nào đó hoặc cùng nghề nghiệp. Trong cộng đồng các cá thể có quan hệ mật thiết, tương tác với nhau nhiều hơn tương tác với những cá thể trong cộng đồng khác. Mặc dù không có định nghĩa tiêu chuẩn nào cho một cluster hoặc cộng đồng nhưng trong lý thuyết đồ thị có thể xem cộng đồng như là một đồ thị con mà ở đó tập đỉnh đại diện cho các thực thể trong cộng đồng, còn tập cung biểu diễn cho tương tác giữa các thực thể đó và mật độ các cung nội bộ lớn hơn mật độ của những cung còn lại của mạng.

Nhiều mạng xã hội được mô tả thuộc tính có chứa cấu trúc cộng đồng. Có nghĩa là chúng được chia thành nhiều nhóm đỉnh với kết nối dày đặc bên trong mỗi nhóm và ít kết nối hơn qua các nhóm khác. Trong đó đỉnh và cạnh đại diện tương ứng cho người dùng mạng xã hội và những tương tác của họ. Những thành viên trong mỗi cộng đồng trong mạng thường chia sẻ những mối quan tâm chung như sở thích về âm nhạc, điện ảnh và thảo luận những chủ đề, họ có xu hướng tương tác thường xuyên với nhau hơn là tương tác với những thành viên bên ngoài. Phát hiện cộng đồng trong một mạng là tích hợp những đỉnh của mạng vào trong những nhóm mà theo cách này những đỉnh (node) bên trong mỗi nhóm kết nối nhiều hơn những đỉnh bên ngoài. Phát hiện cấu trúc cộng đồng là nền tảng để phát hiện ra mối quan hệ giữa cấu trúc và chức năng trong mạng phức hợp và cho những ứng dụng thực tế như trong sinh học: phát hiện ra cấu trúc protein, mạng xã hội. Phát hiện những cộng đồng giúp cho chúng ta có hiểu biết sâu sắc về cấu trúc nội bộ của nó, cũng như các nguyên tắc tổ chức của nó. Hơn thế nữa, việc phát hiện ra cấu trúc cộng đồng giúp ích rất nhiều trong các lĩnh vực khác nhau: như trong kinh doanh, các nhà kinh doanh tìm ra được các nhóm người mua hàng khác nhau trong hệ thống mua hàng của mình để từ đó ứng với mỗi nhóm có chiến lược, chính sách bán hàng cụ thể, biết được sản phẩm nào đang bán chạy để từ đó cung cấp sản phẩm, dịch vụ đúng với thị hiếu của người tiêu dùng. Ngoài ra, việc phát hiện cộng đồng trong mạng cũng giúp việc ngăn ngừa tiềm năng nguy cơ lây lan vi rút,

những "bệnh" tuyên truyền trên mạng xã hội. Qua việc phát hiện những cộng đồng trên mạng xã hội, giúp ta tìm được nhanh chóng những "người quan trọng" (key player) trong những nhóm người đó vì những key player đóng vài trò quan trọng trong việc lan truyền nhanh thông tin trên mạng.

1.3 MỘT SỐ MẠNG XÃ HỘI TIÊU BIỂU

Một số mạng xã hội phổ biến trên thế giới và ở Việt Nam được giới thiệu trong phần này.

1.3.1 Facebook

Facebook là một phương tiện truyền thông xã hội và dịch vụ mạng xã hội trực tuyến, một tiện tích có tính xã hội để kết nối mọi người với bạn bè và những người đang sống, học tập và làm việc cùng nhau. Facebook được sử dụng để giữ liên lạc với bạn bè, cho phép tải không giới hạn hình ảnh, đưa các liên kết và video…[1]

Mark Zuckerberg sáng lập Facebook cùng với Eduardo Saverin, Andrew McCollm, Dustin Moskovitz và Chris Hughes khi ông còn là sinh viên tại Đại học Harvard.

Facebook được bắt đầu bằng phiên bản của Đại học Harvard với tên gọi Facemash. Mark Zuckerberg, khi đang học năm thứ hai tại Harvard, đã dựng nên Facemash vào năm 2003. Việc đăng ký thành viên ban đầu giới hạn trong những sinh viên của Đại học Harvard, và trong tháng đầu tiên, hơn một nửa số sinh viên đại học tại Harvard đã đăng ký dịch vụ này. Sau đó Eduardo Saverin (lĩnh vực kinh doanh), Dustin Moskovitz (lập trình viên), Andrew McCollum (đồ họa) và Chris Hughes nhanh chóng tham gia cùng với Zuckerberg để giúp quảng bá mạng xã hội trên website.

Người dùng truy cập Facebook.com và ứng dụng thông qua trình duyệt và Internet. Facebook cho phép người dùng lựa chọn cài đặt bảo mật của riêng mình và lựa chọn những người có thể nhìn thấy phần cụ thể của tiểu sử của họ. Tuy nhiên ứng dụng không được đặt tại máy chủ của Facebook mà được lưu trên máy chủ của chính người

[1] https://vi.wikipedia.org/wiki/Facebook

tạo ra ứng dụng đó. Facebook Platform cũng cung cấp một giao diện cho người viết ứng dụng.

1.3.2 Twitter

Twitter cũng là một trong những mạng xã hội lớn nhất thế giới và là trang web được truy cập nhiều thứ năm trên thế giới. Người dùng có thể chia sẻ tin nhắn văn bản ngắn, hình ảnh và video trong các bài đăng (trước đây là " tweet ") và thích hoặc đăng lại nội dung của người dùng khác. Twitter cũng bao gồm nhắn tin trực tiếp , gọi video và âm thanh, dấu trang, danh sách và cộng đồng cũng như Spaces, một tính năng âm thanh xã hội. Người dùng có thể bỏ phiếu về ngữ cảnh được thêm bởi người dùng được phê duyệt bằng tính năng Ghi chú cộng đồng. Một đặc điểm nổi bật của Twitter là mỗi người dùng chỉ được phép đăng một tin (tweet) có chiều dài không vượt quá 280 ký tự.[2]

Ngày nay Twitter giữ một vai trò quan trọng trong xã hội, chính trị, truyền thông, thể thao và nhiều lĩnh vực khác. Hiện nay có hơn 200 triệu người dùng Twitter và gửi đi 400 triệu đoạn tweet mỗi ngày.

1.3.3 MySpace

Myspace[3] là một trang mạng xã hội được thành lập từ năm 2003 tại California, Hoa Kỳ, cung cấp mạng lưới thông tin tương tác giữa người dùng với bạn bè của họ, cho phép người dùng tạo những hồ sơ cá nhân, viết blog, lập nhóm, tải hình ảnh lên, lưu trữ nhạc và video cho giới trẻ cũng như người lớn trên khắp thế giới.

1.3.4 LinkedIn

LinkedIn[4] là một trang mạng dịch vụ xã hội, được thành lập từ năm 2002 bởi Reid Hoffman – hiện nay là Chủ tịch Hội đồng quản trị. Người dùng chủ yếu là những thành

[2] https://en.wikipedia.org/wiki/Twitter

[3] https://vi.wikipedia.org/wiki/Myspace

[4] https://vi.wikipedia.org/wiki/LinkedIn

viên chuyên nghiệp về hệ thống mạng. LinkedIn cho phép người dùng (người làm việc và nhà tuyển dụng) tạo hồ sơ bao gồm một sơ yếu lý lịch mô tả kinh nghiệm làm việc, trình độ học vấn và đào tạo, kỹ năng và ảnh cá nhân của họ. Nhà tuyển dụng có thể liệt kê các công việc và tìm kiếm các ứng viên tiềm năng. Người dùng có thể tìm thấy công việc, con người và cơ hội kinh doanh được đề xuất bởi một người nào đó trong mạng lưới liên hệ của người khác. Người dùng có thể lưu (tức là đánh dấu) các công việc mà họ muốn đăng ký. Người dùng cũng có khả năng theo dõi các công ty khác nhau. LinkedIn cho phép các thành viên thực hiện "kết nối" với nhau trong một mạng xã hội trực tuyến có thể đại diện cho các mối quan hệ trong thế giới thực.

LinkedIn có đến 56% người sử dụng đến từ bên ngoài nước Mỹ. Các quốc gia có số người sử dụng chiếm đa số có Mỹ, Ấn Độ, U.K và Brasil. Các nước châu Âu có hoạt động mạnh nhất gồm Hà Lan, Pháp và Ý.

Nam giới chiếm tỉ lệ 61% trên tổng số thành viên. Người dùng các lứa tuổi từ 25-34 và 35-54 chiếm 36%, trong khi độ tuổi 18-24 tuổi chiếm 21%.

1.4 THU THẬP DỮ LIỆU TỪ MẠNG XÃ HỘI

Hiện nay các mạng xã hội lớn đều cho phép truy xuất dữ liệu qua các API dạng Webservice nhưng yêu cầu phải lưu ý về các chính sách sử dụng và thu thập dữ liệu. Ví dụ, Facebook chỉ cho phép truy xuất dữ liệu của chính bản thân và các Page hay Group mở. Ngoài ra, một số Partner được phép sử dụng một số kênh mất chi phí. Bên cạnh Facebook, Twitter cho phép người dùng thu thập và sử dụng toàn bộ dữ liệu trên Twitter. Trong khi đó, LinkedIn thì lại bắt người dùng trả phí cho việc đăng tuyển và tìm ứng cử viên công việc.

1.4.1 Thu thập dữ liệu từ Twitter

Search API

Cho phép người dùng thu thập dữ liệu dựa trên từ khóa tìm kiếm và một số tiêu chí lọc dữ liệu, sau đó dữ liệu sẽ được trả về trong 7 ngày gần nhất.

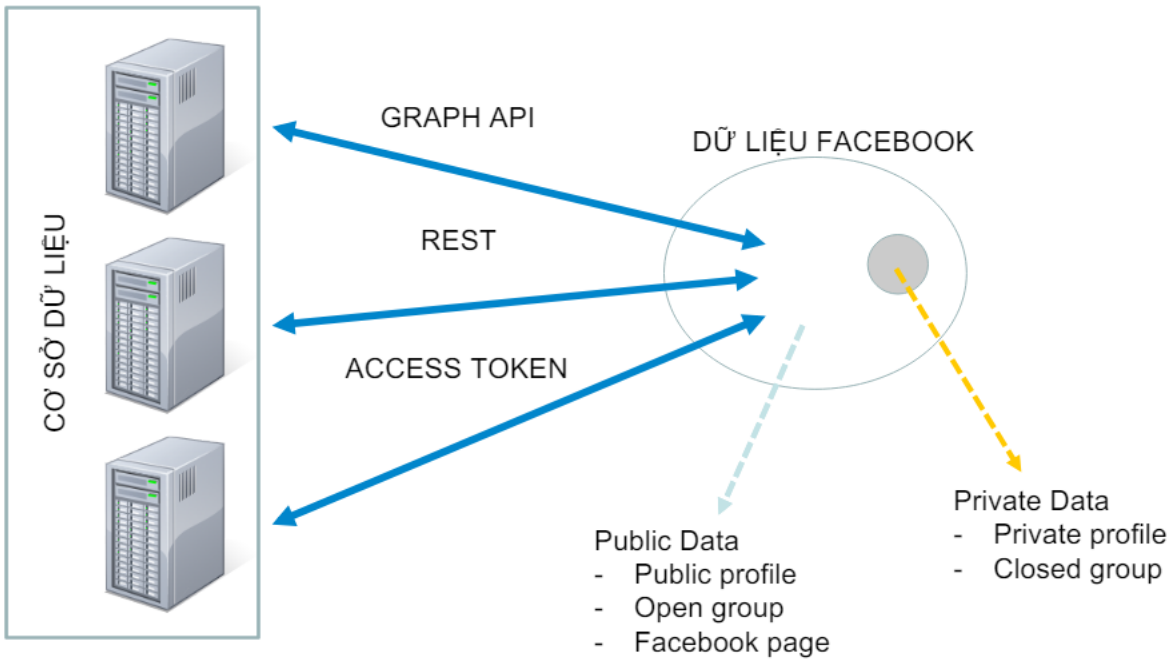
Streaming API

Người dùng được phép truy xuất để lấy dữ liệu Twitter theo thời gian thực. Đồng thời cho phép thiết lập các tham số liên quan đến địa điểm đưa các tweet.

Thư viện

Tùy vào khả năng và hiểu biết, con người có thể sử dụng các ngôn ngữ lập trình khác nhau để hỗt trợ như Java (Twitter4J, jTwitter) hay Python (Tweepy).

1.4.2 Thu thập dữ liệu từ Facebook



Hình 1.2. Thu thập dữ liệu từ Facebook

Graph API

Graph API tương tác với dữ liệu Open group, page, user tham gia app và tài khoản của chính bản thân.

Access Token

Gồm:

- User access token: chỉ truy xuất đến thông tin cá nhân và một số thông tin của bạn bè trực tiếp.

- App access token: chỉ truy xuất đến thông tin của các user tham gia vào app (với điều kiện user cho phép)
- Page access token: chỉ truy xuất vào thông tin của page.

Thư viện

Một số ngôn ngữ lập trình hỗ trợ như Java (RestFB, Facebook4J) hay Python (Facebook SDK for Python).

1.5 BÀI TẬP

1. Cho biết các mặt tích cực và tiêu cực của mạng xã hội?
2. Cho biết cách thu thập dữ liệu từ Facebook, Twitter và LinkedIn.
3. So sánh tính năng của Facebook, Twitter, LinkedIn?
4. Cho biết cách ứng dụng Facebook trong hoạt động kinh doanh, quảng cáo?
5. Cho biết ứng dụng của các bài toán phân tích mạng xã hội vào các mặt của đời sống?

BÀI 2. ĐỒ THỊ MẠNG XÃ HỘI

Học xong bài này, sinh viên sẽ đạt được những mục tiêu sau:

- *Hiểu biết cơ bản về lý thuyết đồ thị cơ bản, tính toán các số đo trong đồ thị mạng xã hội, đồ thị có dấu và cách sử dụng NetworkX để tính toán các số đo trong đồ thị mạng xã hội.*

- *Thể hiện thái độ tích cực, kiên nhẫn, hợp tác và sẵn sàng học hỏi trong quá trình học tập và ứng dụng kiến thức về Đồ thị mạng xã hội.*

2.1 LÝ THUYẾT ĐỒ THỊ CƠ BẢN

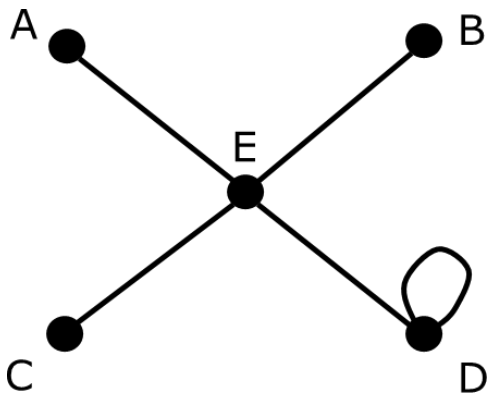
2.1.1 Đồ thị

Đồ thị $G = (V, E)$ gồm một tập V gọi là tập đỉnh và một tập E gọi là tập cạnh hay cung. Tập $E \subseteq V^2$ gồm các cặp phần tử của V. Giả sử u và v là hai đỉnh của đồ thị G $(u, v \in V)$, nếu cặp đỉnh (u, v) không được sắp thứ tự thì (u, v) gọi là cạnh nối hai đỉnh u và v. Ngược lại, nếu cặp đỉnh (u, v) được sắp thứ tự thì (u, v) gọi là cạnh có hướng (hay cung), trong đó u được gọi là đỉnh đầu và v được gọi là đỉnh cuối.

Đồ thị vô hướng là đồ thị chỉ chứa các cạnh trong đồ thị vô hướng, cạnh (u, v) tương đương với cạnh (v, u).

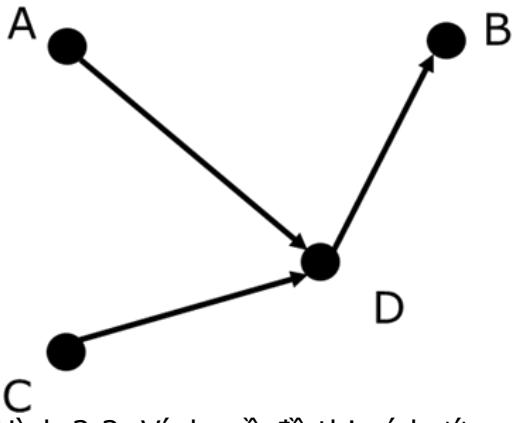
Đồ thị có hướng là đồ thị chỉ chứa các cạnh có hướng (cung). Trong đồ thị có hướng, cung (u, v) khác với cung (v, u).

Về mặt hình học, mỗi đỉnh trong đồ thị vô hướng được biểu diễn bởi một điểm, mỗi cạnh được biểu diễn bởi đường nối giữa hai điểm. Hình 2.1 biểu diễn đồ thị vô hướng có tập đỉnh gồm 5 đỉnh {A, B, C, D, E}, và tập cạnh gồm 5 cạnh {(A, E), (B, E), (C, E), (D, E), (D, D)}.



Hình 2.1. Ví dụ về đồ thị vô hướng

Trong đồ thị có hướng, đỉnh cũng được biểu diễn bởi một điểm, tuy nhiên, mỗi cạnh được biểu diễn bởi một đường có hướng (mũi tên) từ đỉnh đầu sang đỉnh cuối. Hình 2.2 biểu diễn đồ thị có hướng có tập đỉnh gồm 3 đỉnh {A, B, C}, và tập cung gồm 3 cung {(A, D), (D, B), (C, D)}.



Hình 2.2. Ví dụ về đồ thị có hướng

Một số khái niệm trên đồ thị:

Khuyên: Cạnh nối một đỉnh với chính nó được gọi là một khuyên. Trong đồ thị trên Hình 2.1, (D, D) là một khuyên.

Đỉnh kề và cạnh liên thuộc: Trong đồ thị $G = (V, E)$, hai đỉnh u, v ∈ V (u ≠ v) được gọi là kề nhau nếu tồn tại cạnh e = (u, v) ∈ E. Khi đó, cạnh e được gọi là liên thuộc với đỉnh u và v.

Đồ thị trong Hình 2.1 có các cặp đỉnh kề sau: A và E, B và E, C và E, D và E. (A, E) là cạnh liên thuộc với đỉnh A và E, (B, E) là cạnh liên thuộc với đỉnh B và E, (C, E) là cạnh liên thuộc với đỉnh C và E, (D, E) là cạnh liên thuộc với đỉnh D và E.

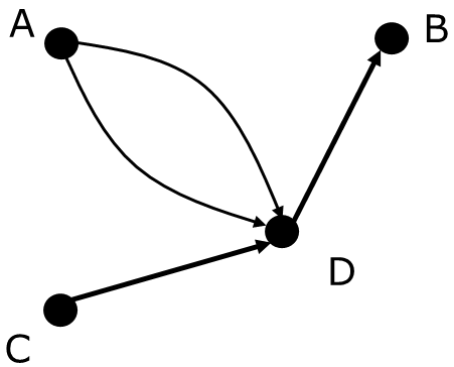
Đồ thị trong Hình 2.2 có các cặp đỉnh kề nhau sau: A và D, B và D, C và D. (A, D) là cung liên thuộc với đỉnh A và D, (B, D) là cung liên thuộc với đỉnh B và D, (C, D) là cung liên thuộc với đỉnh C và D.

Cạnh (cung) kề nhau: Hai cung e1 và e2 ($e1 \neq e2$) được gọi là kề nhau nếu chúng có đỉnh chung (nếu e1 và e2 là cung thì không phụ thuộc vào việc đỉnh chung đó là đỉnh đầu hay đỉnh cuối của cung e1, đỉnh đầu hay đỉnh cuối của cung e2).

Đồ thị trong Hình 2.1 có các cặp cạnh kề nhau sau: (A, E) và (B, E), (A, E) và (C, E), (A, E) và (D, E), (B, E) và (C, E), (B, E) và (D, E), (C, E) và (D, E).

Đồ thị trong Hình 2.2 có các cặp cạnh kề nhau sau: (A, D) và (B, D), (A, D) và (C, D), (B, D) và (C, D).

Cạnh (cung) song song: Hai cạnh (cung) được gọi là song song nếu nó nối hai cặp đỉnh giống nhau. Đồ thị trong Hình 2.3 có cung song song nối giữa hai đỉnh A và D.

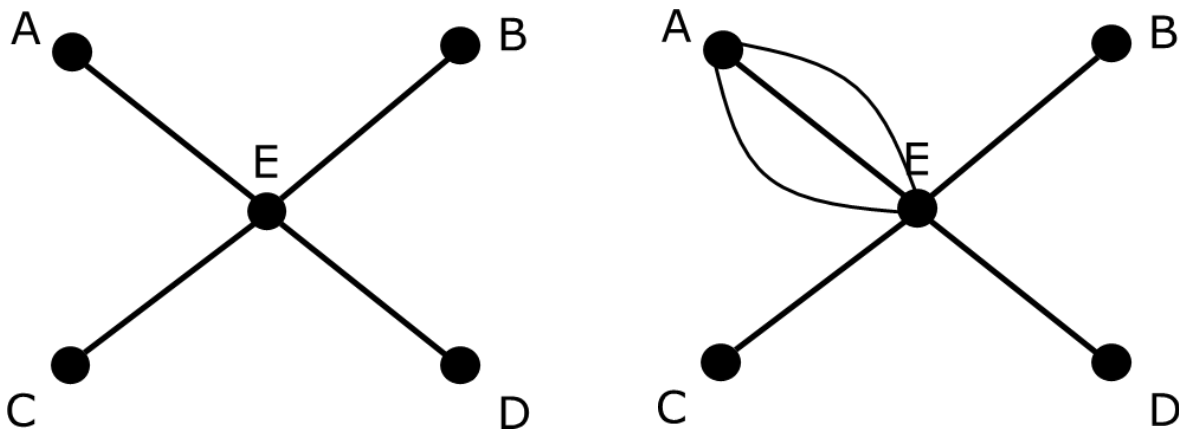


Hình 2.3. Ví dụ về cung song song

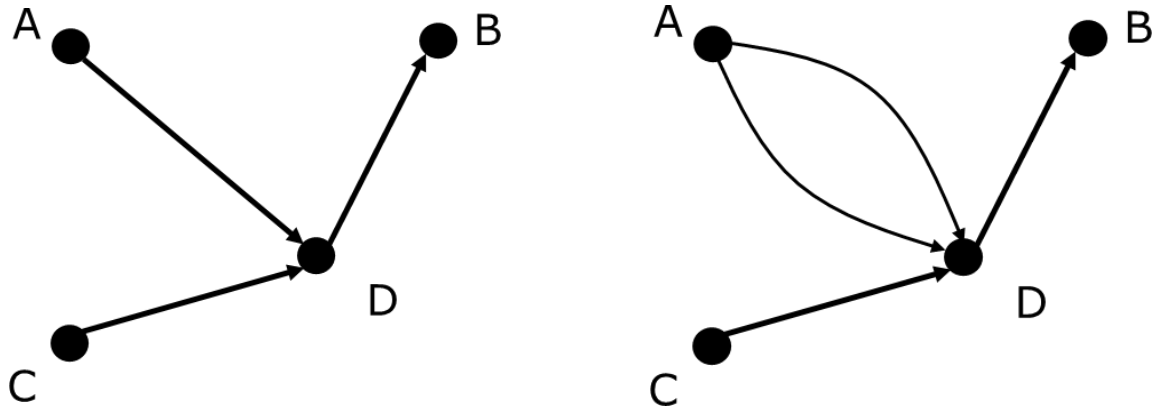
2.1.2 Một số dạng đồ thị đặc biệt

Đơn đồ thị: là đồ thị không chứa khuyên và mỗi cặp đỉnh chỉ được nối bởi một cạnh duy nhất.

Đa đồ thị: là đồ thị mà mỗi cặp đỉnh có thể được nối bởi nhiều hơn một cạnh. Ví dụ về đơn đồ thị vô hướng và đa đồ thị vô hướng được biểu diễn trong Hình 2.4 – Hình 2.5.

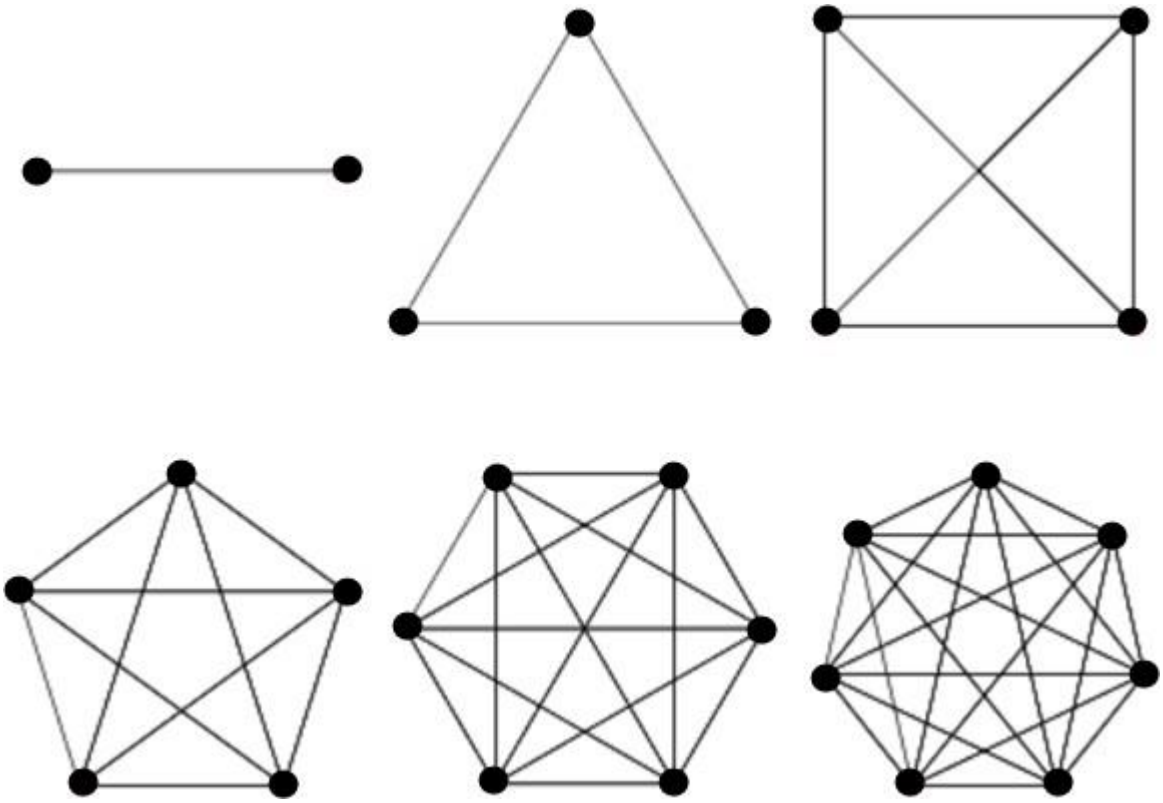


Hình 2.4. (a) Đơn đồ thị vô hướng, (b) Đa đồ thị vô hướng



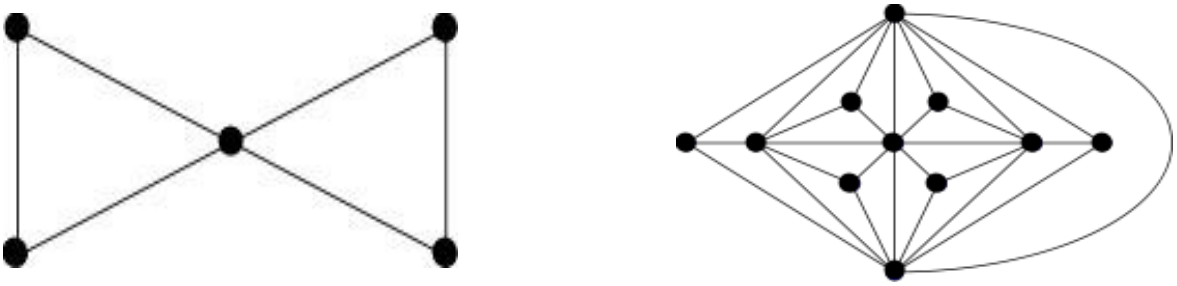
Hình 2.5. (a) Đơn đồ thị có hướng, (b) Đa đồ thị có hướng

Đồ thị đầy đủ: Đồ thị G = (V, E) được gọi là đồ thị đầy đủ nếu mỗi cặp đỉnh được nối với nhau bằng đúng một cạnh (cung). Hình 2.6 là một số ví dụ về đồ thị đầy đủ.



Hình 2.6. Ví dụ về đồ thị đầy đủ

Đồ thị phẳng: đồ thị $G = (V, E)$ được gọi là đồ thị phẳng nếu có thể biểu diễn hình học đồ thị G trên một mặt phẳng nào đó mà các cạnh của đồ thị chỉ cắt nhau ở đỉnh.



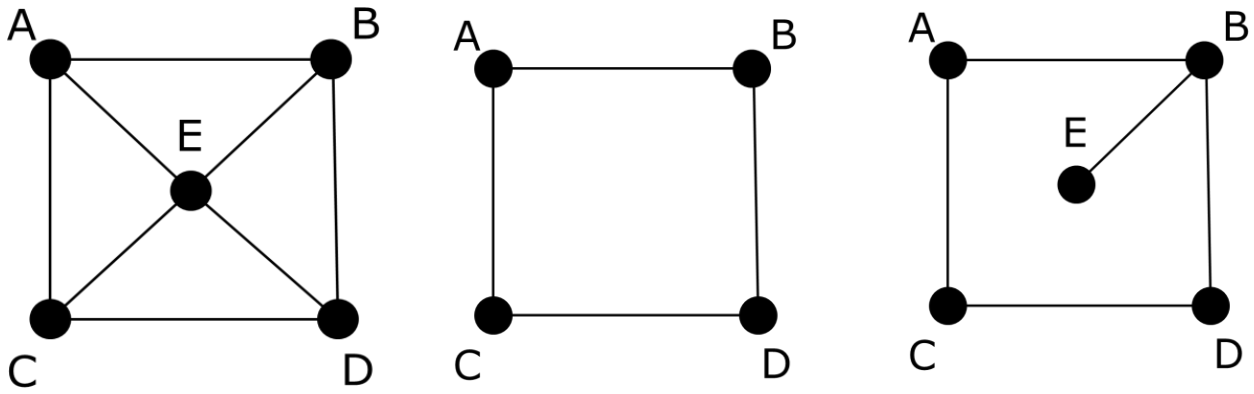
(a) (b)

Hình 2.7. Ví dụ về đồ thị phẳng:
(a) Đồ thị cánh bướm (đồ thị hình nơ) (b) Đồ thị Goldner–Harary

Đồ thị con: Đồ thị $G_A = (V_A, E_A)$ được gọi là đồ thị con của đồ thị $G = (V_G, E_G)$ nếu V_A là tập con của V_G và các cung trong E_A là các cạnh/cung của G mà hai đỉnh nó liên thuộc thuộc tập V_A.

Đồ thị bộ phận: Đồ thị $G_1 = (V_G, E_1)$ là đồ thị bộ phận của đồ thị $G = (V_G, E_G)$ nếu E_1 là tập con của E_G.

Hình 2.8 là ví dụ về đồ thị con và đồ thị bộ phận. Đồ thị G trong Hình 2.8.a có tập đỉnh $V_G = \{A, B, C, D, E\}$ và tập cạnh $E_G = \{(A,B), (A,C), (A,E), (B,D), (B,E), (C,D), (C,E), (D,E)\}$. Đồ thị G_1 trong hình 2.8.b có tập đỉnh $V_{G1} = \{A, B, C, D\}$ và tập cạnh $E_{G1} = \{(A,B), (A,C), (B,D), (C,D)\}$. Do $V_{G1} \subset V_G$, đồ thị G_1 là đồ thị con của đồ thị G. Đồ thị G_2 trong hình 2.8.c có tập đỉnh $V_{G2} = \{A, B, C, D, E\}$ và tập cạnh $E_{G2} = \{(A,B), (A,C), (B,D), (B,E), (C,D)\}$. Do $V_G = V_{G2}$ và $E_{G2} \subset E_G$, đồ thị G2 là đồ thị bộ phận của đồ thị G.



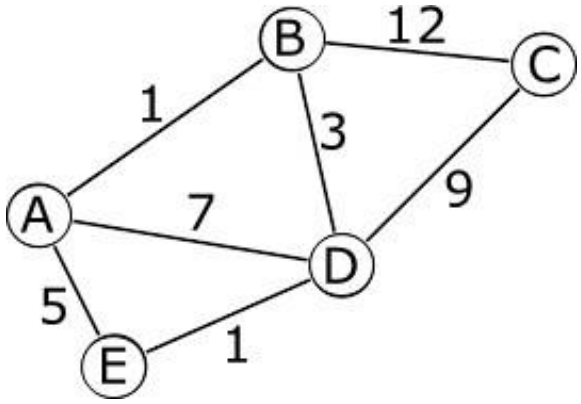
(a) (b) (c)

Hình 2.8. Ví dụ về đồ thị con và đồ thị bộ phận

Đồ thị bộ phận con: là đồ thị vừa là đồ thị con vừa là đồ thị bộ phận.

Đồ thị có trọng số: Đồ thị có trọng số là đồ thị mà mỗi cạnh (cung) (u, v) có một giá trị c(u,v) gọi là trọng số của cạnh. Về mặt hình học, trọng số của mỗi cạnh (cung) được ghi trên mỗi cạnh (cung). Hình 2.9 là ví dụ về đồ thị có trọng số, trong đó giá trị trọng số của mỗi cạnh được thể hiện trong bảng sau:

Cạnh	Trọng số
(A,B)	1
(A,D)	7
(A,E)	5
(B,C)	12
(B,D)	3
(C,D)	9
(D,E)	1



Hình 2.9. Ví dụ đồ thị có trọng số

2.1.3 Bậc của đỉnh đồ thị

Bậc của đỉnh: Trong đồ thị vô hướng (hoặc có hướng), bậc của đỉnh v là số cạnh liên thuộc với đỉnh v.

Đồ thị trong Hình 2.9 có bậc của các đỉnh lần lượt như sau: bậc(A) = 3, bậc (B) = 3, bậc (C) = 2, bậc (D) = 4, bậc (E) = 2.

Đỉnh treo và đỉnh cô lập: Nếu bậc của đỉnh bằng 1, đỉnh được gọi là đỉnh treo.

Nếu bậc của đỉnh bằng 0, đỉnh được gọi là đỉnh cô lập.

Cạnh (cung) treo: là cạnh (cung) có ít nhất một đầu là đỉnh treo.

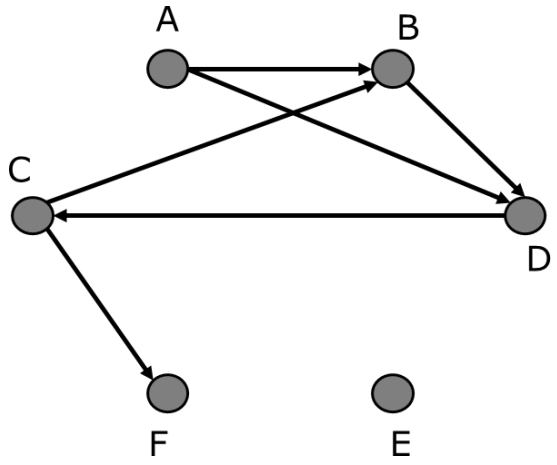
Nửa bậc trong: nửa bậc trong của đỉnh v, ký hiệu d-(v) là số cung có v là đỉnh cuối (tương ứng với số cung đi vào v).

Nửa bậc ngoài: nửa bậc ngoài của đỉnh v, ký hiệu d+(v) là số cung có v là đỉnh đầu (tương ứng với số cung đi ra từ v).

Trong đồ thị có hướng, bậc của đỉnh v, ký hiệu d(v), bằng $d^-(v) + d^+(v)$.

Đồ thị trong Hình 2.10 có nửa bậc trong, nửa bậc ngoài, và bậc của các đỉnh thể hiện trong bảng bên dưới. Đỉnh E của đồ thị đã cho được gọi là đỉnh cô lập, đỉnh F được gọi là đỉnh treo. Cung (C, F) được gọi là cung treo.

Đỉnh	**A**	**B**	**C**	**D**	**E**	**F**
Nửa bậc trong	0	2	1	2	0	1
Nửa bậc ngoài	2	1	2	1	0	0
Bậc	2	3	3	3	0	1



Hình 2.10. Ví dụ về bậc của đồ thị có hướng

Một số tính chất về bậc của đồ thị:

1. Tổng số bậc của tất cả các đỉnh gấp đôi số cạnh.
2. Số đỉnh có bậc lẻ luôn là một số chẵn.
3. Nếu đồ thị có nhiều hơn hai đỉnh thì có ít nhất hai đỉnh cùng bậc.
4. Nếu một đồ thị với n đỉnh (n>2) có đúng hai đỉnh cùng bậc thì hai đỉnh này không thể có bậc 0 hoặc n-1.
5. Luôn tồn tại đồ thị n đỉnh (n>2) mà 3 đỉnh bất kỳ của đồ thị đều không cùng bậc.
6. Cho đồ thị G=(V,E) với ít nhất kn+1 đỉnh, mỗi đỉnh có bậc không bé hơn (k-1)n+1, luôn tồn tại đồ thị con đầy đủ của G gồm k+1 đỉnh.

2.1.4 Đường đi và chu trình

Giả sử G = (V, E) là một đồ thị vô hướng (hoặc có hướng).

Đường đi: dãy $v_0, v_1, \dots, v_n$ $(v_i \in V, i = 0,1, \dots, n)$ được gọi là đường đi từ v_0 đến v_n nếu $\forall i\ (1 \le i \ge n - 1)$ cặp đỉnh v_i và v_{i+1} kề nhau, nghĩa là $(v_i, v_{i+1}) \in$ E. Khi đó, v_0 được gọi là đỉnh bắt đầu của đường đi và v_n được gọi là đỉnh kết thúc của đường đi. Độ dài của đường đi là số cạnh xuất hiện trong dãy $v_0, v_1, \dots, v_n$.

Đường đi sơ cấp: là đường đi mà các đỉnh trong đường đi không bị lặp lại.

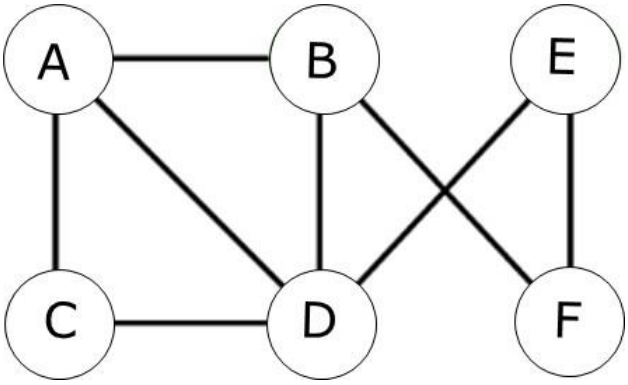
Chu trình: là đường đi có đỉnh bắt đầu và đỉnh kết thúc trùng nhau.

Chu trình sơ cấp: là chu trình có các đỉnh không bị lặp lại, trừ đỉnh đầu và đỉnh cuối.

Cho đồ thị vô hướng như trong Hình 2.11, dãy {A, B, F, E} là một đường đi từ A đến E của đồ thị. Tuy nhiên, dãy {A, B, E, F} không phải là đường đi của đồ thị. Đường đi {A, B, F, E} là đường đi sơ cấp, đường đi {A, B, D, E, F, B} không phải là đường đi sơ cấp.

Đồ thị trong Hình 2.11 có một số chu trình như sau: {A, B, D, A}, {A, C, D, A}, {A, B, F, E, D, A}, {A, B, F, E, D, C, A}, {A, D, B, F, E, D, C, A}. Trong số đó, {A,

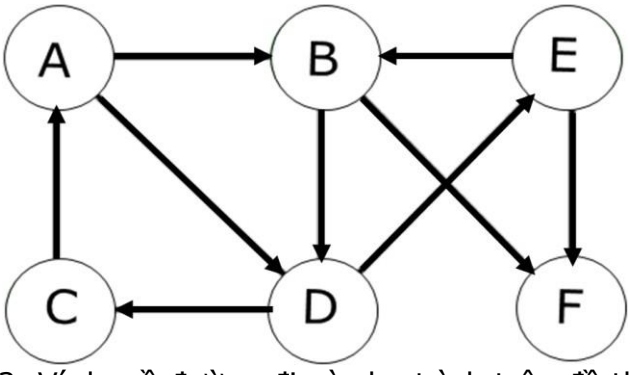
D, B, F, E, D, C, A} không phải là chu trình sơ cấp, còn lại các chu trình khác đều là chu trình sơ cấp.



Hình 2.11. Ví dụ về đường đi và chu trình trên đồ thị vô hướng

Cho đồ thị có hướng trong Hình 2.12, một số đường đi từ đỉnh A đến F như sau: {A, B, F}, {A, B, D, E, F}, {A, D, E, F}, {A, B, D, E, B, F}. Trong số các đường đi từ A đến F vừa liệt kê ở trên, {A, B, F}, {A, B, D, E, F}, {A, D, E, F} là đường sơ cấp, còn {A, B, D, E, B, F} không phải là đường sơ cấp. {A, D, B, F} không phải là một đường đi từ A đến F.

Đồ thị trong Hình 2.12 có một số chu trình như sau: {A, D, C, A}, {A, B, D, C, A}.



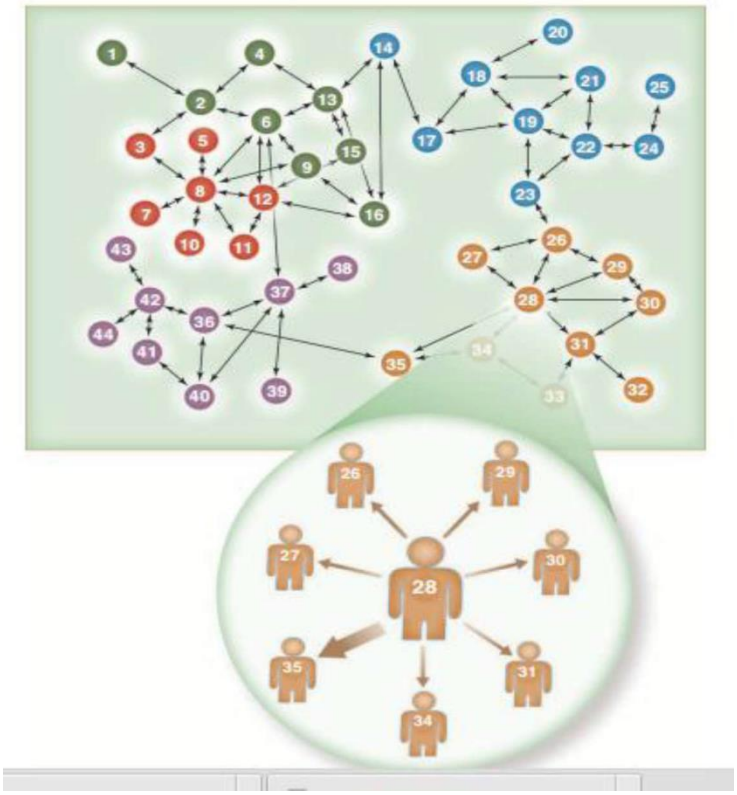
Hình 2.12. Ví dụ về đường đi và chu trình trên đồ thị có hướng

Một số tính chất:

1. Giả sử G là đồ thị vô hướng với n đỉnh (n>2) và các đỉnh đều có bậc không nhỏ hơn 2. Khi đó, G chứa ít nhất một chu trình sơ cấp.
2. Giả sử G là đồ thị vô hướng với n đỉnh (n>3) và các đỉnh đều có bậc không nhỏ hơn 3. Khi đó, G chứa ít nhất một chu trình sơ cấp có độ dài chẵn.

2.2 BIỂU DIỄN MẠNG XÃ HỘI BẰNG ĐỒ THỊ

Mạng xã hội xác định một tập hữu hạn các người dùng (actor) và các liên kết giữa chúng. Actor có thể là cá nhân, tổ chức, công ty,… Ví dụ actor là những thành viên trong một cộng đồng, các nhân viên trong công ty, những nhà nghiên cứu trong một tổ chức. Một cung nối hai actor trong một mạng xã hội được gọi là liên kết hay quan hệ. Cung được xác định bởi kiểu của liên kết giữa các actor. Ví dụ giữa những công ty có thể có các quan hệ giao dịch mua bán, hợp đồng kinh doanh. Giữa nhân viên trong công ty, có thể có quan hệ thứ bậc ví dụ ai là "cấp trên" của ai. Giữa các quốc gia có thể có các quan hệ ngoại giao, quan hệ giao thương,…



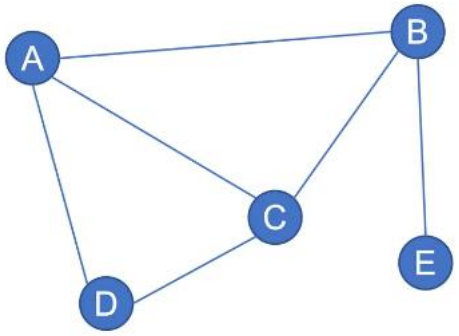
Hình 2.13. Mạng xã hội được biểu diễn bằng đồ thị

Đồ thị G được định nghĩa là cặp (V, E) trong đó V là tập các đỉnh và E là tập các cung nối các đỉnh lại với nhau. Mạng xã hội được biểu diễn bằng đồ thị trong đó mỗi đỉnh đại diện cho một actor, còn những liên kết giữa các actor được biểu diễn bằng các cung của đồ thị.

Biểu diễn đồ thị bằng ma trận kề.

Ma trận kề của đồ thị vô hướng là ma trận đối xứng. Mỗi phần tử của ma trận kề phản ánh một cung giữa hai actor và được ký hiệu như sau:

$$x_{ij} = \begin{cases} 1 & Khi\ có\ liên\ kết\ giữa\ đỉnh\ x_i\ và\ đỉnh\ x_j \\ 0 & khi\ không\ có\ liên\ kết\ giữa\ đỉnh\ x_i\ và\ đỉnh\ x_j \end{cases}$$



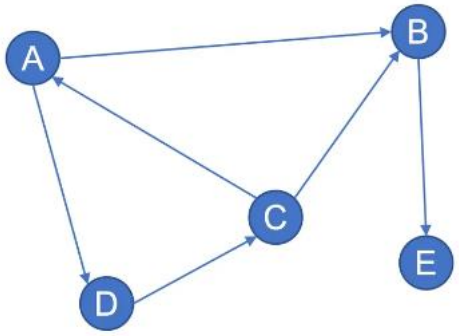
Hình 2.14. Mạng gồm 5 đỉnh

Ma trận kề biểu diễn đồ thị vô hướng ở Hình 2.14 được mô tả như sau:

$$A = \begin{pmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

Ma trận kề của đồ thị có hướng là ma trận không đối xứng và được xác định như sau:

$$x_{ij} = \begin{cases} 1 \ \ khi\ có\ liên\ kết\ nối\ từ\ đỉnh\ n_i\ đến\ n_j \\ 1 \ \ khi\ có\ liên\ kết\ từ\ đỉnh\ n_j\ đến\ đỉnh\ n_i \\ 0 \ \ khi\ không\ có\ liên\ kết\ nối \end{cases}$$



Hình 2.15. Mạng có hướng

Ma trận kề biểu diễn đồ thị có hướng trong Hình 2.15 được trình bày như sau:

$$A = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

2.3 TÍNH TOÁN SỐ ĐO TRONG ĐỒ THỊ MẠNG XÃ HỘI

2.3.1 Mật độ của mạng

Mạng xã hội có nhiều số đo (measure) khác nhau. Một trong những số đo quan trọng là mật độ mạng (density). Khi hệ số gắn kết của mạng càng lớn, mức độ gắn kết, sự chặt chẽ của các mối quan hệ giữa các actor trong mạng càng lớn, và do đó, sự tương trợ, hỗ trợ… giữa các actor càng nhiều. Do vậy, ảnh hưởng của mạng xã hội lên hành vi của actor sẽ mạnh mẽ hơn nếu mạng xã hội có mật độ cao và ảnh hưởng ít nếu mạng xã hội có mật độ thấp.

Một cách tổng quát, tính gắn kết của mạng xã hội được đo bằng tỷ lệ giữa tổng các mối liên hệ thực tế trong mạng xã hội và tổng các mối quan hệ có thể có của mạng xã hội. Số đo mật độ mạng xã hội được tính bằng công thức sau:

$$\frac{k}{n(n-1)/2}$$

Trong đó:

k: tổng các đường liên kết thực tế của toàn mạng

n: tổng các tác nhân (actor) trong mạng xã hội

(n-1)/2: tổng các mối liên kết khả dĩ có trong mạng xã hội

Giá trị của số đo mật độ mạng nằm trong đoạn [0,1].

Giá trị của số đo mật độ càng gần đến 1 thì tính gắn kết của mạng xã hội càng mạnh và do đó sự truyền nhận thông tin giữa các thành viên trong mạng xã hội diễn ra càng tốt.

Giá trị của số đo mật độ càng gần đến 0 thì tính gắn kết của mạng xã hội càng yếu và do đó sự truyền nhận thông tin giữa các thành viên trong mạng xã hội sẽ yếu.

2.3.2 Số đo bậc trung tâm (Degree centrality)

Số đo này giúp ta đo số lượng của các mối quan hệ trực tiếp của một tác nhân nào đó với các thành viên khác trong mạng xã hội. Giá trị của hệ số này nằm trong đoạn [0,1]. Khi giá trị số đo bậc càng gần tới 1 thì tính trung tâm trực tiếp của tác nhân càng lớn. Số đo bậc trung tâm được tính bằng công thức sau:

$$C_D(v) = \frac{\deg(v)}{n-1}$$

Trong đó:

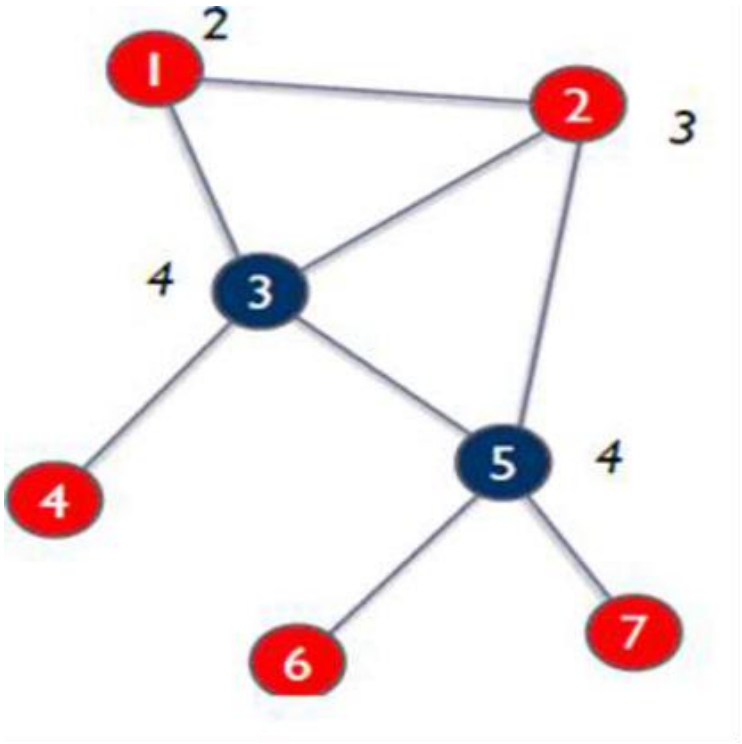
n: là số đỉnh của đồ thị

deg(v): tổng số các liên kết trực tiếp đến đỉnh v (bậc của đỉnh)

Trong mạng xã hội thì số đo bậc trung tâm của một actor là số lượng các actor có quan hệ trực tiếp với actor đang xét. Nói cách khác đây chính là bậc của một đỉnh trong đồ thị mạng xã hội. Bậc của đỉnh là số cung vào và số cung ra của đỉnh trong đồ thị có hướng. Trong đồ thị vô hướng thì tổng số cung vào và cung ra là bậc của đỉnh và thường được sử dụng làm thước đo mức độ kết nối của đỉnh với các đỉnh lân cận.

Số đo bậc trung tâm cho ta mức độ phổ biến hay mức độ rộng rãi trong quan hệ của một actor. Một actor có bậc trung tâm càng cao thì khả năng actor đó có vai trò càng lớn trong mạng; actor đó cũng có sức ảnh hưởng đến actor khác nên có thể dùng số đo bặc trung tâm để xác định người quan trọng (key players).

Độ phức tạp để tính số đo bậc trung tâm là $O(V^2)$ nếu đồ thị là dày và O(E) nếu đồ thị thưa.

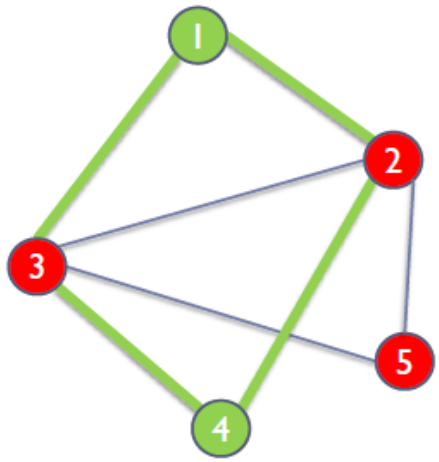


Hình 2.16. Đồ thị với bậc của các node

Trong đồ thị trong Hình 2.16, các nút 3 và 5 có số đo bậc trung tâm cao nhất.

2.3.3 Đường đi ngắn nhất

Đường đi ngắn nhất giữa hai đỉnh là con đường kết nối hai đỉnh và có chiều dài từ đỉnh này đến đỉnh kia là ngắn nhất.



Hình 2.17. Đồ thị với đường đi ngắn nhất

Trong ví dụ ở Hình 2.17, giữa các node 1 và 4 có hai con đường ngắn nhất có độ dài là 2: 1-2-4 và 1-3-4.

Nhờ đường đi ngắn nhất nên tốc độ liên lạc, trao đổi thông tin diễn ra nhanh chóng.

Các node quan trọng thường nằm trên các đường đi ngắn nhất. Bên cạnh đó đường đi ngắn nhất còn giúp xác định các điểm gắn kết mạng. Khi các đỉnh này mất đi, thì việc gắn kết toàn mạng sẽ sụp đổ. Trong Hình 2.17 khi các nút 3 hoặc 5 biến mất thì mạng sẽ sụp đổ.

2.3.4 Số đo trung tâm gần gũi (closeness centrality)

Điểm yếu của số đo bậc (degree) là số đo này chỉ xét các mối quan hệ trực tiếp của actor. Do vậy nếu actor có số đo bậc trung tâm trực tiếp cao thì chưa chắc actor này là người "gần gũi" với mọi actor 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 tác nhân càng gần gũi với các actor khác trong mạng xã hội bao nhiêu thì actor đó

càng dễ có nhiều thông tin và càng có nhiều uy thế. Do đó, actor này sẽ dễ gây ảnh hưởng lên toàn bộ mạng xã hội. Số đo trung tâm gần gũi được tính theo công thức sau:

$$C_C(v) = \frac{1}{\sum_{t \in V/v} d_G(v,t)}$$

Trong đó: $d_G(v,t)$ là chiều dài của đường đi ngắn nhất đi từ đỉnh v tới đỉnh t.

Số đo trung tâm gần gũi tương đối được tính bằng công thức:

$$Cc(x) = (n-1)C_c(v)$$

Số đo mức độ gần gũi của một đỉnh đến tất cả các đỉnh khác mà đỉnh này có thể đi đến. Trong mạng xã hội, số độ này tương ứng với thời gian cần thiết để thông tin truyền từ một actor này đến actor. Khoảng cách này càng nhỏ thì khả năng truyền tin của actor càng lớn và actor có vai trò càng quan trọng.

Số đo mức độ gần gũi của một đỉnh đến tất cả các đỉnh khác mà đỉnh này có thể đi đến. Trong mạng xã hội, số độ này tương ứng với thời gian cần thiết để thông tin truyền từ một actor này đến actor. Khoảng cách này càng nhỏ thì khả năng truyền tin của actor càng lớn và actor có vai trò càng quan trọng.

Có thể tính số đo trung tâm gần gũi cho đồ thị có hướng và đồ thị vô hướng.

Có hai khả năng cho đồ thị có hướng:

- Các cung ra (mức độ gần gũi của đỉnh lựa chọn với tất cả các đỉnh khác: khoảng cách từ đỉnh được chọn đi đến tất cả đỉnh khác).
- Các cung vào (mức độ gần gũi của đỉnh lựa chọn đến tất cả đỉnh khác: khoảng cách từ tất cả đỉnh khác đi đến đỉnh được chọn).

2.3.5 Số đo trung tâm trung gian (betweenness centrality)

Số đo trung tâm trung gian xác định một tác nhân nào đó trong mạng có thể ít gắn kết với các thành viên khác trong mạng xã hội (số đo bậc trung tâm thấp), cũng không "gần gũi" lắm với mọi thành viên khác (số đo trung tâm lân cận thấp), nhưng lại là "cầu nối" (bridge) hay "nhà trung gian" cần thiest trong mọi cuộc trao đổi trong mạng. Nếu

một actor đóng vai trò trung gian càng lớn trong mạng lưới, actor đó sẽ càng ở vị trí thuận lợi trong việc "kiểm soát" mọi giao dịch, mọi thông tin trong mạng xã hội; actor đó cũng tác động đến mạng lưới một cách dễ dàng bằng cách lọc hoặc truyền thông tin trong mạng theo hướng có lợi; đồng thời actor đó cũng đứng ở vị trí tốt nhất để thúc đẩy sự phối hợp giữa các thành viên khác trong mạng lưới. Số đo trung tâm trung gian được tính bằng công thức sau:

$$C_B(v) = \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

Trong đó:

- σ_{st} là số đường đi ngắn nhất đi từ đỉnh s đến đỉnh t của toàn mạng.
- $\sigma_{st}(v)$ là số đường đi ngắn nhất đi từ đỉnh s đến đỉnh t và đi qua đỉnh v

Số đo trung tâm trung gian của một actor là xác suất để thông tin truyền qua lại giữa hai actor bất kỳ trong mạng phải đi qua actor này. Số đo này càng lớn thì tầm quan trọng của actor càng lớn.

Số đo trung tâm trung gian tương đối được tính theo cách sau:

Đối với đồ thị vô hướng:

$$C_B(x) = \frac{c_B(x)}{(n-1)(n-2)/2}$$

Đối với đồ thị có hướng:

$$C_B(x) = \frac{c_B(x)}{(n-1)(n-2)}$$

2.3.6 Số đo gom cụm (clustering centrality)

Trọng mạng xã hội, số đo gom cụm được Watts và Strogatz đề xuất và dùng làm tiêu chuẩn để đo mức độ gắn kết giữa các actor trong mạng. Số đo gom cụm của một actor được xác định bởi các actor láng giềng có mối liên kết trực tiếp với nhau. Số đo gom cụm được tính như sau:

Số đo gom cụm cho đồ thị có hướng:

$$C_i = \frac{|\{e_{jk}\}|}{k_i(k_i - 1)}$$

Số đo gom cụm cho đồ thị vô hướng:

$$C_i = \frac{2|\{e_{jk}\}|}{k_i(k_i - 1)}$$

Số đo gom cụm trung bình cho toàn mạng xã hội:

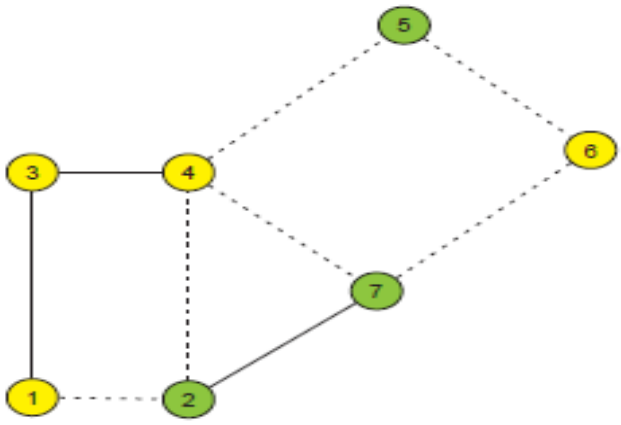
$$\bar{C} = \frac{1}{n}\sum_{i=1}^{n} C_i$$

2.4 ĐỒ THỊ CÓ DẤU

2.4.1 Định nghĩa

Đồ thị có dấu (signed graph) là cặp có thứ tự (G, σ) với $G = (V, L)$ là đồ thị có tập đỉnh V và tập cung L.

$\sigma: L \to \{$p, cho dòng n$\}$ là hàm dấu để gán dấu p cho dòng. Dấu p có thể là dấu cộng hay dấu trừ. Cung ứng với dấu cộng được vẽ liền nét, cung có dấu âm được vẽ bằng nét đứt. Hình 2.18 là đồ thị có dấu.



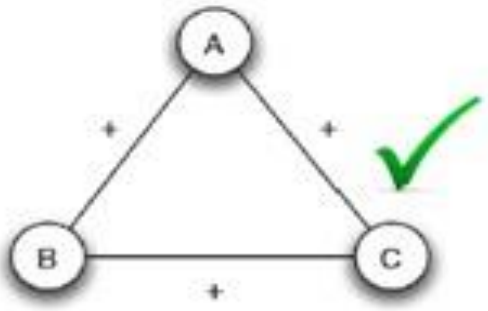
Hình 2.18. Đồ thị có dấu

Cung dương thể hiện quan hệ bạn bè, cung âm thể hiện quan hệ đối thủ.

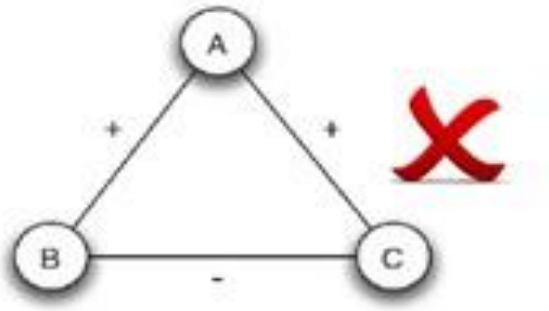
- Bài toán quan trọng trong nghiên cứu mạng xã hội có dấu là hiểu tác động giữa hai lực âm dương;
- Dùng lý thuyết cân bằng cấu trúc (structural balance theory) để nghiên cứu sự cân bằng của đồ thị có dấu.

2.4.2 Cân bằng cấu trúc

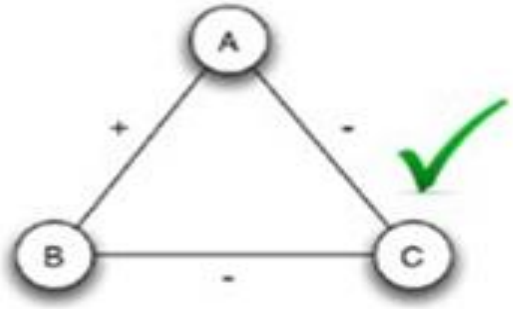
Giả sử ta có mạng xã hội trên một tập hợp người, trong đó mọi người đều quen biết nhau. Đồ thị biểu diễn mạng xã hội là đồ thị đầy đủ hay clique. Theo lý thuyết cân bằng cấu trúc (lý thuyết tâm lý xã hội của Heider vào những năm 1940), ta có các phát biểu sau:

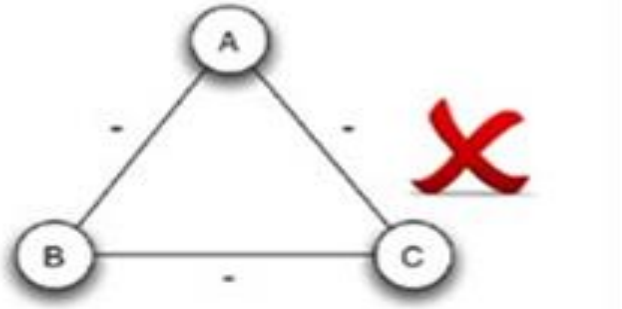


A, B, C là bạn lẫn nhau đồ thị cân bằng



A là bạn của B, C nhưng B không là bạn của C đồ thị không cân bằng





A, B là bạn lẫn nhau

A, B không là bạn của C

đồ thị cân bằng

A, B, C không là bạn của nhau

đồ thị không cân bằng

Hình 2.19. Cân bằng cấu trúc của đồ thị có dấu

– Đối với bất kỳ hai người nào, ta có thể có các cung âm hay dương;

– Đối với ba người hình thành tam giác và phải có một hay cả ba cung nối chúng đều dương.

Các tam giác mất cân bằng là nguồn gốc của căng thẳng hay không hòa hợp.

• Cần dàn xếp để giảm thiểu đến mức nhỏ nhất trạng thái tam giác mất cân bằng;

• Tính chất cân bằng cấu trúc: Với mọi bộ 3 node, nếu chúng ta xét 3 cạnh nối với chúng, hoặc cả ba cạnh là dương, hoặc một cạnh duy nhất là dương;

• Đồ thị cân bằng nếu nó thỏa tính chất trên.

2.4.3 Phân hoạch đồ thị có dấu

Bài toán: phân hoạch các đỉnh của đồ thị có dấu sao cho các cung nối với các đỉnh thuộc về cùng cluster là cung dương và cung nối hai đỉnh thuộc về cluster kia là cung âm.

Nếu phân hoạch được đồ thị có dấu theo cách trên, chúng ta gọi đồ thị có dấu là phân hoạch được hay gom cụm được. Đồ thị thỏa điều kiện trên được gọi là đồ thị cân bằng.

Ví dụ về đồ thị có dấu trong đó cung dương biểu diễn cho bạn bè và cung âm biểu diễn cho đối thủ. Lúc đó đồ thị âm dương cân bằng có hai cụm: một cụm là đối tác, cụm kia là đối thủ.

• Giữa các bạn của A trong X, sẽ chỉ có các cung dương;

• Giữa X và Y, chỉ có các cung âm.

2.4.4 Tính chất của đồ thị có dấu

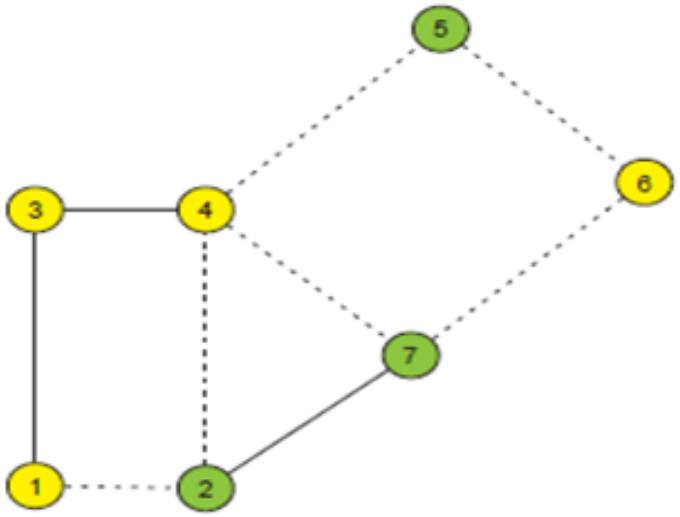
Dấu của đường đi trong đồ thị là dấu của tích các dấu của cung trên đường đi của nó.

Do vậy, dấu của đường đi là dương nếu nó có chứa một số chẵn các cung âm, ngược lại nó sẽ là âm.

Định lý 1: Đồ thị có dấu là cân bằng nếu và chỉ nếu mỗi cặp đỉnh trên tất cả các đường đi giữa chúng đều có cùng dấu.

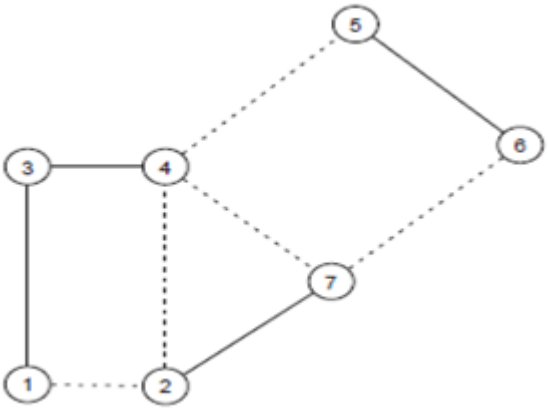
Định lý 2: Đồ thị có dấu là cân bằng nếu và chỉ nếu từng nửa vòng tròn là dương.

Ví dụ về đồ thị có dấu cân bằng:

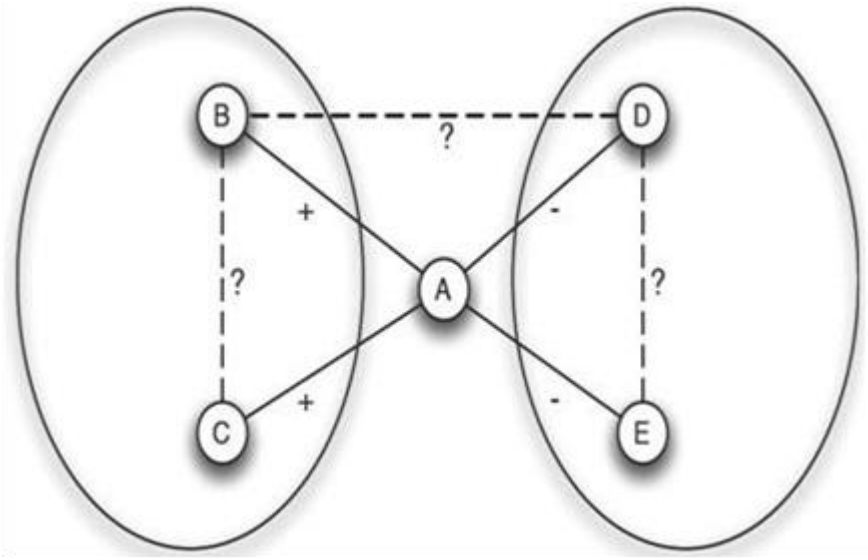


Hình 2.20. Đồ thị có dấu cân bằng

Ví dụ đồ thị có dấu không cân bằng:



Hình 2.21. Đồ thị có dấu không cân bằng



Hình 2.22. Phân hoạch đồ thị có dấu

Giải thích định lý cân bằng:

a) Trường hợp 1 (đồ thị không có cung âm)

Dễ chứng minh, không đối kháng

b) Trường hợp 2 (đồ thị có ít nhất một cung âm) Giả sử node A được kết với cung này

Giả sử A và bạn của A đi đến X thì tất cả đối thủ của A đi đến Y. Giữa các bạn của A trong X, chỉ có các cung dương. Giữa X và Y, chỉ có các cung âm.

Ví dụ ứng dụng: Xét một mạng xã hội với quan hệ bạn bè (friend) hay không phải là bạn bè (đối thủ), quan hệ này xảy ra giữa các cặp cá nhân. Giả định có một tin tức có hai dạng thức đúng, sai. Nếu ai đó loan tin tức này đến một người bạn thì tin tức đó sẽ có cùng dạng thức với tin tức người đó đã nhận được. Nhưng nếu loan tin tức này đến đối thủ thì tin tức sẽ bị thay đổi dạng thức.

Nếu hệ thống cân bằng, mọi người sẽ chỉ nhận một phiên bản của tin tức (Định lý 1 - tất cả các chuỗi có cùng dấu); cùng vậy, người phát tin tức sẽ nhận lại tin tức khi nó quay về người phát tin theo cùng dạng thức (Định lý 2 - mỗi nửa vòng tròn là dương).

Lưu ý: sự xung đột thường xảy ra ở các cụm khác nhau.

Hai định lý trên cho phép phân hoạch đồ thị có dấu thành 2 cluster.

Đồ thị dạng tam giác cân bằng

Đồ thị dạng tam giác không cân bằng

Hình 2.23. Đồ thị cân bằng và không cân bằng

Theo hai định lý cân bằng trên, ta có bốn đồ thị đầu tiên trong Hình 2.23 là cân bằng, còn 4 đồ thị cuối cùng là không cân bằng

Có thể giải thích 4 trường hợp đầu với các phát biểu sau:

- Bạn của bạn cũng là bạn;
- Đối thủ của bạn cũng là đối thủ của ta;

- Bạn của đối thủ củng là đối thủ;
- Đối thủ của đối thủ là bạn.

Ba phát biểu đầu tiên là hiển nhiên, nhưng phát biểu cuối, có người cho rằng:

- Đối thủ của đối thủ là đối thủ;
- Trường hợp này được vẽ ở hình cuối.

2.4.5 Một số ứng dụng đồ thị có dấu

Đối với các cộng đồng trực tuyến (vd: Amazon, Epinions), người dùng có thể biểu diễn các quan hệ âm, dương (ví dụ tin hay không tin) vào người khác. Tuy vậy mọi người không quen biết nhau. Thậm chí khi người dùng A tin hay không tin vào người dủng B, chúng ta không biết người dùng B suy nghĩ gì về người dùng A. Có thể sử dụng lý thuyết cân bằng cấu trúc để giải thích như sau:

Nếu người dùng A không tin người dùng B và người dùng B không tin người dùng C thì người dùng A tin người dùng C (giả đỉnh lý thuyết cân bằng đúng).

A không tin C: có thể người dùng A suy nghĩ anh ta tốt hơn B, người dùng B suy nghĩ anh ta tốt hơn C, rồi người dùng A tốt hơn C.

2.5 SỬ DỤNG NETWORKX ĐỂ TÍNH TOÁN SỐ ĐO TRONG ĐỒ THỊ MẠNG XÃ HỘI

NetworkX cung cấp nhiều phép đo trung tâm khác nhau[5].

2.5.1 Tính degree centrality

Số đo bậc đánh giá tầm quan trọng dựa trên số lượng kết nối của một nút. Số đo bậc là tỷ lệ các nút trong mạng mà một nút được kết nối. Một nút có càng nhiều kết

[5] https://networkx.org/documentation/stable/reference/algorithms/centrality.html

nối, thì tỷ lệ các nút mà chúng được kết nối sẽ càng cao, vì vậy số lượng kết nối và số đo bậc thực sự có thể được sử dụng thay thế cho nhau.

1. Chúng ta có thể tính số đo bậc của từng nút trong mạng:

```
degcent = nx.degree_centrality(G)
degcent
{'@kmg3445t': 0.0008605851979345956,
 '@code_kunst': 0.011187607573149742,
 '@highgrnd': 0.0008605851979345956,
 '@youngjay_93': 0.0008605851979345956,
 '@sobeompark': 0.0008605851979345956,
 '@justhiseung': 0.0008605851979345956,
 '@hwajilla': 0.0008605851979345956,
 '@blobyblo': 0.0034423407917383822,
 '@minddonyy': 0.0008605851979345956,
 '@iuiive': 0.0008605851979345956,
 ...
 }
```

2. Chúng ta có thể sử dụng nó để tạo một DataFrame pandas khác, được sắp xếp theo mức độ trung tâm giảm dần:

```
degcent_df = pd.DataFrame(degcent, index=[0]).T
degcent_df.columns = ['degree_centrality']
degcent_df.sort_values('degree_centrality', inplace=True,
ascending=False)
degcent_df.head()
```

Ta thấy một DataFrame của các tài khoản Twitter và số đo bậc của chúng như sau:

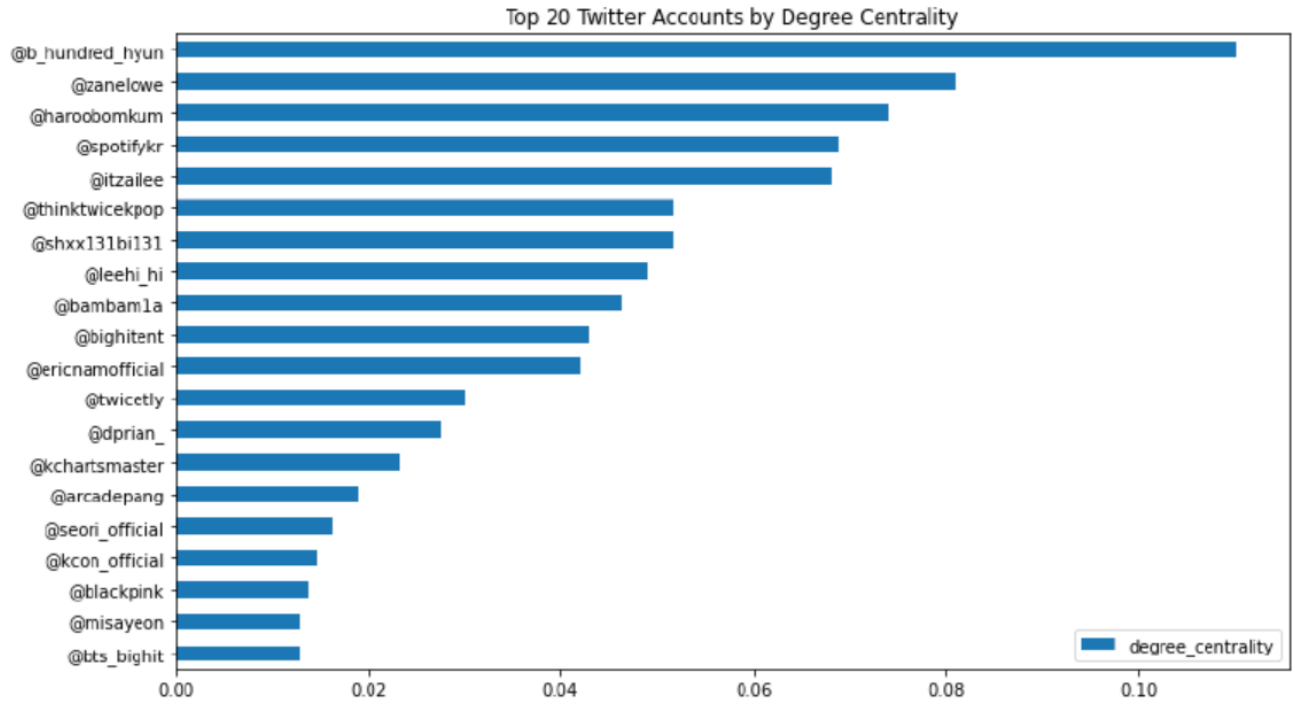
	degree_centrality
@b_hundred_hyun	0.110155
@zanelowe	0.080895
@haroobomkum	0.074010
@spotifykr	0.068847
@itzailee	0.067986

Hình 2.24. pandas DataFrame của số đo bậc của các nút

3. Cuối cùng, chúng ta có thể trực quan dữ liệu dưới dạng biểu đồ cột ngang như sau:

```
title = 'Top 20 Twitter Accounts by Degree Centrality'
_= degcent_df[0:20].plot.barh(title=title, figsize=(12,7))
plt.gca().invert_yaxis()
```

Ta được biểu đồ cột ngang của các tài khoản Twitter với số đo bậc như trong Hình 2.25.



Hình 2.25. Biểu đồ cột ngang của các tài khoản Twitter với số đo bậc

2.5.2 Tính betweenness centrality

Số đo trung tâm (Betweenness centrality) liên quan đến cách thông tin lưu thông qua một mạng lưới. Nếu một nút nằm giữa hai nút khác, thì thông tin từ bất kỳ nút nào trong hai nút đó đều phải đi qua nút nằm giữa chúng. Thông tin di chuyển qua nút nằm ở giữa. Nút đó có thể được coi là một nút thắt cổ chai, hoặc là một vị trí thuận lợi. Việc nắm giữ thông tin mà người khác cần có thể mang lại lợi thế chiến lược.

Tuy nhiên, thông thường, các nút có số đo trung tâm cao nằm giữa nhiều nút, không chỉ giữa hai nút. Điều này thường được thấy trong một mạng lưới khởi nghiệp, nơi một nút trung tâm được kết nối với hàng tá nút khác. Hãy lấy ví dụ về một người có ảnh hưởng trên mạng xã hội. Người đó có thể được kết nối với 22 triệu người theo dõi, nhưng những người theo dõi đó có thể không biết nhau. Chắc chắn họ biết người có ảnh hưởng (hoặc là một bot không chính thống). Người có ảnh hưởng đó là một nút trung tâm và số đo trung tâm sẽ cho thấy điều đó.

Trước khi chúng ta xem cách tính số đo trung tâm, cần lưu ý rằng việc tính toán số đo trung tâm rất tốn thời gian đối với các mạng lưới lớn hoặc dày đặc. Nếu mạng lưới của bạn quá lớn hoặc dày đặc và khiến việc tính toán số đo trung tâm quá chậm đến mức không còn hữu ích nữa, hãy cân nhắc sử dụng một số đo khác để tính mức độ quan trọng.

Chúng ta có thể tính số đo trung tâm của mỗi nút trong mạng như sau:

```
betwcent = nx.betweenness_centrality(G)
betwcent
{'@kmg3445t': 0.0,
 '@code_kunst': 0.016037572215773392,
 '@highgrnd': 0.0,
 '@youngjay_93': 0.0,
 '@sobeompark': 0.0,
 '@justhiseung': 0.0,
 '@hwajilla': 0.0,
 '@blobyblo': 0.02836579219003866,
```

```
 '@minddonyy': 0.0,
 '@iuiive': 0.0,
 '@wgyenny': 0.0,
 '@wondergirls': 0.0013446180439736057,
 '@wg_lim': 0.0026862711087984274,
 ...
 }
```

Sử dụng nó để tạo một pandas DataFrame khác, sắp xếp theo thứ tự giảm dần:

```
betwcent_df = pd.DataFrame(betwcent, index=[0]).T
betwcent_df.columns = ['betweenness_centrality']
betwcent_df.sort_values('betweenness_centrality',
inplace=True, ascending=False)
betwcent_df.head()
```

Thu được một dataframe của các tài khoản Twitter và số đo trung tâm của chúng như sau:

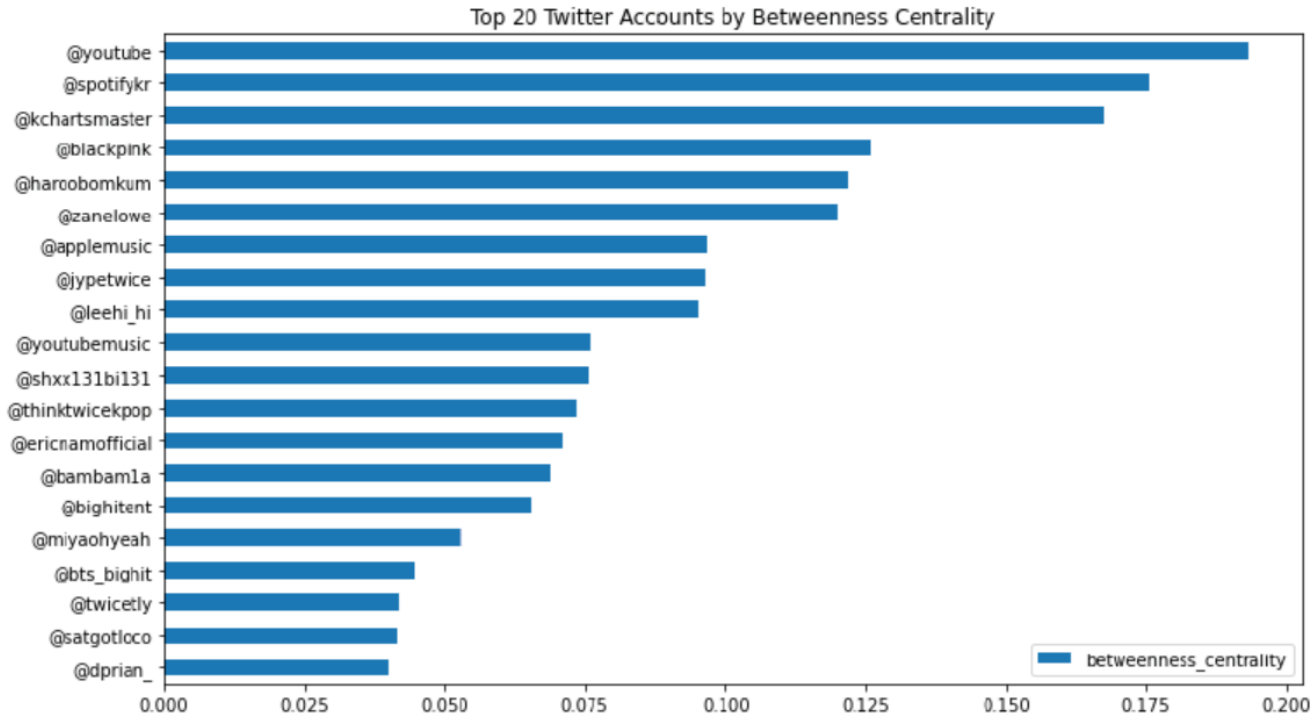
	betweenness_centrality
@youtube	0.193090
@spotifykr	0.175619
@kchartsmaster	0.167481
@blackpink	0.125805
@haroobomkum	0.121789

Hình 2.26. pandas DataFrame với số đo trung tâm của các nút

Cuối cùng, chúng ta có thể trực quan dữ liệu qua biểu đồ cột ngang bằng lệnh sau:

```
title = 'Top 20 Twitter Accounts by Betweenness Centrality'
_= betwcent_df[0:20].plot.barh(title=title, figsize=(12,7))
plt.gca().invert_yaxis()
```

Biểu đồ cột ngang của các tài khoản Twitter với số đo trung tâm được thể hiện trong Hình 2.27.



Hình 2.27. Biểu đồ cột ngang của các tài khoản Twitter với số đo trung tâm

2.5.3 Tính closeness centrality

Số đo gần gũi (Closeness centrality) liên quan đến mức độ gần của các nút với các nút khác, và điều đó liên quan đến một khái niệm được gọi là đường dẫn ngắn nhất (shortest path). Tuy nhiên, việc tính toán đường đi ngắn nhất lại rất tốn nhiều tính toán (và chậm) đối với các mạng lưới lớn hoặc dày đặc. Do đó, tính số đo gần gũi thậm chí còn chậm hơn tính số đo trung gian (betweenness centrality). Nếu việc nhận kết quả từ tính số đo gần gũi quá chậm do quy mô và mật độ của mạng lưới, có thể chọn một số đo khác để đánh giá tầm quan trọng.

Chúng ta có thể tính số đo gần gũi của mỗi nút trong mạng như sau:

```
closecent = nx.closeness_centrality(G)
closecent
```

```
{'@kmg3445t': 0.12710883458078617,
 '@code_kunst': 0.15176930794223495,
 '@highgrnd': 0.12710883458078617,
 '@youngjay_93': 0.12710883458078617,
 '@sobeompark': 0.12710883458078617,
 '@justhiseung': 0.12710883458078617,
 '@hwajilla': 0.12710883458078617,
 '@blobyblo': 0.18711010406907921,
 '@minddonyy': 0.12710883458078617,
 '@iuiive': 0.12710883458078617,
 '@wgyenny': 0.07940034854856182,
 ...
 }
```

Chúng ta có thể tạo một pandas DataFrame khác, sắp xếp bởi số đo gần gũi theo chiều giảm dần:

```
closecent_df = pd.DataFrame(closecent, index=[0]).T
closecent_df.columns = ['closeness_centrality']
closecent_df.sort_values('closeness_centrality',
inplace=True,ascending=False)
closecent_df.head()
```

Như vậy dataframe của các tài khoản Twitter và số đo gần gũi của chúng như sau:

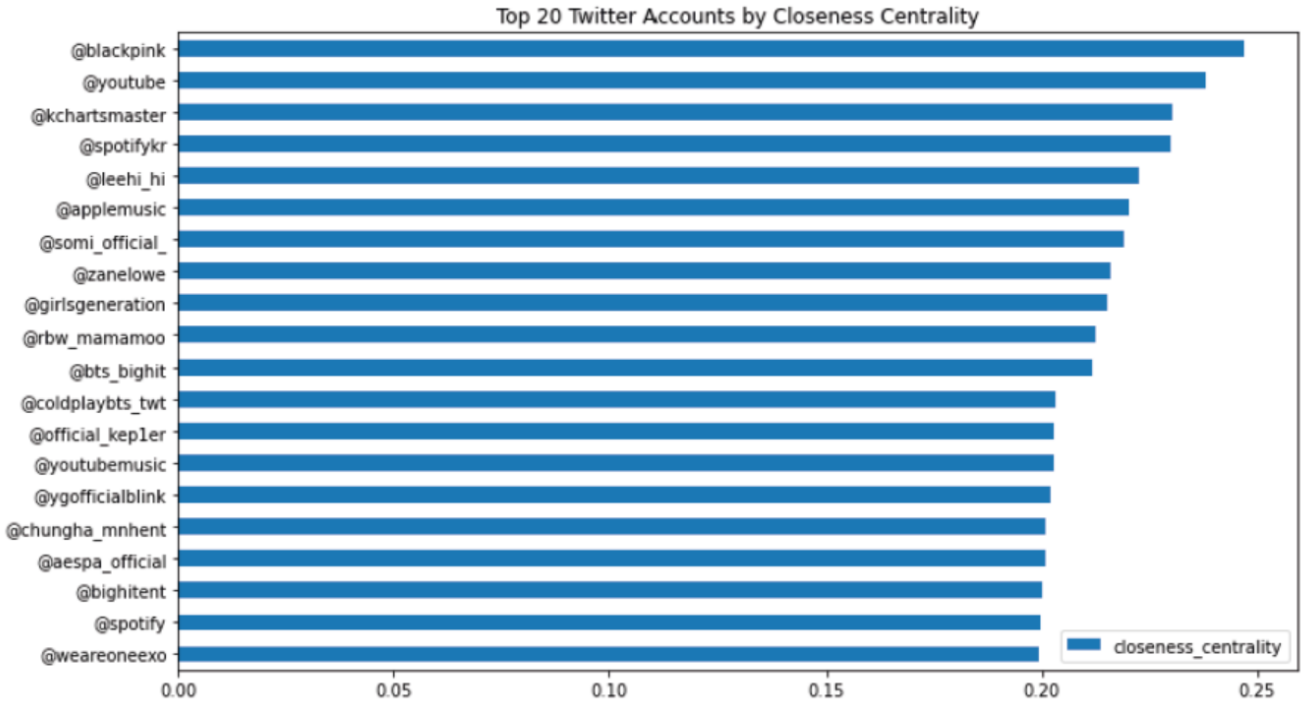
	closeness_centrality
@blackpink	0.247134
@youtube	0.238254
@kchartsmaster	0.230364
@spotifykr	0.229991
@leehi_hi	0.222560

Hình 2.28. pandas DataFrame với số đo gần gũi của các nút

Cuối cùng, có thể trực quan dữ liệu bằng biểu đồ cột ngang bằng lệnh sau:

```
title = 'Top 20 Twitter Accounts by Closeness Centrality'
_= closecent_df[0:20].plot.barh(title=title, figsize=(12,7))
plt.gca().invert_yaxis()
```

Biểu đồ cột ngang của các tài khoản Twitter với số đo gần gũi trong Hình 2.29.



Hình 2.29. Biểu đồ cột ngang của các tài khoản Twitter với số đo gần gũi

2.5.4 PageRank

Công thức toán học PageRank tính đến số lượng liên kết đến và đi ra của không chỉ một nút đang xét mà còn cả các nút liên kết.[6]

PageRank là một thuật toán rất nhanh, phù hợp cho các mạng lưới lớn và nhỏ, và rất hữu ích như một thước đo tầm quan trọng. PageRank hữu ích ngay cả đối với các mạng lưới lớn và dày đặc.

Có thể tính toán điểm PageRank của mỗi nút trong mạng.

[6] http://ilpubs.stanford.edu:8090/422/1/1999-66.pdf

```
pagerank = nx.pagerank(G)
pagerank
{'@kmg3445t': 0.00047123124840596525,
 '@code_kunst': 0.005226313735064201,
 '@highgrnd': 0.00047123124840596525,
 '@youngjay_93': 0.00047123124840596525,
 '@sobeompark': 0.00047123124840596525,
 '@justhiseung': 0.00047123124840596525,
 '@hwajilla': 0.00047123124840596525,
 '@blobyblo': 0.0014007295303692594,
 '@minddonyy': 0.00047123124840596525,
 ...
 }
```

Có thể sử dụng để tạo một DataFrame Pandas khác, được sắp xếp theo PageRank theo thứ tự giảm dần:

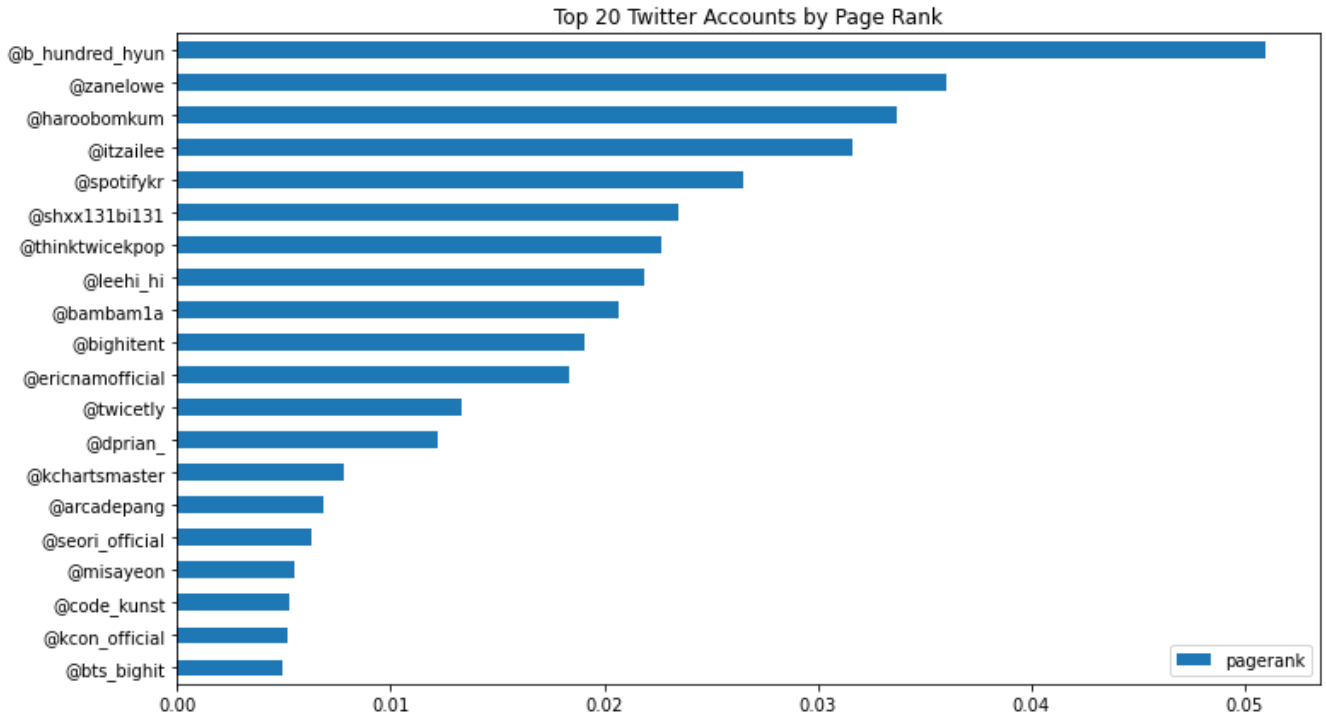
```
pagerank_df = pd.DataFrame(pagerank, index=[0]).T
pagerank_df.columns = ['pagerank']
pagerank_df.sort_values('pagerank', inplace=True, ascending=False)
pagerank_df.head()
```

Điều này sẽ hiển thị một bảng dữ liệu các tài khoản Twitter và điểm PageRank của chúng.

	pagerank
@b_hundred_hyun	0.050979
@zanelowe	0.036025
@haroobomkum	0.033742
@itzailee	0.031641
@spotifykr	0.026531

Hình 2.30. pandas DataFrame of nodes' Pag

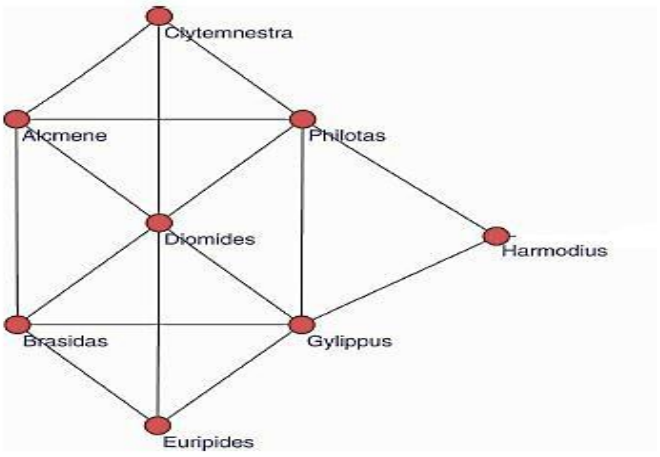
Cuối cùng, chúng ta có thể trực quan bằng biểu đồ ngang như sau:



Hình 2.31. Biểu đồ các tài khoản Twitter accounts bởi pagerank

2.6 BÀI TẬP

Câu 1: Cho mạng xã hội sau đây:

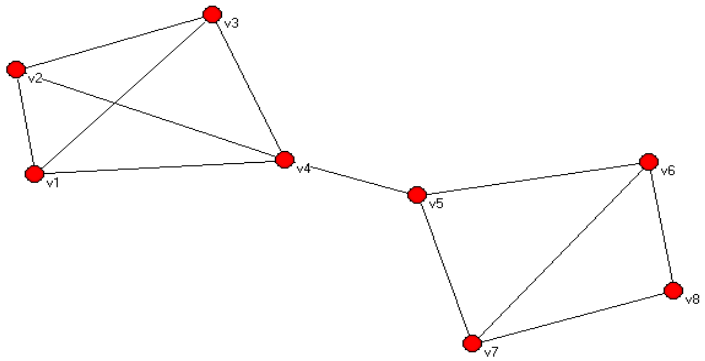


1.1. Cho biết công thức tính và ý nghĩa của các độ đo trung tâm là độ đo theo bậc (degree), độ đo trung gian (betweenness centrality), độ đo gần gũi (closeness centrality).

1.2. Dùng các công thức ở câu 1.1. để tính ma trận đường đi ngắn nhất phục vụ việc tính bằng tay các độ đo theo bậc, theo betweenness centrality và theo closeness centrality. Cho biết node có độ đo theo bậc, theo betweenness và theo closeness centrality lớn nhất. Nhận xét.

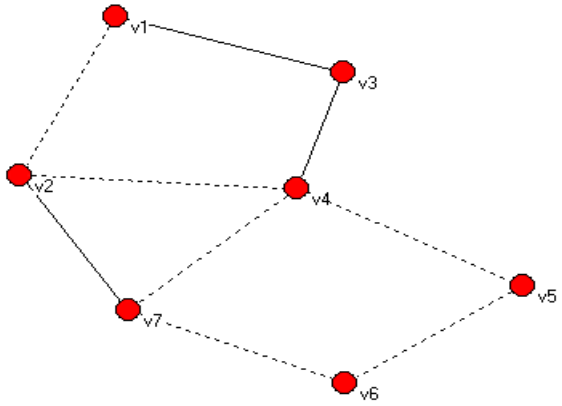
Câu 2: Cho biết ý tưởng một thuật toán tìm tất cả các đường đi ngắn nhất giữa 2 node trong mạng xã hội lớn, theo anh chị chúng ta có các cải tiến gì so với các thuật toán truyền thống?

Câu 3: Cho mạng xã hội:



Tìm số đo gom cụm cho các đỉnh trong mạng xã hội trên?

Câu 4: Cho đồ thị âm dương sau đây:

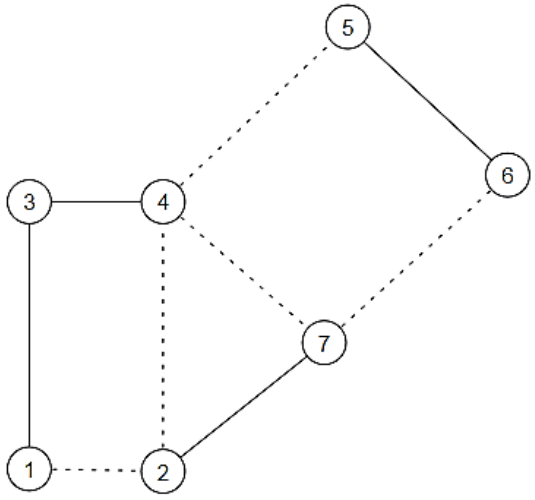


Cạnh đứt nét là cạnh âm, cạnh liền nét là cạnh đương

4.1. Cho biết ý tưởng của giải thuật tìm tất cả các chu trình, ứng dụng để tìm tất cả chu trình của đồ thị trên. Kiểm tra trong các chu trình đó có chứa một số chẵn cạnh âm (điều kiện đồ thị cân bằng).

4.2. Phân hoạch tập đỉnh của đồ thị ra làm 2 tập hợp đỉnh thỏa điều kiện đồ thị có dấu cân bằng.

Câu 5: Cho biết đồ thị sau có cân bằng hay không?



BÀI 3. CỘNG ĐỒNG MẠNG XÃ HỘI

Học xong bài này, sinh viên sẽ đạt được những mục tiêu sau:

- *Hiểu biết khái niệm về cộng đồng mạng xã hội.*
- *Biết cách khám phá cộng đồng mạng xã hội.*
- *Thể hiện thái độ tích cực, kiên nhẫn, hợp tác và sẵn sàng học hỏi trong quá trình học tập và ứng dụng kiến thức về Mạng xã hội.*

3.1 KHÁI NIỆM CỘNG ĐỒNG

Cộng đồng được tạo từ các cá nhân sao cho các cá nhân trong cùng một nhóm sẽ tương tác với nhau thường xuyên hơn các cá nhân nằm ngoài nhóm. Khám phá cộng đồng là tìm các nhóm trong mạng xã hội với hàm thành viên của nhóm không được xác định trước. Chúng ta cần khám phá các cộng đồng trên mạng xã hội vì:

- Con người có tương tác trong xã hội;
- Mạng xã hội cho phép con người mở rộng đời sống xã hội theo nhiều cách khác nhau;
- Trong thế giới thực, việc tìm và gặp bạn bè nhằm tìm bạn có cùng sở thích khó hơn tìm và trao đổi trong mạng xã hội;
- Việc tương tác giữa các cá nhân giúp xác định cộng đồng. Có ba tiếp cận để khám phá cộng đồng trong mạng xã hội:
- Dựa trên cấu trúc của mạng xã hội để khám phá cộng đồng;
- Dựa trên thông tin trao đổi theo các tương tác trên mạng xã hội;
- Dựa trên vừa cấu trúc và nội dung trao đổi.

3.2 KHÁM PHÁ CỘNG ĐỒNG

Một trong những bài toán thường gặp khi phân tích một mạng xã hội là việc phát hiện ra các cộng đồng theo một số tính chất nào đó trên mạng xã hội. Có nhiều phương pháp tiếp cận cho bài toán khám phá cộng đồng. Sau đây là các hướng tiếp cận chính:

- Node-Centric Community: Mỗi node trong cộng đồng sẽ thỏa một số tính nào đó;
- Group-Centric Community: Xem tất cả các liên kết trong một nhóm là một liên kết duy nhất. Trong cộng đồng thì mỗi nhóm sẽ phải thỏa một số tính chất nào đó. Ta không xem xét từng node mà xem một nhóm các node;
- Network-Centric Community: Chia mạng xã hội thành những tập con các node không liên thông nhau;
- Hierarchy-Centric Community: Xây dựng một cấu trúc phân cấp từ mạng xã hội.

Mỗi mạng xã hội đều có thể phân chia thành các đơn vị là các nhóm các node có liên kết chặt chẽ với nhau. Trong quần thể động vật, cộng đồng có thể là một nhóm động vật, trong mạng xã hội về khách hàng với hoạt động mua bán hàng thì cộng đồng có thể là các nhóm khách hàng khác nhau được phân nhóm dựa trên một số độ nào đó để tạo nhóm và thiết lập các chính sách bán hàng phù hợp cho từng nhóm.

Bài toán phát hiện cấu trúc cộng đồng trong mạng là bài toán được quan tâm cả trong toán học, sinh học, xã hội học. Có rất nhiều thuật toán được đề xuất để giải quyết. Sau đây là một số thuật toán:

- Thuật toán phát hiện cấu trúc dựa trên độ tương tự của đỉnh, thuật toán Girvan – Newman;
- Thuật toán phát hiện cấu trúc cộng đồng dựa trên độ đo tương tự các node;
- Thuật toán lan truyền nhãn.

3.2.1 Modularity

Số đo modularity (Q), được đề xuất bởi Girvan và Newman dùng làm thước đo khám phá cộng đồng.

Thuật toán này theo hướng Node-Centric Community.

Sức mạnh của cộng đồng được tính bằng công thức sau:

$$\sum_{i \in Cl, j \in Cl} (A_{ij} - d_i d_j / 2m)$$

Modularity:

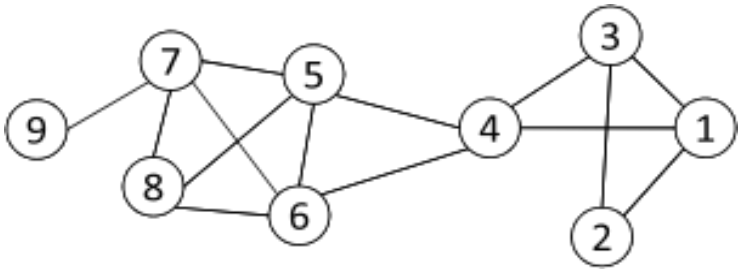
$$QQ = \frac{1}{2m} \sum_{l=1}^{k} \sum_{i \in Cl, j \in Cl} (A_{ij} - d_i d_j / 2m)$$

Trong đó:

m là số cạnh của đồ thị

d_i, d_j là bậc của đỉnh i và đỉnh j

Giá trị Modularity càng lớn, cộng đồng có cấu trúc càng tốt. Xét đồ thị trong Hình 3.1 để khám phá cộng đồng.



Hình 3.1. Đồ thị để khám phá cộng đồng

Modularity matrix được xác định bởi:

$$B = A - dd^T / 2m$$

$$B_{ij} = A_{ij} - d_i d_j / 2m$$

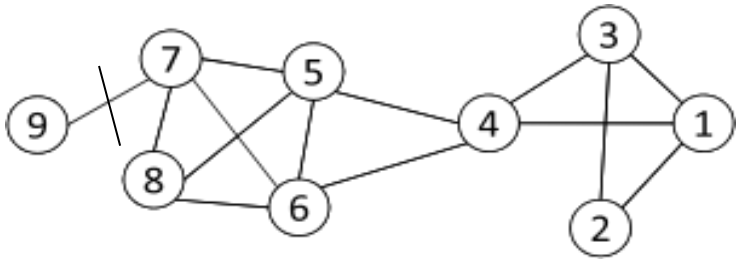
Ma trận Modularity của đồ thị trong Hình 3.1 được trình bày trong Hình 3.2.

$$B = \begin{bmatrix} -0.32 & 0.79 & 0.68 & 0.57 & -0.43 & -0.43 & -0.43 & -0.32 & -0.11 \\ 0.79 & -0.14 & 0.79 & -0.29 & -0.29 & -0.29 & -0.29 & -0.21 & -0.07 \\ 0.68 & 0.79 & -0.32 & 0.57 & -0.43 & -0.43 & -0.43 & -0.32 & -0.11 \\ 0.57 & -0.29 & 0.57 & -0.57 & 0.43 & 0.43 & -0.57 & -0.43 & -0.14 \\ -0.43 & -0.29 & -0.43 & 0.43 & -0.57 & 0.43 & 0.43 & 0.57 & -0.14 \\ -0.43 & -0.29 & -0.43 & 0.43 & 0.43 & -0.57 & 0.43 & 0.57 & -0.14 \\ -0.43 & -0.29 & -0.43 & -0.57 & 0.43 & 0.43 & -0.57 & 0.57 & 0.86 \\ -0.32 & -0.21 & -0.32 & -0.43 & 0.57 & 0.57 & 0.57 & -0.32 & -0.11 \\ -0.11 & -0.07 & -0.11 & -0.14 & -0.14 & -0.14 & 0.86 & -0.11 & -0.04 \end{bmatrix}$$

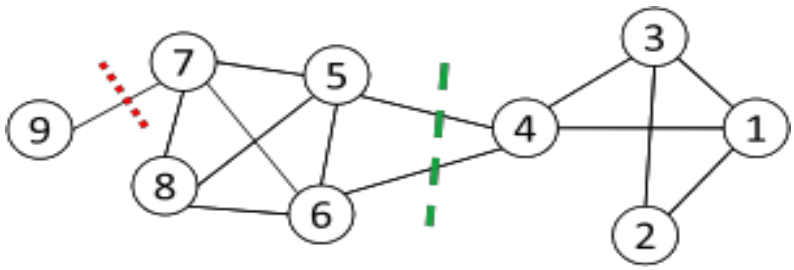
Hình 3.2. Ma trận Modularity

3.2.2 Nhát cắt

- Nhát cắt sẽ phân hoạch các đỉnh của đồ thị thành hai tập rời nhau (xem Hình 3.3);
- Bài toán nhát cắt tối thiểu: tìm một phân hoạch đồ thị sao cho số các cung của hai tập là tối thiểu;
- Từ các phân hoạch, chúng ta có cộng đồng;
- Thuật toán này theo hướng Node-Centric Community.



Hình 3.3. Ma trận modularity Ratio Cut & Normalized Cut



Hình 3.4. Nhát cắt phân hoạch đồ thị

- Nhát cắt tối thiểu thường cho kết quả là một phân hoạch không cân bằng, trong đó một tập hợp chứa một đỉnh (singleton), ví dụ node 9 trong Hình 3.4.

Ta có thể thay đổi hàm mục tiêu để xem xét kích thước của cộng đồng. Có hai hàm mục tiêu là:

$$Ratio\ Cut\ (\pi) = \frac{1}{k}\sum_{i=1}^{k}\frac{cut\ (C_i, \overline{C_i})}{|C_i|}$$

$$Normalize\ Cut\ (\pi) = \frac{1}{k}\sum_{i=1}^{k}\frac{cut\ (C_i, \overline{C_i})}{vol\ (C_i)}$$

Trong đó:

C_i: cộng đồng

$|C_i|$: số node trong cộng đồng C_i

$vol\ (C_i)$: tổng số bậc của các node trong cộng đồng C_i

Với phân hoạch π_1 tại cung <7,9>, ta có:

$$Ratio\ Cut\ (\pi_1) = \frac{1}{2}\left(\frac{1}{1}+\frac{1}{8}\right) = \frac{9}{16} = 0.56$$

$$Normalized\ Cut\ (\pi_1) = \frac{1}{2}\left(\frac{1}{1}+\frac{1}{27}\right) = \frac{14}{27} = 0.52$$

Với phân hoạch π_2 tại cung <4,5> và <4,6>, ta có:

$$Ratio\ Cut\ (\pi_2) = \frac{1}{2}\left(\frac{2}{4}+\frac{2}{5}\right) = \frac{9}{20} = 0.45 < Ratio\ Cut\ (\pi_1)$$

$$Normalized\ Cut\ (\pi_2) = \frac{1}{2}\left(\frac{2}{12}+\frac{2}{16}\right) = \frac{7}{48} = 0.15 < Normalized\ Cut\ (\pi_1)$$

Do số đo $Normalized\ Cut\ (\pi_2)$ có giá trị nhỏ hơn nên ta chọn phân hoạch $\boldsymbol{\pi_2}$, đó là phân hoạch tại cung <4,5> và <4,6>.

3.2.3 Thuật toán Girvan Newman

Thuật toán Girvan Newman do Michelle Girvan và Mark Newman đề xuất vào năm 2002. Số đ trung gian (betweenness centrality) được sử dụng để phát hiện cấu trúc cộng đồng trong các mạng, đặc biệt là trong thuật toán Girvan Newman. Ý tưởng của thuật toán là tính số đo "edge betweenness" của mỗi cung và sau đó là loại bỏ các cung có số đo "egde betweenness" cao nhất. Số đo "edge betweenness" của cung e được tính bằng tỷ số giữa số đường đi ngắn nhất của các cặp cung đi qua cung e và số đường đi ngắn nhất giữa các cặp node trong mạng. Số đo này được tính bằng công thức sau:

$$BC(e) = \sum_{\substack{u,w \in V \\ u \neq w}} \frac{\sigma_{uw}(e)}{\sigma_{uw}}$$

Thuật toán được thực hiện như sau:

Đầu vào: Một mạng xã hội G(V, E), trong đó V là tập đỉnh, E là tập cạnh

Đầu ra: Tập các cộng đồng $V_1, V_2, \dots V_k$ $(\cup_{i=1}^{k} V_i = V)$

Quá trình thực hiện:

Bước 1: Tính "edge betweenness" của tất cả các cung trong mạng xã hội theo công thức đã được trình bày về "edge betweenness" ở phần trên.

Bước 2: Tìm cung với "edge betweenness" cao nhất và xóa cung này.

Bước 3: Tính lại "edge betweenness" cho những cung bị ảnh hưởng.

Bước 4: Quay lại bước 2, cho đến khi không còn cung nào.

Ví dụ: Cho mạng xã hội như Hình 3.1.

Bước 1: Tính "edge betweeness" cho tất cả các cung trong mạng:

Edge betweenness

******* Xét cung <1,1>

KL: Cung <1,1> có edge betweenness 0.00

******* Xét cung <1,2>

- Cung <1,2> nằm trong Shortest Path = 1;2;

- Số đường đi ngắn nhất đi qua đỉnh = 1 và đỉnh 2 là = 1

Gia tăng Edge Betweenness là = 1.00

- Cung <1,2> nằm trong Shortest Path = 4;1;2;
- Số đường đi ngắn nhất đi qua đỉnh = 4 và đỉnh 2 là = 2

Gia tăng Edge Betweenness là = 0.50

- Cung <1,2> nằm trong Shortest Path = 5;4;1;2;
- Số đường đi ngắn nhất đi qua đỉnh = 5 và đỉnh 2 là = 2

Gia tăng Edge Betweenness là = 0.50

- Cung <1,2> nằm trong Shortest Path = 6;4;1;2;
- Số đường đi ngắn nhất đi qua đỉnh = 6 và đỉnh 2 là = 2

Gia tăng Edge Betweenness là = 0.50

- Cung <1,2> nằm trong Shortest Path = 7;5;4;1;2;
- Số đường đi ngắn nhất đi qua đỉnh = 7 và đỉnh 2 là = 4

Gia tăng Edge Betweenness là = 0.25

- Cung <1,2> nằm trong Shortest Path = 7;6;4;1;2;
- Số đường đi ngắn nhất đi qua đỉnh = 7 và đỉnh 2 là = 4

Gia tăng Edge Betweenness là = 0.25

- Cung <1,2> nằm trong Shortest Path = 8;5;4;1;2;
- Số đường đi ngắn nhất đi qua đỉnh = 8 và đỉnh 2 là = 4

Gia tăng Edge Betweenness là = 0.25

- Cung <1,2> nằm trong Shortest Path = 8;6;4;1;2;
- Số đường đi ngắn nhất đi qua đỉnh = 8 và đỉnh 2 là = 4

Gia tăng Edge Betweenness là = 0.25

- Cung <1,2> nằm trong Shortest Path = 9;7;5;4;1;2;

- Số đường đi ngắn nhất đi qua đỉnh = 9 và đỉnh 2 là = 4

Gia tăng Edge Betweenness là =0.25

- Cung <1,2> nằm trong Shortest Path =9;7;6;4;1;2;
- Số đường đi ngắn nhất đi qua đỉnh =9 và đỉnh 2 là =4

Gia tăng Edge Betweenness là =0.25

KL: Cung <1,2> có edge betweenness 4.00

******* Xét cung <1,3>

- Cung <1,3> nằm trong Shortest Path = 1;3;
- Số đường đi ngắn nhất đi qua đỉnh = 1 và đỉnh 3 là = 1

Gia tăng Edge Betweenness là = 1.00

KL: Cung <1,3> có edge betweenness 1.00

******* Xét cung <1,4>

- Cung <1,4> nằm trong Shortest Path = 1;4;
- Số đường đi ngắn nhất đi qua đỉnh = 1 và đỉnh 4 là = 1

Gia tăng Edge Betweenness là = 1.00

- Cung <1,4> nằm trong Shortest Path = 1;4;5;
- Số đường đi ngắn nhất đi qua đỉnh = 1 và đỉnh 5 là = 1

Gia tăng Edge Betweenness là = 1.00

- Cung <1,4> nằm trong Shortest Path = 1;4;6;
- Số đường đi ngắn nhất đi qua đỉnh = 1 và đỉnh 6 là = 1

Gia tăng Edge Betweenness là = 1.00

- Cung <1,4> nằm trong Shortest Path = 1;4;5;7;
- Số đường đi ngắn nhất đi qua đỉnh =1 và đỉnh 7 là = 2

Gia tăng Edge Betweenness là = 0.50

- Cung <1,4> nằm trong Shortest Path = 1;4;6;7;
- Số đường đi ngắn nhất đi qua đỉnh =1 và đỉnh 7 là = 2

Gia tăng Edge Betweenness là = 0.50

- Cung <1,4> nằm trong Shortest Path = 1;4;5;8;
- Số đường đi ngắn nhất đi qua đỉnh = 1 và đỉnh 8 là = 2

Gia tăng Edge Betweenness là = 0.50

- Cung <1,4> nằm trong Shortest Path = 1;4;6;8;
- Số đường đi ngắn nhất đi qua đỉnh = 1 và đỉnh 8 là = 2

Gia tăng Edge Betweenness là = 0.50

- Cung <1,4> nằm trong Shortest Path = 1;4;5;7;9;
- Số đường đi ngắn nhất đi qua đỉnh = 1 và đỉnh 9 là = 2

Gia tăng Edge Betweenness là = 0.50

- Cung <1,4> nằm trong Shortest Path = 1;4;6;7;9;
- Số đường đi ngắn nhất đi qua đỉnh =1 và đỉnh 9 là = 2

Gia tăng Edge Betweenness là = 0.50

- Cung <1,4> nằm trong Shortest Path = 2;1;4;
- Số đường đi ngắn nhất đi qua đỉnh = 2 và đỉnh 4 là = 2

Gia tăng Edge Betweenness là = 0.50

- Cung <1,4> nằm trong Shortest Path = 2;1;4;5;
- Số đường đi ngắn nhất đi qua đỉnh = 2 và đỉnh 5 là = 2

Gia tăng Edge Betweenness là = 0.50

- Cung <1,4> nằm trong Shortest Path = 2;1;4;6;
- Số đường đi ngắn nhất đi qua đỉnh =2 và đỉnh 6 là =2

Gia tăng Edge Betweenness là = 0.50

- Cung <1,4> nằm trong Shortest Path = 2;1;4;5;7;
- Số đường đi ngắn nhất đi qua đỉnh = 2 và đỉnh 7 là = 4

Gia tăng Edge Betweenness là = 0.25

- Cung <1,4> nằm trong Shortest Path = 2;1;4;6;7;
- Số đường đi ngắn nhất đi qua đỉnh = 2 và đỉnh 7 là = 4

Gia tăng Edge Betweenness là = 0.25

- Cung <1,4> nằm trong Shortest Path = 2;1;4;5;8;
- Số đường đi ngắn nhất đi qua đỉnh = 2 và đỉnh 8 là = 4

Gia tăng Edge Betweenness là = 0.25

- Cung <1,4> nằm trong Shortest Path = 2;1;4;6;8;
- Số đường đi ngắn nhất đi qua đỉnh = 2 và đỉnh 8 là = 4

Gia tăng Edge Betweenness là = 0.25

- Cung <1,4> nằm trong Shortest Path = 2;1;4;5;7;9;
- Số đường đi ngắn nhất đi qua đỉnh = 2 và đỉnh 9 là = 4

Gia tăng Edge Betweenness là = 0.25

- Cung <1,4> nằm trong Shortest Path = 2;1;4;6;7;9;
- Số đường đi ngắn nhất đi qua đỉnh = 2 và đỉnh 9 là = 4

Gia tăng Edge Betweenness là = 0.25

KL: Cung <1,4> có edge betweenness 9.00

Ta có được bảng "edge betweenness" của cung như trong Bảng 3.1:

Bảng 3.1. Bảng Edge Betweenness

	1	2	3	4	5	6	7	8	9
1	0	4	1	9	0	0	0	0	0
2	4	0	4	0	0	0	0	0	0
3	1	4	0	9	0	0	0	0	0
4	9	0	9	0	10	10	0	0	0
5	0	0	0	10	0	1	6	3	0
6	0	0	0	10	1	0	6	3	0
7	0	0	0	0	6	6	0	2	8
8	0	0	0	0	3	3	2	0	0
9	0	0	0	0	0	0	8	0	0

Trong Bảng 3.1 có hai cung <4,5> và <4,6> có "edge betweenness" bằng 10.

Bước 2: chọn cung <4,5> để loại bỏ vì cung <4,5> có giá trị edge betweenness cao nhất.

Bước 3: Tính lại "edge betweenness" sau khi đã loại bỏ cung <4,5>. Lúc này cung <4,6> trở thành cung có "edge betweenness" cao nhất, CB(4,6) = 20.

Sau khi bỏ cung này, ta có hai miền liên thông là: {1, 2, 3, 4} và {5, 6, 7, 8, 9}.

Thuật toán đươc thực hiện theo sơ đồ ở Hình 3.5. Tiến trình phân hoạch tạo cộng đồng của thuật toán Girvan Newman:

Hình 3.5. Tiến trình phân hoạch tạo cộng đồng của thuật toán Girvan Newman

Theo cách tương tự như đã mô tả ở trên, thuật toán được thực hiện cho đến khi không còn cung nào được loại bỏ. Ta có thể tìm các miền liên thông để lấy ra các cộng đồng. Thuật toán Girvan Newman tạo phân nhóm theo kiến trúc phân cấp.

Thuật toán Girvan Newman có độ phức tạp trong trường hợp xấu nhất là O (t^2n), trong đó t là số cung của của một mạng và n là số node. Trong một đồ thị đầy đủ (complete graph) thì $t = \frac{1}{2}n(n-1)$ và độ phức tạp trong trường hợp xấu nhất là O(n^5). Sau mỗi bước, số lượng những kết nối và những node để xem xét sẽ giảm dần, đặc biệt là cho những mạng có cấu trúc cộng đồng mạnh.

Thuật toán này vẫn còn xử lý chậm đối với những mạng lớn với hàng triệu node.

Hoạt động của thuật toán Girvan-Newman

1. Đầu tiên import thuật toán:

```
from networkx.algorithms.community import girvan_newman
```

2. Tiếp theo, chúng ta cần truyền biểu đồ vào thuật toán như một tham số. Khi thực hiện việc này, thuật toán sẽ trả về kết quả của mỗi lần phân chia lặp lại, chúng ta có thể nghiên cứu chúng bằng cách chuyển đổi kết quả thành một danh sách.

```
communities = girvan_newman(G)
communities = list(communities)
```

3. Số lần lặp tối đa mà thuật toán có thể thực hiện trước mỗi cộng đồng

```
len(communities)
76
```

4. Giả sử thấy rằng lần lặp phân chia thứ mười mang lại kết quả tốt nhất. Đặt kết quả của lần lặp thứ mười làm nhóm cộng đồng cuối cùng của mình, sau đó visualize các cộng đồng giống.

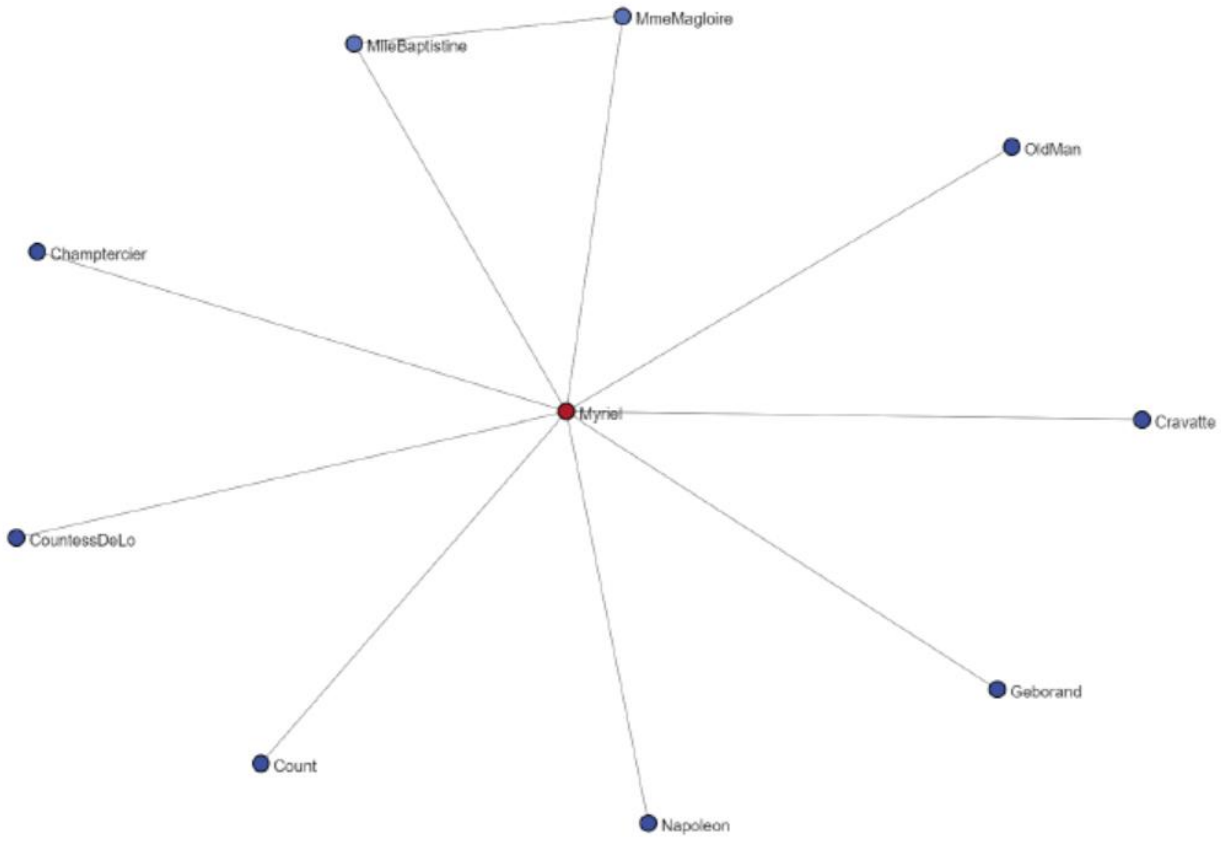
```
communities = communities[9]
```

5. So sánh cộng đồng này với những cộng đồng khác như sau:

```
community = communities[0]
G_community = G.subgraph(community)
```

```
draw_graph(G_community, show_names=True, node_size=5)
```

Chúng ta nhận được đầu ra như sau:

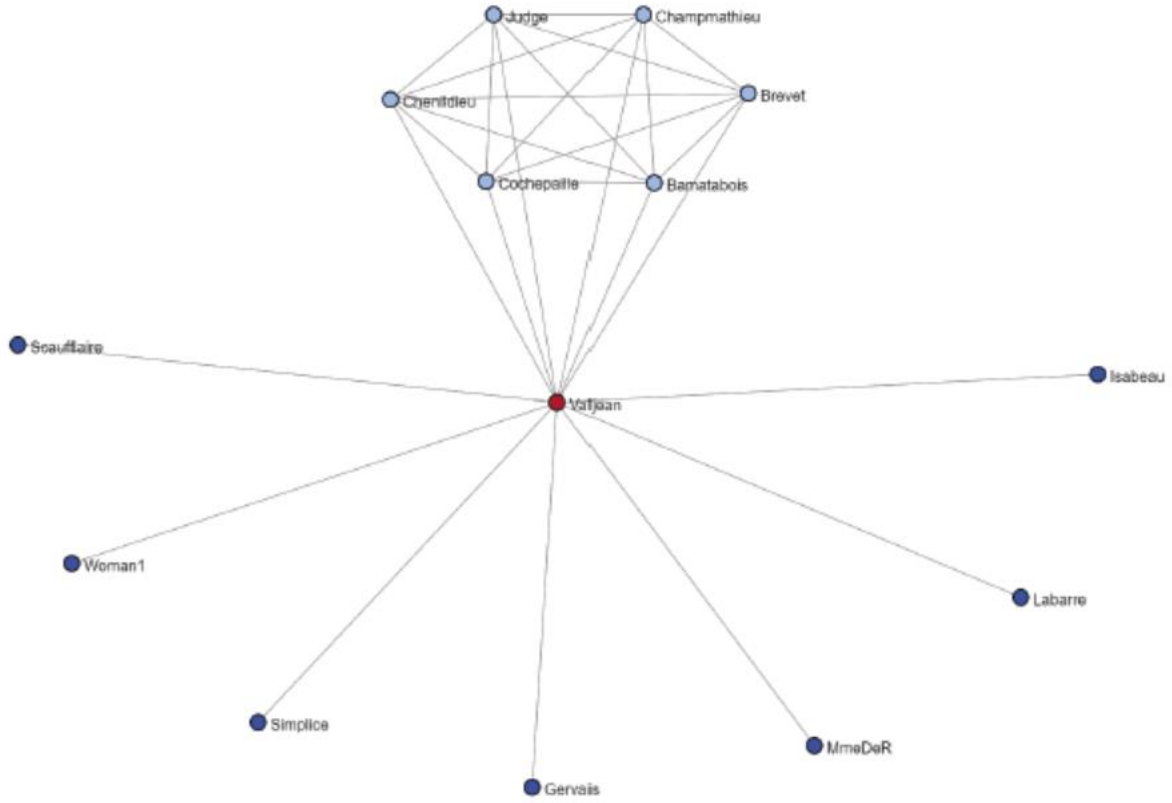


Hình 3.6. Girvan-Newman khám phá cộng đồng mạng Les Miserable, cộng đồng 0

6. Cộng đồng khác

```
community = communities[1]
G_community = G.subgraph(community)
draw_graph(G_community, show_names=True, node_size=5)
```

Việc này tạo ra trực quan mạng sau:

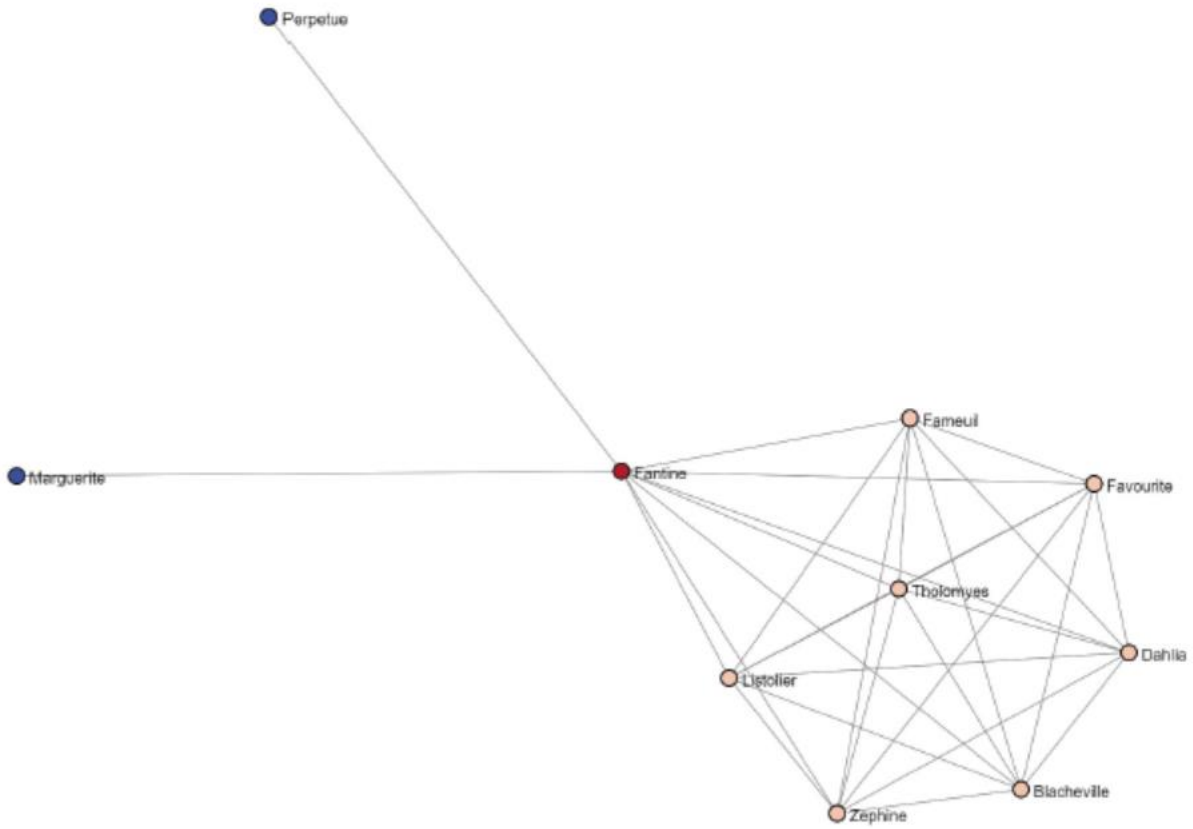


Hình 3.7. Girvan-Newman khám phá cộng đồng mạng Les Miserable, cộng đồng 1

Hình 3.7 cho kết quả có vẻ rất tốt. Không có gì lạ khi các cộng đồng có một nhóm kết nối chặt chẽ, cũng như một số nút ít kết nối hơn.

Và một cộng đồng khác:

```
community = communities[2]
G_community = G.subgraph(community)
draw_graph(G_community, show_names=True, node_size=5)
```



Hình 3.8. Girvan-Newman khám phá cộng đồng mạng Les Miserable, cộng đồng 2

Hình 3.8 tương tự như cộng đồng cuối cùng. Chúng ta có một nhóm các nút được kết nối chặt chẽ và hai nút có một cạnh duy nhất.

3.2.4 Thuật toán dựa trên độ tương tự của node (node similarity)

Độ tương tự của node dựa trên các node láng giềng. Thuật toán này theo hướng Node-Centric Community.

Hai node có cấu trúc tương tự nhau nếu chúng có chung tập các node láng giềng.

Ví dụ: Đồ thị trong Hình 3.1

Các node 1 và node 3 có cấu trúc tương tự. Các node 5 và 6 cũng có cấu trúc tương tự.

Ta cùng các số đo Jaccard và Cosine để tính độ tương tự của các node.

Jaccard Similarity

$$Jaccard\ (v_i, v_j) = \frac{|N_i \cap N_j|}{|N_i \cup N_j|}$$

Cosine similarity

$$Cosine\ (v_i, v_j) = \frac{|N_i \cap N_j|}{\sqrt{|N_i|.|N_j|}}$$

Độ tương tự của node 4 và node 6 trong Hình 3.1 được tính như sau:

$$Jaccard\ (4,6) = \frac{|\{5\}|}{|\{1,3,4,5,6,7,8\}|} = \frac{1}{7}$$

$$Cosine\ (4,6) = \frac{1}{\sqrt{4.4}} = \frac{1}{4}$$

Sau khi có ma trận, khoảng cách giữa các node, chúng ta có thể dùng các giải thuật k-means để khám phá các cluster.

Số đo tương tự giữa các node phản ánh về sự gần gũi của các node. Xét node i và node j. Do node i có thể gửi tài nguyên đến node j. Các node láng giềng của node i đóng vai trò những node trung gian để vận chuyển. Số đo tương tự của node i và node j được tính theo công thức:

$$S_{ij} = \sum_{z \in T(i) \cap T(j)} \frac{1}{k(z)}$$

Trong đó, T(i) là tập những node láng giềng của node i, và node z là node chung của $T(i) \cap T(j)$. Còn k(z) là số đo bậc của node z. S_{ij} bằng 0 khi node i không kết nối trực tiếp với node j.

Đối với mạng có n node, đầu tiên chúng ta tính độ tương tự của các cặp node (i, j) theo công thức nêu trên và lưu giá trị này vào ma trận S cấp n x n, trong đó S_{ij} là độ tương tự của node i và node j.

Ý tưởng của thuật toán: Thuật toán lặp đi lặp lại việc hợp nhất cộng đồng có chứa một node với các cộng đồng có chứa node tương tự lớn nhất với node đó để tạo cộng đồng mới

Đầu vào: Một mạng xã hội G(V, E), trong đó V là tập nút, E là tập cạnh và ma trận S đo độ tương tự của các nút trong mạng.

Đầu ra: Tập các cộng đồng V1, V2,…, Vk (với $\cup_{i=1}^{k} V_i = V$)

Quá trình thực hiện:

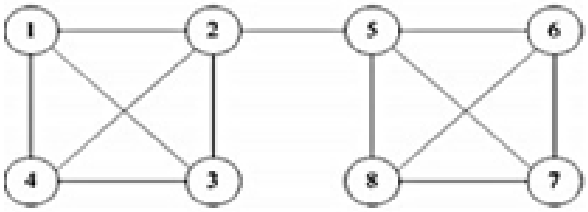
Bước 1: Ban đầu, mỗi node là một cộng đồng, thuật toán chọn một node bất kỳ làm node đầu tiên.

Bước 2: Hợp nhất cộng đồng có chứa node này với cộng đồng có chứa node có độ tương tự lớn nhất với node này để tạo cộng đồng mới.

Bước 3: Xác định node kế tiếp để xử lý: chọn node có độ tương tự lớn nhất với node hiện tại. Nếu node mới này không có trong cộng đồng hiện tại, thì đi đến bước 2. Ngược lại, chọn ngẫu nhiên một node mới (chưa được xét đến) làm node đầu tiên và thực hiện bước 2 để thực hiện hợp nhất cộng cồng.

Bước 4: Quay lại bước 2 và bước 3 cho đến khi không còn node nào chưa được đi qua.

Ví dụ: cho mạng xã hội như Hình 3.9:



Hình 3.9. Đồ thị tính số đo tương tự node

Ban đầu tính độ tương tự của các node:

Xét nút 1 và 2, ta có $\tau(1) \cap \tau(2) = \{4, 3\}$. Ta có degree(4) = 3 và degree(3) = 3. Do đó ta có S12 = 2/3.

Xét đỉnh 1 và 6, do không có kết nối nên S16 = 0. Tương tự xét cho các cặp đỉnh khác. Cuối cùng ta có ma trận S như trong Hình 3.10:

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	0	2/3	7/12	7/12	0	0	0	0
2	2/3	0	2/3	2/3	0	0	0	0
3	7/12	2/3	0	7/12	0	0	0	0
4	7/12	2/3	7/12	0	0	0	0	0
5	0	0	0	0	0	2/3	2/3	2/3
6	0	0	0	0	2/3	0	7/12	7/12
7	0	0	0	0	2/3	7/12	0	7/12
8	0	0	0	0	2/3	7/12	7/12	0

Hình 3.10. Ma trận số đo tương tự của node

Bước 1: Ta có 8 cộng đồng bao gồm {1}, {2}, {3}, {4}, {5}, {6}, {7}, {8} và chọn ngẫu nhiên node 1.

Bước 2: Node 1 có node 2 với độ tương tự với node 1 cao nhất nên hợp {1} và {2} thành {1, 2}.

Bước 3: Các node 1, 3, 4 cùng có độ tương tự bằng nhau so với node 2. Chọn ngẫu nhiên node 3 để xét tiếp. Vì node {3} không nằm trong cộng đồng {1, 2} nên quay lại bước 2, ta hợp nhất cộng đồng {1, 2} và {3} thành {1, 2, 3}. Có node 2 có độ tương đồng cao nhất với node 3. Nhưng do node 3 thuộc vào cộng đồng {1, 2, 3} đang xét nên chọn một node mới ngẫu nhiên để xét tiếp. Chọn node 4, node 2 có độ tương tự cao nhất với nó. Nên hợp nhất cộng đồng {1, 2, 3} và {4} thành {1, 2, 3, 4}. Chọn tiếp node có độ tương đồng cao nhất với 4 làm node xét tiếp theo thì đó là node 2. Do node 2 thuộc vào cộng đồng đang xét. Chọn ngẫu nhiên node khác, chọn node 5. Thuật toán tiếp tục thực hiện cho đến khi nào không còn node nào chưa xét nữa thì thuật toán sẽ dừng.

Kết thúc thuật toán ta tìm được 2 cộng đồng là:{1, 2, 3, 4} và {5, 6, 7, 8}.

Chi phí tính toán của thuật toán này bao gồm: Chi phí tính độ tương tự, tìm kiếm các nút tiếp theo để xử lý, và hợp nhất cộng đồng lại với nhau. Tính độ tương đồng chỉ tính những node láng giềng. Độ phức tạp khi tính độ tương đồng của k node láng giềng là O(k). Độ phức tạp để tính toán độ tương đồng của mạng n có k node là O(nk). Thuật toán cần không gian bộ nhớ ít nhất là O(nk).

3.2.5 Thuật toán khám phá cộng đồng bằng lan truyền nhãn (Label Propagation Community Detection)

Nhiều thuật toán phát hiện cộng đồng đã được đề xuất và sử dụng với mức độ hiệu quả khác nhau trong nghiên cứu phát hiện cộng đồng. Thuật toán Label Propagation (LPA) lần đầu tiên được được đề xuất bởi Raghavan et al. (2007). LPA sử dụng các định danh duy nhất của các đỉnh là nhãn và lan truyền nhãn dựa trên điều kiện là đến càng nhiều đỉnh lân cận và mỗi đỉnh chọn một nhãn từ vùng lân cận để làm nhãn của đỉnh.

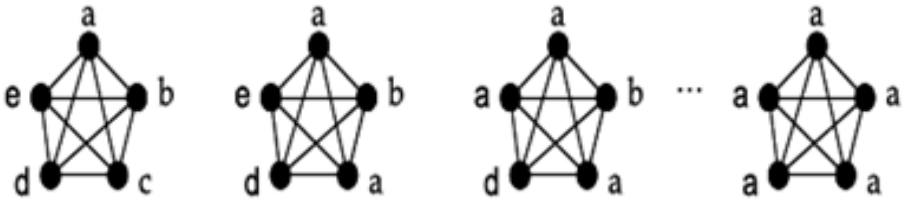
Thuật toán LPA được tóm tắt như sau:

Đỉnh x có các láng giềng $x_1, x_2, x_3, \dots, x_n$ và mỗi láng giềng được gán nhãn nhằm cho biết đỉnh đó thuộc về cộng đồng nào.

Mỗi đỉnh trong đồ thị mạng xã hội sẽ chọn một cộng đồng để tham gia và số lượng láng giềng tối đa thuộc về một cộng đồng là ngẫu nhiên

Lúc đầu, mỗi đỉnh được khởi tạo với một nhãn cộng đồng duy nhất. Nhãn (gọi là định danh) và các nhãn cộng đồng được lan truyền qua mạng. Tại mỗi bước lan truyền nhãn, mỗi đỉnh sẽ cập nhật nhãn cộng đồng của nó dựa trên nhãn của láng giềng làm nhãn lan truyền.

Các nhóm đỉnh sẽ kết nối chặt chẽ nhằm đạt được sự đồng thuận về một nhãn như trong Hình 3.11:



Hình 3.11. Các nhãn đỉnh được cập nhật từ trái sang phải

(Nguồn: Raghavan et al. 2007)

Vào cuối tiến trình lan truyền nhãn, hầu hết các nhãn biến mất và chỉ một vài nhãn cộng đồng còn tồn tại trong mạng xã hội. Thuật toán lan truyền nhãn sẽ hội tụ khi mỗi đỉnh có nhãn chính là nhãn của láng giềng của nó.

Vào cuối tiến trình, các đỉnh kết nối với cùng một nhãn mẫu trở thành một cộng đồng.

Theo thuật toán LPA, các cộng đồng là các đỉnh có nhãn giống nhau vào cuối tiến trình hội tụ.

Việc cập nhật nhãn cộng đồng cho đỉnh có thể được thực hiện đồng bộ hoặc không đồng bộ trong tiến trình lan truyền nhãn.

Trong cập nhật đồng bộ, một đỉnh x tại lần lặp thứ t việc cập nhật nhãn cộng đồng của đỉnh này dựa trên nhãn của các đỉnh lân cận tại thời điểm lặp t - 1. Vì thế:

$$C_x(t) = f\,(C_{x1}(t-1), C_{x2}(t-1), \ldots C_{xk}(t-1))$$

Trong đó $C_x(t)$ là nhãn của đỉnh x tại thời điểm t.

Tiến trình cập nhật không đồng bộ, một đỉnh ở lần lặp lại thứ t, sẽ cập nhật nhãn của nó dựa trên các nhãn của láng giềng tại thời điểm lặp t cũng như t - 1. Nhãn mới của các đỉnh bị cập nhật trong lần lặp hiện hành t và các đỉnh không bị cập nhật trong thời gian hiện hành t, thì các nhãn trong lần lặp thứ t - 1 sẽ được chọn. Do đó có thể biểu diễn việc cập nhật không đồng bộ như sau:

$$C_x(t) = f\,(C_{x,1}(t), \ldots, C_{x,m}(t), C_{x,m+1}(t-1)\,..C_{x,k}(t-1))$$

Trong đó x_{i1} đến x_{im} là láng giềng của đỉnh x đã được cập nhật trong lần lặp hiện tại, trong khi $x_{i(m+!)}$ đến x_{ik} là những láng giềng chưa được cập nhật.

Có thể tóm tắt thuật toán LPA như sau:

Bước 1. Khởi tạo các nhãn tại tất cả các đỉnh trong mạng. Đối với một đỉnh cho trước x, Cx (0) = x.

Bước 2. Đặt t = 1.

Bước 3. Sắp xếp các đỉnh trong mạng theo thứ tự ngẫu nhiên và đưa đỉnh vào X.

Bước 4. Với mỗi $x \in X$ được chọn theo thứ tự cụ thể, thực hiện:

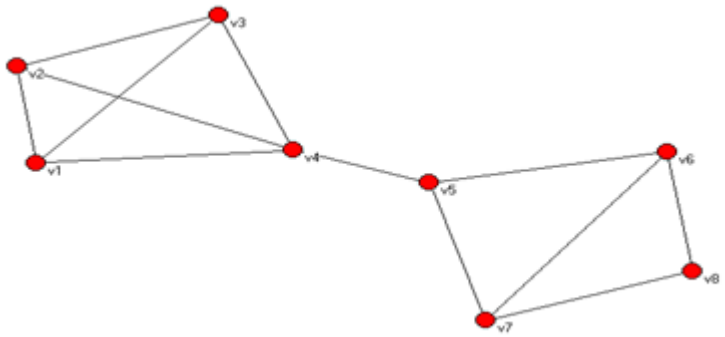
$$C_x(t) = f(C_{x,1}(t),\ldots,C_{x,m}(t),C_{x,m+1}(t-1)..C_{x,k}(t-1))$$

Hàm f trả về nhãn xảy ra với tần suất cao nhất trong số những người láng giềng và liên kết bị chia ngẫu nhiên.

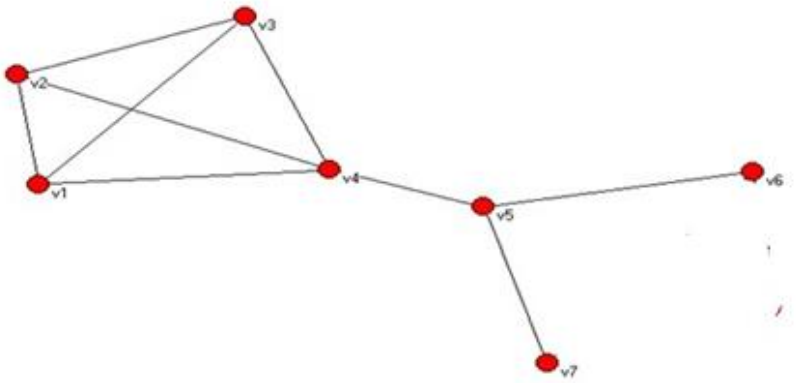
Bước 5. Nếu mỗi đỉnh có một nhãn là nhãn mà đại đa số láng giềng của đỉnh đó, thì dừng thuật toán, ngược lại, đặt t = t + 1 và quay về bước 3.

3.3 BÀI TẬP

Câu 1: Hãy dùng thuật toán Givan Newman để khám phá cộng đồng cho các đỉnh trong đồ thị sau:



Câu 2: Hãy dùng thuật toán Label Propagation để khám phá cộng đồng cho các đỉnh trong đồ thị sau:


v3
v2
v1
v4
v5
v6
v7

BÀI 4. DỰ ĐOÁN LIÊN KẾT MẠNG XÃ HỘI

Học xong bài này, sinh viên sẽ đạt được những mục tiêu sau:

- *Hiểu biết cơ bản tổng quan về dự đoán liên kết.*

- *Nắm được một số phương pháp tiếp cận dự đoán liên kết, điểm tương đồng giữa 2 đỉnh và các ứng dụng của dự đoán liên kết.*

- *Thể hiện thái độ tích cực, kiên nhẫn, hợp tác và sẵn sàng học hỏi trong quá trình học tập và ứng dụng kiến thức về Mạng xã hội.*

4.1 TỔNG QUAN VỀ DỰ ĐOÁN LIÊN KẾT

4.1.1 Giới thiệu

Hiện nay, dữ liệu thu được từ các mạng xã hội rất phong phú và rất quan trọng đối với các nhà phân tích dữ liệu. Một trong nhiều câu hỏi mà các nhà phân tích dữ liệu đặt ra là "Có các quan hệ nào giữa các cá nhân trong tổ chức?" và "những người có khả năng tương tác trực tiếp với người này là ai?". Để trả lời các câu hỏi này chúng ta cần nghiên cứu bài toán dự đoán liên kết (link prediction) trong mạng xã hội.

Bài toán dự đoán liên kết nhằm mục đích nhận định có tồn tại hay không mối liên kết trong tương lai giữa hai actor trên mạng xã hội. Để giải bài toán dự đoán liên kết, chúng ta dựa trên các dữ liệu quan sát, đây là các liên kết hiện có trên mạng xã hội để suy đoán khả năng xuất hiện các liên kết mới trong tương lai. Dự đoán liên kết thường được ứng dụng trong business intelligence, cộng tác… nhằm dự đoán các mối quan hệ trong tương lai từ dữ liệu hiện có.

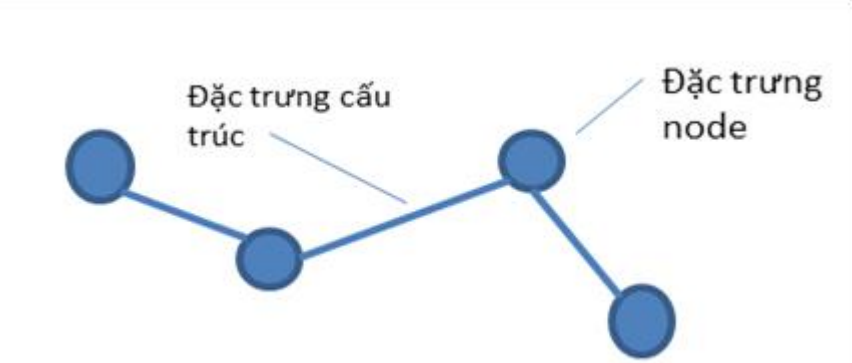
Hầu hết các phương pháp khai phá dữ liệu hiện nay đều làm việc trên một hoặc nhiều bảng dữ liệu, trong đó dòng là bản ghi và cột là các thuộc tính. Các bản ghi có thể xem là các vector có các giá trị là các giá trị của thuộc tính tương ứng. Các bảng

dữ liệu có thể kết nối với nhau để thành một bảng dữ liệu duy nhất cho khai phá dữ liệu.

Ngày càng xuất hiện nhiều hệ thống kết nối thông tin, ví dụ mạng xã hội, mạng tương tác, mạng trích dẫn. Trong các mạng này, dữ liệu liên kết các node đóng vai trò quan trọng và chúng ta có thể dự đoán khả năng xảy ra liên kết.

4.1.2 Nhiệm vụ dự đoán liên kết

Bài toán dự đoán liên kết cho các mạng xã hội dựa trên đặc điểm của node (node features) và đặc điểm cấu trúc của mạng xã hội (structural features) (xem Hình 4.1).



Hình 4.1. Đặc trưng cấu trúc mạng

Mạng xã hội được được biểu diễn theo cấu trúc đồ thị, trong đó mỗi node đại diện cho dữ liệu và một liên kết đại diện cho mối quan hệ giữa hai dữ liệu, nói cách khác, các node đại diện cho các yếu tố cấu thành và liên kết đại diện cho mối quan hệ giữa chúng. Ngoài ra, mỗi node cũng có thể có một liên kết đến dữ liệu là vector cấu trúc trong mô hình mạng.

Phương pháp tiếp cận học có giám sát dựa vào việc học một bộ phân loại nhị phân sẽ được dùng để dự đoán xem có tồn tại hay không một liên kết giữa một cặp của các node (Hassan et al., 2006).

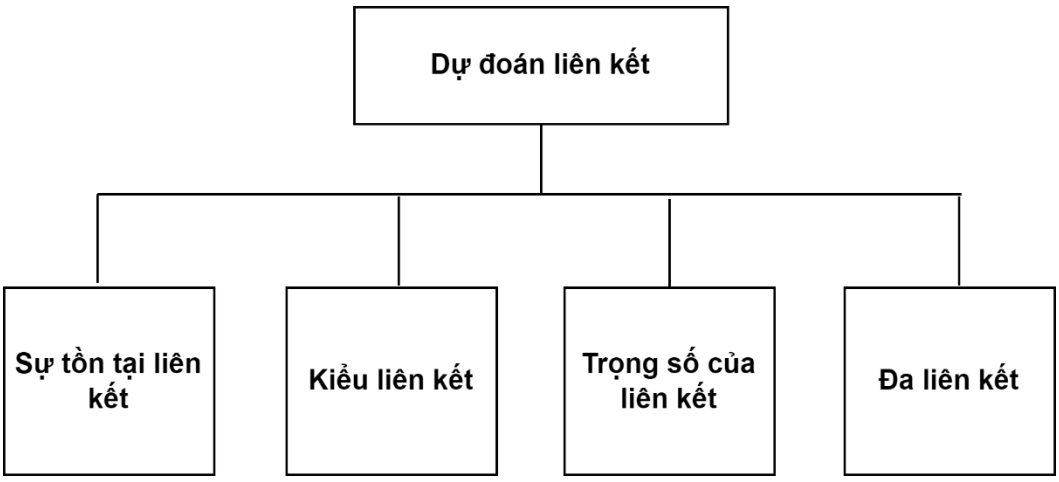
Hơn nữa, phân loại xem một liên kết tồn tại hay không có thể được thực hiện bằng cách sử dụng các thuật toán học có giám sát/phân loại khác nhau như các thuật toán trên cây quyết định, K láng giềng gần nhất hoặc máy vector hỗ trợ (SVM).

Phân loại dựa trên các thuộc tính của node và thuộc tính của đồ thị (cung, đường đi,...).

- Thuộc tính của node bao gồm số láng giềng, sở thích, mô hình chủ đề, cộng đồng, dữ liệu nhân khẩu học (vị trí địa lý)…;
- Thuộc tính dựa trên đồ thị: chiều dài của đường đi ngắn nhất, chồng chéo vùng lân cận, tầm quan trọng, thời điểm liên kết…;
- Mô hình đồ thị có hướng so với mô hình đồ thị vô hướng (ví dụ Markov Networks), chẳng hạn như các mạng Bayesian và PRMs cho phép dễ dàng nắm bắt sự phụ thuộc của sự tồn tại liên kết trên các thuộc tính và hạn chế xác suất phụ thuộc đồ thị có hướng.

Hình 4.2 cho thấy 4 bài toán trong dự đoán liên kết.

Hầu hết các nghiên cứu về dự đoán liên kết đều tập trung vào vấn đề có tồn tại liên kết, nghĩa là dự đoán trong tương lai có xuất hiện liên kết giữa hai node trong mạng xã hội hay không?

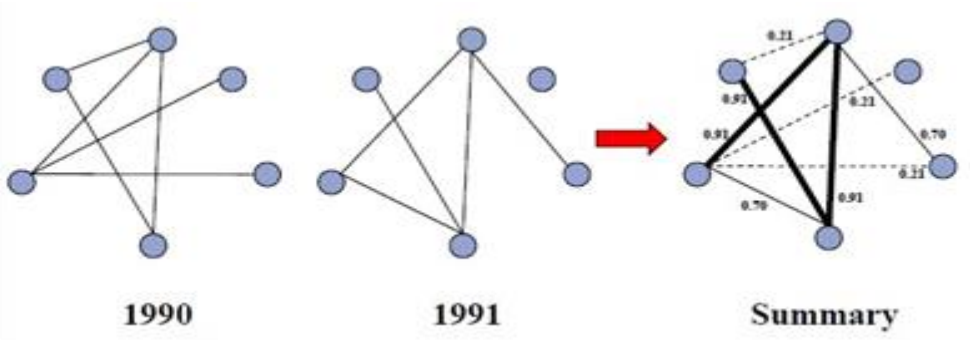


Hình 4.2. Bốn nhiệm vụ dự đoán liên kết

Bài toán dự đoán tồn tại liên kết có thể dễ dàng mở rộng sang kiểu liên kết có trọng và đa liên kết (nhiều hơn một liên kết giữa cùng cặp nút trong một mạng xã hội).

Bài toán dự đoán kiểu liên kết có điểm mới là dự đoán hình thức hoặc vai trò của liên kết sẽ xuất hiện trong tương lai giữa hai actor.

Khi dự báo liên kết, ta có thể áp dụng mô hình đồ thị xác suất (probabilistic graphic model), trong đó mỗi cung sẽ có trọng số là xác suất (xem Hình 4.3).



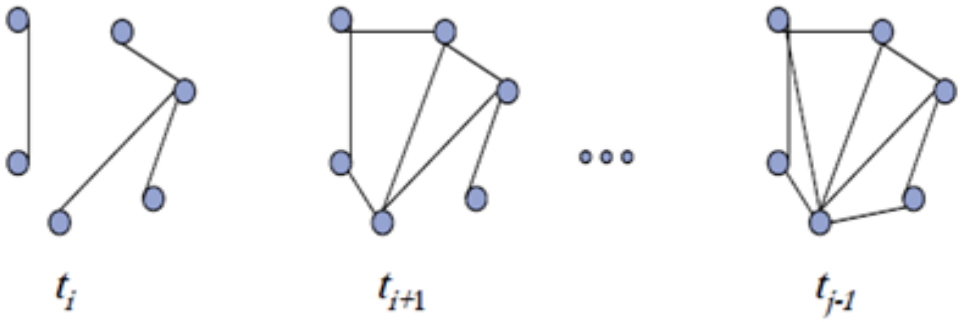
Hình 4.3. Áp dụng mô hình xác suất để dự đoán liên kết

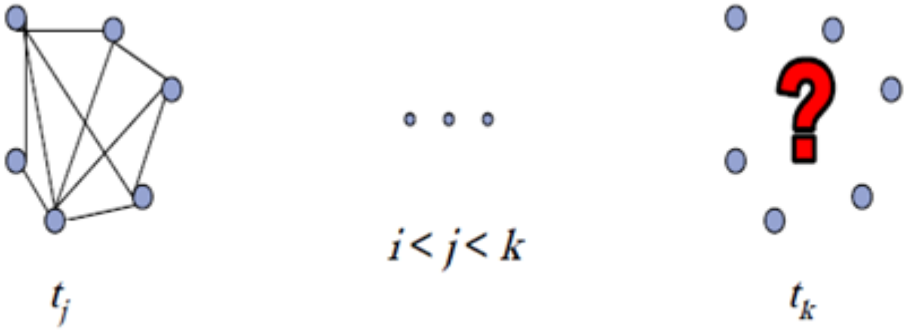
4.1.3 Mô tả bài toán dự đoán liên kết

Bài toán dự đoán liên kết thường được mô tả bởi dữ liệu được biểu diễn dưới dạng đồ thị với các đỉnh đại diện cho các thực thể và các cung đại diện cho các liên kết hay nói cách khác là tương tác giữa các node

Thông thường, dữ liệu được biểu diễn dưới dạng một tập hợp các liên kết. Một liên kết bao gồm thực thể được liên kết qua một số mối quan hệ. Một liên kết được sử dụng để thể hiện một loạt các mối quan hệ khác nhau như: giao tiếp trực tiếp, đồng xảy ra, hoặc chia sẻ các thuộc tính chung.

Hình 4.4 biểu diễn trực quan bài toán dự đoán liên kết. Trong đó dấu chấm hỏi nhằm trả lời câu hỏi có liên kết mới nào sẽ xuất hiện hay không?





Hình 4.4. Biểu diễn trực quan bài toán dự đoán liên kết

Bài toán dự đoán liên kết được phát biểu như sau: Cho một tập dữ liệu $V = \{v_i\}^n{}_{i=1}$, được tổ chức dưới hình thức của một mạng xã hội $G = (V, E)$, trong đó E là tập hợp các liên kết được quan sát. Mục đích của bài toán là dự đoán có tồn tại hay không một liên kết chưa thấy $e_{ij} \notin E$ giữa một cặp tùy ý của các node $< v_i, v_j >$ trong mạng dữ liệu. Với một hình chụp nhanh (snapshot) của một mạng xã hội tại thời gian t, bài toán đặt ra là dự đoán chính xác các cung sẽ được bổ sung vào mạng trong khoảng thời gian từ thời gian t đến thời gian t' trong tương lai.

Cũng có thể phát biểu bài toán như sau: Cho thể hiện đồ thị mạng xã hội tại thời điểm t(i) được mô hình hóa thành một đồ thị vô hướng có trọng số Gt(i) = (V, E).

Trọng số của một cạnh có giá trị bằng tổng số lần liên lạc của hai node.

Mạng xã hội đang phát triển được mô hình hóa thành một đồ thị đang phát triển EG, trong đó:

$$EG = \{Gt(i) | i = 1 \dots p, t(i) < t(i+1)\}$$

Kết quả của sự dự đoán là một đồ thị Gt(p+1).

Mô hình này tập trung vào vấn đề xác định các cung được thêm vào thể hiện đồ thị dự đoán, thừa nhận số node là không đổi trong suốt quá trình phát triển của mạng.

Việc dự đoán chỉ giới hạn ở dự đoán các liên kết sẽ tồn tại hay không tồn tại, không dự đoán trọng số của các liên kết.

Dự đoán liên kết là bài toán quan trọng trong phân tích mạng xã hội cũng như các ứng dụng trong các lĩnh vực khác như truy cập thông tin, tin sinh học và thương mại điện tử…

Có nhiều kỹ thuật dự đoán liên kết khác nhau. Hiện nay có 3 phương pháp dự đoán liên kết đại diện như sau: phương pháp đầu tiên dựa trên học máy (machine learning) để tạo mô hình phân loại nhị phân. Thứ hai, phương pháp tiếp cận dựa trên mô hình cấu trúc topo. Cuối cùng là tiếp cận theo mô hình xác suất Markov, Bayes, Random (xem Hình 4.5).

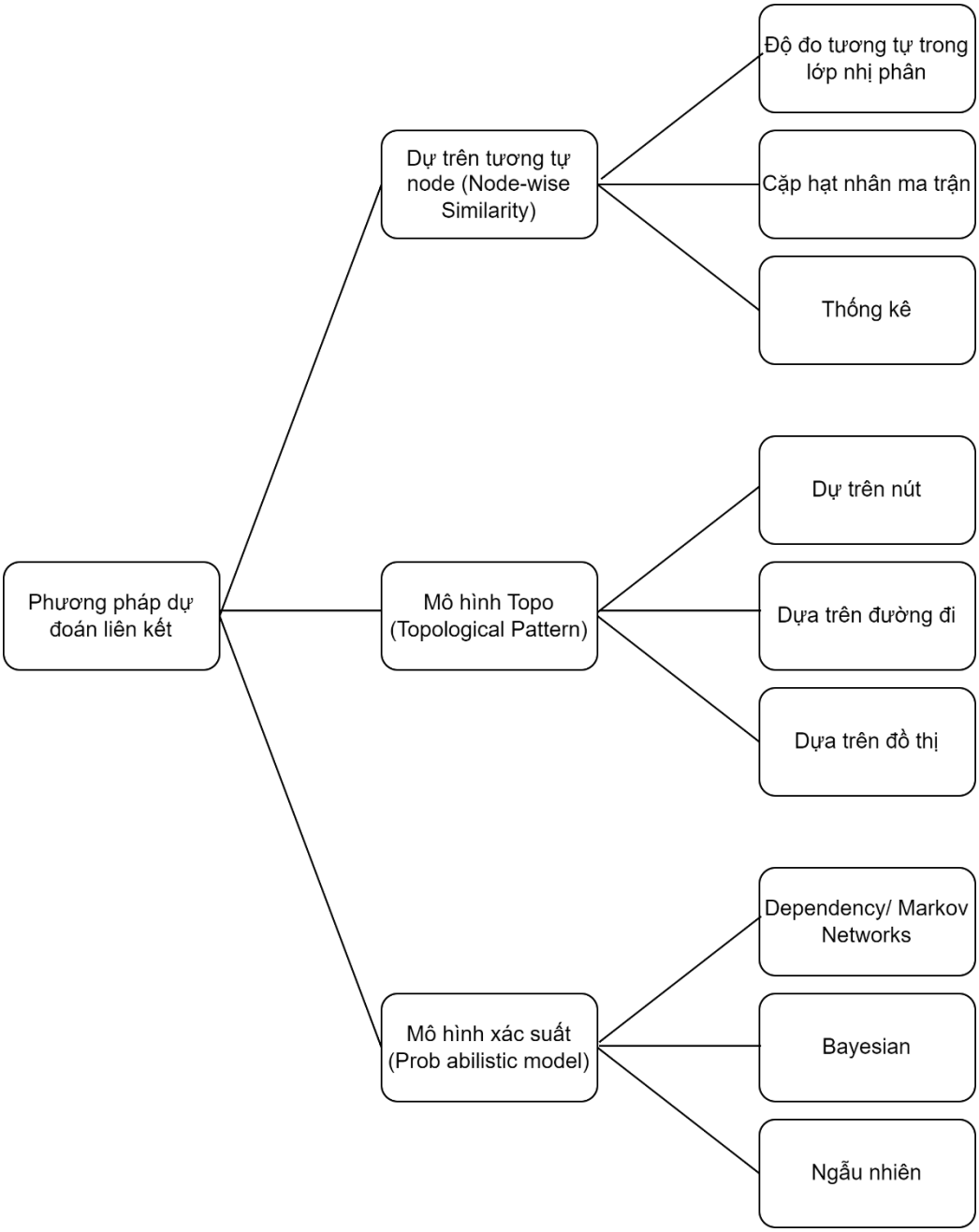
4.2 CÁC CÁCH TIẾP CẬN DỰ ĐOÁN LIÊN KẾT

Dự đoán liên kết sử dụng bằng các phương pháp học máy có giám sát. Mohammad Al Hasan, Vineet Chaoji, Saeed Salem và Mohammed J. Zaki sử dụng một số đặc trưng cấu trúc mạng và thuộc tính của node thực hiện dự đoán trên mạng đồng tác giả BIOBASE và DBLP.

- Năm 2009, các tác giả Anet Potgieter, Kurt A. April, Richard J. E, Cooke và Isaac O. Osunmakinde dự đoán liên kết sử dụng mạng Bayes. Thử nghiệm trên mạng kết bạn Pussokram.

- Năm 2011 Naoki Shibata, Yuya Kajikawa, Ichiro Sakata xây dựng mô hình dự đoán trên mạng trích dẫn và sử dụng mô hình phân lớp SVM. Ngoài ra, L. Backstrom và J. Leskovec cũng đã nghiên cứu về dự báo và khuyến nghị liên kết trong mạng xã hội, trong đó phát triển thuật toán dựa trên giải thuật Supervised Random Walks, sử dụng các thông tin cấu trúc mạng kết hợp với các thuộc tính của node. Hệ thống được thử nghiệm trên mạng Facebook.

4.3 MỘT SỐ PHƯƠNG PHÁP DỰ ĐOÁN LIÊN KẾT



Hình 4.5. Tóm tắt phương pháp dự đoán liên kết

4.4 ĐIỂM TƯƠNG ĐỒNG GIỮA 2 ĐỈNH

4.4.1 Khoảng cách đồ thị

Đây là phương pháp dự đoán liên kết dựa trên sự tương tự của node. Phương pháp này xác định một trọng số thể hiện liên kết là $score(x, y)$ của cặp node x, y dựa trên các đồ thị đầu vào, sau đó tạo sinh một danh sách các node được xếp hạng theo thứ tự giảm dần của $score(x, y)$. Khoảng cách này phản ánh sự "tương tự" giữa các nút x và y. Khoảng cách chính là chiều dài của con đường ngắn nhất nối 2 node x và node y.

4.4.2 Láng giềng chung

Đây là phương pháp dự đoán liên kết dựa trên mô hình topo. Việc xác định $score(x, y)$ dựa trên số các láng giềng chung của node x và node y. Newman, đã tính số lượng này trong mạng cộng tác để xác định sự tương quan giữa số các láng giềng chung của node x và node y tại thời điểm t, và khả năng họ sẽ cộng tác với nhau trong tương lai. Công thức tính điểm như sau:

$$score(x, y) = |\Gamma x \cap \Gamma(y)|$$

4.4.3 Hệ số Jaccard

Đây là phương pháp dự đoán liên kết dựa trên mô hình topo. Hệ số Jaccard được sử dụng khi truy vấn thông tin xác suất mà node x và node y có chung đặc điểm f. Hệ số Jaccard được sử dụng để so sánh sự giống nhau và sự đa dạng của những láng giềng trong mạng G theo công thức:

$$score(x, y) = \frac{|\Gamma(x) \cap \Gamma(y)|}{|\Gamma(x) \cup \Gamma(y)|}$$

4.4.4 Hệ số Adamic/Adar

Đây là phương pháp dự đoán liên kết dựa trên mô hình topo. Hệ số này cho phép tính láng giềng chung nhưng cho trọng số cao hơn các láng giềng không chung. Công thức như sau:

$$score(x, y) = \sum_{z \epsilon P(x) \cap P(y)} \frac{1}{\log |\Gamma(y)|}$$

$$score(x,y) = \sum_{z \epsilon P(x) \cap P(y)} \frac{1}{\log |\Gamma(y)|}$$

$$score^*_{unweighted}(x,y) = |\{z: z \in \Gamma(y) \cap S_x^{\langle\delta\rangle}\}|$$

$$score^*_{weighted}(x,y) = \sum_{z \in \Gamma(y) \cap S_x^{\langle\delta\rangle}} score(x,z)$$

4.4.5 Preferential attachment

Đây là phương pháp dự đoán liên kết dựa trên mô hình topo. Ý tưởng cơ bản là xác suất mà một cung mới liên quan đến node x tỷ lệ thuận với số các láng giềng của node x là $\Gamma(y)$. Xác suất đồng tác giả của x và y tương quan với tích giữa số cộng tác viên của node x và cộng tác viên của node y. Công thức này được tính như sau:

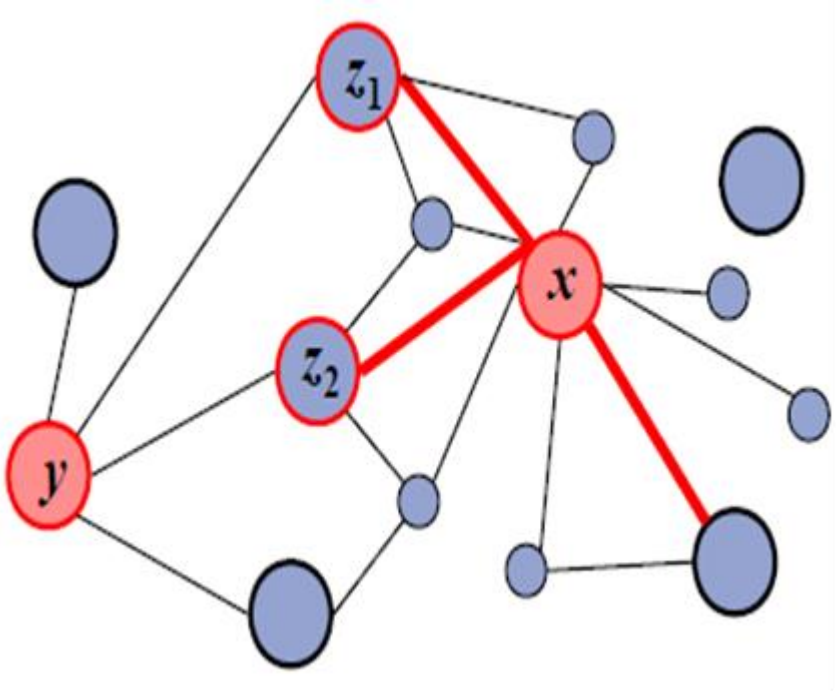
$$score(x,y) = |\Gamma(x)|.|\Gamma(y)|$$

4.4.6 SimRank

Đây là phương pháp dự đoán liên kết dựa trên mô hình topo. Sỉmank được định nghĩa đệ quy như sau:

Hai nút là tương tự dựa trên số láng giềng tương tự. Số đo này được xác định theo tham số $\gamma \epsilon [0,1]$ như sau:

$$score(x,y) = \gamma . \frac{\sum_{a \epsilon \Gamma(x)} \sum_{b \epsilon \Gamma(y)} score(a,b)}{|\Gamma(x)|.|\Gamma(y)|}$$



Hình 4.6. Mức độ tương đồng giữa các nút (cung tô đậm biểu diễn liên kết mạnh mẽ)

4.4.7 Dựa trên các thuộc tính node và cung

Đây là phương pháp dự đoán liên kết dựa trên sự tương tự của node. Node và các thuộc tính của cung đóng một vai trò quan trọng để dự đoán liên kết. Lưu ý rằng, trong một mạng xã hội, các liên kết được trực tiếp thúc đẩy bởi các tiện ích của cá nhân đại diện cho các node và tiện ích là một chức năng của các thuộc tính đỉnh và cạnh. Theo nghiên cứu của Hasan và cộng sự cho thấy, các thuộc tính node hoặc cung giúp tăng đáng kể độ chính xác của dự đoán liên kết. Ví dụ, dự đoán liên kết trong một mạng xã hội đồng tác giả, các thuộc tính như mức độ chồng chéo giữa các từ khóa nghiên cứu được sử dụng bởi một cặp của các tác giả là các thuộc tính xếp hạng cao nhất đối với một số bộ dữ liệu. Ở đây thuộc tính đỉnh là từ khóa với giả định một cặp tác giả "gần" nhau theo nghĩa mạng xã hội, nếu công việc nghiên cứu của họ phát triển xung quanh một tập các từ khóa phổ biến.

4.5 CÁC ỨNG DỤNG CỦA DỰ ĐOÁN LIÊN KẾT

- Dự đoán những thay đổi trong các mối quan hệ trước khi chúng xảy ra là rất có lợi cho một tổ chức;
- Xác định cấu trúc của một mạng lưới tội phạm và dự đoán các đường liên kết trong một mạng lưới tội phạm bằng cách sử dụng dữ liệu quá khứ;
- Cải thiện phân tích siêu văn bản để lấy thông tin và các công cự tìm kiếm bằng cách dự đoán các liên kết có thể xuất hiện giữa các thực thể;
- Dự đoán các trang web mà người sử dụng sẽ truy xuất tiếp theo để nâng cao hiệu quả và điều chỉnh trang web. Đây cũng là giải pháp kiểm soát truy cập các trang web có nội dung không lành mạnh;
- Dự đoán sự lan truyền của một thực thể thông qua mạng. Ví dụ như dịch bệnh, thông tin, chẳng hạn như xu hướng thời trang hoặc tin đồn trên mạng xã hội.

4.6 BÀI TẬP

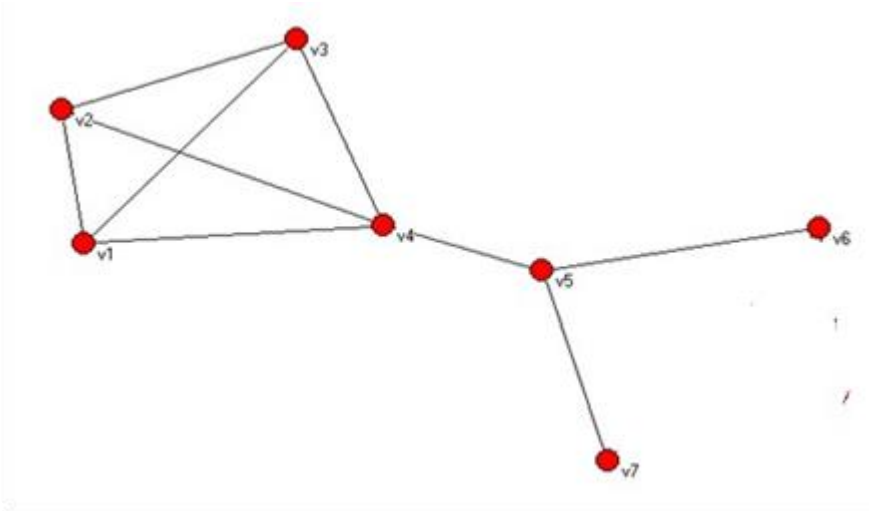
Câu 1: Cho biết các ứng dụng của dự đoán liên kết trên mạng xã hội.

Câu 2: Tại sao lại có nhiều mô hình dự đoán liên kết?

Câu 3: Cho mạng xã hội có đồ thị như sau:

Tính khoảng cách đồ thị, láng giềng chung, hệ số Jaccard, hệ số Adamic/Adar… của các cặp nút trong đồ thị trên.

Lập bảng so sánh các giá trị tính được và cho nhận xét về các cặp đỉnh đã có cung nối và các cặp đỉnh không có cung nối.


v3
v2
v1
v4
v5
v6
v7

BÀI 5. PHÂN TÍCH MẠNG XÃ HỘI

Học xong bài này, sinh viên sẽ đạt được những mục tiêu sau:

- *Hiểu biết tổng quan về phân tích Mạng xã hội.*

- *Biết được các cách phân tích mạng xã hội, các công cụ hỗ trợ phân tích mạng xã hội và một số chiến lược áp dụng trong khai thác dữ liệu.*

- *Thể hiện thái độ tích cực, kiên nhẫn, hợp tác và sẵn sàng học hỏi trong quá trình học tập và ứng dụng kiến thức về Mạng xã hội.*

5.1 TỔNG QUAN

Mạng xã hội được định nghĩa là một mạng lưới các tương tác, trong đó các nút (node) đại diện cho mọi người và các cạnh (edge) đại diện cho các tương tác giữa những người đó. Khái niệm về mạng xã hội và các chiến lược phân tích chúng đã xuất hiện từ nhiều thập kỷ trước. Thống kê, lý thuyết đồ thị và xã hội học là những nền tảng cơ bản cho sự phát triển của lĩnh vực mạng xã hội và được sử dụng trong nhiều lĩnh vực như kinh doanh, kinh tế và khoa học thông tin. Phân tích mạng xã hội tương tự như phân tích đồ thị do sự hiện diện của cấu trúc (topology) giống như một đồ thị của mạng xã hội. Phân tích đồ thị bao gồm một số chiến lược nhưng không hoàn toàn phù hợp để phân tích mạng xã hội do đặc điểm phức tạp của chúng. Một mạng xã hội có kích thước rất lớn, bao gồm hàng triệu cạnh và nút, trong đó mỗi nút thường sở hữu một số thuộc tính. Các chiến lược phân tích đồ thị cũ không thể xử lý được mạng xã hội đồ thị phức tạp và lớn này.

Mạng xã hội bao gồm nhiều loại khác nhau, chẳng hạn như mạng email, mạng cộng tác và mạng điện thoại. Tuy nhiên, các mạng xã hội trực tuyến gần đây như Twitter, Facebook và LinkedIn đã nhanh chóng đạt được sự phổ biến rộng rãi hơn chỉ trong một thời gian ngắn với số lượng người dùng khổng lồ. Một cuộc khảo sát cho thấy Facebook đã vượt qua mốc 500 triệu người dùng vào năm 2010. Mạng xã hội đóng vai trò như một nền tảng được công nhận rộng rãi với nguồn dữ liệu phong phú, hỗ trợ đáng kể

cho lĩnh vực marketing của các thương hiệu khác nhau, giúp ứng phó với những thay đổi trong marketing, nâng cao thương hiệu thông qua quảng bá và thu hút được lượng lớn khách hàng. Đặc biệt, vai trò của mạng xã hội rất quan trọng trong lĩnh vực ứng dụng chăm sóc sức khỏe. Do đó, lĩnh vực chăm sóc sức khỏe cần khám phá những phương thức mới để kiểm soát hoạt động của các nhà cung cấp dịch vụ và đo lường các thực tiễn tốt nhất nhằm mang lại sự hài lòng và cải thiện kết quả sức khỏe. Phân tích mạng xã hội (SNA) tập trung vào việc đánh giá mối quan hệ giữa các cá nhân, những người được kết nối bởi một hoặc nhiều nút phụ thuộc lẫn nhau, chẳng hạn như tình bạn, tình yêu, lòng tin, hợp tác hoặc giao tiếp. Phân tích mạng xã hội có thể cung cấp cái nhìn sâu sắc để đánh giá và hiểu các mạng lưới giao tiếp chuyên biệt và do đó, phát triển các can thiệp hiệu quả vào mạng lưới để nâng cao hiệu quả hoạt động của nhà cung cấp và cuối cùng là cải thiện các kết quả liên quan đến sức khỏe. Biểu đồ minh họa của SNA được hiển thị trong Hình 5.1.

Hình 5.1. Phân tích mạng xã hội (SNA)

Để minh họa, chúng ta hãy xem xét việc ứng dụng mạng xã hội trực tuyến trong việc phân tích các bệnh truyền nhiễm có nguồn gốc từ các tác nhân sinh học như cúm, thủy đậu, sởi và các vi rút lây truyền qua đường tình dục giữa người với người.

Các nghiên cứu gần đây đã quan sát thấy sự ra đời của một số mô hình SNA nhằm giải thích quá trình hình thành ý kiến trong một cộng đồng dân cư, dựa trên việc xem xét một số lý thuyết xã hội. Những mô hình này có một số đặc điểm chung với các mô hình về sự lây lan và dịch bệnh. Nhìn chung, mọi người được coi là các tác nhân với một trạng thái nhất định và được kết nối bởi một mạng lưới xã hội. Các liên kết xã hội được biểu diễn bằng đồ thị đầy đủ hoặc bằng các mạng phức tạp hợp lý hơn. Trạng thái của một nút thường được xác định bằng các biến, có thể là rời rạc hoặc liên tục, với khả năng lựa chọn một trong hai tùy chọn. Bản chất của các cá nhân thay đổi theo thời gian, phụ thuộc vào một số quy tắc cập nhật, chủ yếu dựa trên sự tương tác với những người xung quanh.

5.2 PHƯƠNG PHÁP PHÂN TÍCH MẠNG XÃ HỘI

5.2.1 Xử lý và phân tích dữ liệu

Một số thư viện hữu ích để làm việc với dữ liệu và bạn sẽ muốn sử dụng các thư viện và các kỹ thuật khác nhau tại các điểm khác nhau của dữ liệu. Ví dụ, khi làm việc với dữ liệu, nó thường hữu ích khi bắt đầu với Exploratory Data Analysis (EDA). Sau đó, bạn sẽ làm sạch, sắp xếp, thực hiện các biến đổi khác nhau cho quá trình tiền xử lý, v.v. Dưới đây là một số thư viện Python có sẵn và cách sử dụng chúng.

5.2.1.1 Pandas

pandas là một trong những thư viện quan trọng nhất sử dụng khi muốn làm bất cứ điều gì với dữ liệu trong Python. Có thể sử dụng pandas cho một số mục đích khác nhau khi làm việc với dữ liệu, chẳng hạn như sau:

- Đọc dữ liệu từ nhiều loại tập tin hoặc từ internet
- EDA
- Trích xuất, biến đổi, tải (ETL)
- Trực quan hóa dữ liệu đơn giản và nhanh chóng

Cài đặt

Nếu làm việc trên Jupyter hoặc Google Colab, pandas đã được cài đặt và có thể bắt đầu sử dụng nó bằng cách chạy câu lệnh này trong notebook:

```
import pandas as pd
```

5.2.1.2 NumPy

NumPy (Numeric Python) là gói cơ bản cho khoa học máy tính trong Python. NumPy có mục đích chung hơn và bao gồm các hàm toán học và những biến đổi khác nhau.

Cài đặt

Giống như pandas, nếu làm việc trong môi trường notebook, NumPy có thể đã được cài đặt. Tuy nhiên, nếu cần có thể làm theo các bước sau để cài đặt theo hướng dẫn: https://numpy.org/install/.

5.2.2 Trực quan hoá dữ liệu

Khả năng biểu diễn trực quan (visualization) của các mạng xã hội là rất quan trọng để hiểu dữ liệu mạng và chuyển tải các kết quả phân tích. Biểu diễn trực quan cho phép giải thích chất lượng của dữ liệu mạng. Với khả năng biểu diễn trực quan, các công cụ phân tích mạng được sử dụng để thay đổi cách bày trí, màu sắc, kích thước và các tính năng khác của mạng xã hội. Tất cả các công cụ ở trên đều chứa khả năng trực quan, NetMiner, igraph, Cytoscape, NetworkX có các tính năng cao cấp để xử lý đồ họa chất lượng cao.

Công nghệ biểu diễn trực quan và tương tác dữ liệu thường bao gồm khả năng phân tích mạng xã hội. Trong công nghệ này, các hình thức khác của dữ liệu trực quan được sử dụng để tương tác với đồ thị mạng xã hội. Những hình thức trực quan bao gồm một loạt các hình tượng biểu đồ, bảng biểu, dòng thời gian và bản đồ. Người dùng có khả năng hiển thị dữ liệu theo các bày trì khác nhau (graph layout) trong khi vẫn áp dụng được các chức năng để khám phá dữ liệu với tính năng tương tác cao. Ví dụ, có thể lọc đồ thị mạng xã hội phức tạp bằng cách sử dụng các chức năng tóm tắt biểu đồ hoặc dòng thời gian để cô lập các phần của đồ thị mạng xã hội cần quan tâm phân tích.

Tương tác dữ liệu trực quan cũng có thể bao gồm khả năng tích hợp dữ liệu và trình bày biểu đồ hoặc các mẫu báo cáo kết quả.

Một số thư viện Python có thể được sử dụng để trực quan hóa dữ liệu như Matplotlib, ngoài ra còn có các thư viện khác như Seaborn, Plotly.

5.2.2.1 Matplotlib

Matplotlib là một thư viện Python để trực quan hóa dữ liệu. Đó là nó. Nếu bạn có dữ liệu, Matplotlib có thể được sử dụng để trực quan dữ liệu này. Thư viện được tích hợp trực tiếp vào pandas, do đó nếu sử dụng pandas thì cũng có thể sử dụng Matplotlib.

Cài đặt

Giống như Pandas và NumPy, nếu làm việc trong môi trường notebook, Matplotlib đã được cài đặt sẵn. Tuy nhiên, nếu cần có thể làm theo các bước hướng dẫn cài đặt: https://matplotlib.org/stable/users/installing/.

Ngoài ra có thể tham khảo thêm hướng dẫn Matplotlib tại: https://matplotlib.org/stable/tutorials/.

5.2.2.2 Seaborn

Seaborn là phần mở rộng của Matplotlib và hỗ trợ trực quan hóa dữ liệu tốt hơn. Tuy nhiên, một số thứ không trực quan trong Matplotlib, thậm chí còn kém hơn trong Seaborn. Ngoài ra, các hình ảnh trực quan mang tính thẩm mỹ cao hơn nhiều so với Matplotlib. Ví dụ một số danh mục hình ảnh trực quan của Seaborn: https://seaborn.pydata.org/examples/.

Cài đặt

Link tham khảo cài đặt Seaborn tại: https://seaborn.pydata.org/installing.html.

5.2.2.3 Plotly

Plotly thể hiện biểu đồ phân tán tương tác và có hình ảnh trực quan hữu ích. Plotly có thể là một công cụ mạnh mẽ khi người dùng cần cách đơn giản để trực quan hóa dữ liệu.

Cài đặt

Để cài đặt Plotly, có thể tham khảo hướng dẫn tại link sau: https://plotly.com/python/gettingstarted/#installation.

5.3 MỘT SỐ CÔNG CỤ PHÂN TÍCH MẠNG XÃ HỘI

5.3.1 Các công cụ phần mềm phổ biến

Trong những năm gần đây, Phân tích mạng xã hội (SNA) đã nhận được sự chú ý ngày càng nhiều trong các lĩnh vực nghiên cứu khác nhau. Điều này là do tính linh hoạt trong vận hành được cung cấp bởi lý thuyết đồ thị, giúp giảm thiểu vô số hiện tượng xuống dạng phân tích cơ bản dựa trên các nút và liên kết. Chắc chắn, các mối quan hệ xã hội, giao thông vận tải, thương mại, chiến lược truyền thông và thậm chí cả não bộ đều có thể được mô phỏng thành một mạng lưới và phân tích. Điều này hỗ trợ cho việc nhìn nhận rõ ràng hơn các nghiên cứu liên quan đến phân tích mạng, mang lại lợi thế trong các trung tâm giáo dục, học viện và đặc biệt là các trường đại học, lĩnh vực y tế.

Một số công cụ đã được phát triển để nhiều người có thể dễ dàng sử dụng SNA. Thư viện SNA và các công cụ đồ họa được cung cấp cho các nhà vật lý, toán học, tin học máy tính, v.v. SNA, là một lĩnh vực nghiên cứu năng động, cũng có thể được sử dụng để khám phá các tương tác của con người và sự lan truyền ý kiến. Ngay cả đối với một số ứng dụng đặc biệt, cũng có sẵn nhiều công cụ và thư viện chuyên dụng. Tuy nhiên, việc lựa chọn công cụ phù hợp cho một nhiệm vụ cụ thể tốn nhiều thời gian, gây bất tiện cho người dùng.

Nhìn chung các phần mềm phân tích mạng xã hội thuộc một trong hai loại phần mềm dựa trên giao diện đồ họa (GUI), hoặc loại phần mềm được xây dựng cho kịch bản, ngôn ngữ lập trình. Loại phần mềm GUI dễ dàng tìm hiểu qua các công cụ mã nguồn mở.

Các phần mềm được sử dụng rộng rãi là NetMiner, UCINet, Pajek (freeware), GUESS, ORA và Cytoscape. Loại phần mềm giao diện hướng khách hàng doanh nghiệp bao gồm: Orgnet, cung cấp đào tạo về việc sử dụng các phần mềm, Keyhubs và KXEN. Một vài nền tảng phần mềm phân tích mạng xã hội khác, chẳng hạn như Idiro SNA Plus, đã

được đặc biệt phát triển cho các ngành công nghiệp đặc thù như viễn thông và game trực tuyến, với yêu cầu phân tích các khối dữ liệu lớn.

5.3.1.1 Gephi

Gephi (https://gephi.org) là một phần mềm mã nguồn mở chuyên dụng để trực quan hóa và khám phá các mạng lưới phức tạp, động và đa tầng. Nó có thể hoạt động trên các hệ điều hành Linux, macOS và Windows. Gephi là một công cụ mạnh mẽ dành cho những người cần nghiên cứu và khám phá các mạng lưới. Giống như Photoshop nhưng dành cho dữ liệu, người dùng tương tác với các tính năng để tùy chỉnh và kiểm soát bố cục, hình dạng và màu sắc nhằm khám phá các thuộc tính ẩn giấu trong mạng lưới.

5.3.1.2 Paijek

Pajek (http://pajek.imfm.si/doku.php) là một phần mềm chạy trên hệ điều hành Windows, được biết đến với khả năng xử lý các mạng lưới lớn. Pajek là phần mềm được sử dụng rộng rãi để vẽ mạng lưới, có các tính năng phân tích chuyên sâu và có thể được sử dụng để tính toán hầu hết các phép đo trung tâm, phát hiện các cộng đồng chính, mô hình khối, v.v. Igraph natomiast là một thư viện lập trình miễn phí để tạo và thao tác với các đồ thị. Nó bao gồm các thuật toán cho các vấn đề lý thuyết đồ thị điển hình như tìm cây ít giao cắt nhất và phân rã mạng, đồng thời thực hiện các tính toán như tìm kiếm cấu trúc cục bộ. Hiệu suất mạnh mẽ của igraph cho phép nó xử lý các đồ thị với số lượng cạnh và nút rất lớn.

5.3.1.3 UCINET

Là một gói phân tích mạng mở rộng, có thể chạy trong Windows. Tuy nhiên, các máy khác như Linux hay Mac cũng có thể sử dụng nó trong VMWare hoặc Parallel hoặc BootCamp. Nó bao gồm một công cụ trực quan có tên NetDraw hiển thị đầu ra hình ảnh của mạng.

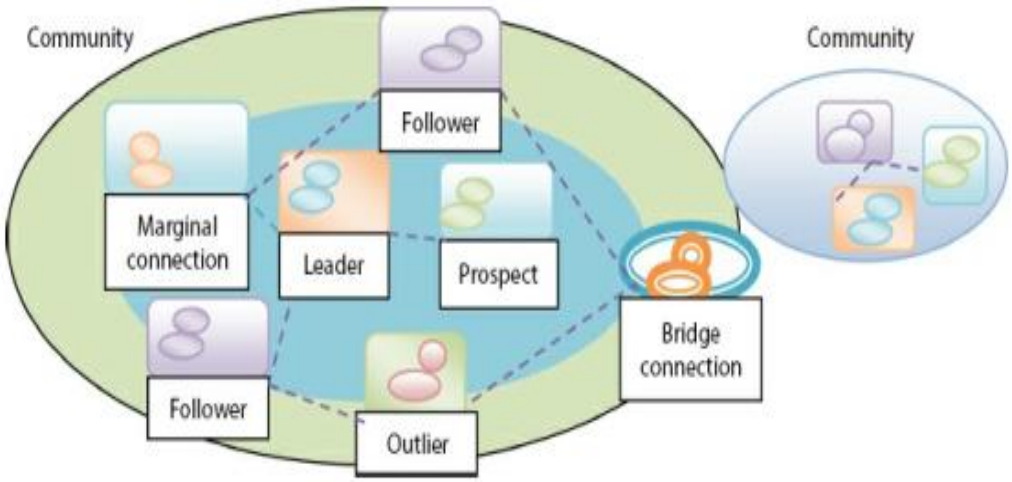
5.3.2 Sử dụng ngôn ngữ lập trình

Ngoài ra, một số bộ thư viện khác rất hữu ích cho việc phân tích và trực quan hóa các loại mạng khác nhau. Trong phần này, chúng ta sẽ thảo luận về một số công cụ và thư viện sẵn có.

5.3.2.1 NetworkX

NetworKit (NetworkX[7]) cung cấp các thuật toán đồ thị và hiệu quả trong việc phân tích các khả năng của mạng. NetworkX cung cấp các hàm để tạo, thao tác và phân tích cấu trúc và thuộc tính của các mạng phức tạp. Với thư viện này, người dùng có thể tải và lưu trữ mạng trong các định dạng dữ liệu phổ biến, tạo ra nhiều loại mạng ngẫu nhiên và theo mẫu, phân tích cấu trúc mạng, xây dựng mô hình mạng, vẽ biểu diễn mạng, v.v. NetworkX có nhiều tính năng nổi bật như hỗ trợ đa đồ thị (MultiGraph), hỗ trợ lưu trữ dữ liệu dạng ký hiệu trên các cạnh (edge attributes), và đồ thị có hướng (DiGraph). Các nút (Node) có thể chứa "bất cứ thứ gì", chẳng hạn như hình ảnh và văn bản. Các cạnh (Edge) có thể chứa thông tin tùy chọn, chẳng hạn như trọng số, thuộc tính thời gian, v.v. NetworkX cung cấp các thuật toán tính toán tiêu chuẩn cho lý thuyết đồ thị, xây dựng mạng, các phép đo độ trung tâm, v.v.

[7] http://networkx.github.io



Hình 5.2. Phân tích mạng xã hội bằng Python

Có thể cài đặt NetworkX theo hướng dẫn sau:

https://networkx.org/documentation/stable/install.html.

scikit-network

scikit-network là gói Python dùng cho việc phân tích các biểu đồ lớn. Trực quan hóa NetworkX chậm và mất nhiều thời gian để kết xuất, ngay cả trên một mạng nhỏ và giao diện của chúng rất đơn điệu và cơ bản. Mặt khác, scikit-network trực quan hình ảnh rất nhanh vì chúng được hiển thị dưới dạng SVG. Trực quan có tốc độ hợp lý, ngay cả khi dữ liệu có hàng trăm hoặc hàng nghìn nút.

Cài đặt

Có thể cài đặt scikit-network theo hướng dẫn sau:

https://scikit-network.readthedocs.io/en/latest/first_steps.html.

5.3.2.2 Igragh

Igraph là một công cụ phân tích mạng hiệu quả, tính di động và dễ sử dụng. Igraph có thể được lập trình bằng R, Python, Mathematica và C/C++.

Cụ thể, giao diện Python của igraph, một thư viện C mã nguồn mở và nhanh để thao tác và phân tích biểu đồ (còn gọi là mạng). Nó có thể được sử dụng để:

- Tạo, thao tác và phân tích mạng.
- Chuyển đổi đồ thị từ/sang NetworkX, Graph-tool và định dạng nhiều file.
- Vẽ đồ thị mạng bằng cách sử dụng Matplotlib và Plotly.

Cài đặt

Cài đặt bằng cách sử dụng pip:

```
pip install igraph
```

Cài đặt bằng cách sử dụng conda:

```
conda install -c conda-forge python-igraph
```

5.4 PHÂN TÍCH MẠNG XÃ HỘI

Ngày nay, mạng đại diện cho một khía cạnh quan trọng trong cuộc sống của con người. Lượng lớn các vấn đề trong thế giới thực bao gồm mối quan hệ giữa các bản ghi dữ liệu. Mạng có tác động đáng kể đến cuộc sống hàng ngày của chúng ta, từ việc cung cấp thông tin có giá trị để gây ảnh hưởng tới cuộc bầu cử. Mạng cho phép người dùng mạng cộng tác và chia sẻ tài nguyên. Nó cho phép người dùng tập trung vào việc bảo vệ và xử lý những thông tin cần thiết của doanh nghiệp. Điều này cho phép các máy tính khác nhau của mạng nhận dữ liệu cần thiết từ điểm trung tâm. Các công ty khác nhau có thể liên kết đến các máy tính của chính họ và cần tổ chức các máy tính vào mạng, do đó bất kỳ máy tính nào trên mạng đều có thể tương tác với bất kỳ máy tính khác. Mạng máy tính cho phép nhân viên hợp tác chia sẻ ý tưởng một cách hiệu quả và dễ dàng hơn, làm tăng năng suất làm việc của nhân viên và tạo ra nhiều lợi nhuận hơn cho công ty. Quan trọng hơn, mạng máy tính cải thiện cách các công ty tương tác với phần còn lại của thế giới. Mạng Logistics, World Wide Web, Internet và mạng xã hội đều là những ví dụ về ứng dụng mạng. Ngoài ra, các loại đồ thị này thường được sử dụng trong ngân hàng - ví dụ, để minh họa các giao dịch tài chính và sự kết nối của

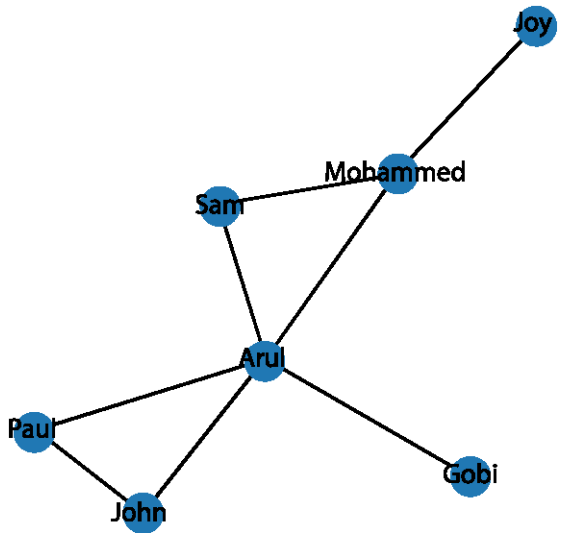
các đối tác trung tâm. Một số các tính năng của đồ thị rất hữu ích trong việc hiểu dữ liệu chứa trong đó. Phân tích mạng giúp hiểu rõ những vấn đề phức tạp.

Lý thuyết mạng là khái niệm về mạng như một sự mô tả về tính đối xứng hoặc mối quan hệ bất đối xứng giữa các đối tượng riêng biệt. Trong nghiên cứu khoa học máy tính và mạng, lý thuyết mạng là một nhánh của lý thuyết đồ thị: một mạng có thể được biểu diễn dưới dạng đồ thị có các thuộc tính trên các nút và/hoặc các cạnh của nó. Điều quan trọng là mục đích của việc phân tích bất kỳ loại mạng nào là làm việc với độ phức tạp của mạng để truy xuất các dữ kiện sẽ không đạt được bằng cách nghiên cứu các chức năng một cách riêng biệt. Ở đây, hai thành phần cơ bản của mạng là các nút và các cạnh.

Nút là một vị trí trên mạng được liên kết với hai hoặc nhiều nút khác các thành phần. Các nút đại diện cho những thứ sẽ được kiểm tra, trong khi các cạnh thể hiện mối quan hệ của chúng. Các nút có thể lưu trữ cả các đặc điểm tự tham chiếu (chẳng hạn như trọng lượng, kích thước, vị trí và bất kỳ thuộc tính nào khác) và thông tin tham chiếu mạng (chẳng hạn như số độ, mức độ trung tâm, v.v.). Các cạnh thể hiện sự kết nối giữa các nút và có thể chứa thêm các đặc điểm, chẳng hạn như trọng lượng, cho biết chiều dài và hướng của kết nối.

Phân tích mạng giống như một thiết kế nghiên cứu được điều chỉnh phù hợp để xác định, nghiên cứu và phân tích các đặc điểm cấu trúc cũng như quan hệ khác nhau. Phân tích mạng thực sự là một kỹ thuật tôpô nhấn mạnh các mẫu của các mối quan hệ actor-actor. Phân tích mạng xã hội (SNA) là một phương pháp nghiên cứu các hệ thống xã hội thông qua việc sử dụng mạng, cũng như lý thuyết đồ thị. Khả năng đánh giá các mạng lưới này và đưa ra quyết định đúng đắn là quan trọng đối với bất kỳ nhà phân tích dữ liệu nào. Nếu các nhà phân tích đang phân tích mối liên hệ xã hội giữa người dùng Facebook, ví dụ các nút đại diện cho những người hướng đến và các cạnh biểu thị sự kết nối giữa những người dùng, chẳng hạn như tình bạn hoặc các thành viên trong nhóm. Nó cho phép mọi người hiểu biết toàn diện về cấu trúc của một liên kết trong mạng xã hội, cấu trúc hoặc quá trình thay đổi trong các sự kiện tự nhiên và thậm chí cả hệ thống sinh học của sinh vật.

Hình 5.3, các vòng tròn biểu thị các nút và các đường nối các vòng tròn là các cạnh. Nếu biểu đồ này đại diện cho một mạng xã hội thì vòng tròn sẽ đại diện cho con người, và một cạnh giữa hai đỉnh có thể biểu thị rằng hai cá nhân đó là bạn bè.



Hình 5.3. Các nút và cạnh trong một mạng xã hội

Trong đó:

Nút dùng để chỉ bất kỳ loại tác nhân hoặc phần tử nào chúng tôi đang cố gắng liên lạc tới. Trong ví dụ này, nó là các cá nhân - được hướng đến. Ở đây, Arul, Gobi, John, Joy, Mohammed và Sam là các nút.

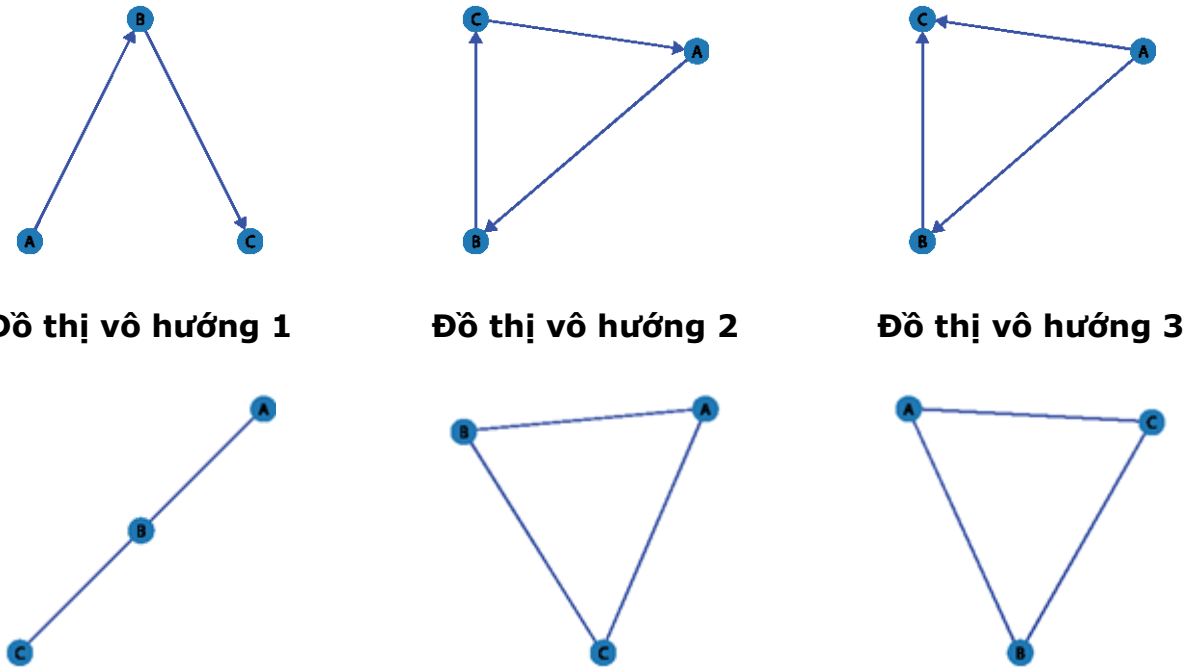
Cạnh là liên kết giữa hai nút. Đó là sự tương tác vật lý giữa các cá nhân được xét trong phân tích. Liên kết chính là đường dẫn thực tế mà những người dùng mạng xã hội được kết nối tới. Ở đây, liên kết kết nối Arul và Sam là cạnh của mạng.

5.4.1 Mạng vô hướng và có hướng

Mạng vô hướng: Các cạnh không có bất kỳ hướng nào.

Mạng có hướng: Các cạnh có hướng.

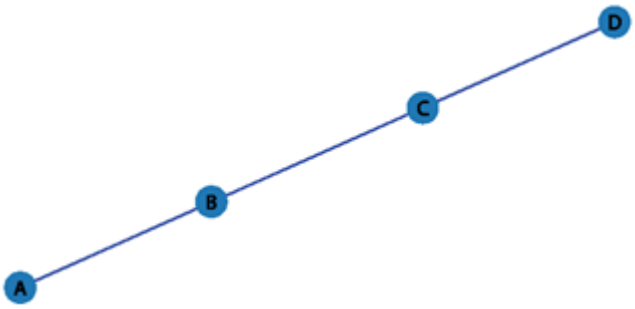
Đồ thị có hướng 1 **Đồ thị có hướng 2** **Đồ thị có hướng 3**



Hình 5.4. So sánh đồ thị có hướng và vô hướng

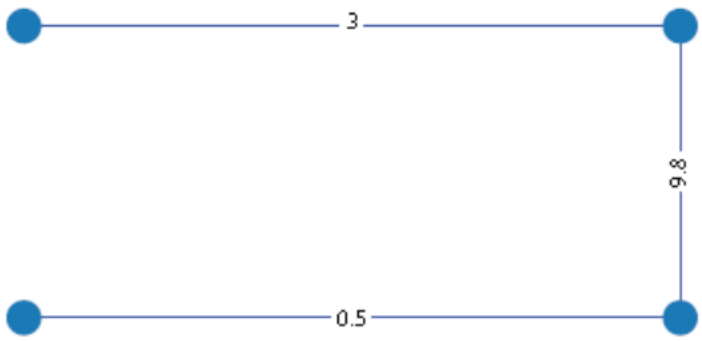
5.4.2 Mạng có trọng số và không trọng số

Mạng không có trọng số: Các cạnh trong biểu đồ không chứa trọng số.



Hình 5.5. Đồ thị không có trọng số

Mạng có trọng số: Cạnh trong biểu đồ chứa giá trị (số), là được gọi là trọng lượng.



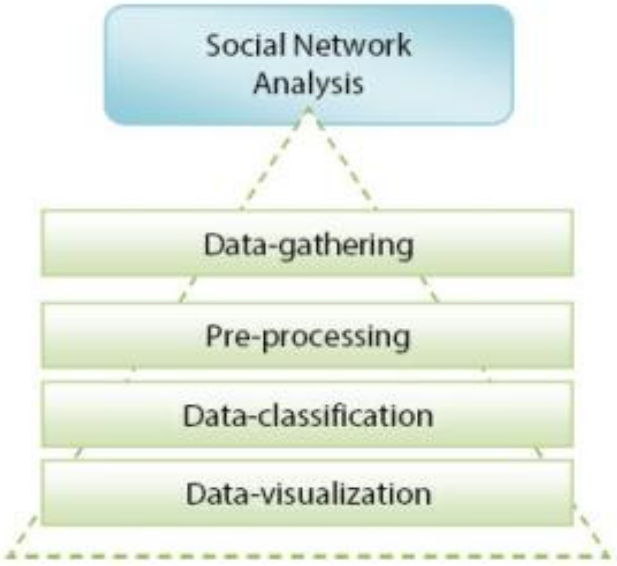
Hình 5.6. Đồ thị có trọng số

5.4.3 Thiết lập một mạng xã hội

Mạng xã hội có thể được xây dựng từ một số tập dữ liệu miễn là vì các kết nối nút-nút có thể được chỉ định. Để chuyển đổi kết quả thành dạng bảng, người dùng có thể đọc dữ liệu từ tệp dữ liệu (ví dụ: Excel) vào một khung dữ liệu Pandas. Sau đó, sử dụng danh sách cạnh trong khung dữ liệu Pandas, các nhà phát triển có thể sử dụng NetworkX để xây dựng một đồ thị có hướng. Cuối cùng, các kỹ thuật trực quan hóa có thể được sử dụng để khám phá.

Phần này mô tả việc ứng dụng SNA sử dụng các thư viện Python vào các tình huống thực tế. Ví dụ, chúng ta có thể xem xét phân tích tình cảm của người dùng mạng xã hội trong tình huống đại dịch COVID hoặc dự đoán và truy vết các bệnh truyền nhiễm. Với sự phát triển mạnh mẽ của công nghệ, dữ liệu mong muốn có thể đạt được chỉ bằng cách nhập từ khóa cần thiết vào công cụ tìm kiếm. Số lượng các trang mạng xã hội có khả năng cung cấp nhiều dữ liệu thông tin hơn giúp ích cho việc đánh giá MHX. Dữ liệu cần thiết cho phân tích được thu thập thông qua việc áp dụng khái niệm khai thác dữ liệu trên các trang mạng xã hội. Những người tạo ra nền tảng truyền thông xã hội, chẳng hạn như Facebook, Reddit, Twitter, cung cấp cho người dùng Giao diện Lập trình Ứng dụng (API) hỗ trợ thu thập dữ liệu mong muốn từ trang web. Thu thập dữ liệu, tiền xử lý và phân loại là những giai đoạn quan trọng trong SNA, được minh họa trong Hình 5.7. Thu thập dữ liệu là bước đầu tiên để thực hiện bất kỳ công việc khai thác dữ liệu nào. Quá trình thu thập dữ liệu là một nhiệm vụ linh hoạt và phụ thuộc vào chủ đề cụ thể mà người dùng quan tâm. Ban đầu, dữ liệu thô được tích lũy từ mạng xã hội bằng cách yêu cầu dữ liệu với từ khóa chính xác.

Sau khi thu thập dữ liệu từ mạng xã hội, dữ liệu được tiền xử lý để thực hiện các quy trình, chẳng hạn như dự đoán hoặc phân tích. Dựa trên ứng dụng, dữ liệu thu thập được xử lý với các giai đoạn tiền xử lý và sau đó có thể được phân loại và trực quan hóa. Ngày nay, trong Python, các phân loại được triển khai cho một ứng dụng chủ yếu là bất kỳ loại phân loại học máy nào hoạt động theo phương pháp học máy có giám sát. Bộ phân loại yêu cầu đào tạo thích hợp bằng cách sử dụng dữ liệu đào tạo được dán nhãn, nếu không thì hiệu suất của bộ phân loại không thể được phân tích. Một trong những bộ phân loại thống kê được sử dụng phổ biến là bộ phân loại Naive Bayes, thường được sử dụng để phân loại cảm tính của mọi người trong điều kiện đại dịch COVID. Loại phân loại này thường tận dụng hiệu quả dữ liệu có sẵn công khai (từ dữ liệu truyền thông cộng đồng) để thực hiện các bài toán dự đoán, phân tích hoặc phân loại.



Hình 5.7. Sơ đồ của Mạng xã hội

Các số liệu của mạng xã hội rất quan trọng trong việc đánh giá hoạt động của phương tiện truyền thông xã hội trên một tổ chức cụ thể. Việc lựa chọn độ đo có ý nghĩa quan trọng giữa hàng trăm số liệu là một thách thức quan trọng. Giá trị thương hiệu của một tổ chức cần được theo dõi liên tục cho những người chăm sóc sử dụng các số

liệu. Một số độ đo thường được sử dụng trong các tổ chức chăm sóc sức khỏe được giải thích dưới đây:

a) *Khối lượng*: Khối lượng hoạt động như một thước đo đơn giản giúp phân tích một số lượng thông điệp từ một thương hiệu cụ thể của một tổ chức và số người đã đăng tin nhắn lên vòng kết nối đã biết của họ.

b) *Danh tiếng*: Sự lan truyền hoặc phổ biến của một hoạt động trên mạng xã hội là được gọi là phạm vi tiếp cận hoặc danh tiếng. Tuy nhiên, để thước đo này có hiệu quả, một số số liệu khác phải được kết hợp với nó. Nói chung, phạm vi tiếp cận số liệu

được sử dụng làm mẫu số trong các phương trình đo liên quan đến truyền thông xã hội.

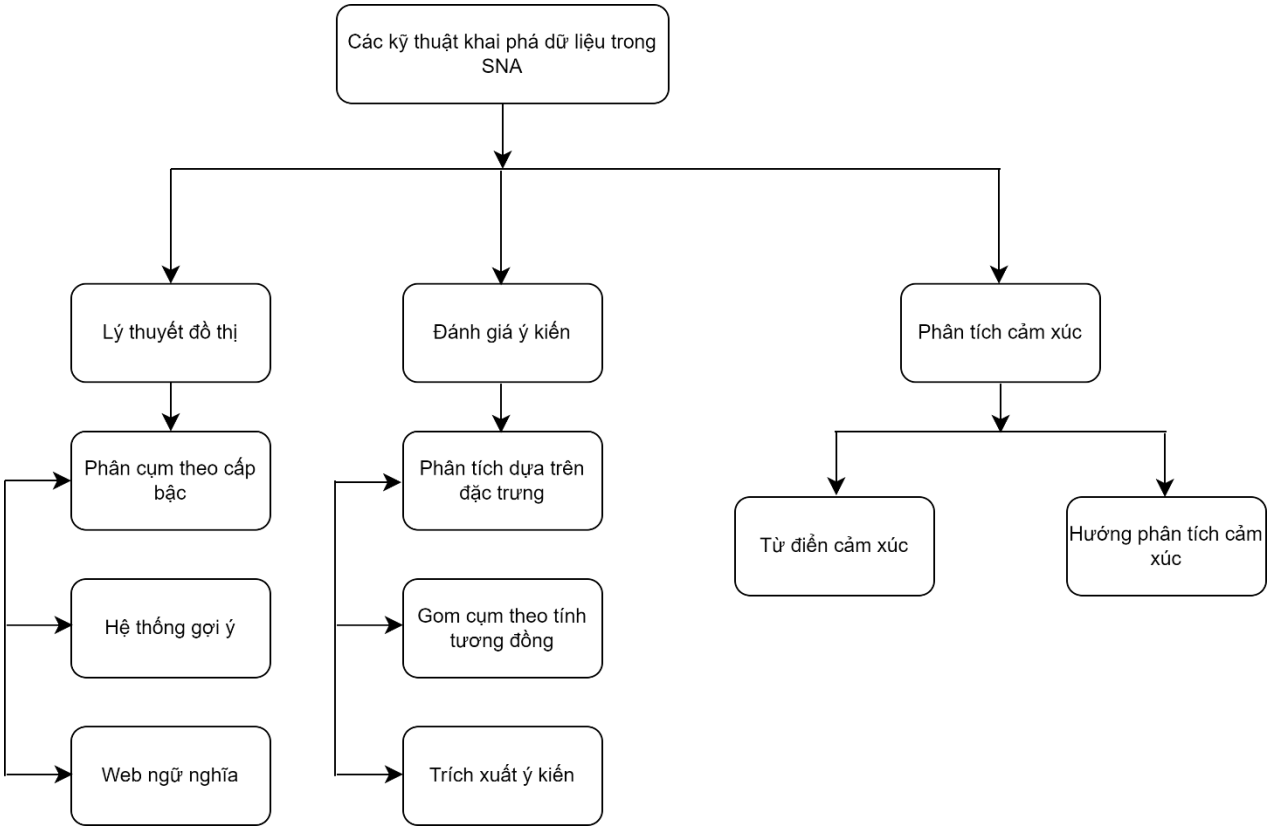
c) *Người dùng chuyên dụng*: Người dùng chuyên dụng cũng là thước đo tốt nhất liên quan đến tìm những người tham gia vào các hoạt động nêu chi tiết cuộc trò chuyện về các sản phẩm hoặc dịch vụ để giới thiệu nó.

d) *Sự thống trị*: Thước đo sự thống trị giúp lựa chọn một người có năng lực cao ảnh hưởng đến công chúng được tiếp cận để phát sóng một sản phẩm cụ thể.

e) *Đánh giá số liệu*: Để vượt trội, hiệu suất phải cao hơn đối với một tổ chức cụ thể được so sánh với những người tham gia. Demographics Pro là một công cụ phân tích hỗ trợ các đại lý có được thông tin cần thiết đặc điểm kỹ thuật truyền thông xã hội với các dự án xã hội trên các trang web phổ biến, như Facebook, Instagram, v.v. Điều này giúp các nhà tiếp thị tập trung vào nền tảng để đạt được lợi ích tối đa.

5.5 MỘT SỐ CHIẾN LƯỢC THÔNG THƯỜNG TRONG KỸ THUẬT KHAI THÁC DỮ LIỆU

Một số mô tả ngắn gọn về hệ thống hiện có được sử dụng để khai thác ý kiến được trình bày chi tiết trong phần này. Sơ đồ thể hiện phương pháp hiện nay được minh họa trong Hình 5.8.

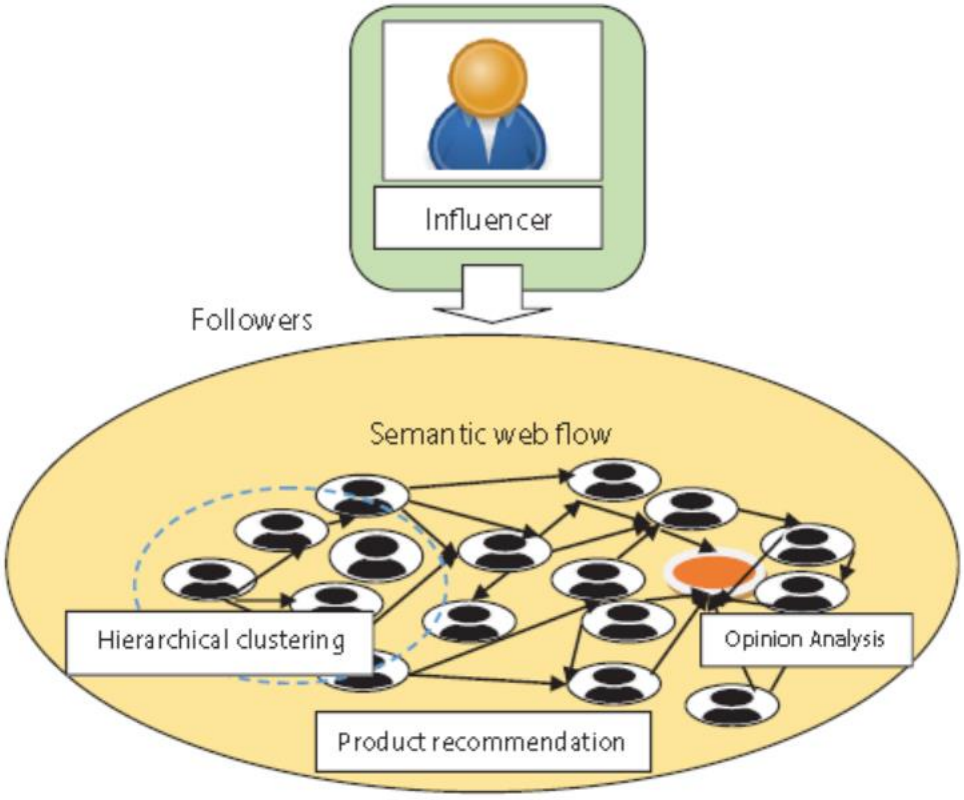


Hình 5.8. Sơ đồ biểu diễn các phương pháp hiện nay

5.5.1 Lý thuyết đồ thị

Lý thuyết đồ thị là một trong những chiến lược chính được sử dụng trong điều tra cộng đồng không chính thức về các số liệu thống kê trong quá khứ của ý tưởng tổ chức giữa các cá nhân. Phương pháp này được áp dụng trong SNA để quyết định những điểm nổi bật quan trọng của tổ chức, chẳng hạn như các kết nối và các nút, người có ảnh hưởng và những người theo dõi. Những người có ảnh hưởng trong cộng đồng không chính thức được coi là khách hàng, người có ảnh hưởng đến hoạt động hàng ngày hoặc đánh giá của các khách hàng khác nhau thông qua người tiêu dùng hoặc ảnh hưởng đến các lựa chọn được lựa chọn bởi các khách hàng khác nhau trên tổ chức. Lý thuyết đồ thị đã tạo ra các tập dữ liệu có phạm vi lớn (như thông tin cộng đồng không chính thức) vì khả năng truyền tải của người ảnh hưởng bỏ qua những ý tưởng cơ bản về mô tả đặc điểm hình ảnh để thực hiện một cách đơn giản trên lưới thông tin. Độ đo trung tâm được sử dụng để điều tra sự mô tả của lực và tác động cấu trúc nên các bó và mối quan hệ qua lại giữa các tổ chức giữa các cá nhân. Các nhà nghiên cứu đã sử dụng cách

đo lường trung tâm được xác định để giải quyết nghiên cứu về khung tổ chức và để phân loại tính khả dụng của nút. Sơ đồ biểu diễn của lý thuyết đồ thị được minh họa trên Hình 5.9.



Hình 5.9. Sơ đồ biểu diễn của lý thuyết đồ thị

5.5.1.1 Phương pháp phân cụm theo cấp bậc

Một cộng đồng được định nghĩa là tập hợp nhóm đông người bên trong một tổ chức lớn hơn. Sự phát triển khu vực địa phương được biết đến là một trong những chất lượng quan trọng về đích đến giữa các tổ chức cá nhân. Các khách hàng với mạng lưới cấu trúc lợi ích tương tự trong tổ chức giữa các cá nhân minh họa cho thiết kế bộ phận vững chắc. Mạng lưới các tổ chức không chính thức, liên quan đến một số mạng khác trong đời thực, rất phức tạp trong mô tả và khá lớn để nhận ra. Việc áp dụng các công cụ phù hợp trong việc nhận biết và hiểu rõ về hiệu suất của mạng lưới tổ chức là rất quan trọng vì điều này có thể được sử dụng để minh họa tính hiệu quả của không gian họ có tại chỗ. Nhiều nhà nghiên cứu đã áp dụng các chiến lược phân cụm khác nhau để xác định

mạng lưới về tổ chức giữa các cá nhân, với việc phân nhóm lũy tiến thường được sử dụng. Phương pháp này là sự tích hợp của nhiều quy trình được sử dụng để gom các trung tâm trong tổ chức thành nhóm, nhằm khám phá ra thế mạnh của các hội thánh độc lập. Từ đó, phương pháp này được sử dụng để phân chia tổ chức thành các mạng lưới. Việc nhóm các đỉnh (vertex) được thực hiện bằng các chiến lược cụm phân cấp khác nhau. Các đỉnh trên biểu đồ có thể được giải quyết bằng cách thêm chúng vào một không gian vector để ước tính khoảng cách từng đôi giữa các đỉnh. Trong một tổ chức địa phương phi chính thức, các thành viên thường xuyên đề xuất các sản phẩm và dịch vụ cho nhau dựa trên chuyên môn của họ về các lĩnh vực cụ thể.

5.5.1.2 Hệ thống gợi ý trong cộng đồng mạng xã hội

Tùy thuộc vào mức độ tương đồng giữa các trung tâm trong các cộng đồng không chính thức, quy trình lọc cộng tác (CF), là một trong ba lớp của khung đề xuất (RS), có thể được sử dụng để dự đoán mối quan hệ giữa các thành viên. Nhược điểm cơ bản của CF là sự thiếu hụt thông tin, phương pháp dựa trên nội dung (một chiến lược RS khác) lại nghiên cứu cấu trúc của thông tin để tạo ra các đề xuất. Tuy nhiên, các phương pháp kết hợp thường đề xuất các mục bằng cách kết hợp các đề xuất CF và dựa trên nội dung.

5.5.1.3 Web ngữ nghĩa cho mạng xã hội

Web ngữ nghĩa (SW) được sử dụng để chia sẻ và tái sử dụng thông tin có thể có trên các ứng dụng và các nút khác nhau của mạng. Đánh giá sâu về SW cải thiện chất lượng thông tin của cộng đồng SW và thúc đẩy sự tích hợp của SW. Friend of a Friend (FOAF) giúp phân tích cách các cộng đồng quy mô địa phương và toàn cầu hình thành và phát triển thành các mạng xã hội quy mô lớn trên SW. Tương tự, khung ứng dụng của các khung đánh giá mạng xã hội dựa trên SW cung cấp trung tâm truyền thông trí tuệ cho việc đánh giá mạng xã hội liên quan đến cấu trúc chung của SW để đạt được hiệu quả thu thập dịch vụ internet. Bên cạnh đó, hệ thống Web-Harvest được nâng cấp mã nguồn mở để thu thập thông tin về mạng xã hội trực tuyến nhằm xem xét các mô hình cải thiện quyền riêng tư và hợp nhất logic trực tuyến. Web ngữ nghĩa là một lĩnh

vực tương đối mới trong SNA và việc nghiên cứu trong lĩnh vực này vẫn đang được phát triển.

5.5.2 Đánh giá ý kiến trên mạng xã hội

Lượng thông tin khổng lồ được tạo ra mỗi thời điểm trên các trang mạng xã hội thông thường bị choáng ngợp với các đánh giá của người dùng trên các chủ đề khác nhau, từ các vấn đề cá nhân đến các vấn đề toàn cầu. Các kỹ thuật khai thác dữ liệu dựa trên phân tích ý kiến được trình bày chi tiết trong phần này.

5.5.2.1 Phân tích dựa trên đặc trưng

Khai thác ý kiến theo đặc điểm (hay theo khía cạnh) - aspect-based opinion mining - tập trung vào việc phân tích những khía cạnh được khách hàng nhắc đến nhiều nhất trong đánh giá. Phương pháp này không chỉ đơn thuần trích xuất các đặc điểm từ đánh giá mà còn phải phân tích độ tin cậy của các bình luận được khách hàng chia sẻ. Do đó, phân tích theo khía cạnh đòi hỏi phải phân loại đánh giá thành tích cực và tiêu cực.

5.5.2.2 Gom cụm theo tính tương đồng

Đánh giá được chia sẻ bởi khách hàng trên mạng xã hội luôn phản ánh ý kiến cá nhân của họ, do đó không thể đảm bảo tính tổng quát. Các đánh giá này sẽ ảnh hưởng đáng kể đến khả năng ra quyết định của người xem. Do đó, kỹ thuật gom cụm theo tính tương đồng là cần thiết cho các kỹ thuật khai thác dữ liệu hiệu quả để đảm bảo tính chính xác của các đánh giá sản phẩm. Những người dùng chia sẻ cùng đánh giá hoặc ý kiến sẽ được xếp vào cùng một nhóm để tạo thành cụm. Kỹ thuật này được gọi là tính tương đồng (homophily) trong tổ chức xã hội.

5.5.2.3 Trích xuất ý kiến

Trích xuất ý kiến (Opinion extraction) là một quá trình quan trọng trong khai thác dữ liệu nhằm phân loại các đánh giá thực sự về sản phẩm, con người hoặc sự vật. Quá trình trích xuất ý kiến giúp lọc ra những thông tin có liên quan nhất do khách hàng chia sẻ.

5.5.3 Phân tích cảm xúc

Phân tích cảm xúc được tích hợp vào quy trình khai thác dữ liệu nhằm hỗ trợ khả năng ra quyết định của người xem.

5.5.3.1 Hướng phân tích cảm xúc

Hướng phân tích cảm xúc (Sentiment orientation): Đối với các sản phẩm được sử dụng rộng rãi, hàng triệu đánh giá của khách hàng có thể ảnh hưởng đến quá trình ra quyết định mua hàng. Trong khi đó, người bán hàng cũng tận dụng phân tích cảm xúc để quảng bá sản phẩm của họ. Để phân tích cảm xúc trong các đánh giá, các kỹ thuật phân loại theo thứ bậc kết hợp với các kỹ thuật học máy đã được sử dụng nhiều gần đây.

5.5.3.2 Từ điển cảm xúc

Từ điển cảm xúc (Sentiment lexicon): Từ điển cảm xúc được định nghĩa là danh sách các từ thể hiện cảm xúc. Phân tích cảm xúc dựa trên từ điển giúp cải thiện hệ thống hỗ trợ quyết định vì nó loại bỏ các đánh giá trung lập, đồng thời tập trung vào các nhận xét tích cực và tiêu cực.

BÀI 6. MỘT SỐ CHƯƠNG TRÌNH PHÂN TÍCH MẠNG XÃ HỘI

Học xong bài này, sinh viên sẽ đạt được những mục tiêu sau:

- *Hiểu biết các chức năng phân tích Mạng xã hội.*
- *Áp dụng phân tích mạng xã hội Facebook.*
- *Thể hiện thái độ tích cực, kiên nhẫn, hợp tác và sẵn sàng học hỏi trong quá trình học tập và ứng dụng kiến thức về Mạng xã hội.*

6.1 CHỨC NĂNG PHÂN TÍCH MẠNG XÃ HỘI

6.1.1 Chương trình tính số đo bậc của các đỉnh

Bậc của một nút thể hiện số lượng kết nối của nút đó. NetworkX cung cấp hàm bậc để xác định bậc của một nút trong mạng xã hội. Bậc của một nút được định nghĩa là số lượng cạnh kết nối với nút đó. Đây là một tham số cơ bản có tác động đến các thuộc tính khác, chẳng hạn như tính trung tâm của một nút. Hàm phân phối của tất cả các nút trong mạng giúp xác định xem mạng có phải là mạng không quy mô hay không. Trong mạng xã hội có hướng, các nút có hai bậc: bậc ra (Out-degree) cho các cạnh đi ra khỏi nút và bậc vào (In-degree) cho các cạnh đi vào nút.

Bậc của một nút được minh họa trong lệnh sau:

nx.degree(G_symmetric, ‘User B’)

Ví dụ:

```
nx.degree(G_graph, ‘User B’)
4
```

Đoạn mã được đề cập ở trên sẽ trả về giá trị 4 vì "User B" chỉ hoạt động với bốn người dùng khác trên mạng.

6.1.2 Chương trình tìm chiều dài đường đi ngắn nhất giữa 2 đỉnh

Để thể hiện luồng thông tin, đường dẫn ngắn nhất hoặc khoảng cách ngắn nhất trong mạng xã hội giữa hai nút bất kỳ, người ta sử dụng các đường dẫn ngắn nhất. Hàm khoảng cách được sử dụng để tính toán khoảng trống giữa bất kỳ nút nào trong mạng. Đường dẫn ngắn nhất là đường dẫn đi qua ít nút nhất. Nó tìm đường dẫn trở kháng tích lũy ngắn nhất giữa hai nút. Đường dẫn có thể chỉ nối hai điểm, vị trí hoặc có thể bao gồm các vị trí bổ sung trong mạng.

Phân cụm và phân cụm trung bình của một nút cho đồ thị đối xứng có thể được thực hiện như sau:

```
nx.average_clustering(G_symmetric)
```

0.8666666666666

```
nx.clustering(G_symmetric,’UserA’)
```

0.6666666666666

Dựa trên chương trình và biểu diễn được cung cấp trong mục 6.1.1 và mục 6.1.2, chúng ta có thể tính toán được đường đi ngắn nhất. Hàm nx.shortest_path(Graph_Type, 'Node1_Info: 'Node2_Info') được sử dụng để tìm đường đi ngắn nhất giữa hai nút bất kỳ và hàm nx.shortest_path_length (Graph, Node1, Node2) được dùng để tính toán độ dài của đường đi giữa hai nút bất kỳ.

```
nx.shortest_path(G_symmetric,'User C','User E')
```

['User C', 'User A', 'User E']

```
nx.shortest_path_length(G_symmetric,'User C','User E')
```

2

```
nx.shortest_path(G_symmetric,'User 0','User E')
```

['User D', 'User A', User E']

```
nx.shortest_path_length(G_symmetric,'User D','User E')
```

2

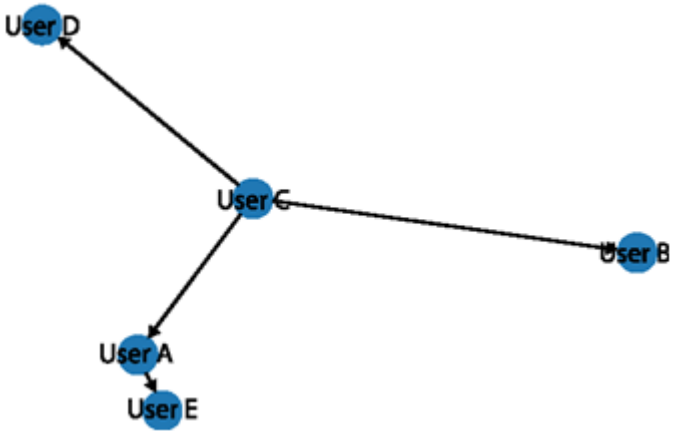
Dựa vào kết quả trong đoạn lệnh trên, đường đi ngắn nhất giữa "user C" và "user E" là ["user C", "user A", "user E"]. Độ dài đường đi ngắn nhất giữa hai nút này được tính toán là 2. Tương tự, độ dài đường đi ngắn nhất giữa "user D" và "user E" cũng là 2.

Để tính toán khoảng cách giữa một nút và tất cả các nút khác trong mạng, thuật toán tìm kiếm theo chiều rộng (BFS) được sử dụng, bắt đầu từ một nút. Thư viện NetworkX cung cấp hàm bfs_tree() cho mục đích này. Trong Hình 6.1, việc thực hiện T = nx.bfs_tree(G_symmetric, "user C") và hiển thị cây BFS cho thấy cách thức đi đến các nút khác trong mạng bắt đầu từ "user C".

6.1.3 Phân bố độ lệch tâm của một nút trong mạng xã hội

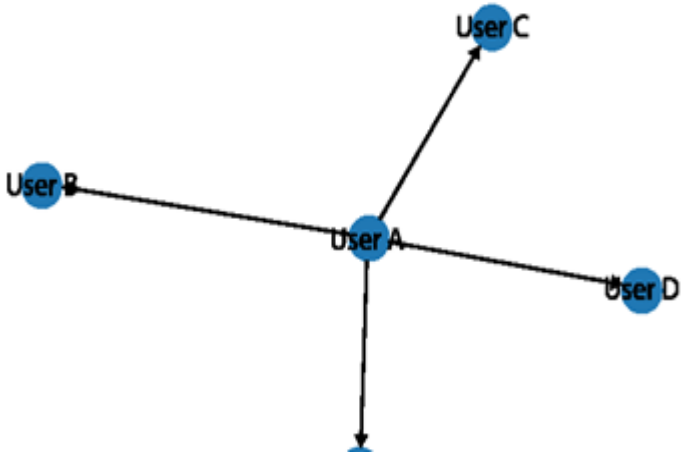
Độ lệch tâm được xác định bởi độ dài của đường đi ngắn nhất dài nhất bắt đầu từ một nút nhất định trong mạng xã hội. Nó cũng là thuật ngữ để chỉ sự khác biệt lớn nhất giữa một cạnh và tất cả các cạnh khác trong mạng. Phân bố độ lệch tâm này được ký hiệu bằng e(V). Trong Hình 6.1, "User C" được chọn làm nút bắt đầu, trong khi trong Hình 6.2, "User A" được chọn làm nút bắt đầu để thực hiện tìm kiếm theo chiều rộng (BFS).

```
Tree_info = nx.bfs_tree(G_symmetric, 'User C')
nx.draw_networkx(Tree_Info)
```



Hình 6.1. Thực hiện thuật toán tìm kiếm theo chiều rộng (BFS) cho User C

```
Tree_info2 = nx.bfs_tree(G_symmetric, 'User A')
nx.draw_networkx(Tree_info2)
```



Hình 6.2. Thực hiện thuật toán tìm kiếm theo chiều rộng (BFS) cho User A

Hàm nx.eccentricity(Graph_Type, 'Node_Info') được sử dụng để tính toán giá trị từ đồ thị được cung cấp trong đoạn lệnh bên dưới.

```
nx.eccentricity(G_symmetric,'User A')
```

1

```
nx.eccentricity(G_symmetric,'User C')
```

2

Theo hàm độ lệch tâm của mạng, nút "User A" có độ lệch tâm là 1, trong khi nút "User C" có độ lệch tâm là 2.

6.1.4 Độ đo trung tâm trong mạng xã hội

Xác định các nút quan trọng trong mạng xã hội đó là vai trò của các độ đo trung tâm (centrality metrics). Các độ đo trung tâm giúp chúng ta tìm ra những nút phổ biến nhất, được yêu thích nhất và có ảnh hưởng nhất trong mạng. Có một vài thuật ngữ then chốt giúp chúng ta tìm hiểu thêm về một nút cụ thể bên trong mạng xã hội:

- Độ Trung Tâm Theo Bậc (Degree Centrality)
- Độ Trung Tâm của Vector Riêng (Eigenvector Centrality)
- Độ Trung Tâm Trung Gian (Betweenness Centrality)

Số lượng cạnh kết nối một nút cụ thể được gọi là độ trung tâm theo bậc của nó. Trong một mạng xã hội, đây có thể là số lượng bạn bè của một người. Độ trung tâm theo bậc được tính toán bằng hàm degree().

Theo đồ thị được cung cấp trong Hình 6.2, biểu thức nx.degree(G_symmetric, "User A") được sử dụng để tính toán độ trung tâm. Kết quả thu được trong trường hợp này là 4. "User A" được kết nối với các nút khác: "User B", "User C", "User D" và "User E".

6.1.5 Tính số đo gần gũi

Tính số đo gần gũi (Closeness centrality) của một nút thể hiện khoảng cách trung bình (nghịch đảo) của nó đến tất cả các nút khác. Các nút có điểm gần cao có khoảng cách ngắn nhất đến tất cả các nút khác. Nút có điểm gần cao luôn luôn có khoảng cách ngắn nhất đến tất cả các nút khác trong mạng. Đây là một số liệu hữu ích trong các mạng xã hội để ước tính tốc độ thông tin truyền giữa hai nút. Kết quả của hàm closeness_centrality() được cung cấp trong hàm:

```
nx.closeness_centrality(G_symmetric)
```

{'User A': l.a,

'User B': l.a,

'User C': a.8,

'User D': a.8,

'User E': 9.6666666666666666}

6.1.6 Chương trình tính số đo trung gian

Số đo trung gian (Betweenness centrality) định lượng tần suất một nút hoạt động như cầu nối giữa hai nút khác trên đường đi ngắn nhất. Nó được tạo ra bởi Linton

Freeman như một phương pháp để đo lường tác động của một người lên sự giao tiếp của những người khác trong một mạng xã hội. Nó cho thấy tần suất một nút xuất hiện trên đường đi ngắn nhất giữa hai điểm. Nói cách khác, nó định lượng mức độ thường xuyên một nút nằm trên đường đi ngắn nhất giữa hai nút khác.

Các nút có độ trung tâm trung gian cao đóng vai trò quan trọng trong việc truyền thông/dòng thông tin của mạng. Chúng có khả năng ảnh hưởng và kiểm soát đáng kể lên những nút khác. Một cá nhân ở vị trí nổi bật như vậy có thể tác động đến toàn bộ nhóm bằng cách che giấu hoặc tô màu cho sự thật trong quá trình truyền thông tin.

Hàm nx.betweenness_centrality() trong NetworkX được sử dụng để tính toán độ trung gian của một mạng và kết quả được thể hiện trong hàm sau:

```
nx.betweenness_centrality(G_symmetric)
```

{'User A': a.166666666666666>

'User B': a.166666666666666 >

'User C': a.e>

'User D': a.e>

'User E': a.e}

Hàm này cho phép chúng ta xác định có chuẩn hóa dữ liệu trung gian hay không, có tính đến các nút đầu cuối trong việc đếm đường đi ngắn nhất hay không.

6.2 NGHIÊN CỨU TRƯỜNG HỢP CỦA FACEBOOK

Trong phần này sử dụng tập dữ liệu mạng ego kết hợp từ Facebook, bao gồm tổng hợp danh sách bạn bè trên Facebook của mười người. Có thể tải file combined.txt cần thiết cho Facebook từ trang web của Đại học Stanford. Để phân tích dữ liệu, có thể sử dụng các API của Facebook/Twitter để truy xuất dữ liệu Facebook/Twitter của riêng bạn. Tập dữ liệu này bao gồm "vòng tròn" (hoặc "danh sách bạn bè") trên Facebook. Ứng dụng Facebook này được sử dụng để thu thập dữ liệu từ những người tham gia khảo sát. Tập dữ liệu bao gồm thuộc tính của nút (hồ sơ), vòng tròn và mạng ego. Dữ liệu Facebook đã được ẩn danh bằng cách thay thế ID Facebook nội bộ của mỗi người

tham gia bằng một giá trị mới. Hơn nữa, mặc dù các vector đặc trưng từ tập dữ liệu này đã được công khai, nhưng việc giải thích chúng hiện đã bị ẩn giấu. Do đó, bằng cách sử dụng dữ liệu ẩn danh, chúng ta có thể xác định liệu hai người có cùng một thành viên nào đó, nhưng không thể biết ý nghĩa của những thành viên đó.

Toàn bộ phân tích phần nghiên cứu được thực hiện bằng cách sử dụng thư viện NetworkX trong Python. Chi tiết phân tích như sau:

```
import networkx as nx
import matplotlib.pyplot as plt
import pa ndas as pd
%matplotlib inline
import warnings ; warnings.simplefilter( 'ignore ')
```

```
df = pd.read_csv( '/content/facebook combined .txt')
```

Trong đoạn lệnh trên, các gói cần thiết như NetworkX, matplotlib.pyplot và pandas được import để phục vụ cho việc phân tích. Pandas là một bộ công cụ phân tích dữ liệu phổ biến dựa trên Python, có thể được nạp bằng lệnh "import pandas as pd". "read_csv" là một hàm cơ bản của Pandas để đọc và thao tác với các file văn bản và csv. Hàm 'read_csv' này đọc tệp facebook_combined.txt.

Hàm info() trong Python cung cấp tóm tắt nội dung của một DataFrame. Phương thức này cung cấp thông tin về kiểu dữ liệu của chỉ mục và cột, các giá trị không null và dung lượng bộ nhớ của một DataFrame. Hàm df.info() hiển thị tên cột, các giá trị không null, số lượng và kiểu dữ liệu, như được minh họa trong Hình 6.3:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 88233 entries, 0 to 88232
Data columns (total 1 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   0 1     88233 non-null  object
dtypes: object(1)
memory usage: 689.4+ KB
```

Hình 6.3. Hàm info() để hiển thị nội dung của DataFrame

Theo mã Python được cung cấp trong Hình 6.4, tổng số nút là 4039 và số cạnh trong tập dữ liệu là 88234.

```
G_fb = nx.read_edgelist("/content/facebook_combined.txt", create_using = nx.Graph(), nodetype=int)
```

```
print(nx.info(G_fb))
```

```
Graph with 4039 nodes and 88234 edges
```

Hình 6.4. Hàm info() để minh họa các nút và cạnh trong tập dữ liệu

Hàm degree_centrality() trả về mức độ trung tâm theo bậc cao nhất trong mạng. Theo chương trình Python được cung cấp trong Hình 6.5, mức độ trung tâm theo bậc này có giá trị là 107. Ngoài ra, khi sử dụng hàm nx.degree(), có thể thấy rõ ràng rằng "User 107" chỉ kết nối với 1045 người dùng khác trong mạng.

```
dg_centrality = nx.degree_centrality(G_info)
sorted_dg_centrality = sorted(dg_centrality.items(), key=operator.itemgetter(1), reverse=True)
sorted_dg_centrality[:10]
```

```
[(107, 0.258791480931154),
 (1684, 0.1961367013372957),
 (1912, 0.1869737493808816 3),
 (3437, 0.13546310054482416),
 (0, 0.08593363051015354),
 (2543, 0.07280832095096582),
 (2347, 0.07206537890044576),
 (1888, 0.0629024269440317),
 (1800, 0.06067360079247152),
 (1663, 0.058197127290737984)]
```

```
nx.degree(G_info, [107])
```

```
DegreeView({107: 1045})
```

Hình 6.5. Hàm Degree_centrality() and nx.degree()

Thuật ngữ average_shortest_path_length() dùng để chỉ số bước trung bình trên đường đi ngắn nhất giữa hai nút bất kỳ trong một mạng. Đây là số liệu đo lường hiệu quả truyền thông tin hoặc lan truyền trên mạng. Giá trị đầu ra của số liệu này là 3.6925068496963913, được hiển thị trong Hình 6.6.

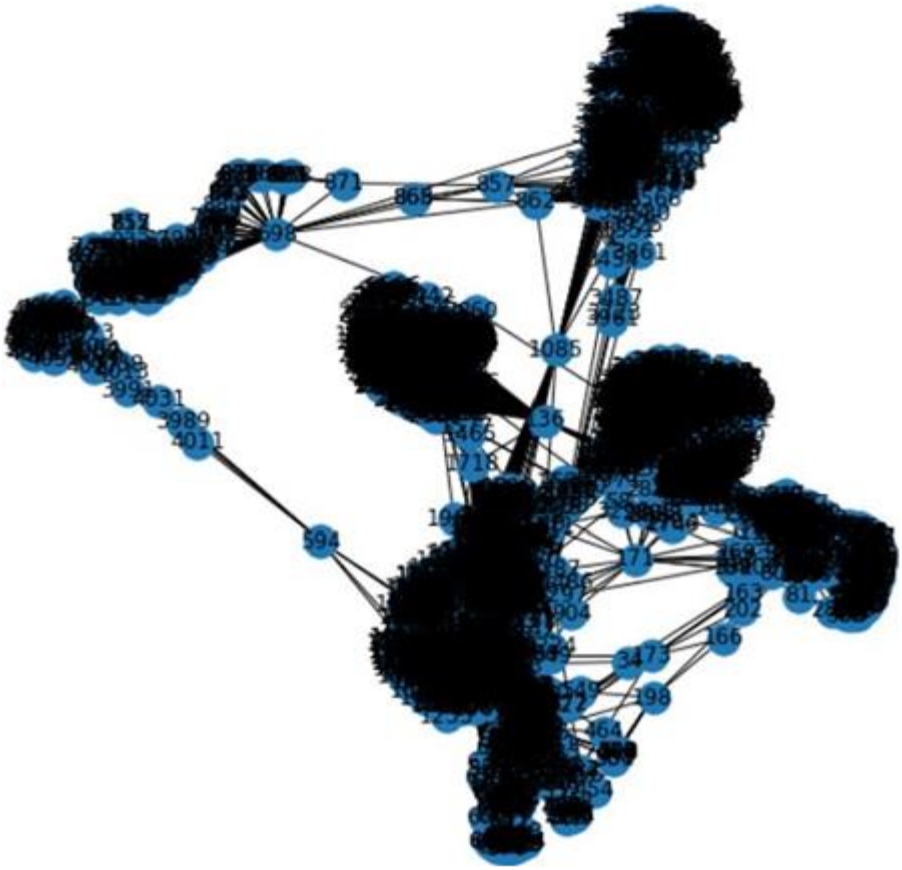
```
print(nx.average_shortest_path_length(G_info))
```

```
3.6925068496963913
```

Hình 6.6. Tính toán đường đi ngắn nhất trung bình giữa hai mạng

Tiếp theo, phương thức nx.draw_networkx() được sử dụng trong đoạn lệnh bên dưới để hiển thị tập dữ liệu Facebook dưới dạng đồ thị (Hình 6.7).

```
plt.figure(figsize=( 10,10))
nx.draw_networkx(G_info);
```

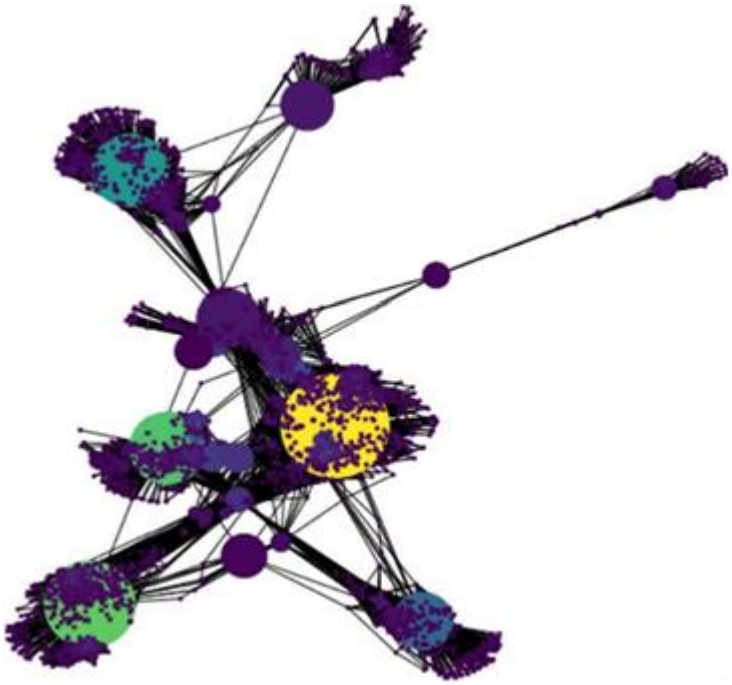


Hình 6.7. Thể hiện trực quan dữ liệu Facebook với draw_networkx()

Để hiển thị mạng theo cách mà màu sắc của nút thay đổi theo độ trung tâm theo bậc (degree centrality) và kích thước của nút thay đổi theo độ trung tâm trung gian (betweenness centrality). Mã Python cho hàm betweenness_centrality() được mô tả trong đoạn lệnh sau.

```
pes = nx.spring_layout(G_info)
betCent=nx.betweenness_centrality(G_info, normalized= True, endpoints=True)
node_color = [2eeee.e * G_info.degree(v) for v in G_info ]
node_size = [v * 1eeee for v in betCent.values()]
plt.figure(figsize=( l0,10))
nx.draw_networkx(G_info, pos=pos, with_labels =False,
node_color =node_color, node_size=node_size )
plt.axis( 'off');
```

Đầu ra của đoạn mã này được thể hiện dưới dạng đồ thị trong Hình 6.8 với các nút có màu sắc khác nhau.



Hình 6.8. Thể hiện trực quan của tập dữ liệu với betweenness_centrality ()

Các nút có độ trung tâm trung gian (betweenness centrality) cao nhất được xác định bằng công thức sorted(). Ngoài ra, nó hiển thị nhãn của năm nút cùng với giá trị độ trung tâm trung gian tương ứng. Trong trường hợp này, các nút 107, 1684, 3437, 1912 và 1085 là những nút có độ trung tâm trung gian cao nhất và chúng kiểm soát luồng thông tin trong mạng. Thông thường, các nút được kết nối nhiều hơn sẽ nằm trên các đường đi ngắn nhất giữa các nút khác. Nút 107 đóng vai trò quan trọng vì nó là nút trung tâm trong các số liệu đo lường trung tâm được kiểm tra. Hàm sorted() trong mã Python được cung cấp trong Hình 6.9.

```
sorted(betCent, key=betCent.get, reverse=True)[:5]
[107, 1684, 3437, 1912, 1085]
```

Hình 6.9. Phương thức sorted() hiển thị các nút theo thứ tự của độ trung tâm

Khái niệm PageRank được sử dụng trong đoạn lệnh sau để ước tính mức độ phổ biến của các liên kết đến trong mạng.

```
g_fb_pr = nx.pagerank(G_info)
top = 10
max_pagera nk = sorted (g_fb_pr.items(), key = lambda v: -v[1])[:top]
max_pagerank
```

Kết quả của hàm PageRank() được đưa ra trong Hình 6.10. Nó cho thấy nút 3437 nổi tiếng hơn các nút khác trong mạng.

```
[(3437, 0.0076145868447496),
 (107, 0.006936420955866117),
 (1684, 0.006367162138306824),
 (0, 0.006289602618466542),
 (1912, 0.003876971600884498),
 (348, 0.002348096972780577),
 (686, 0.002219359259800019),
 (3980, 0.0021703235790099928),
 (414, 0.001800299047070226),
 (698, 0.0013171153138368812)]
```

Hình 6.10. Các nút phổ biến theo phương pháp PageRank()

Tài liệu tham khảo

[1]. Knickerbocker, David. Network Science with Python: Explore the networks around us using network science, social network analysis, and machine learning. Packt Publishing Ltd, 2023.

[2]. Knoke, David, and Song Yang. Social network analysis. SAGE publications, 2019.

[3]. Đỗ, Phúc. "Phân tích Mạng xã hội và Ứng dụng.", Đại học Quốc gia Tp.HCM (2017).

[4]. Nguyễn Thị Hải Bình, "Cấu trúc dữ liệu và giải thuật", HUTECH, 2021.