Mục lục

[Mục lục 1](#_Toc426548341)

[Danh mục các ký hiệu và chữ viết tắt 3](#_Toc426548342)

[Danh mục các bảng 4](#_Toc426548343)

[Danh mục các hình vẽ, đồ thị 5](#_Toc426548344)

[MỞ ĐẦU 6](#_Toc426548345)

[Giới thiệu 6](#_Toc426548346)

[Phát biểu bài toán 8](#_Toc426548347)

[Ứng dụng của bài toán 8](#_Toc426548348)

[Thách thức của bài toán 9](#_Toc426548349)

[Mục tiêu của luận văn 11](#_Toc426548350)

[Đối tượng và phạm vi nghiên cứu 12](#_Toc426548351)

[Đóng góp của luận văn 12](#_Toc426548352)

[Cấu trúc của luận văn 13](#_Toc426548353)

[Chương 1. TỔNG QUAN CÁC HƯỚNG TIẾP CẬN 15](#_Toc426548354)

[1.1 Hướng tiếp cận phân tích ma trận 15](#_Toc426548355)

[1.1.1 Phương pháp lan truyền độ tin cậy 15](#_Toc426548356)

[1.1.2 Phương pháp phân tích phổ đồ thị 17](#_Toc426548357)

[1.2 Hướng tiếp cận máy học 18](#_Toc426548358)

[1.2.1 Phương pháp khai thác đặc trưng dựa vào lý thuyết cân bằng 18](#_Toc426548359)

[1.2.2 Phương pháp HOC (Higher Order Cycle) 22](#_Toc426548360)

[1.2.3 Phương pháp khai thác đặc trưng PNR (Positive Negative Ratio) 23](#_Toc426548361)

[1.2.4 Phương pháp khai thác đặc trưng latent 24](#_Toc426548362)

[1.3 Kết luận 25](#_Toc426548363)

[Chương 2. CƠ SỞ LÝ THUYẾT 27](#_Toc426548364)

[2.1 Mô hình dự đoán tổng quát 27](#_Toc426548365)

[2.2 Tiền xử lý dữ liệu 28](#_Toc426548366)

[2.3 Đặc trưng dự đoán loại liên kết 28](#_Toc426548367)

[2.3.1 Đặc trưng explicit 29](#_Toc426548368)

[2.3.2 Đặc trưng latent 31](#_Toc426548369)

[2.3.3 Đặc trưng Positive Negative Ratio 35](#_Toc426548370)

[2.3.4 Đánh giá ưu nhược điểm các loại đặc trưng 37](#_Toc426548371)

[2.4 Phương pháp máy học 38](#_Toc426548372)

[2.4.1 Phương pháp transfer learning 38](#_Toc426548373)

[2.4.2 Phương pháp SVM 40](#_Toc426548374)

[2.5 Kết luận 41](#_Toc426548375)

[Chương 3. MÔ HÌNH DỰ ĐOÁN LOẠI LIÊN KẾT “POSITIVE” VÀ “NEGATIVE” 42](#_Toc426548376)

[3.1 Giới thiệu 42](#_Toc426548377)

[3.1.1 Ý tưởng 42](#_Toc426548378)

[3.1.2 Thành phần dữ liệu và các định nghĩa 43](#_Toc426548379)

[3.2 Phương pháp Transfer AdaBoost with SVM 45](#_Toc426548380)

[3.3 Kết luận 48](#_Toc426548381)

[Chương 4. KẾT QUẢ THỰC NGHIỆM VÀ ĐÁNH GIÁ 50](#_Toc426548382)

[4.1 Dữ liệu thực nghiệm 50](#_Toc426548383)

[4.2 Các độ đo đánh giá 51](#_Toc426548384)

[4.3 Các phương pháp thực nghiệm 52](#_Toc426548385)

[4.4 Kết quả thực nghiệm 54](#_Toc426548386)

[4.4.1 So sánh độ đo accuracy 54](#_Toc426548387)

[4.4.2 So sánh đường cong PR 57](#_Toc426548388)

[4.4.3 So sánh thời gian thực hiện 58](#_Toc426548389)

[4.5 Chương trình minh họa 59](#_Toc426548390)

[4.6 Kết luận 60](#_Toc426548391)

[KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 61](#_Toc426548392)

[Kết quả đạt được 61](#_Toc426548393)

[Hướng phát triển 62](#_Toc426548394)

[DANH MỤC CÔNG BỐ KHOA HỌC CỦA TÁC GIẢ 63](#_Toc426548395)

[TÀI LIỆU THAM KHẢO 64](#_Toc426548396)

[PHỤ LỤC 66](#_Toc426548397)

[Phụ lục 1: Kết quả thực nghiệm chi tiết 66](#_Toc426548398)

[Phụ lục 2: Nội dung bài báo được chấp nhận đăng trong hội nghị KES 74](#_Toc426548399)

# Danh mục các ký hiệu và chữ viết tắt

E Tập cạnh của đồ thị mạng xã hội

G Mô hình đồ thị mạng xã hội

HOC Higher Order Cycle

NMTF Nonnegative Matrix Tri-Factorization

PLSA Probabilistic Latent Semantic Analysis

PNR Positive Negative Ratio

PR Precision-Recall

SVM Support Vector Machine

TAS Transfer AdaBoost with SVM

V Tập đỉnh của đồ thị mạng xã hội

# Danh mục các

[Bảng 1.1. Danh sách các ma trận sử dụng trong mô hình lan truyền độ tin cậy 15](#_Toc426548400)

[Bảng 1.2. Danh sách các mẫu hạt nhân lan truyền 16](#_Toc426548401)

Y[Bảng 3.1. Thành phần dữ liệu của bài toán dự đoán loại liên kết 44](#_Toc426548402)

Y[Bảng 4.1. Thống kê dữ liệu thực nghiệm của 3 mạng xã hội 50](#_Toc426548403)

[Bảng 4.2. Thống kê dữ liệu cân bằng của 3 mạng xã hội 50](#_Toc426548404)

[Bảng 4.3. Các phương pháp thực nghiệm (**** **** có, **** **** không) 53](#_Toc426548405)

[Bảng 4.4. Độ đo accuracy (%) trung bình của các phương pháp thực nghiệm (E – Epinions, S – Slashdot, W – Wiki) 54](#_Toc426548406)

[Bảng 4.5. Thời gian thực hiện trung bình (phút) của quá trình rút trích đặc trưng (E – Epinions, S – Slashdot, W – Wiki) 58](#_Toc426548407)

[Bảng 4.6. Thời gian thực hiện trung bình (phút) của quá trình huấn luyện và kiểm tra dữ liệu với SVM (E – Epinions, S – Slashdot, W – Wiki) 58](#_Toc426548408)

[Bảng 4.7. Thời gian thực hiện trung bình (phút) của quá trình huấn luyện và kiểm tra dữ liệu với TAS (E – Epinions, S – Slashdot, W – Wiki) 59](#_Toc426548409)

# Danh mục các hình vẽ

[Hình 0.1. Mô hình đồ thị mạng xã hội 6](#_Toc426548410)

[Hình 0.2. Mô hình bài toán dự đoán loại liên kết “*positive*” và “*negative*” 8](#_Toc426548411)

Y[Hình 1.1. Mô hình lan truyền độ tin cậy 16](#_Toc426548412)

[Hình 1.2. Mô hình phân tích phổ đồ thị 17](#_Toc426548413)

[Hình 1.3. Bộ ba cân bằng tuân theo quy luật (1) [3] 19](#_Toc426548414)

[Hình 1.4. Bộ ba cân bằng tuân theo quy luật (2), (3), (4) [3] 19](#_Toc426548415)

[Hình 1.5. Bộ ba không cân bằng [3] 19](#_Toc426548416)

[Hình 1.6. Các hình thái liên kết của bộ ba theo lý thuyết cân bằng [22] 20](#_Toc426548417)

[Hình 1.7. Mô hình máy học khai thác đặc trưng dựa vào lý thuyết cân bằng 22](#_Toc426548418)

Y[Hình 2.1. Mô hình tổng quát dự đoán dấu của liên kết 27](#_Toc426548419)

[Hình 2.2. Bốn loại bộ ba 30](#_Toc426548420)

[Hình 2.3. Mã giả thuật toán tối ưu hóa [23] 35](#_Toc426548421)

Y[Hình 3.1. Mô hình dự đoán loại liên kết “*positive*” và “*negative*” cho mạng xã hội mới thành lập 43](#_Toc426548422)

[Hình 3.2. Mã giả thuật toán TAS 47](#_Toc426548423)

Y[Hình 4.1. Độ đo accuracy trung bình 55](#_Toc426548424)

[Hình 4.2. Đường cong PR trung bình 57](#_Toc426548425)

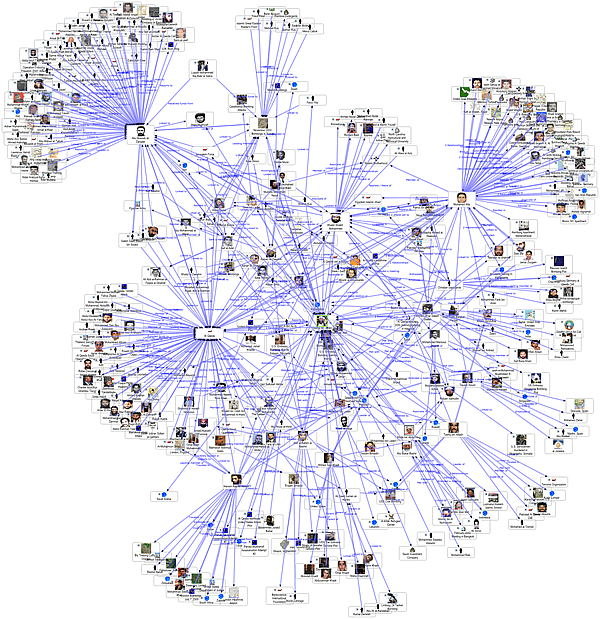
[Hình 4.3. Chương trình minh họa 59](#_Toc426548426)

# MỞ ĐẦU

## Giới thiệu

Kể từ khi ra đời, internet không ngừng phát triển và mở rộng đã mang lại rất nhiều tiện ích hữu dụng như: hệ thống thư điện tử (*email*), trò chuyện trực tuyến (*chat*), tìm kiếm dữ liệu (*search engine*), … Những năm gần đây, sự ra đời của mạng xã hội trực tuyến là một trong những sự kiện thú vị nhất. Có rất nhiều mạng xã hội trực tuyến phổ biến hiện nay như: *Facebook*, *Twitter*, *LinkedIn*, … Với sự phát triển nhanh chóng về số người sử dụng trên toàn thế giới, mạng xã hội trực tuyến như một mô hình thu nhỏ của thế giới thực. Do đó, mạng xã hội trở thành nơi cất giữ thông tin và các mối quan hệ giữa các cá nhân, doanh nghiệp, … Những thông tin này tạo thành “*đám mây tri thức*”. Việc tìm hiểu và khai thác hiệu quả những thông tin này sẽ tạo tiền đề cho nhiều ứng dụng khác như: hệ thống tư vấn, hệ thống tìm kiếm thông tin, tiếp thị trực tuyến, điều tra tội phạm, dự đoán sự phát triển của mạng xã hội [22], … Vì vậy, việc khai thác những thông tin trên mạng xã hội để áp dụng vào thực tiễn ngày càng trở nên quan trọng.

Mạng xã hội được định nghĩa như là một mô hình mạng (*đồ thị*) được cấu tạo bởi các đỉnh và các cạnh. Các đỉnh là tập các đối tượng và các cạnh là tập các liên kết thể hiện mối quan hệ hoặc sự tương tác giữa các đối tượng này.



Hình 0.1. Mô hình đồ thị mạng xã hội[[1]](#footnote-2)

Dự đoán liên kết trong mạng xã hội là bài toán khá mới và thu hút sự quan tâm nghiên cứu của các nhà khoa học, nhất là khi các mạng xã hội ngày càng phát triển mạnh mẽ trên phạm vi toàn thế giới như hiện nay.

Bài toán dự đoán liên kết trong mạng xã hội có thể chia thành 4 bài toán con như sau:

* Dự đoán sự tồn tại của liên kết.
* Dự đoán loại liên kết.
* Dự đoán trọng số liên kết.
* Dự đoán số lượng liên kết.

Hầu hết các nghiên cứu dự đoán liên kết chỉ xem xét đến các mối quan hệ tích cực (*positive*) tồn tại trong mạng. Từ đó, các nghiên cứu này tập trung vào việc xác định sự tồn tại của liên kết (*dự đoán một liên kết mới giữa 2 đối tượng trong mạng xã hội trong tương lai có tồn tại hay không*) bởi vì từ bài toán này dễ mở rộng sang bài toán dự đoán trọng số của liên kết (*những liên kết có trọng số khác nhau*) và số lượng liên kết (*có hơn một liên kết giữa cùng một cặp đỉnh trong mạng xã hội*). Nhưng trong khoảng vài năm gần đây, các nhà khoa học bắt đầu quan tâm đến bài toán dự đoán loại liên kết, cụ thể bài toán dự đoán loại liên kết “*positive*” và liên kết “*negative*” trong mạng xã hội. Bởi lẽ, trong thực tế, mối liên kết giữa các đối tượng trong mạng xã hội luôn hàm chứa song song hai loại liên kết: positive (*mối quan hệ tích cực giữa bạn bè*) và negative (*mối quan hệ tiêu cực giữa hai cá nhân có quan điểm trái ngược nhau*). Từ các liên kết “*negative*”, các nghiên cứu có thể suy dẫn ra những thông tin hữu ích liên quan đến các liên kết hàm ẩn khác, đóng góp một phần không nhỏ vào quá trình và kết quả dự đoán liên kết trong toàn mạng. Vì thế, nếu chỉ xem xét các liên kết “*positive*” mà bỏ qua các liên kết “*negative*” thì sẽ là một thiếu sót của bài toán dự đoán liên kết trong mạng xã hội.

Do đó, luận văn sẽ tập trung nghiên cứu và giải quyết *bài toán dự đoán loại liên kết trong mạng xã hội*. Cụ thể là bài toán dự đoán loại liên kết “*positive*” (liên kết có dấu hiệu cạnh là “+”) và liên kết “*negative*” (liên kết có dấu hiệu cạnh là “”) trong mạng xã hội. Trong đó, đầu vào thường là thông tin dấu hiệu cạnh có sẵn của một đồ thị mạng xã hội và đầu ra là dấu hiệu cạnh của tập cạnh chưa được gán nhãn trong đồ thị đó.

## Phát biểu bài toán

Mạng xã hội là một mô hình mạng (*đồ thị*) G được cấu tạo bởi các đỉnh và các cạnh:

* Các đỉnh là tập các đối tượng V.
* Các cạnh là tập các liên kết E.

Luận văn quy định rằng liên kết “*positive*” sẽ được biểu diễn bằng một cạnh có dấu “+” trên đồ thị G và ngược lại liên kết “*negative*” sẽ được biểu diễn bằng một cạnh có dấu “”.

Bài toán dự đoán loại liên kết “*positive*” và “*negative*” trong mạng xã hội có *input* và *output* như sau:

* ***Input***: Thông tin dấu hiệu cạnh có sẵn của một đồ thị mạng xã hội tại một thời điểm cụ thể.
* ***Output***: Dấu hiệu cạnh của tập cạnh chưa được gán nhãn trong đồ thị đó.



Hình 0.2. Mô hình bài toán dự đoán loại liên kết “*positive*” và “*negative*”

## Ứng dụng của bài toán

Mạng xã hội là một đối tượng động, có cấu trúc thay đổi thường xuyên theo thời gian. Việc dự đoán loại liên kết cũng là một bài toán dự đoán liên kết trong mạng xã hội [16]. Vì thế, khi giải quyết được bài toán dự đoán liên kết này, luận văn sẽ dự đoán được cấu trúc của mạng xã hội trong tương lai (*cách thức mạng xã hội hoạt động và tiến hóa theo thời gian*) để ứng dụng vào các lĩnh vực thực tế như:

* Xác định cấu trúc mạng lưới tội phạm và dự đoán các liên kết có thể tồn tại giữa các đối tượng. Từ đó, có thể ứng dụng vào lĩnh vực an ninh là điều tra và truy tìm tội phạm.
* Khắc phục vấn đề thưa thớt dữ liệu trong các hệ thống giới thiệu sản phẩm cho người dùng sử dụng cơ cấu lọc cộng tác. Nhờ đó, có thể cải thiện chất lượng quảng cáo sản phẩm của hệ thống.
* Ứng dụng cải thiện chức năng giới thiệu bạn mới phù hợp trên mạng xã hội hoặc đề cử việc cộng tác giữa các nhà khoa học và các nhà nghiên cứu.
* Dự đoán các trang web mà người dùng có thể viếng thăm nhằm cải thiện hiệu quả việc tìm kiếm thông tin.
* Các hệ thống tìm kiếm nâng cao.

Vào năm 1995, mạng xã hội ra đời và trở nên phổ biến từ những năm 2000. Cho đến nay, mạng xã hội đã phát triển với tốc độ không tưởng với sự gia tăng không ngừng về số người sử dụng và lượng thông tin chia sẻ. Mạng xã hội trở thành một kho lưu trữ dữ liệu khổng lồ tiềm ẩn nhiều thông tin quý báu cần được khai thác và sử dụng.

Dự đoán liên kết là một phần của lĩnh vực phân tích mạng xã hội nhằm khai thác những lợi ích rút trích được từ kho dữ liệu khổng lồ này. Bài toán dự đoán liên kết trong mạng xã hội nói chung, cụ thể là bài toán dự đoán loại liên kết trong mạng xã hội là một trong những bài toán thu hút được nhiều sự quan tâm của các nhà khoa học hiện nay vì tính mới mẻ và những ứng dụng to lớn mà nó mạng lại trong thực tế.

## Thách thức của bài toán

Bản thân bài toán dự đoán liên kết trong mạng xã hội nói chung đã có những thách thức nhất định [16]. Khi xem xét mạng xã hội trong ngữ cảnh có hai loại liên kết “*positive*” và “*negative*”, điều này sẽ làm bản chất và độ phức tạp của nhiều vấn đề liên quan đến cấu trúc mạng thay đổi so với việc chỉ xét liên kết “*positive*” trong mạng xã hội. Chẳng hạn như trong các vấn đề sau:

* Tính biến động và độ lớn của mạng xã hội: Mạng xã hội là loại dữ liệu có tính biến động rất cao và khả năng bùng nổ về kích thước là rất lớn. Số lượng người tham gia từ hàng trăm nghìn đến hàng trăm triệu người. Đồng thời số lượng liên kết thêm vào hoặc bỏ đi cũng luôn luôn thay đổi. Điều này cũng đủ làm cho những đặc trưng cũng như cấu trúc của mạng trở nên phức tạp và thường xuyên biến động. Với sự xuất hiện thêm loại liên kết “*negative*” thì sự phức tạp và độ biến động này càng cao hơn nữa. Vì thế, các thuật toán áp dụng vào môi trường mạng xã hội có 2 loại liên kết “*positive*” và “*negative*” đều phải trả giá cao về độ phức tạp.
* Tính không đầy đủ của thông tin được chia sẻ: Giá trị lớn nhất của mạng xã hội là những thông tin mà người dùng chia sẻ cho nhau. Những thông tin này chính là cơ sở, là nhân tố mang tính quyết định đối với những bài toán khai thác mạng trong đó có bài toán dự đoán liên kết. Tuy nhiên, người dùng không phải lúc nào cũng chia sẻ đầy đủ những thông tin của họ. Vì thế, hầu hết các dữ liệu mạng xã hội đều là dữ liệu không đầy đủ. Những thông tin còn thiếu này, bằng những cách nào đó phải được suy diễn thêm ra nhằm đáp ứng cho yêu cầu của bài toán. Đặc biệt, đối với bài toán dự đoán loại liên kết “*positive*” và “*negative*”, điều này là một trong những yếu tố ảnh hưởng đến độ chính xác và tính hiệu quả.
* Sự khác biệt lớn giữa các mạng xã hội: Một thách thức nữa của bài toán dự đoán liên kết trong mạng xã hội có 2 loại liên kết “*positive*” và “*negative*” đó là sự khác biệt khá lớn giữa tính chất của những thông tin trong từng mạng xã hội. Cách định nghĩa của đỉnh và cạnh trong các mạng cũng sẽ khác nhau. Ví dụ, ở hai mạng xã hội *Epinions[[2]](#footnote-3)* và *Slashdot[[3]](#footnote-4)* thì khái niệm đỉnh cạnh được xác định như sau:
  + *Epinions*: đỉnh là người dùng tham gia bình luận sản phẩm, cạnh là liên kết thể hiện sự tin tưởng hay không tin tưởng của người dùng đối với bài bình luận sản phẩm của người dùng khác.
  + *Slashdot*: đỉnh là người dùng tham gia vào mạng, cạnh là liên kết thể hiện mối quan hệ (*“friend” hoặc “foe”*) của người dùng đối với những người dùng khác.

Do đó, việc làm thế nào để tìm ra những đặc trưng mang tính tổng quát nhất không phụ thuộc vào ngữ cảnh riêng của từng mạng xã hội cũng là một vấn đề đáng quan tâm của bài toán dự đoán loại liên kết “*positive*” và “*negative*”.

Vì vậy, không phải phương pháp nào được đề xuất cho mạng xã hội chỉ xét liên kết “*positive*” đều có thể áp dụng trong mạng xã hội có 2 loại liên kết “*positive*” và “*negative*”. Ngoài ra, dự đoán loại liên kết “*positive*” và “*negative*” trong mạng xã hội có sự kết nối với lý thuyết cân bằng thuộc lĩnh vực nghiên cứu xã hội trong khi các bài toán dự đoán liên kết truyền thống không có sự liên quan nào với lý thuyết này.

Bên cạnh những vấn đề trên, bài toán dự đoán loại liên kết “*positive*” và “*negative*” trong mạng xã hội còn tồn tại một vấn đề quan trọng khác. Các nghiên cứu trước đây về bài toán này [16] đều yêu cầu có nhiều thông tin dấu hiệu cạnh cho dữ liệu đầu vào: “*Dấu hiệu của tất cả các liên kết ngoại trừ liên kết cần dự đoán đều biết trước*”. Yêu cầu này đôi khi không thực tế, nhất là đối với các ***mạng xã hội mới thành lập***, thông tin dấu hiệu cạnh có sẵn trong mạng này rất khan hiếm và gây nhiều khó khăn trong việc xây dựng một bộ phân lớp dấu hiệu cạnh tốt từ lượng thông tin có sẵn này.

Tóm lại, từ tất cả những vấn đề còn tồn tại trên và những đặc điểm khác biệt của bài toán dự đoán loại liên kết “*positive*” và “*negative*” trong mạng xã hội so với các bài toán dự đoán liên kết truyền thống, bài toán này trở thành một bài toán đầy hứa hẹn và hấp dẫn.

## Mục tiêu của luận văn

* Nghiên cứu đề xuất cải tiến phương pháp dự đoán loại liên kết bằng cách sử dụng sự hỗ trợ của đồ thị mạng xã hội khác theo mô hình máy học. Mục tiêu của phương pháp đề xuất là nâng cao hiệu quả dự đoán loại liên kết “*positive*” và “*negative*” cho mạng xã hội mới thành lập theo 2 tiêu chí:
* Nâng cao độ chính xác dự đoán.
* Rút ngắn thời gian rút trích đặc trưng, quá trình huấn luyện và kiểm tra dữ liệu.
* Nghiên cứu bộ dữ liệu của 3 mạng xã hội trực tuyến: *Epinions*, *Slashdot* và *Wikipedia*. Từ đó, tiến hành xây dựng dữ liệu thực nghiệm trên 3 mạng xã hội này.

## Đối tượng và phạm vi nghiên cứu

* Dấu hiệu của các liên kết trong đồ thị xã hội của 3 mạng trực tuyến *Epinions*, *Slashdot* và *Wikipedia*.
  + Dữ liệu của 3 mạng xã hội trên được công khai và cho phép tải về tại *Stanford Large Network Dataset Collection*.
  + Đường dẫn tải về: http://snap.stanford.edu/data/index.html.

## Đóng góp của luận văn

Do hầu hết các nghiên cứu trước đây về bài toán dự đoán loại liên kết “*positive*” và “*negative*” trong mạng xã hội [16] đều có yêu cầu cao về dữ liệu đầu vào: “*Dấu hiệu của tất cả các liên kết ngoại trừ liên kết cần dự đoán đều biết trước*” nên luận văn sẽ xem xét vấn đề dự đoán loại liên kết trong một ngữ cảnh thực tế hơn. Giả sử, luận văn có một mạng xã hội mới thành lập, thông tin dấu hiệu cạnh có sẵn trong mạng này rất khan hiếm. Từ lượng thông tin có sẵn này, việc xây dựng một bộ phân lớp dấu hiệu cạnh tốt sẽ gặp nhiều khó khăn. Để giải quyết vấn đề này, luận văn xem xét đến sự trợ giúp của một mạng xã hội khác có nhiều thông tin dấu hiệu cạnh. Phương pháp học này là phương pháp “*transfer learning*” [18,19].

Dựa theo phương pháp “*transfer learning*”, luận văn đề xuất phương pháp **Transfer AdaBoost with SVM** (***TAS***). Phương pháp này là cải tiến phương pháp học của ***Freund & Schapire*** [9] và kết hợp với việc thiết kế bộ phân lớp thành phần là ***RBFSVM*** (*SVM sử dụng nhân RBF*) [15].

Ngoài ra, nghiên cứu trước đây về mạng xã hội mới thành lập đề xuất sử dụng 2 loại đặc trưng là ***explicit*** và ***latent*** [23]. Tuy nhiên, trong nghiên cứu này, luận văn đề xuất thay thế đặc trưng ***latent*** bằng đặc trưng thích hợp với mô hình “*transfer learning*” hơn là ***Positive Negative Ratio*** (*PNR*) [21]. Đặc trưng ***PNR*** được xây dựng dựa trên lý thuyết ra quyết định trong mạng xã hội (*thuộc lĩnh vực tâm lý xã hội*). Đặc trưng này có tính khái quát cao và vượt trội hơn các đặc trưng được đề xuất trước đây về cả 3 tiêu chí: *độ chính xác*, *tổng quát* và *tốc độ*.

Kết quả thực nghiệm trên ba mạng xã hội đã chứng minh phương pháp được đề xuất trong luận văn (*phương pháp* ***TAS*** *sử dụng các đặc trưng* ***explicit*** *và* ***PNR***) có thể cải thiện độ chính xác dự đoán và giảm thời gian rút trích đặc trưng, huấn luyện và kiểm tra dữ liệu hơn phương pháp được đề xuất trước đây về mạng xã hội mới thành lập.

Kết quả nghiên cứu của luận văn được chấp nhận đăng trong kỷ yếu hội nghị khoa học quốc tế KES 2015 (*The Nineteenth International Conference on Knowledge-Based and Intelligent Information & Engineering Systems*): Anh-Thu Nguyen-Thi, Phuc Quang Nguyen, Thanh Duc Ngo, Tu-Anh Nguyen-Hoang, “Transfer AdaBoost SVM for Link Prediction in Newly Signed Social Networks using Explicit and PNR Features”, 7-9/9/2015, Singapore.

## Cấu trúc của luận văn

***Bố cục của luận văn gồm 5 phần với nội dung chính như sau***:

* Chương 1. Tổng Quan Các Hướng Tiếp Cận

Chương này trình bày các hướng tiếp cận chính của bài toán dự đoán loại liên kết trong mạng xã hội. Đồng thời, phân tích và đánh giá các công trình nghiên cứu tiêu biểu đã được công bố có liên quan đến bài toán dự đoán loại liên kết này. Từ đó, xác định những vấn đề còn tồn tại và thách thức của bài toán dự đoán loại liên kết cho mạng xã hội mới thành lập mà luận văn sẽ tập trung giải quyết.

* Chương 2. Cơ Sở Lý Thuyết

Trong chương này, luận văn sẽ trình bày những cơ sở lý thuyết cần thiết để xây dựng phương pháp dự đoán loại liên kết “*positive*” và “*negative*” cho mạng xã hội mới thành lập. Nội dung trình bày bao gồm:

* Mô hình tổng quát dùng để dự đoán loại liên kết “*positive*” và “*negative*” trong mạng xã hội.
* Các đặc trưng được sử dụng cho bài toán dự đoán loại liên kết “*positive*” và “*negative*” trong mạng xã hội.
* Phương pháp máy học (*transfer learning và SVM*) xây dựng mô hình dự đoán dấu của liên kết.
* Chương 3. Mô Hình Dự Đoán Loại Liên Kết “Positive” Và “Negative”

Nội dung chính của chương là trình bày chi tiết về mô hình dự đoán loại liên kết mà luận văn đề xuất để nâng cao hiệu quả dự đoán loại liên kết “*positive*” và “*negative*” cho mạng xã hội mới thành lập. Mục tiêu chính của phương pháp đề xuất là:

* Nâng cao độ chính xác dự đoán.
* Rút ngắn thời gian rút trích đặc trưng, quá trình huấn luyện và kiểm tra dữ liệu.
* Chương 4. Kết Quả Thực Nghiệm Và Đánh Giá

Trình bày kết quả thực nghiệm của phương pháp đề xuất và các phương pháp cơ sở (*phương pháp của các công trình nghiên cứu liên quan đã được công bố*). Từ đó, tiến hành phân tích, đánh giá và so sánh giữa các phương pháp với nhau.

* Kết Luận Và Hướng Phát Triển

Luận văn đưa ra những kết luận từ quá trình nghiên cứu và kết quả thực nghiệm đạt được. Từ đó, đưa ra những định hướng nghiên cứu mở rộng cho đề tài trong tương lai.

# Chương 1. TỔNG QUAN CÁC HƯỚNG TIẾP CẬN

Mạng xã hội là một mô hình mạng (*đồ thị G*) được cấu tạo bởi các đỉnh (*tập đối tượng V*) và các cạnh (*tập các liên kết E*). Liên kết “*positive*” và “*negative*” được biểu diễn lần lượt bằng các cạnh có dấu “+” và dấu “”. Các nghiên cứu hiện tại về bài toán dự đoán loại liên kết “*positive*” và “*negative*” trong mạng xã hội có thể phân loại thành 2 hướng tiếp cận chính như sau [21]:

* Phân tích ma trận (Guha [11] – 2004, Kunegis [12] – 2010).
* Máy học (Leskovec [13,14] – 2010, Kai-Yang [4] – 2011, Tuyen [21] – 2013, Ye [23] – 2013).

1. **Hướng tiếp cận phân tích ma trận**
2. ***Phương pháp lan truyền độ tin cậy***

Phương pháp của Guha và các cộng sự [11] là nghiên cứu tiên phong về bài toán dự đoán loại liên kết trong mạng xã hội. Dựa vào ý tưởng của các thuật toán lan truyền trên mạng, Guha đã phát triển một mô hình lan truyền độ tin cậy để dự đoán mức độ tin cậy hoặc không tin cậy giữa các cặp đỉnh trong mạng xã hội. Các ma trận sử dụng trong mô hình lan truyền độ tin cậy của Guha được liệt kê trong ***Bảng 1.1*** như sau:

Bảng 1.1. Danh sách các ma trận sử dụng trong mô hình lan truyền độ tin cậy

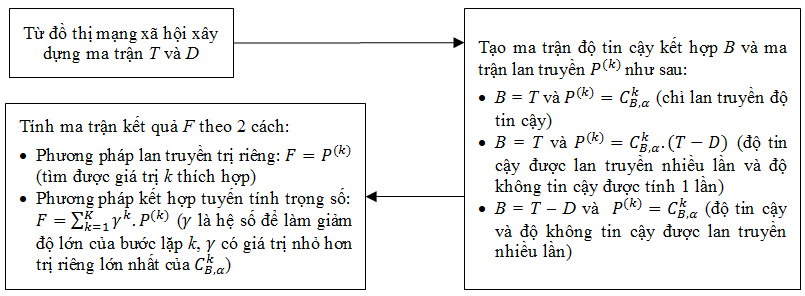
|  |  |  |
| --- | --- | --- |
| **Tên** | **Diễn giải** | **Giá trị** |
| *T* | Ma trận tin cậy ( thể hiện mức độ tin tưởng giữa *u* và *v*) | [0,1] |
| *D* | Ma trận không tin cậy ( thể hiện mức độ không tin tưởng giữa *u* và *v*) | [0,1] |
| *B* | Ma trận độ tin cậy kết hợp |  |
|  | Ma trận lan truyền hạt nhân kết hợp | Là bộ trọng số của 4 lan truyền hạt nhân |
|  | Ma trận lan truyền qua *k* bước |  |
| *F* | Ma trận kết quả |  |

Trong mô hình lan truyền độ tin cậy, ma trận sử dụng 4 mẫu hạt nhân lan truyền được liệt kê trong ***Bảng 1.2*** như sau:

Bảng 1.2. Danh sách các mẫu hạt nhân lan truyền

|  |  |  |
| --- | --- | --- |
| **Toán tử** | **Mẫu hạt nhân lan truyền** | **Diễn giải** |
|  | Lan truyền trực tiếp |  |
|  | Đồng trích dẫn |  |
|  | Chuyển vị độ tin cậy | Nếu thì khi sẽ ngầm định cả việc |
|  | Cặp tin cậy | Nếu thì khi sẽ ngầm định cả việc |

Quy trình hoạt động của mô hình lan truyền độ tin cậy [11] gồm có các bước thực hiện được biểu diễn chi tiết trong ***Hình 1.1*** như sau:

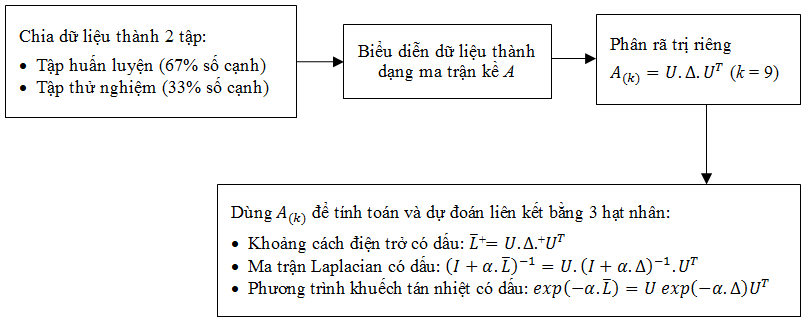


Hình 1.1. Mô hình lan truyền độ tin cậy

Kết quả thực nghiệm của [11] cho thấy *k* là một giá trị nhỏ vừa thì có độ tin cậy cao hơn so với *k* lớn. Tuy nhiên, giá trị *k* càng lớn thì kết quả càng mang tính toàn cục hơn. Vì vậy, để giải quyết vấn đề này, tác giả Guha đề xuất 2 phương pháp để tính ma trận kết quả *F* như trên. Từ đó, tác giả Guha đã cải tiến và tìm ra chỉ số lặp tối ưu cũng như những tham số thích hợp cho mô hình lan truyền này thông qua việc thử nghiệm nhiều lần trên dữ liệu mạng xã hội *Epinions*.

1. ***Phương pháp phân tích phổ đồ thị***

Dựa vào phương pháp phân tích phổ đồ thị, tác giả Kunegis và các cộng sự [12] đề xuất sử dụng các nhân được dẫn xuất từ biến thể có dấu của ma trận Laplacian (*khoảng cách điện trở có dấu, ma trận Laplacian có dấu và phương trình khuếch tán nhiệt có dấu*) để dự đoán loại liên kết. Quy trình hoạt động dựa vào phương pháp phân tích phổ đồ thị [12] gồm có các bước thực hiện được biểu diễn chi tiết trong ***Hình 1.2*** như sau:



Hình 1.2. Mô hình phân tích phổ đồ thị

Sau đó tác giả Kunegis thực hiện so sánh phương pháp đề xuất trên với ba phương pháp khác:

* Rút gọn hạng ma trận: Sử dụng *A(k)* thay cho *A*.
* Tổng lũy thừa: Đa thức bậc 4 của *A(k)*.
* Hàm mũ ma trận [12]:

(1.1)

Trong đó:

* : trị riêng của ma trận kề .
* : là các ma trận phân rã từ .

Tất cả các thực nghiệm của [12] được chạy trên dữ liệu *Slashdot*. Kết quả cho thấy ba phương pháp sử dụng nhân được đề xuất cho kết quả tốt hơn các phương pháp trước đó. Trong đó, đạt kết quả tốt nhất là *ma trận Laplacian có dấu*. Tuy nhiên, tác giả Kunegis không đề cập đến phương pháp học có giám sát nào để đưa ra hệ số cho mô hình của mình và đồng thời tính chất về dữ liệu cũng chưa được quan tâm đến.

1. **Hướng tiếp cận máy học**
2. ***Phương pháp khai thác đặc trưng dựa vào lý thuyết cân bằng***

Để giải quyết bài toán dự đoán loại liên kết trong mạng xã hội theo phương pháp máy học, trong công trình nghiên cưnguyếtứu [13,14], tác giả Leskovec và các cộng sự tập trung nghiên cứu xây dựng các đặc trưng dựa vào lý thuyết cân bằng.

* Lý thuyết cân bằng

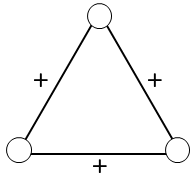
Lý thuyết cân bằng là một lý thuyết thuộc lĩnh vực nghiên cứu xã hội xuất hiện từ giữa thế kỷ 20, được Heider phát biểu thành hệ thống vào năm 1940. Sau đó, lý thuyết trong lĩnh vực xã hội này được Cartwright và Harary [3] chuyển sang “*ngôn ngữ*” lý thuyết đồ thị.

Lý thuyết cân bằng xem xét tính chất cân bằng của một mạng xã hội thông qua những cách thức liên kết có thể có của bộ ba phần tử trong mạng xã hội. Từ đó, chỉ ra những cách liên kết nào của bộ ba là cân bằng hay không cân bằng.

Một bộ ba được gọi là cân bằng nếu số liên kết “*positive”* giữa chúng là số lẻ. Hay nói một cách khác bộ ba đó phải tuân theo những quy luật cân bằng sau:

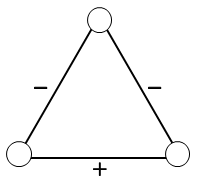
1. Bạn của bạn mình là bạn của mình (*a là bạn của b, b là bạn của c thì a là bạn của c*).
2. Bạn của kẻ thù là kẻ thù của mình (*a là bạn của b, mà b là kẻ thù của c, vậy a là kẻ thù của c*).
3. Kẻ thù của bạn mình là kẻ thù của mình (*a là bạn của b, c là kẻ thù của b, vậy c là kẻ thù của a*).
4. Kẻ thù của kẻ thù mình là bạn của mình (*a là kẻ thù của b, b là kẻ thù của c, vậy c là bạn của a*).

Bộ ba cân bằng tuân theo quy luật (1) được biểu diễn trong ***Hình 1.3*** như sau:



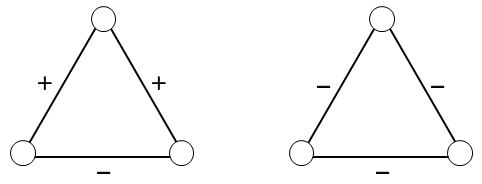
Hình 1.3. Bộ ba cân bằng tuân theo quy luật (1) [3]

Bộ ba cân bằng tuân theo quy luật (2), (3), (4) được biểu diễn trong ***Hình 1.4*** như sau:



Hình 1.4. Bộ ba cân bằng tuân theo quy luật (2), (3), (4) [3]

Đối với bộ ba không cân bằng thì không được xét trong quy luật cân bằng. Bộ ba không cân bằng được biểu diễn trong ***Hình 1.5*** như sau:



Hình 1.5. Bộ ba không cân bằng [3]

* Lý thuyết trạng thái

Lý thuyết trạng thái được phát triển dựa trên nền tảng lý thuyết cân bằng. Trong đó, lý thuyết trạng thái cho rằng một liên kết có nhiều ý nghĩa hơn trong thuyết cân bằng. Cụ thể là:

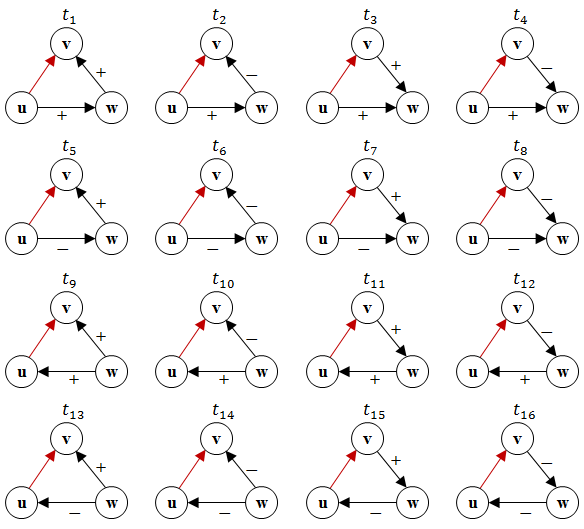
* *a* xem *b* là bạn. (*giống với thuyết cân bằng*)
* *b* có trạng thái cao hơn *a*. (*điểm khác biệt với thuyết cân bằng*)

Ngược lại đối với liên kết . Từ đó, bậc của trạng thái có thể được lan truyền qua nhiều lần thông qua đường đi trong toàn mạng xã hội.

* Vì thế, tạo ra kết quả dự đoán khác hơn so với lý thuyết cân bằng.
* Nội dung nghiên cứu của công trình [14]

Trong công trình nghiên cứu này, tác giả Leskovec tiến hành so sánh và kết hợp giữa *lý thuyết cân bằng* với *lý thuyết trạng thái* để giải quyết bài toán dự đoán loại liên kết trong mạng xã hội. Đây là công trình đầu tiên trong lĩnh vực này nghiên cứu đặc trưng của dữ liệu để đưa ra phương pháp dự đoán.

Cụ thể, tác giả Leskovec đề xuất một định nghĩa tường minh cho vấn đề dự đoán dấu của liên kết. Tác giả nhận thấy có một sự tương thích lớn giữa *lý thuyết cân bằng* và *lý thuyết trạng thái* với bài toán dự đoán loại liên kết trong mạng xã hội. Nghiên cứu [14] đã phân tích và đưa ra những thống kê và thực nghiệm nhằm mô hình hóa *lý thuyết trạng thái* và *lý thuyết cân bằng* của xã hội [3]. Tác giả Leskovec đã đưa ra 16 đặc trưng ứng với 16 loại bộ ba theo *lý thuyết cân bằng*. Hình thái liên kết của 16 loại bộ ba được biểu diễn cụ thể trong ***Hình 1.6*** như sau:



Hình 1.6. Các hình thái liên kết của bộ ba theo lý thuyết cân bằng [22]

* Nội dung nghiên cứu của công trình [13]

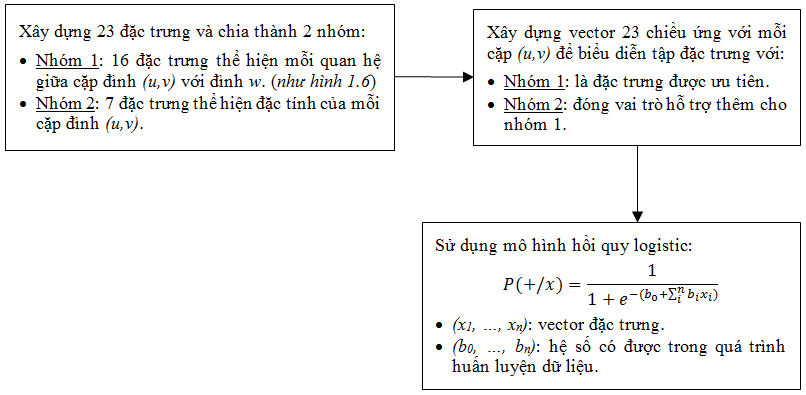
Trong công trình tiếp theo [13], Leskovec và các cộng sự đã xây dựng một tập đặc trưng khá đa dạng (*23 chiều*), đồng thời đề xuất ra một mô hình máy học có giám sát để dự đoán dấu liên kết dựa trên tập đặc trưng được xây dựng.

Tập đặc trưng được xây dựng gồm 16 đặc trưng được tạo thành dựa theo *lý thuyết cân bằng* [14] và bổ sung thêm 7 đặc trưng thể hiện đặc tính của cặp đỉnh *(u,v)* như sau:

* : Số cạnh dương đi vào đỉnh *v*
* : Số cạnh âm đi vào đỉnh *v*
* : Số cạnh dương xuất phát từ đỉnh *u*
* : Số cạnh âm xuất phát từ đỉnh *u*
* *C(u,v)*: Số láng giềng chung
* : Tổng số cạnh đi vào *v*
* : Tổng số cạnh đi ra *u*

Ý tưởng: “*Dấu của một cạnh (u, v) là dấu được dự đoán sau khi đã tối thiểu hóa số lượng các bộ ba không cân bằng liên quan đến cạnh (u, v) (những bộ ba không tuân theo lý thuyết cân bằng xã hội)*”.

Quy trình hoạt động của mô hình máy học khai thác đặc trưng dựa vào lý thuyết cân bằng [13] gồm có các bước thực hiện được biểu diễn chi tiết trong ***Hình 1.7*** như sau:



Hình 1.7. Mô hình máy học khai thác đặc trưng dựa vào lý thuyết cân bằng

Mô hình dự đoán của Leskovec cho kết quả tốt hơn khi số lượng láng giềng chung tối thiểu *Em* giữa các cặp đỉnh đủ lớn. Do đó, đối với phương pháp này, số lượng láng giềng chung giữa hai đỉnh đóng vai trò quyết định tính hiệu quả cho cả bài toán.

1. ***Phương pháp HOC (Higher Order Cycle)***

Dựa trên ý tưởng của Leskovec [13] kết hợp với ý tưởng lan truyền của tác giả Guha [11] và các cộng sự, nhóm tác giả Kai-Yang, Nagarajan, Ambuj [4] đã đề xuất tập đặc trưng mới bằng cách biến đổi các đặc trưng cục bộ thành đặc trưng mang tính toàn cục hơn.

Ý tưởng: “*Xem xét mối quan hệ của cặp nút đang xét trong chu trình có k đỉnh (chu trình này có số cạnh dương là số lẻ để đảm bảo tính cân bằng) thay vì chỉ xem xét mối quan hệ của bộ ba đỉnh chứa cặp nút đang xét*”.

Đặc điểm của phương pháp ***HOC***:

* Với k = 3, số đặc trưng sẽ là 42 = 16 (*trở về phương pháp của tác giả Leskovec*).
* Với k tăng, số đặc trưng sẽ tăng theo hàm mũ 4k-1.
* Độ phức tạp của bài toán là O(4n).

Do đó trong [4], thí nghiệm chỉ xét trường hợp k  5 và sử dụng công thức hội quy logistic để dự đoán:

(1.2)

Trong đó:

* : là phép biến đổi.
* : là các hệ số có được trong quá trình huấn luyện dữ liệu.

Thực nghiệm của [4] cho thấy phương pháp ***HOC*** đã cho kết quả khả quan khi *Em* nhỏ song chi phí của phương pháp này là khá cao dù đã được giảm từ *O(4n)* xuống còn *O(2n)* dựa vào tính đối xứng của ma trận đặc trưng.

1. ***Phương pháp khai thác đặc trưng PNR (Positive Negative Ratio)***

Trong công trình nghiên cứu [21], tác giả Tuyến đã chứng minh rằng đồ thị biểu diễn mạng xã hội có những đặc tính cơ bản liên hệ mật thiết đến cấu trúc và cơ chế vận hành của mạng xã hội. Do đó, tác giả Tuyến đã sử dụng các quy luật hành vi và tâm lý xã hội có ảnh hưởng đến việc dự đoán loại liên kết “*positive*” và “*negative*” làm cơ sở cho quá trình rút trích đặc trưng ***PNR***.

Đặc trưng ***PNR*** là đặc trưng 2 chiều được xây dựng từ bốn thông tin cơ bản tại mỗi đỉnh trong đồ thị mạng:

* Số cạnh dương đi vào.
* Số cạnh dương đi ra.
* Số cạnh âm đi vào.
* Số cạnh âm đi ra.

Trong công trình [21], tác giả Tuyến sử dụng mô hình hội quy logistic được đề xuất bởi [13] để dự đoán:

(1.3)

Trong đó:

* : là vector đặc trưng.
* : là các hệ số có được trong quá trình huấn luyện dữ liệu.

Tất cả các thực nghiệm của [21] được chạy trên dữ liệu *Epinions*, *Slashdot* và *Wiki*. Kết quả cho thấy rằng, đặc trưng ***PNR*** có độ khái quát cao và ổn định trên nhiều dữ liệu mạng xã hội khác nhau. Do đó, đặc trưng ***PNR*** đảm bảo đạt được độ chính xác cao hơn so với các loại đặc trưng được công bố trước đây. Mặt khác, do quá trình tính toán xây dựng đặc trưng đơn giản và số chiều đặc trưng thấp (*2 chiều*) nên đặc trưng ***PNR*** có khả năng rút ngắn thời gian rút trích đặc trưng, quá trình huấn luyện và kiểm tra dữ liệu trên tập dữ liệu lớn.

1. ***Phương pháp khai thác đặc trưng latent***

Công trình nghiên cứu [23] của tác giả Ye giải bài toán dự đoán loại liên kết “*positive*” và “*negative*” trong một ngữ cảnh hoàn toàn mới là mạng xã hội mới thành lập. Vì dữ liệu của mạng xã hội mới thành lập rất khan hiếm nên tác giả Ye tận dụng sự hỗ trợ của một mạng xã hội khác có nhiều thông tin dấu hiệu cạnh. Tác giả Ye xây dựng đặc trưng ***latent*** thể hiện mô hình chung phổ biến giữa 2 đồ thị mạng xã hội này.

Bên cạnh đặc trưng ***latent***, tác giả Ye còn đề xuất sử dụng loại đặc trưng ***explicit*** thể hiện tính chất đặc trưng của các cạnh trong đồ thị mạng xã hội. Đặc trưng ***explicit*** bao gồm 4 đặc trưng cơ bản:

* Độ đo của đỉnh.
* Độ trung tâm của đỉnh.
* Số lượng bộ ba.
* Số láng giềng chung.

Bốn đặc trưng cơ bản này được xây dựng dựa theo công trình [13]. Tuy nhiên, để phù hợp với đặc điểm của mạng xã hội mới thành lập (*dữ liệu có sẵn rất khan hiếm*), tác giả Ye tiến hành rút trích đặc trưng ***explicit*** trên đồ thị mạng xã hội chỉ dựa vào thông tin kết nối mà bỏ qua dấu hiệu cạnh của liên kết.

Tập dữ liệu huấn luyện được sử dụng trong công trình [23] bao gồm:

* Dữ liệu của mạng xã hội mới thành lập.
* Dữ liệu của mạng xã hội có nhiều thông tin dấu hiệu cạnh dùng để hỗ trợ.

Từ tập dữ liệu huấn luyện này, tác giả Ye tiến hành rút trích 2 loại đặc trưng: ***explicit*** và ***latent***. Sau đó, sử dụng mô hình máy học ***SVM*** cho quá trình dự đoán dấu hiệu của liên kết.

Trong nghiên cứu [23], tác giả Ye tiến hành chạy thực nghiệm trên dữ liệu của 3 mạng xã hội: *Epinions*, *Slashdot* và *Wiki*. Kết quả thực nghiệm trong [23] cho thấy rằng, 2 loại đặc trưng mà tác giả Ye đề xuất khá phù hợp với mô hình dự đoán loại liên kết cho mạng xã hội mới thành lập. Tuy nhiên, đặc trưng ***latent*** có số chiều khá lớn và quá trình xây dựng đặc trưng này phải trải qua nhiều bước tính toán phức tạp. Do đó, mô hình dự đoán loại liên kết sử dụng đặc trưng ***latent*** phải trả giá cao về chi phí tính toán.

1. **Kết luận**

Dựa vào các công trình nghiên cứu được trình bày trên, có thể thấy rằng bài toán dự đoán loại liên kết trong mạng xã hội có nhiều thách thức và đầy hứa hẹn. Luận văn có thể đánh giá ưu nhược điểm của 2 hướng tiếp cận chính như sau:

* Tiếp cận phân tích ma trận
  + Trả giá cao về độ phức tạp (*do sử dụng các phép toán trên ma trận có kích thước lớn*) nên không thích hợp với mạng xã hội lớn có độ biến động cao.
  + Không quan tâm đến đặc trưng của dữ liệu nên hiệu quả chưa cao.
* Tiếp cận máy học
  + Có độ phức tạp thấp hơn và độ chính xác cao (*do sử dụng hiệu quả các đặc trưng của dữ liệu*).

Tuy nhiên, hầu hết các công trình được khảo sát (*ngoại trừ công trình [*23*]*) đều có yêu cầu cao về dữ liệu đầu vào là một mạng xã hội có sẵn nhiều thông tin dấu hiệu cạnh. Điều này đôi khi không khả thi khi dự đoán loại liên kết trong một mạng xã hội mới thành lập (*thông tin dấu hiệu cạnh có sẵn rất khan hiếm*).

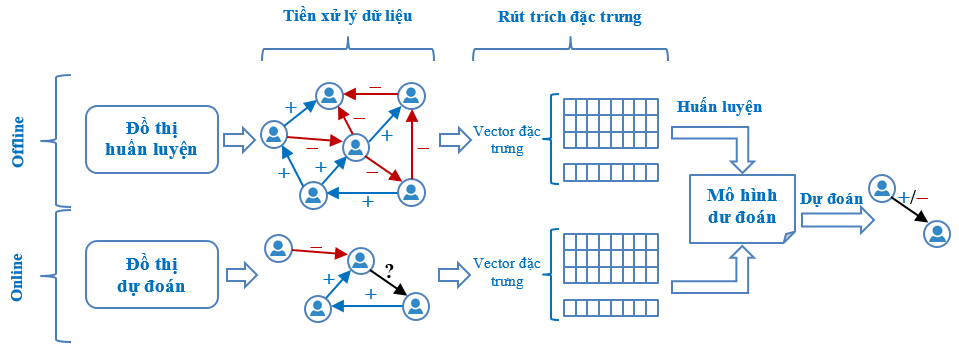
Do đó, đề tài này tập trung vào ***nghiên cứu bài toán dự đoán loại liên kết “positive” và “negative” trong mạng xã hội mới thành lập*** nhằm gia tăng hiệu suất dự đoán loại liên kết theo hướng tiếp cận máy học. Vì thông tin dấu hiệu có sẵn của mạng xã hội mới thành lập rất khan hiếm nên gặp rất nhiều khó khăn trong việc xây dựng một bộ phân lớp dấu hiệu cạnh tốt. Vì thế, dựa vào ý tưởng của công trình [23], luận văn nghiên cứu hướng giải quyết theo phương pháp “*transfer learning*” là tận dụng sự hỗ trợ của một mạng xã hội khác có nhiều thông tin dấu hiệu cạnh.

Đồng thời, tìm hiểu, phân tích và đề xuất sử dụng những đặc trưng có tính khái quát cao phù hợp với mô hình máy học. Từ đó, có thể nâng cao kết quả dự đoán, giảm thời gian rút trích đặc trưng, huấn luyện và kiểm tra dữ liệu.

# Chương 2. CƠ SỞ LÝ THUYẾT

1. **Mô hình dự đoán tổng quát**

Mô hình tổng quát giải quyết bài toán dự đoán loại liên kết “*positive*” và “*negative*” trong mạng xã hội theo hướng tiếp cận máy học hoạt động với 2 quá trình: *huấn luyện (thực hiện offline)* và *dự đoán (thực hiện online)*. Toàn bộ quy trình hoạt động của mô hình được biểu diễn chi tiết trong ***Hình 2.1*** như sau:



Hình 2.1. Mô hình tổng quát dự đoán dấu của liên kết

Chi tiết hoạt động của quá trình huấn luyện và quá trình dự đoán như sau:

* ***Quá trình huấn luyện*** có nhiệm vụ xây dựng mô hình dự đoán dấu (+/) của liên kết qua 3 bước: tiền xử lý dữ liệu, rút trích đặc trưng và huấn luyện dữ liệu.
  + Tiền xử lý dữ liệu

Mạng xã hội được biểu diễn thành đồ thị trong bước này. Các đối tượng (*người dùng, bình luận, …*) trong mạng xã hội được biểu diễn thành các đỉnh. Mối quan hệ giữa các đối tượng được biểu diễn thành các cạnh. Loại quan hệ (*positive hay negative*) giữa các đối tượng sẽ được mã hóa thành dấu hiệu cạnh (+/). Sau đó, tiến hành cân bằng dữ liệu huấn luyện sao cho số lượng liên kết “*positive*” và “*negative*” là bằng nhau dựa theo phương pháp của tác giả Guha [11].

* + Rút trích đặc trưng

Vì không phải tất cả các thông tin trong mạng xã hội đều hữu ích nên bước rút trích đặc trưng là cần thiết cho việc loại bỏ dữ liệu dư thừa và mã hóa các thông tin cần thiết của các cạnh thành vector đặc trưng. Chất

lượng của vector đặc trưng đóng vai trò quan trọng trong quá trình dự đoán dấu của liên kết.

* + Huấn luyện dữ liệu

Vector đặc trưng sau khi rút trích từ mạng xã hội trở thành dữ liệu đầu vào cho thuật toán máy học. Mỗi vector đặc trưng được dán nhãn là “*positive*” hoặc “*negative*”. Vector đặc trưng và nhãn của chúng được sử dụng để đào tạo ra mô hình dự đoán.

* ***Quá trình dự đoán*** có các bước tiền xử lý dữ liệu và rút trích đặc trưng tương tự như quá trình huấn luyện. Sau đó, mô hình dự đoán được sử dụng để dự đoán dấu của liên kết là (+/).

1. **Tiền xử lý dữ liệu**

Dữ liệu của phần lớn các mạng xã hội có 2 loại liên kết “*positive*” và “*negative*” thường có sự mất cân bằng: 80% liên kết trong mạng là liên kết “*positive*” và 20% còn lại là liên kết “*negative*” [13]. Dữ liệu bị mất cân bằng có thể dẫn đến sự sai lệch trong kết quả huấn luyện. Ví dụ như mô hình huấn luyện học từ dữ liệu mất cân bằng này có khuynh hướng dự đoán ra loại liên kết “*positive*” nhiều hơn.

Để khắc phục tình trạng mất cân bằng giữa số lượng liên kết “*positive*” và liên kết “*negative*” này, luận văn áp dụng phương pháp của tác giả Guha [11]. Ý tưởng chính của phương pháp này là:

“*Với dữ liệu của một mạng xã hội, giữ lại tất cả các liên kết* ***negative*** *của mạng này. Tương ứng với mỗi liên kết* ***negative****, tiến hành chọn ngẫu nhiên một liên kết* ***positive****.*”

Sau quá trình cân bằng dữ liệu, dữ liệu sẽ có 50% liên kết “*positive*” và 50% còn lại là liên kết “*negative*”. Tập dữ liệu cân bằng này sẽ được sử dụng cho quá trình huấn luyện và dự đoán.

1. **Đặc trưng dự đoán loại liên kết**

Trong phần này, luận văn sẽ trình bày các loại đặc trưng được sử dụng trong các công trình nghiên cứu trước đây về bài toán dự đoán loại liên kết trong mạng xã hội. Đầu tiên, là 2 loại đặc trưng ***explicit*** và ***latent*** được công bố bởi [23] (*đây là công trình tiên phong giải bài toán dự đoán loại liên kết cho mạng xã hội mới thành lập*). Mặc khác, luận văn còn tìm hiểu về đặc trưng ***PNR*** được công bố bởi [21] vì những ưu điểm nội bật của loại đặc trưng này.

1. ***Đặc trưng explicit***

Trong đồ thị có hướng, xét cạnh xuất phát từ đỉnh đến đỉnh . Đặc trưng ***explicit*** của cạnh bao gồm 4 đặc trưng cơ bản sau:

* Độ đo của đỉnh

Là đặc trưng dựa vào bậc của đỉnh, thể hiện mối quan hệ cục bộ của 1 đỉnh với các đỉnh còn lại trong đồ thị, được biểu diễn bởi 2 thông số:

* : số cạnh đi từ đỉnh .
* : số cạnh đi vào đỉnh .

Hai đặc trưng trên thể hiện mật độ kết nối của 2 đỉnh (*cấu thành nên cạnh e*) với các đỉnh khác trong đồ thị.

* Độ trung tâm của đỉnh

Thể hiện tầm quan trọng của đỉnh đó trong đồ thị, là giá trị xác định số lần đỉnh đó là cầu nối trong đoạn đường đi ngắn nhất giữa 2 đỉnh khác. Những đỉnh có độ đo này cao thường có nhiều khả năng xuất hiện trong 1 đoạn đường đi ngắn nhất bất kỳ giữa 2 đỉnh trong đồ thị.

Ứng dụng của đặc trưng này là dùng để xác định tầm ảnh hưởng của 1 người trong mạng xã hội thông qua việc định lượng số lần kiểm soát của người đó đối với quá trình giao tiếp giữa những người khác trong mạng xã hội. Đặc trưng này thể hiện sự yêu thích của người đó khi tham gia vào mạng xã hội.

Xét đỉnh , ta có ***công thức (2.1)*** tính độ trung tâm như sau:

(2.1)

Trong đó:

* : số đường đi ngắn nhất đi từ đỉnh đến đỉnh .
* : số đường đi ngắn nhất đi từ đỉnh  đến đỉnh  thông qua .
* : điều kiện là mỗi cặp đỉnh sao cho .

Với mỗi cạnh , luận văn sử dụng và như là 2 đặc trưng về độ trung tâm của 2 đỉnh cấu thành nên cạnh .

* Số lượng bộ ba

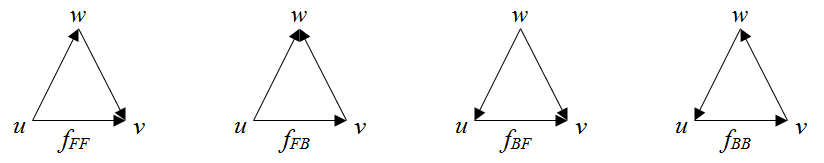
Đặc trưng này có nguồn gốc từ lý thuyết cân bằng thuộc lĩnh vực nghiên cứu xã hội, luận văn sử dụng đặc trưng này để dự đoán loại liên kết giữa đỉnh và thông qua mối quan hệ của 2 đỉnh này với 1 đỉnh thứ 3.

Xét 1 bộ 3 đỉnh gồm 2 đỉnh với đỉnh :

* Cạnh thứ 1: được tạo từ 2 đỉnh và .
* Cạnh thứ 2: được tạo từ đỉnh với (*có 2 trường hợp xảy ra là đi từ đến hoặc từ đến* ).
* Cạnh thứ 3: được tạo từ đỉnh  với  (*tương tự như cạnh thứ 2 cũng có 2 trường hợp*).
* Có 1  2  2 = 4 loại bộ ba.

Gọi:

* Cạnh đi từ đến hoặc đi từ đến là cạnh tiến (*forward edge*), ký hiệu là *F*.
* Cạnh đi từ đến hoặc đi từ đến là cạnh lùi (*backward edge*), ký hiệu là *B*.
* Bốn loại bộ ba được biểu diễn trong ***Hình 2.2*** như sau:



Hình 2.2. Bốn loại bộ ba

Luận văn sử dụng 4 đặc trưng để lưu số lượng 4 loại bộ ba trên.

* Số láng giềng chung

Số láng giềng chung là đặc trưng thể hiện mối quan hệ của 2 đỉnh và trong đồ thị, nghĩa là xác định số đỉnh (*số hàng xóm chung*) có liên kết cùng lúc với cả 2 đỉnh trên.

Luận văn sử dụng đặc trưng để lưu số lượng các đỉnh  có liên kết cùng lúc với cả 2 đỉnh  và .

Tóm lại, đặc trưng ***explicit*** của cạnh  bao gồm 9 đặc trưng như độ đo của đỉnh ( và ), độ trung tâm của đỉnh ( và ), số lượng bộ ba () và số láng giềng chung ().

1. ***Đặc trưng latent***

* Giới thiệu

Trong công trình nghiên cứu [23], tác giả Ye sử dụng phương pháp ***Nonnegative Matrix Tri-Factorization*** (*NMTF*) để xây dựng đặc trưng ***latent*** thể hiện mô hình chung phổ biến giữa 2 ma trận kề của 2 đồ thị mạng. Trong đó:

* *Đồ thị mục tiêu*: là đồ thị của mạng xã hội cần phải dự đoán dấu hiệu liên kết.
* *Đồ thị hỗ trợ*: là đồ thị của mạng xã hội dùng để hỗ trợ.

Tác giả đề xuất chỉ sử dụng thông tin kết nối của 2 đồ thị trên mà bỏ qua dấu hiệu cạnh của liên kết. Do đó, đồ thị hỗ trợ và đồ thị mục tiêu được biểu diễn như sau:

* Đồ thị hỗ trợ:
* : tập đỉnh.
* : tập cạnh.
* : số lượng phần tử của tập đỉnh .
* : là ma trận kề của (*với mỗi cặp đỉnh thì nếu và ngược lại thì* ).
* Đồ thị mục tiêu:
* : tập đỉnh.
* : tập cạnh.
* : số lượng phần tử của tập đỉnh .
* : là ma trận kề của (*với mỗi cặp đỉnh thì nếu và ngược lại thì* ).

Áp dụng phương pháp ***NMTF*** lên 2 ma trận kề và , ta có phép tính xấp xỉ được biểu diễn bởi ***công thức (2.2)*** và ***công thức (2.3)*** như sau:

(2.2)

(2.3)

Trong đó:

* : là không gian đặc trưng.
* ; : là 4 ma trận đặc trưng.

Đặc trưng ***latent*** của các đỉnh trong đồ thị hỗ trợ và đồ thị mục tiêu được biểu diễn trên cùng một không gian thông qua 4 ma trận đặc trưng .

* : dòng thứ thể hiện đặc trưng liên kết ra của đỉnh  trên không gian .
* : dòng thứ thể hiện đặc trưng liên kết vào của đỉnh trên không gian .
* : dòng thứ thể hiện đặc trưng liên kết ra của đỉnh trên không gian .
* : dòng thứ thể hiện đặc trưng liên kết vào của đỉnh trên không gian .
* Quá trình xây dựng đặc trưng

***Công thức (2.4)*** dùng để tìm không gian dựa vào phương pháp ***NMTF*** như sau:

(2.4)

Trong đó:

* là chuẩn *Frobenius* của ma trận.
* Ví dụ: ( là ma trận có kích thước *m  n*).
* là không gian tiềm ẩn của đồ thị hỗ trợ và đồ thị mục tiêu.
* là tham số giới hạn độ lớn của các phần tử trong ma trận .

Để có thể xây dựng được các đặc trưng ***latent*** tốt cần đảm bảo 2 yếu tố sau:

* *Tối ưu hóa giá trị* : tìm giá trị nhỏ nhất để đảm bảo phép tính xấp xỉ ma trận trong phương pháp ***NMTF*** có sai số chấp nhận được.
* *Tham số phù hợp*: đảm bảo cho giá trị của các phần tử trong 4 ma trận đặc trưng () không tiến đến 0 khi các giá trị trong quá lớn.

Trong công trình [23], tác giả Ye giời thiệu **thuật toán tối ưu hóa** giá trị của (*giá trị nhỏ nhất của* ) để xây dựng 4 ma trận đặc trưng () và không gian tiềm ẩn với ý tưởng chính sau:

“*Cập nhật giá trị của các phần tử trong 1 ma trận đặc trưng theo* ***nguyên lý nhân tử Lagrange*** *trong khi giữ cho giá trị của các phần tử trong các ma trận đặc trưng còn lại không đổi. Áp dụng lần lượt cho từng ma trận đặc trưng.*”

Để áp dụng ***nguyên lý nhân tử Lagrange*** cho thuật toán tối ưu hóa, ***công thức (2.4)*** được biến đổi thành ***công thức (2.5)*** như sau:

(2.5)

Trong đó:

* (Vì )
* Tương tự cho: và

Ngoài ra, dựa vào [23], luận văn áp dụng các ***công thức (2.6), (2.7), (2.8), (2.9)*** và ***(2.10)*** để cập nhật giá trị trong thuật toán tối ưu hóa:

(2.6)

(2.7)

(2.8)

(2.9)

(2.10)

Trong đó:

* : là 2 ma trận kề.
* : là không gian đặc trưng.
* ; : là 4 ma trận đặc trưng.

Trong công trình nghiên cứu [23], tác giả Ye đã xây dựng một thuật toán lặp dùng để cập nhật giá trị của không gian đặc trưng và 4 ma trận đặc trưng cho đến khi giá trị hội tụ. Mã giả thuật toán tối ưu hóa này được trình bày chi tiết trong ***Hình 2.3*** sau:

ITERATIVE UPDATE ALGORITHM FOR Ep. 2.5

//input: Hai ma trận kề ,  và tham số .

//output: Ma trận đặc trưng  và không gian .

1. Khởi tạo giá trị ban đầu cho theo [24] như sau:
2. là kết quả gom nhóm của Probabilistic Latent Semantic Analysis (PLSA)[[4]](#footnote-5). Với được khởi tạo bởi giá trị (kết quả gom nhóm khi áp dụng thuật toán PLSA lên và ).
3. là kết quả phân lớp của thuật toán phân lớp bất kỳ với tập dữ liệu huấn luyện là và .
4. được khởi tạo như sau: mỗi giá trị trong ma trận đều bằng nhau và tổng của các giá trị trên một dòng .
5. while ( chưa hội tụ) do
6. Cập nhật giá trị của theo công thức (2.6).
7. Cập nhật giá trị của theo công thức (2.7).
8. Cập nhật giá trị của theo công thức (2.8).
9. Cập nhật giá trị của theo công thức (2.9).
10. Cập nhật giá trị của  theo công thức (2.10).
11. Chuẩn hóa giá trị của về đoạn .

Hình 2.3. Mã giả thuật toán tối ưu hóa [23]

Giải thích mã giả thuật toán tối ưu hóa trên:

* Dòng 1  4: dùng để khởi tạo giá trị ban đầu của 4 ma trận đặc trưng và không gian đặc trưng .
* Dòng 5  11: dùng để cập nhật giá trị của các ma trận đặc trưng và không gian đặc trưng cho đến khi hội tụ.

Tóm lại, mỗi cạnh có đặc trưng ***latent*** là và thể hiện đặc trưng liên kết ra của đỉnh và đặc trưng liên kết vào của đỉnh . Tương tự cho trường hợp cạnh của .

1. ***Đặc trưng Positive Negative Ratio***

Theo xu hướng phát triển hiện nay, mạng xã hội ngày càng mở rộng và số lượng người tham gia vô cùng lớn. Do đó, mạng xã hội trở thành mô hình ảo của xã hội thực. Mô hình này vận hành theo quy luật hành vi xã hội và tâm lý học. Do đó, đặc trưng ***Positive Negative Ratio*** (*PNR*) [21] sử dụng các quy luật hành vi và tâm lý xã hội có ảnh hưởng đến việc dự đoán loại liên kết “***positive***” và “***negative***” (*tâm lý bầy đàn; thói quen và hành vi ra quyết định; quy luật nhân quả*) làm cơ sở cho quá trình rút trích đặc trưng.

* Tâm lý bầy đàn

Bởi vì sự kết nối rộng lớn và sự lan truyền thông tin nhanh chóng trong mạng xã hội khiến cho hành vi ra quyết định của một đối tượng trong mạng xã hội chịu sự tác động rất lớn từ *tâm lý bầy đàn* hay *tâm lý đám đông*. Nghĩa là tuân theo quy luật sau:

* Một đỉnh có nhiều liên kết vào là liên kết “*positive*” thì có khuynh hướng nhận thêm nhiều liên kết “*positive*” và ngược lại.
* Thói quen và hành vi ra quyết định

“*Thói quen về hành vi*” cũng ảnh hưởng đến hành vi ra quyết định của đối tượng trong mạng xã hội. Nghĩa là, một đối tượng có thói quen tạo nhiều liên kết “*positive*” đến các đối tượng khác trong mạng xã hội; thì trong tương lai, đối tượng này sẽ tiếp tục tạo thêm các liên kết “*positive*” và ngược lại.

* Quy luật nhân quả

Quy luật hành vi và tâm lý xã hội này cũng ảnh hưởng đến hành vi ra quyết định của đối tượng. Nghĩa là, một đối tượng đã tạo nhiều liên kết “*positive*” đến các đối tượng khác trong mạng xã hội sẽ có nhiều khả năng nhận lại nhiều liên kết vào là liên kết “*positive*” và ngược lại.

Trong công trình nghiên cứu [21], tác giả sử dụng 2 tỷ số được định nghĩa bởi ***công thức (2.11)*** và ***công thức (2.12)*** để thể hiện những hành vi mang tính bản chất của các đối tượng trong mạng xã hội (*tâm lý bầy đàn; thói quen và hành vi ra quyết định; quy luật nhân quả*):

(2.11)

(2.12)

Trong đó:

* : số lượng liên kết ra của đỉnh là liên kết “*positive*”.
* : số lượng liên kết ra của đỉnh là liên kết “*negative*”.
* : số lượng liên kết vào của đỉnh là liên kết “*positive*”.
* : số lượng liên kết vào của đỉnh là liên kết “*negative*”.
* : là tham số với giá trị vô cùng nhỏ đảm bảo mẫu số khác 0.

Tóm lại, đặc trưng ***PNR*** của cạnh bao gồm 2 tỷ số: (*là tỷ lệ loại liên kết ra của đỉnh* ) và (*là tỷ lệ loại liên kết vào của đỉnh* ).

1. ***Đánh giá ưu nhược điểm các loại đặc trưng***

Dựa vào quá trình xây dựng những đặc trưng trên, có thể thấy rằng bên cạnh những đặc trưng ***explicit*** truyền thống, đặc trưng ***latent*** của [23] đề xuất có nhiều nhược điểm không phù hợp với mô hình dự đoán loại liên kết “*positive*” và “*negative*”. Những nhược điểm có thể được liệt kê như sau:

* Quá trình xây dựng các ma trận đặc trưng phải trải qua nhiều bước từ khởi tạo giá trị ma trận bằng ***PLSA*** và thuật toán phân lớp đến các bước lặp tính toán để cập nhật giá trị ma trận. Do đó, mô hình dự đoán phải trả giá cao về chi phí tính toán và mất nhiều thời gian xây dựng loại đặc trưng này.
* Đặc trưng ***latent*** được biểu diễn bởi một vector đặc trưng chiều. Trong công trình [23], tác giả đề xuất sử dụng . Như vậy, đặc trưng ***latent*** có 60 chiều và lớn hơn nhiều so với các đặc trưng thông thường khác. Từ đó, dẫn đến quá trình huấn luyện dữ liệu cho mô hình máy học mất nhiều thời gian.
* Đặc trưng ***latent*** được xây dựng dựa trên 2 ma trận kề của đồ thị hỗ trợ và đồ thị mục tiêu. Đặc trưng này thể hiện mô hình chung phổ biến giữa 2 đồ thị trên. Do đó, khi đồ thị hỗ trợ và đồ thị mục tiêu có sự khác biệt lớn, đặc trưng ***latent*** sẽ trở nên tệ đi và không có đóng góp gì nhiều cho mô hình dự đoán.

Mặt khác, đặc trưng ***PNR*** của [21] lại có nhiều ưu điểm vượt trội phù hợp với mô hình dự đoán như:

* Đặc trưng ***PNR*** được xây dựng dựa trên các quy luật hành vi xã hội trong thực tế. Những quy luật này phù hợp với mô hình mạng xã hội vì phản ánh được bản chất quá trình hình thành và xu hướng phát triển của các liên kết trong mạng. Do đó, đặc trưng ***PNR*** đạt được độ ổn định cao trên các mạng xã hội khác nhau.
* Đặc trưng ***PNR*** được biểu diễn bởi vector đặc trưng 2 chiều. Quá trình cài đặt tính toán của đặc trưng này chỉ thực hiện một vài phép toán đơn giản. Do đó, đặc trưng ***PNR*** có nhiều lợi thế hơn đặc trưng ***latent*** về độ phức tạp và thời gian tính toán cho nên phù hợp với mô hình dự đoán hơn.

Bởi vì những nhược điểm của đặc trưng ***latent*** và ưu điểm vượt trội của đặc trưng ***PNR***, luận văn đề xuất sử dụng đặc trưng ***PNR*** thay thế cho đặc trưng ***latent*** trong mô hình dự đoán loại liên kết cho mạng xã hội mới thành lập. Kết quả thí nghiệm cũng chứng minh cho những phân tích trên, việc sử dụng đặc trưng ***PNR*** cho kết quả dự đoán cao, rút ngắn thời gian rút trích và huấn luyện dữ liệu so với phương pháp cũ.

1. **Phương pháp máy học**
2. ***Phương pháp transfer learning***

Phương pháp máy học theo hướng tiếp cận *transfer learning* dùng để giải quyết bài toán phân lớp có dữ liệu huấn luyện hạn chế. Ý tưởng chính của hướng tiếp cận này là tận dụng sự hỗ trợ của một tập dữ liệu lớn khác bằng cách bổ sung tập dữ liệu này vào bộ dữ liệu huấn luyện. Sau đó, phát triển thuật toán máy học phù hợp có thể tận dụng toàn bộ thông tin từ tập dữ liệu huấn luyện mới này.

Công trình nghiên cứu của *Freund and Schapire* vào năm 1997 [9] là công trình tiên phong phát triển thuật toán ***AdaBoost*** theo hướng tiếp cận *transfer learning*. Ý tưởng cổ điển của phương pháp này là xây dựng một mô hình máy học có khả năng tăng độ chính xác dự đoán của mô hình phân lớp yếu bằng cách điều chỉnh trọng số của các trường hợp đào tạo để học được mô hình phân lớp phù hợp hơn. Vector trọng số của thuật toán ***AdaBoost*** được cập nhật trong ***công thức (2.13)*** [9] sau:

(2.13)

Trong đó:

* : là vector trọng số.
* : là số lần lặp.
* : là thứ tự lặp.
* : là hệ số cập nhật.
* : là nhãn lớp đi kèm với vector đặc trưng.
* : là nhãn lớp dự đoán được từ bộ phân lớp thành phần.
* : là vector đặc trưng.
* : là tham số quy tắc.
* : là kích thước của tập dữ liệu huấn luyện.

Do hướng tiếp cận *transfer learning* là bổ sung thêm dữ liệu cho tập dữ liệu huấn luyện nên thuật toán ***AdaBoost*** cổ điển [9] sẽ gặp khó khăn khi có sự khác biệt lớn giữa dữ liệu gốc với dữ liệu bổ sung. Khi đó, mô hình máy học sẽ không cho kết quả tốt. Tác giả Dai đã phát triển thuật toán ***Transfer AdaBoost*** [5] từ nền tảng của thuật toán ***AdaBoost*** [9] để giải quyết vấn đề này. Tác giả Dai đưa ra quy tắc điều chỉnh trọng số khác nhau giữa các trường hợp đào tạo của dữ liệu gốc và dữ liệu bổ sung. Vector trọng số của thuật toán ***Transfer AdaBoost*** được cập nhật trong ***công thức (2.14)*** [5] sau:

(2.14)

Trong đó:

* : là vector trọng số.
* : là số lần lặp.
* : là thứ tự lặp.
* : là hệ số cập nhật.
* : là nhãn lớp đi kèm với vector đặc trưng.
* : là nhãn lớp dự đoán được từ bộ phân lớp thành phần.
* : là vector đặc trưng.
* : là kích thước của tập dữ liệu bổ sung.
* : là kích thước của tập dữ liệu gốc.

Tóm lại, phương pháp máy học theo hướng tiếp cận *transfer learning* có thể khắc phục được hạn chế của mô hình phân lớp yếu do tập dữ liệu huấn luyện có kích thước nhỏ gây ra. Tuy nhiên, ý tưởng của hướng tiếp cận này là bổ sung thêm dữ liệu cho tập dữ liệu huấn luyện và thực hiện phân lớp nhiều lần để điều chỉnh trọng số phù hợp cho các trường hợp đào tạo. Do đó, hạn chế của phương pháp máy học theo hướng tiếp cận này là phải trả giá cao về chi phí tính toán và thời gian huấn luyện.

1. ***Phương pháp SVM***

***Support Vector Machine*** (*SVM*) [17] là một phương pháp phân lớp dựa trên lý thuyết học thống kê được đề xuất lần đầu tiên bởi Vapnik vào năm 1995. ***SVM*** bao gồm một tập hợp các phương pháp máy học có giám sát liên quan đến nhau để *phân loại* và *phân tích hội quy*. ***SVM*** dạng chuẩn nhận dữ liệu vào và phân loại chúng vào 2 lớp khác nhau. Do đó, ***SVM*** là một thuật toán phân loại nhị phân. Hàm số tuyến tính cho bộ phân lớp nhị phân được định nghĩa trong ***công thức (2.15)*** [17] sau:

(2.15)

Trong đó:

* : là vector trọng số hay vector chuẩn của siêu phẳng phân cách.
* : là ký hiệu chuyển vị.
* : là độ lệch.
* : là vector đặc trưng.
* : là hàm ánh xạ từ không gian ban đầu sang không gian đặc trưng.

Hàm ánh xạ dữ liệu huấn luyện từ không gian ban đầu sang không gian đặc trưng còn được gọi là nhân của ***SVM***. Hiện tại, phương pháp ***SVM*** có 5 nhân phổ biến thường được sử dụng là:

* Linear
* Quadratic
* Polynomial
* RBF
* mlp

Trong đó, thuật toán máy học ***SVM*** thường sử dụng nhân RBF bởi vì so với các nhân khác thì RBF cho kết quả tốt hơn về độ chính xác cũng như thời gian thực hiện [2].

1. **Kết luận**

Trong chương này, luận văn đã tìm hiểu những cơ sở lý thuyết cần thiết để giải bài toán dự đoán loại liên kết “*positive*” và “*negative*” cho mạng xã hội.

* Trước tiên, luận văn tìm hiểu các bước hoạt động của mô hình dự đoán dấu của liên kết theo phương pháp máy học.
* Sau đó, luận văn tìm hiểu cách thức tiền xử lý dữ liệu cho quá trình rút trích đặc trưng.
* Tiếp theo, luận văn tiến hành tìm hiểu và phân tích các loại đặc trưng dùng để dự đoán loại liên kết đã được công bố trong các công trình nghiên cứu trước đây. Từ đó, luận văn lựa chọn sử dụng các loại đặc trưng phù hợp với mô hình dự đoán loại liên kết cho mạng xã hội mới thành lập.
* Cuối cùng, tìm hiểu các thuật toán máy học cần thiết để xây dựng một mô hình dự đoán loại liên kết phù hợp với mạng xã hội mới thành lập.

# Chương 3. MÔ HÌNH DỰ ĐOÁN LOẠI LIÊN KẾT “POSITIVE” VÀ “NEGATIVE”

1. **Giới thiệu**
2. ***Ý tưởng***

Vì mạng xã hội mới thành lập có thông tin dấu hiệu cạnh rất khan hiếm nên gây khó khăn trong việc xây dựng một mô hình dự đoán dấu tốt. Do đó, luận văn chọn hướng nghiên cứu theo phương pháp “*transfer learning*”. Trong hướng tiếp cận này, phương pháp ***TrAdaBoost*** [5] của Dai đã khắc phục được những nhược điểm của phương pháp ***AdaBoost*** truyền thống bằng cách tận dụng những thông tin có ích và hạn chế sự ảnh hưởng của những dữ liệu gây nhiễu.

Vì vậy, luận văn nghiên cứu phương pháp cơ sở ***TrAdaBoost*** và cải tiến phương pháp này bằng cách sử dụng ***RBFSVM*** (*SVM với nhân RBF*) như là một bộ phân lớp thành phần. Phương pháp do luận văn đề xuất gọi là **Transfer AdaBoost with SVM** (***TAS***).

Phương pháp ***TAS*** tận dụng sự hỗ trợ của một mạng xã hội khác có nhiều thông tin dấu hiệu cạnh để xây dựng một mô hình dự đoán dấu tốt cho mạng xã hội mới thành lập. Như vậy, tập dữ liệu huấn luyện theo phương pháp “*transfer learning*” bao gồm dữ liệu của 2 đồ thị mạng.

* Đồ thị mục tiêu: đồ thị của mạng xã hội mới thành lập có thông tin dấu hiệu cạnh rất khan hiếm.
* Đồ thị hỗ trợ: đồ thị của một mạng xã hội khác có nhiều thông tin dấu hiệu cạnh.

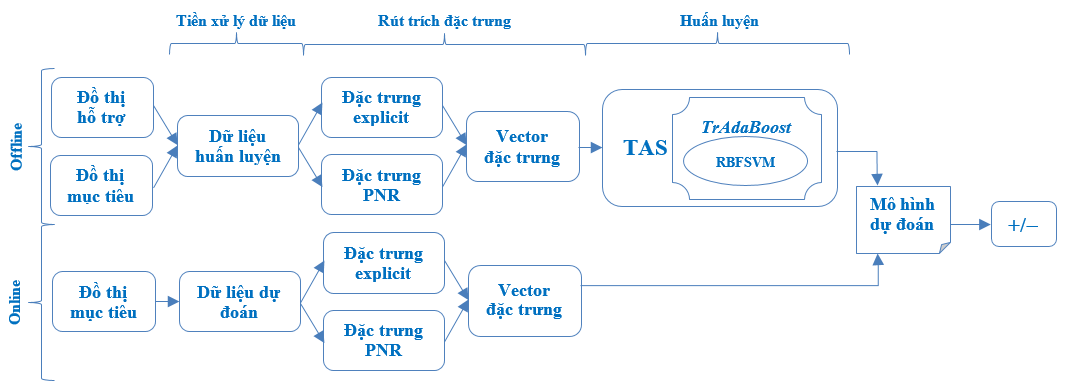
Bên cạnh đó, luận văn đã tìm hiểu và đề xuất sử dụng loại đặc trưng có tính khái quát cao và có thể áp dụng trên nhiều mạng xã hội. Trong nghiên cứu trước đây về vấn đề dự đoán dấu của liên kết trong mạng xã hội mới thành lập [23], các tác giả đề xuất sử dụng 2 loại đặc trưng: ***explicit*** và ***latent***.

* Đặc trưng ***explicit*** (*độ đo của đỉnh, độ trung tâm của đỉnh, số lượng bộ ba và số láng giềng chung*) thể hiện tính chất đặc trưng của các trường hợp cạnh.
* Đặc trưng ***latent*** thể hiện mô hình chung phổ biến giữa đồ thị hỗ trợ và đồ thị mục tiêu. Do đó, đóng góp của đặc trưng ***latent*** cho mô hình máy học sẽ trở nên không hiệu quả khi độ khác biệt giữa đồ thị hỗ trợ và đồ thị mục tiêu cao.

Trong một nghiên cứu khác về bài toán dự đoán loại liên kết trong mạng xã hội [21], tác giả đã xây dựng đặc trưng ***PNR*** để dự đoán dấu hiệu liên kết bằng cách sử dụng *mô hình hội quy logistic*. Đặc trưng ***PNR*** được xây dựng dựa trên lý thuyết ra quyết định thuộc về lĩnh vực tâm lý xã hội (*tâm lý bầy đàn; thói quen và hành vi ra quyết định; quy luật nhân quả*) nên có độ khái quát cao và có thể áp dụng trên nhiều mạng xã hội.

Vì thế, luận văn đề xuất sử dụng đặc trưng ***PNR*** thay thế cho đặc trưng ***latent*** để giải bài toán dự đoán loại liên kết trong một ngữ cảnh mới là mạng xã hội mới thành lập.

Tóm lại, ***quá trình huấn luyện*** của mô hình dự đoán loại liên kết “*positive*” và “*negative*” trong mạng xã hội mới thành lập do luận văn đề xuất sẽ có vài điểm khác biệt so với ***quá trình huấn luyện*** trong mô hình dự đoán tổng quát. Quy trình hoạt động cụ thể được biểu diễn trực quan hóa trong ***Hình 3.1*** như sau:



Hình 3.1. Mô hình dự đoán loại liên kết “*positive*” và “*negative*” cho mạng xã hội mới thành lập

1. ***Thành phần dữ liệu và các định nghĩa***

Bài toán dự đoán loại liên kết “*positive*” và “*negative*” lấy dữ liệu từ các mạng xã hội có liên kết được gán nhãn là “+” hay “”. Thành phần dữ liệu cần quan tâm cho bài toán này là: đối tượng tham gia mạng (*đỉnh*), mối quan hệ giữa các đối tượng (*cạnh/liên kết*) và thông tin dấu hiệu của liên kết (+/). Do đó, dữ liệu được lấy từ các mạng xã hội gồm các thành phần trong ***Bảng 3.1*** sau:

Bảng 3.1. Thành phần dữ liệu của bài toán dự đoán loại liên kết

|  |  |  |  |
| --- | --- | --- | --- |
| **Cạnh** | **Đỉnh bắt đầu** | **Đỉnh kết thúc** | **Dấu hiệu** |
| 1 | 0 | 1 | 1 |
| 2 | 0 | 8 | 1 |
| 3 | 0 | 102 | 1 |
| … | … | … | … |
| n | a | b | 1 |

Trong bài toán dự đoán loại liên kết, mạng xã hội được biểu diễn thông qua mô hình đồ thị có hướng. Trong mô hình dự đoán được đề xuất, mạng xã hội mới thành lập được biểu diễn thành *đồ thị mục tiêu*. Đồ thị này là đồ thị có hướng . Với:

* : là tập đỉnh của đồ thị
* : là tập cạnh đã biết dấu hiệu của đồ thị
* : là tập cạnh chưa biết dấu hiệu của đồ thị
* : là tập biểu diễn dấu hiệu cạnh của đồ thị (+1/1)

Vì thông tin dấu hiệu cạnh của mạng xã hội mới thành lập không đủ để xây dựng một mô hình dự đoán dấu tốt nên luận văn tận dụng sự hỗ trợ của một mạng xã hội có nhiều thông tin dấu hiệu cạnh khác. Mạng xã hội này được biểu diễn thông qua đồ thị có hướng . Đồ thị này được gọi là *đồ thị hỗ trợ* với:

* : là tập đỉnh của đồ thị
* : là tập cạnh đã biết dấu hiệu của đồ thị

Tập dữ liệu của *đồ thị mục tiêu* được sử dụng để làm tập dữ liệu kiểm tra. Trong khi đó, tập dữ liệu huấn luyện được xây dựng dựa trên cả 2 đồ thị trên. Với:

Mỗi trường hợp cạnh sẽ được mã hóa thông tin và biểu diễn thông qua vector đặc trưng. Mỗi vector đặc trưng sẽ có dấu hiệu là +1 hoặc 1. Các vector đặc trưng và dấu hiệu của chúng sẽ được dùng trong quá trình huấn luyện để tạo ra mô hình dự đoán.

1. **Phương pháp Transfer AdaBoost with SVM**

Từ tập dữ liệu huấn luyện được xây dựng, luận văn không áp dụng một phương pháp máy học thông thường (*ví dụ như SVM*) mà đi theo hướng tiếp cận “*transfer learning*” thông qua ***TAS***. Lý do là vì giữa *đồ thị hỗ trợ* và *đồ thị mục tiêu* tồn tại nhiều điểm khác biệt trong sự phân bố trường hợp cạnh và tính chất thông tin của dấu hiệu liên kết.

Do đó, *đồ thị hỗ trợ* không chỉ chứa những thông tin hữu ích mà còn bao gồm các thông tin gây nhiễu.

* Những thông tin hữu ích của *đồ thị hỗ trợ* giúp xây dựng một mô hình dự đoán tốt.
* Trong khi đó, những thông tin gây nhiễu thì không giúp ích gì nhiều cho mô hình dự đoán mà còn làm cho kết quả dự đoán trở nên xấu hơn.
* Vì thế, tuy tập dữ liệu huấn luyện có nhiều thông tin dấu hiệu cạnh, nhưng một mô hình dự đoán được học từ theo phương pháp máy học thông thường sẽ không cho kết quả tốt.

Yêu cầu đặt ra là cần một mô hình dự đoán có thể tận dụng được sự hỗ trợ của cả *đồ thị mục tiêu* lẫn *đồ thị hỗ trợ*. Để xây dựng được mô hình dự đoán đó, luận văn chọn hướng tiếp cận “*transfer learning*”, sử dụng ý tưởng Transfer AdaBoost của Dai [5] và AdaBoost với bộ phân lớp thành phần SVM của Li [15] để xây dựng thuật toán ***TAS***.

* Transfer AdaBoost

Ý tưởng cổ điển của phương pháp AdaBoost (*Freund and Schapire, 1997*) [9] là xây dựng một mô hình máy học có khả năng tăng độ chính xác dự đoán của mô hình phân lớp yếu bằng cách điều chỉnh trọng số của các trường hợp đào tạo để học được mô hình phân lớp phù hợp hơn. Nhưng đối với các trường hợp đào tạo của *đồ thị hỗ trợ* (), một số trường hợp có sự phân bố rất khác biệt với các trường hợp đào tạo của *đồ thị mục tiêu* (). Theo phương pháp [9], những trường hợp này vẫn được điều chỉnh trọng số giống với các trường hợp khác bởi mô hình máy học. Điều này sẽ làm cho mô hình máy học cho kết quả không tốt. Do đó, luận văn sử dụng ý tưởng ***Transfer AdaBoost*** của Dai [5] để giải quyết vấn đề này. Đối với các trường hợp của có sự phân bố khác biệt với các trường hợp của , chúng sẽ được gán trọng số nhỏ. Ngược lại, các trường hợp của có sự phân bố tương tự với các trường hợp của sẽ được gán trọng số lớn hơn nhằm thể hiện sự quan trọng của chúng.

* AdaBoost with SVM

Trong các nghiên cứu về “*transfer learning*” trước đây, bộ phân lớp thành phần thường được dùng là *Decision Tree* [7] và *Neural Networks* [20]. Tuy nhiên, vẫn còn tồn tại một số vấn đề khi sử dụng 2 bộ phân lớp thành phần trên. Đối với *Decision Tree*, các tác giả của [7] vẫn chưa đưa ra được phương pháp xác định kích thước cây phù hợp. Bộ phân lớp thành phần *Neural Networks* lại khó kiểm soát độ phức tạp nên dễ gặp tình trạng “*overfitting*”. Trong nghiên cứu gần đây của tác giả Li [15], sự phân bố độ chính xác và tính đa dạng của bộ phân lớp thành phần ***RBFSVM*** (*SVM với nhân RBF*) với bộ tham số được điều chỉnh giá trị thích hợp cho kết quả đầy hứa hẹn. Với ý nghĩa của các giá trị tham số như sau:

* : tham số kiểm soát độ phức tạp và quá trình huấn luyện bị lỗi.
* : tham số tự do của hàm số cơ sở Gaussian.

Công trình nghiên cứu của Li [15] chứng minh rằng phương pháp “*transfer learning*” sử dụng bộ phân lớp thành phần ***RBFSVM*** cho kết quả khả quan và vượt trội hơn *Decision Tree* và *Neural Networks*. Vì vậy, luận văn kết hợp thiết kế ***RBFSVM*** trở thành bộ phân lớp thành phần.

* Vì thế, thuật toán ***TAS*** do luận văn đề xuất là sự kết hợp của thuật toán ***Transfer AdaBoost*** sử dụng ***RBFSVM*** làm bộ phân lớp thành phần. Mã giả thuật toán ***TAS*** được trình bày chi tiết trong ***Hình 3.2*** sau:

TRANSFER ADABOOST WITH SVM (TAS) Algorithm

//input: Hai tập dữ liệu đã biết dấu hiệu cạnh và , tập dữ liệu chưa biết dấu hiệu cạnh , bộ phân lớp thành phần RBFSVM và số lần lặp .

//output: Mô hình dự đoán .

1. Cho ,
2. Khởi tạo vector trọng số . Giá trị khởi tạo của vector trọng số có thể tùy chỉnh phù hợp với từng trường hợp khác nhau.
3. for do
4. Gán
5. Gọi bộ phân lớp thành phần RBFSVM (được học từ bộ dữ liệu huấn luyện với vector trọng số ) để dự đoán cho 2 tập dữ liệu và , .
6. Tính toán lỗi của trên :
7. Nếu thì chọn các thông số thích hợp mới ( và ) theo [15]. Sau đó, quay lại bước 5.
8. Gán ,
9. Cập nhật vector trọng số :

Hình 3.2. Mã giả thuật toán TAS

Giải thích mã giả thuật toán TAS trên:

* Dòng 1: tìm kích thước của 2 tập dữ liệu và .
* Dòng 2: khởi tạo vector trọng số với các giá trị được dùng để lưu trọng số của các cạnh trong , còn lưu trọng số của các cạnh trong .
* Dòng 3  9: là quá trình sử dụng RBFSVM để dự đoán, tính toán lỗi và cập nhật vector trọng số .
  + Quá trình tính toán lỗi dựa vào và của cạnh . Với:
    - : dấu hiệu của cạnh được dự đoán bởi .
    - : dấu hiệu của cạnh trong đồ thị.
  + Quy tắc cập nhật vector trọng số:
    - Do *đồ thị hỗ trợ* chỉ đóng vai trò tương trợ cho *đồ thị mục tiêu* nên trọng số của trường hợp cạnh trong *đồ thị hỗ trợ* luôn luôn nhỏ hơn trọng số của trường hợp cạnh trong *đồ thị mục tiêu*.
      * Do đó, trong *đồ thị mục tiêu*, trọng số của mỗi cạnh bất kỳ luôn luôn tăng bởi hệ số .
      * Ngược lại, trong *đồ thị hỗ trợ*, trọng số của mỗi cạnh bất kỳ luôn luôn giảm bởi hệ số .
    - Sau lần lặp, những cạnh thuộc *đồ thị hỗ trợ* có sự phân bố tương tự với những cạnh thuộc *đồ thị mục tiêu* sẽ có trọng số lớn hơn những cạnh có sự phân bố khác biệt.

1. **Kết luận**

Trong chương này, luận văn đã giới thiệu ý tưởng tổng quan và cơ sở để xây dựng mô hình dự đoán loại liên kết “*positive*” và “*negative*” cho mạng xã hội mới thành lập. Luận văn đã trình bày chi tiết các bước:

* Xây dựng tập dữ liệu huấn luyện.
* Rút trích loại đặc trưng phù hợp với mô hình dự đoán.
* Thuật toán TAS sử dụng cho quá trình huấn luyện.

Dựa vào nền tảng khoa học đã phân tích trên, mô hình dự đoán loại liên kết cho mạng xã hội mới thành lập do luận văn đề xuất giúp cải thiện độ chính xác dự đoán và giảm thời gian rút trích đặc trưng, huấn luyện và dự đoán.

# Chương 4. KẾT QUẢ THỰC NGHIỆM VÀ ĐÁNH GIÁ

1. **Dữ liệu thực nghiệm**

Luận văn sử dụng dữ liệu của 3 mạng xã hội trực tuyến Epinions, Slashdot và Wiki để tiến hành thực nghiệm. Dữ liệu của 3 mạng xã hội này được tải về từ *Stanford Large Network Dataset Collection[[5]](#footnote-6)*. Bởi vì dữ liệu gốc của 3 mạng xã hội này quá lớn và thưa thớt nên luận văn thực hiện tiền xử lý dữ liệu theo [23]. Nghĩa là, trong tất cả các đỉnh của đồ thị mạng Epinions, luận văn chọn 19.987 đỉnh có độ đo cao nhất. Tương tự với cách trên, luận văn chọn 15.999 đỉnh từ Slashdot và 6.998 đỉnh từ Wiki.

Dữ liệu thực nghiệm bao gồm tập đỉnh được chọn và các liên kết (*cạnh*) được tạo bởi tập đỉnh đó. Số liệu thống kê dữ liệu thực nghiệm của 3 mạng xã hội được thể hiện trong ***Bảng 4.1*** sau:

Bảng 4.1. Thống kê dữ liệu thực nghiệm của 3 mạng xã hội

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Số đỉnh** | **Số cạnh** | **Số cạnh “positive”** | **Số cạnh “negative”** | **(%) Cạnh “positive”** |
| **Epinions** | 19.987 | 634.209 | 555.601 | 78.608 | 87,6 |
| **Slashdot** | 15.999 | 371.122 | 283.993 | 87.129 | 76,5 |
| **Wiki** | 6.998 | 113.844 | 83.832 | 30.012 | 73,6 |

Từ số liệu thống kê trong ***Bảng 4.1***, luận văn có thể kết luận rằng tỷ lệ liên kết “*positive*” luôn chiếm xấp xỉ 80% trong cả 3 mạng. Luận văn khắc phục tình trạng mất cân bằng giữa số lượng liên kết “*positive*” và liên kết “*negative*” này bằng cách áp dụng phương pháp của tác giả Guha [11]. Số liệu thống kê dữ liệu thực nghiệm của 3 mạng xã hội sau khi cân bằng được thể hiện trong ***Bảng 4.2*** sau:

Bảng 4.2. Thống kê dữ liệu cân bằng của 3 mạng xã hội

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Số đỉnh** | **Số cạnh** | **Số cạnh “positive”** | **Số cạnh “negative”** |
| **Epinions** | 18.960 | 157.216 | 78.608 | 78.608 |
| **Slashdot** | 15.965 | 174.258 | 87.129 | 87.129 |
| **Wiki** | 5.432 | 46.124 | 23.062 | 23.062 |

Luận văn sẽ tiến hành thực nghiệm trên từng cặp đồ thị mạng đôi một được lấy từ 3 mạng xã hội trên. Trong cặp đồ thị mạng xã hội đó, một đồ thị mạng được xem là đồ thị hỗ trợ và đồ thị còn lại là đồ thị mục tiêu. Sau đó, tiến hành hoán vị đồ thị hỗ trợ và đồ thị mục tiêu. Như vậy, luận văn sẽ tiến hành thực nghiệm trên 6 cặp đồ thị mạng xã hội (*đồ thị hỗ trợ  đồ thị mục tiêu*): Slashdot – Epinions, Wiki – Epinions, Epinions – Slashdot, Wiki – Slashdot, Epinions – Wiki, Slashdot – Wiki.

Trong quá trình thực nghiệm, luận văn sử dụng phương pháp ***4-fold cross validation*** tương tự như [23] để huấn luyện và kiểm tra trên từng cặp đồ thị hỗ trợ và đồ thị mục tiêu. Trong đó:

* Đồ thị mục tiêu là đồ thị của mạng xã hội mới thành lập có thông tin dấu hiệu cạnh rất khan hiếm. Do đó, dựa theo [23] để giả định thành mạng xã hội mới thành lập, luận văn chỉ lấy 10% dữ liệu làm tập dữ liệu huấn luyện.
* Đồ thị hỗ trợ là đồ thị của mạng xã hội có nhiều thông tin dấu hiệu cạnh nên toàn bộ dữ liệu đều được sử dụng làm tập dữ liệu huấn luyện.

Cụ thể, với đồ thị mục tiêu, luận văn chia dữ liệu thành 4 phần bằng nhau, mỗi phần đều có sự phân bố cân bằng về số lượng liên kết “***positive***” và “***negative***”. Một phần dữ liệu sẽ được chọn làm tập dữ liệu kiểm tra . Sau đó, chọn ngẫu nhiên 10% dữ liệu trong 3 phần dữ liệu còn lại làm tập dữ liệu . Tập dữ liệu kết hợp với dữ liệu của đồ thị hỗ trợ sẽ trở thành bộ dữ liệu huấn luyện. Luận văn tiến hành thực nghiệm lần lượt trên 4 tập dữ liệu kiểm tra được chia từ dữ liệu của đồ thị mục tiêu để lấy kết quả dự đoán của từng phần, sau đó tính kết quả dự đoán trung bình.

1. **Các độ đo đánh giá**

Để đánh giá hiệu quả của phương pháp dự đoán đề xuất, luận văn sử dụng độ đo *accuracy*. Đây là độ đo cơ bản thường được dùng để đánh giá hiệu suất dự đoán của một phương pháp.

Công thức tính độ đo *accuracy[[6]](#footnote-7)* được thể hiện trong ***công thức (4.1)*** như sau:

 (4.1)

Trong đó:

* : số lượng các liên kết “*positive*” được gán nhãn đúng.
* : số lượng các liên kết “*negative*” được gán nhãn đúng.
* : số lượng các liên kết “*positive*” bị gán nhãn sai.
* : số lượng các liên kết “*negative*” bị gán nhãn sai.

Bên cạnh độ đo thông dụng *accuracy*, luận văn sử dụng thêm đường cong Precision-Recall (*PR*) [6]. Bởi vì, nếu chỉ sử dụng duy nhất độ đo *accuracy* thì chưa đủ. Trong trường hợp thuật toán học bị lệch thì độ đo *accuracy* không phản ánh đầy đủ hiệu năng của phương pháp dự đoán. *Đường cong PR* cung cấp thêm thông tin hình ảnh về hiệu suất dự đoán của phương pháp. Ngoài ra, luận văn còn tiến hành so sánh thời gian rút trích đặc trưng, huấn luyện và kiểm tra dữ liệu giữa các phương pháp thực nghiệm.

1. **Các phương pháp thực nghiệm**

Để đánh giá hiệu quả dự đoán, luận văn tiến hành cài đặt các phương pháp thực nghiệm trên môi trường MATLAB. Tất cả các thực nghiệm của luận văn đều được thực hiện trên hệ thống máy ảo có cấu hình như nhau là 2.90 GHz CPU và 12G RAM. Luận văn tiến hành cài đặt thực nghiệm 4 phương pháp cơ sở (*Target, Combine + Latent [*23*], TAS + Latent, Combine + PNR*) dùng để so sánh với phương pháp luận văn đề xuất (*TAS + PNR*).

Dựa vào các phương pháp thực nghiệm này, luận văn sẽ tiến hành so sánh hiệu quả khi sử dụng phương pháp máy học SVM thông thường với phương pháp TAS mà luận văn đề xuất. Đồng thời, luận văn cũng so sánh hiệu quả khi sử dụng các loại đặc trưng được đề xuất bởi [23] với các loại đặc trưng mà luận văn đề xuất sử dụng.

Chi tiết cài đặt các phương pháp thực nghiệm được trình bày chi tiết trong ***Bảng 4.3*** sau:

Bảng 4.3. Các phương pháp thực nghiệm (  có,   không)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Phương pháp thực nghiệm** | **Dữ liệu thực nghiệm** | | **Đặc trưng** | | **Phương pháp máy học** | |
| **Đồ thị hỗ trợ** | **Đồ thị mục tiêu** | **Explicit + Latent** | **Explicit + PNR** | **SVM (RBF)** | **TAS** |
| **Target** | **** | **** | **** | **** | **** | **** |
| **Combine + Latent** [23] | **** | **** | **** | **** | **** | **** |
| **TAS + Latent** | **** | **** | **** | **** | **** | **** |
| **Combine + PNR** | **** | **** | **** | **** | **** | **** |
| **TAS + PNR** | **** | **** | **** | **** | **** | **** |

Ba phương pháp đầu (*Target, Combine + Latent, TAS + Latent*) đều sử dụng những đặc trưng cũ được đề xuất bởi [23], chỉ khác nhau về tập dữ liệu huấn luyện và phương pháp máy học. Dựa vào 3 phương pháp này, luận văn tiến hành so sánh hiệu suất dự đoán khi:

* Chỉ sử dụng dữ liệu của đồ thị mục tiêu để xây dựng tập huấn luyện (*Target*).
* Tận dụng thêm dữ liệu của đồ thị hỗ trợ để huấn luyện với phương pháp máy học SVM thông dụng (*Combine + Latent được đề xuất bởi [*23*]*).
* Sử dụng dữ liệu của đồ thị hỗ trợ và đồ thị mục tiêu để huấn luyện với thuật toán TAS (*TAS + Latent*).
* Từ đó, luận văn có thể đánh giá hiệu quả dự đoán của phương pháp TAS do luận văn đề xuất với các phương pháp truyền thống khác.

Hai phương pháp cuối cùng (*Combine + PNR và TAS + PNR*) dùng các loại đặc trưng mà luận văn đề xuất sử dụng. Cụ thể, là sử dụng kết hợp 2 loại đặc trưng *explicit* và *PNR* thay vì sử dụng *explicit* và *latent* được đề xuất bởi [23].

* Từ đó, luận văn có thể đánh giá hiệu quả dự đoán khi dùng các loại đặc trưng được đề xuất bởi [23] với các loại đặc trưng mà luận văn đề xuất sử dụng.

Luận văn sử dụng độ đo *accuracy* và *đương cong PR* để so sánh hiệu quả dự đoán của 4 phương pháp cơ sở với phương pháp luận văn đề xuất (*TAS + PNR*). Ngoài ra, để so sánh hiệu quả dự đoán khi thay thế đặc trưng *latent* bởi đặc trưng *PNR*, luận văn sẽ tiến hành so sánh thời gian rút trích đặc trưng, huấn luyện và kiểm tra dữ liệu giữa các phương pháp: *Combine + Latent* với *Combine + PNR*, *TAS + Latent* với *TAS + PNR*.

1. **Kết quả thực nghiệm**

Phương pháp luận văn đề xuất (***TAS + PNR***) được đánh giá qua 3 tiêu chí: ***accuracy***, ***đường cong PR*** và ***thời gian thực hiện*** (*thời gian rút trích đặc trưng, huấn luyện và kiểm tra dữ liệu*). Trong quá trình thực nghiệm, phương pháp ***TAS + PNR*** được so sánh với 4 phương pháp cơ sở trên 6 cặp (*đồ thị hỗ trợ  đồ thị mục tiêu*): Slashdot – Epinions, Wiki – Epinions, Epinions – Slashdot, Wiki – Slashdot, Epinions – Wiki và Slashdot – Wiki.

1. ***So sánh độ đo accuracy***

Độ chính xác dự đoán (*độ đo accuracy*) trung bình của các phương pháp *Target*, *Combine + Latent*, *TAS + Latent*, *Combine + PNR*, *TAS + PNR* trên 6 cặp đồ thị được thể hiện đầy đủ trong ***Bảng 4.4*** sau:

Bảng 4.4. Độ đo accuracy (%) trung bình của các phương pháp thực nghiệm (E – Epinions, S – Slashdot, W – Wiki)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Phương pháp** | **E – S** | **E – W** | **S – E** | **S – W** | **W – E** | **W – S** |
| Target | 50.39 | 50.77 | 57.24 | 50.01 | 57.24 | 51.64 |
| Combine + Latent | 53.38 | 52.78 | 62.59 | 51.31 | 62.41 | 54.76 |
| TAS + Latent | 61.41 | 53.25 | 65.32 | 52.25 | 64.74 | 56.63 |
| Combine + PNR | 83.88 | 80.17 | 91.78 | 84.50 | 89.11 | 86.98 |
| TAS + PNR | **87.85** | **85.34** | **93.49** | **87.06** | **93.17** | **90.01** |

Độ chính xác dự đoán trung bình của các phương pháp *Target*, *Combine + Latent*, *TAS + Latent*, *Combine + PNR*, *TAS + PNR* trên 6 cặp đồ thị với 4 tập dữ liệu kiểm tra khác nhau được biểu diễn thành biểu đồ cột trong ***Hình 4.1***.

**Hình 4.**1**.** Độ đo accuracy trung bình

Dựa vào các kết quả đạt được trên, luận văn có thể kết luận rằng phương pháp *TAS + PNR* cho độ chính xác dự đoán cao nhất trên 6 cặp đồ thị trong tất cả các trường hợp. Cụ thể, phương pháp *TAS + PNR* giúp tăng độ chính xác dự đoán thêm 40% so với phương pháp cơ sở (*Target*).

Trong cặp đồ thị đầu tiên, *Epinions* là đồ thị hỗ trợ và *Slashdot* là đồ thị mục tiêu. Luận văn sẽ tiến hành so sánh và phân tích các phương pháp theo 2 tiêu chí:

* Tiêu chí 1: đánh giá hiệu quả dự đoán khi tận dụng thêm dữ liệu của đồ thị hỗ trợ và áp dụng thuật toán *TAS*.
* Tiêu chí 2: so sánh hiệu quả dự đoán khi thay thế đặc trưng *latent* bởi đặc trưng *PNR*.

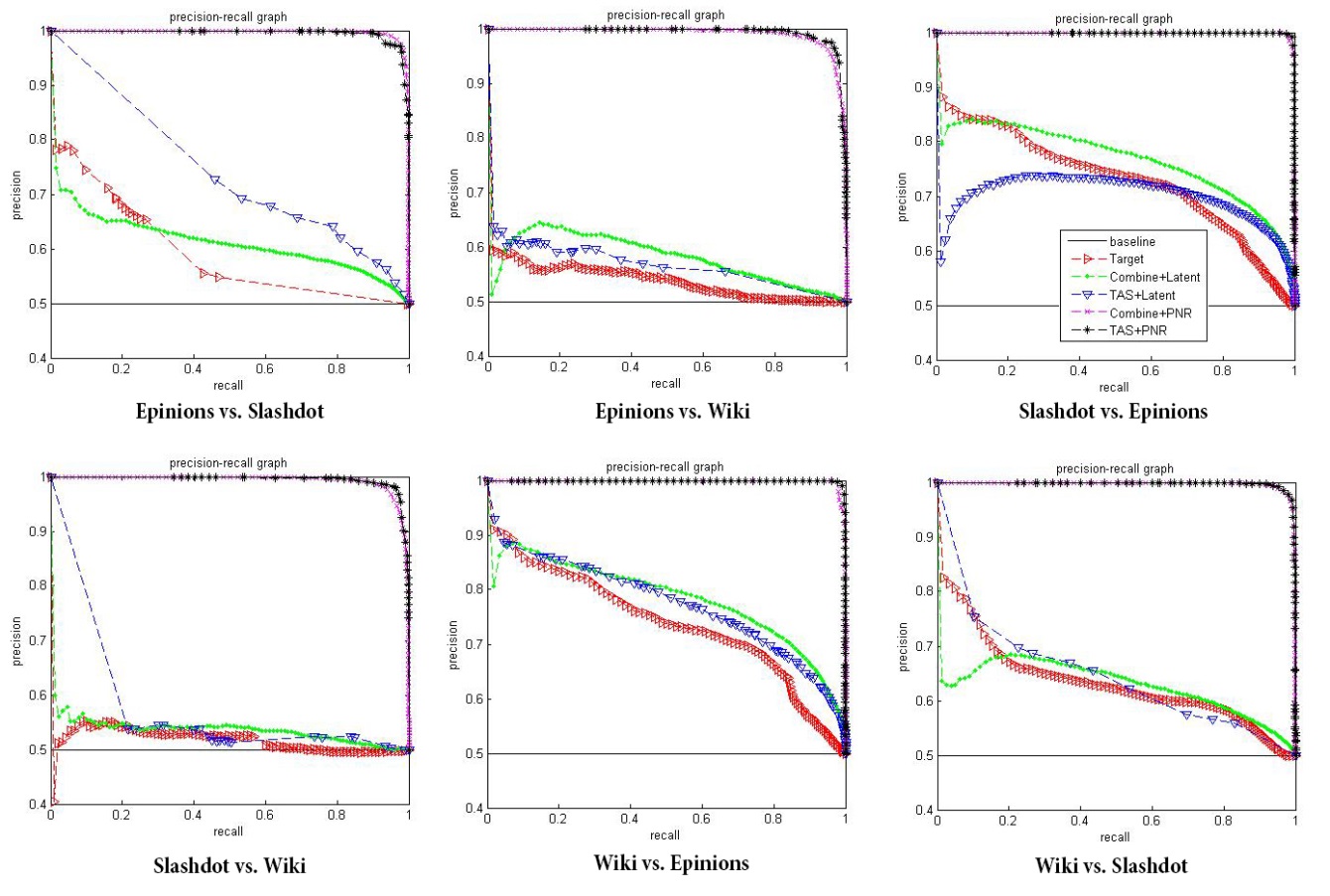
Những phân tích và đánh giá cụ thể của luận văn như sau:

* Tiêu chí 1: phương pháp *Target* cho độ chính xác dự đoán tệ nhất bởi vì phương pháp này chỉ sử dụng lượng dữ liệu khan hiếm của đồ thị mục tiêu để xây dựng tập huấn luyện. Phương pháp *Combine + Latent* cho độ chính xác dự đoán cao hơn phương pháp *Target* vì đã kết hợp thêm dữ liệu của đồ thị hỗ trợ để xây dựng tập huấn luyện. Mặc dù, tập huấn luyện đã có nhiều thông tin hơn nhưng vẫn không cho kết quả tốt nhất. Bởi vì sự phân bố khác nhau giữa đồ thị hỗ trợ và đồ thị mục tiêu nên không phải tất cả dữ liệu của đồ thị hỗ trợ cũng có ích, một phần dữ liệu của đồ thị hỗ trợ gây nhiễu và không giúp ích gì cho mô hình dự đoán. Do đó, phương pháp *TAS + Latent* cho kết quả tốt hơn phương pháp *Combine + Latent* khi tận dụng được những dữ liệu có ích của đồ thị hỗ trợ và hạn chế sự ảnh hưởng của những dữ liệu gây nhiễu.
* Tiêu chí 2: thuật toán *TAS* đã khắc phục được nhược điểm của thuật toán máy học thông thường nhưng kết quả dự đoán vẫn chưa thật sự tốt vì những khuyết điểm của đặc trưng *latent* được sử dụng trong mô hình dự đoán. Đặc trưng *latent* trở nên tệ và không giúp ích gì nhiều cho mô hình dự đoán khi sự khác biệt giữa đồ thị hỗ trợ và đồ thị mục tiêu lớn. Do đó, luận văn đề xuất sử dụng thay thế đặc trưng *latent* bởi đặc trưng *PNR* có độ khái quát cao hơn và có thể áp dụng trên nhiều mạng xã hội. Vì những ưu điểm vượt trội của đặc trưng *PNR* so với đặc trưng *latent* nên 2 phương pháp *Combine + PNR* và *TAS + PNR* cho kết quả cao hơn rất nhiều so với *Combine + Latent* và *TAS + Latent*.

Cùng với thuật toán *TAS* được đề xuất, luận văn tiến hành nghiên cứu và lựa chọn đặc trưng *PNR* thay thế đặc trưng *latent*. Phương pháp *TAS + PNR* được đề xuất trong luận văn cho độ chính xác dự đoán tốt nhất so với các phương pháp cơ sở. Luận văn có thể đưa ra những giải thích tương tự cho các trường hợp còn lại.

1. ***So sánh đường cong PR***

*Đường cong PR* trung bình của các phương pháp thí nghiệm *Target*, *Combine + Latent*, *TAS + Latent*, *Combine + PNR*, *TAS + PNR* trên 6 cặp đồ thị được biểu diễn trong ***Hình 4.2***.



**Hình 4.**2**.** Đường cong PR trung bình

Dựa vào *đường cong PR* được biểu diễn trong các hình trên, luận văn có thể kết luận rằng phương pháp *TAS + PNR* cho hiệu suất dự đoán tốt nhất và ổn định. Bởi vì, *đường cong PR* của phương pháp *TAS + PNR* chiếm diện tích lớn nhất.

Ngoài ra, do những tính chất vượt trội của đặc trưng *PNR* so với đặc trưng *latent* nên *đường cong PR* của 2 phương pháp *Combine + PNR* và *TAS + PNR* tốt hơn rất nhiều so với 3 phương pháp *Target*, *Combine + Latent* và *TAS + Latent*. Điều này chứng tỏ rằng đặc trưng *PNR* phù hợp với mô hình dự đoán loại liên kết hơn đặc trưng *latent*.

Bởi vì, đặc trưng *latent* thể hiện mô hình chung phổ biến giữa 2 đồ thị mạng xã hội. Do đó, khi cấu trúc của 2 đồ thị mạng xã hội có nhiều khác biệt, đóng góp của đặc trưng *latent* sẽ thấp và làm cho thuật toán máy học bị lệch. Vì thế, *đường cong PR* của phương pháp *TAS + Latent* có đôi khi thấp hơn phương pháp *Combine + Latent*.

1. ***So sánh thời gian thực hiện***

Để đánh giá hiệu quả dự đoán khi dùng các loại đặc trưng được đề xuất bởi [23] với các loại đặc trưng của luận văn đề xuất (*cụ thể là thay thế đặc trưng latent bởi đặc trưng PNR*), luận văn tiến hành so sánh thời gian thực hiện (*thời gian rút trích đặc trưng, huấn luyện và kiểm tra dữ liệu*) của các phương pháp.

* Đầu tiên, luận văn sẽ so sánh thời gian rút trích đặc trưng trung bình theo 2 cách: đặc trưng *explicit* kết hợp với đặc trưng *latent* [23], đặc trưng *explicit* kết hợp với đặc trưng *PNR* (*luận văn đề xuất sử dụng*). ***Bảng 4.5*** sau thể hiện thời gian rút trích đặc trưng trung bình.

Bảng 4.5. Thời gian thực hiện trung bình (phút) của quá trình rút trích đặc trưng (E – Epinions, S – Slashdot, W – Wiki)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **E – S** | **E – W** | **S – E** | **S – W** | **W – E** | **W – S** |
| **Explicit + Latent** | 196.71 | 147.70 | 90.01 | 60.64 | 43.48 | 62.51 |
| **Explicit + PNR** | **141.87** | **116.04** | **48.80** | **34.22** | **22.87** | **34.12** |

* Sau đó, luận văn sẽ so sánh thời gian huấn luyện và kiểm tra dữ liệu trung bình với 2 phương pháp máy học: SVM thông dụng và TAS do luận văn đề xuất. ***Bảng 4.6*** sau thể hiện thời gian huấn luyện và kiểm tra dữ liệu trung bình với SVM. Tương tự, ***Bảng 4.7*** thể hiện thời gian của phương pháp TAS.

Bảng 4.6. Thời gian thực hiện trung bình (phút) của quá trình huấn luyện và kiểm tra dữ liệu với SVM (E – Epinions, S – Slashdot, W – Wiki)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **E – S** | **E – W** | **S – E** | **S – W** | **W – E** | **W – S** |
| **Combine + Latent** | 635.32 | 867.72 | 3586.28 | 198.74 | 1523.24 | 846.33 |
| **Combine + PNR** | **10.58** | **8.83** | **17.93** | **19.80** | **2.64** | **2.55** |

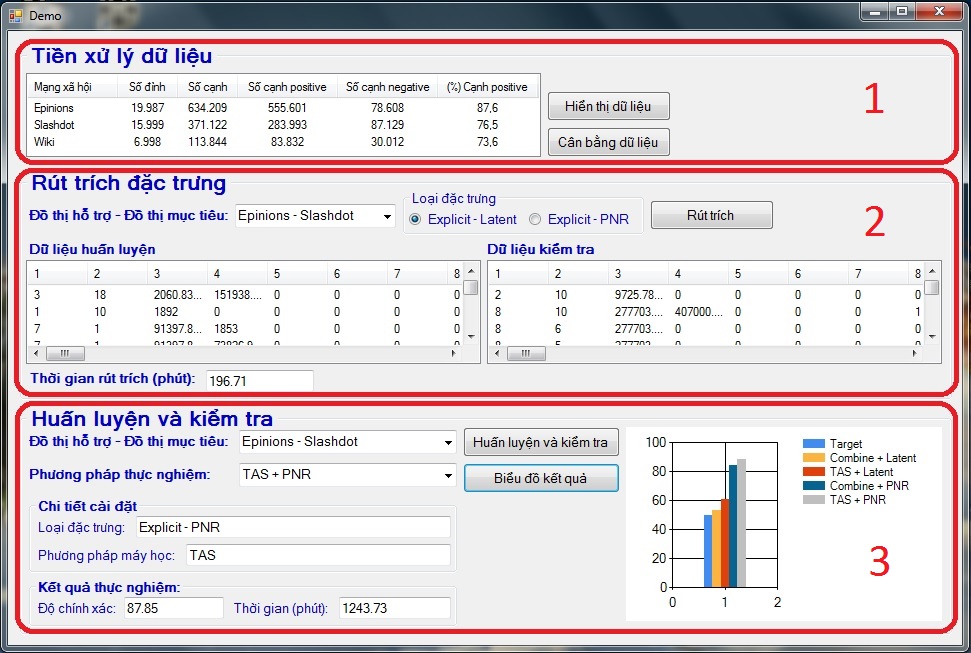
Bảng 4.7. Thời gian thực hiện trung bình (phút) của quá trình huấn luyện và kiểm tra dữ liệu với TAS (E – Epinions, S – Slashdot, W – Wiki)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **E – S** | **E – W** | **S – E** | **S – W** | **W – E** | **W – S** |
| **TAS + Latent** | 5832.03 | 3549.95 | 7271.38 | 5367.53 | 939.63 | 1100.59 |
| **TAS + PNR** | **1243.73** | **762.93** | **1865.63** | **1089.94** | **268.72** | **242.11** |

Dựa vào số liệu của tất cả các bảng thể hiện thời gian thực hiện trên, luận văn có thể kết luận rằng khi thay thế đặc trưng *latent* bởi đặc trưng *PNR* thì tốc độ rút trích đặc trưng, huấn luyện và kiểm tra dữ liệu trở nên nhanh hơn. Thời gian thực hiện giảm được rất nhiều bởi vì so với đặc trưng *latent* thì đặc trưng *PNR* có 2 ưu điểm nổi bật là quá trình tính toán xây dựng đặc trưng đơn giản và số chiều đặc trưng thấp.

1. **Chương trình minh họa**

Quá trình thực nghiệm giải bài toán dự đoán loại liên kết “*positive*” và “*negative*” cho mạng xã hội được biểu diễn trực quan trong chương trình minh họa. Giao diện của chương trình minh họa được thể hiện trong ***Hình 4.3*** sau:



**Hình 4.**3**.** Chương trình minh họa

Giao diện của chương trình gồm 3 phần thể hiện các bước cần thiết để thực hiện thực nghiệm:

* Phần 1: minh họa bước tiền xử lý dữ liệu thực nghiệm của 3 mạng xã hội Epinions, Slashdot và Wiki.
* Phần 2: minh họa bước rút trích các loại đặc trưng từ dữ liệu huấn luyện và dữ liệu kiểm tra. Những đặc trưng này là dữ liệu đầu vào cho mô hình máy học.
* Phần 3: minh họa bước huấn luyện và kiểm tra dữ liệu của mô hình máy học. Từ đó, luận văn sẽ thu được kết quả dự đoán của các phương pháp thực nghiệm.

1. **Kết luận**

Dựa vào tất cả các kết quả thực nghiệm trên, luận văn đã khẳng định tính hiệu quả của phương pháp *TAS + PNR* với bài toán dự đoán loại liên kết trong mạng xã hội mới thành lập. Cụ thể hơn, phương pháp *TAS + PNR* giúp cải thiện độ chính xác dự đoán 40% so với phương pháp cơ sở. Đồng thời, giảm thiểu thời gian rút trích đặc trưng, huấn luyện và kiểm tra dữ liệu của mô hình dự đoán khi thay thế đặc trưng *latent* bởi đặc trưng *PNR*. Phương pháp *TAS + PNR* có thể giải quyết được các vấn đề sau:

* Khắc phục được việc khan hiếm thông tin của dữ liệu huấn luyện trong mạng xã hội mới thành lập.
* Hạn chế sự ảnh hưởng của những dữ liệu gây nhiễu trong đồ thị hỗ trợ.
* Lựa chọn đặc trưng phù hợp giúp cải thiện độ chính xác dự đoán cũng như thời gian thực hiện của quá trình rút trích đặc trưng, huấn luyện và kiểm tra dữ liệu trong mô hình dự đoán.

Do đó, phương pháp *TAS + PNR* do luận văn đề xuất là một phương pháp có giá trị, độ tương thích cao và phù hợp giải quyết bài toán dự đoán loại liên kết cho mạng xã hội mới thành lập.

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## Kết quả đạt được

Bài toán dự đoán loại liên kết trong mạng xã hội là một bài toán thú vị có nhiều thách thức và khả năng ứng dụng thực tế trong thời đại bùng nổ của mạng xã hội. Tuy nhiên, bởi vì những đặc trưng khá chuyên biệt của dữ liệu trong bài toán dự đoán loại liên kết nên bài toán này có nhiều điểm khác biệt so với các bài toán con về dự đoán liên kết trong mạng xã hội. Vì thế, đây là một hướng nghiên cứu khá mới mẻ với nhiều thách thức. Phần lớn các nghiên cứu trước đây về bài toán dự đoán loại liên kết “*positive*” và “*negative*” trong mạng xã hội đều có yêu cầu cao về thông tin dấu hiệu cạnh cho dữ liệu đầu vào. Do đó, luận văn xem xét bài toán trong một ngữ cảnh thực tế hơn là dự đoán loại liên kết trong mạng xã hội mới thành lập. Trong mạng xã hội mới thành lập, thông tin dấu hiệu cạnh có sẵn rất khan hiếm và gặp nhiều khó khăn trong việc xây dựng một bộ phân lớp dấu hiệu cạnh tốt. Để giải quyết vấn đề này, luận văn tìm hiểu những phương pháp của các nghiên cứu liên quan nhằm đề xuất một phương pháp mới. Cụ thể, luận văn đã hoàn thành những công việc sau:

* Tìm hiểu về mô hình dự đoán loại liên kết của các công trình nghiên cứu tiêu biểu đã được công bố. Thông qua quá trình phân tích và đánh giá, luận văn xác định những vấn đề và thách thức còn tồn tại của bài toán để tập trung giải quyết.
* Đề xuất mô hình dự đoán loại liên kết phù hợp với dữ liệu của mạng xã hội mới thành lập. Luận văn xây dựng phương pháp *TAS* theo hướng nghiên cứu “*transfer learning*” cho mô hình dự đoán nhằm khắc phục tình trạng khan hiếm dữ liệu trong mạng xã hội mới thành lập. Bên cạnh đó, luận văn còn tìm hiểu và đề xuất sử dụng loại đặc trưng có tính khái quát cao thay thế cho loại đặc trưng cũ trước đây. Mô hình dự đoán được đề xuất trong luận văn giúp nâng cao độ chính xác dự đoán. Đồng thời, rút ngắn thời gian rút trích đặc trưng, huấn luyện và kiểm tra dữ liệu.
* Tiến hành cài đặt và thực hiện nhiều thực nghiệm chứng minh những ưu điểm của phương pháp đề xuất so với các phương pháp mới hiện tại.

Kết quả nghiên cứu của luận văn được chấp nhận đăng trong kỷ yếu hội nghị khoa học quốc tế KES 2015 (*The Nineteenth International Conference on Knowledge-Based and Intelligent Information & Engineering Systems*): Anh-Thu Nguyen-Thi, Phuc Quang Nguyen, Thanh Duc Ngo, Tu-Anh Nguyen-Hoang, “Transfer AdaBoost SVM for Link Prediction in Newly Signed Social Networks using Explicit and PNR Features”, 7-9/9/2015, Singapore.

## Hướng phát triển

Dựa vào những kết quả đạt được cũng như các phân tích trong quá trình nghiên cứu và thực nghiệm, trong tương lai, đề tài có thể phát triển theo những hướng sau:

* Tìm hiểu và nghiên cứu phát triển một phiên bản cải tiến của phương pháp *TAS* để giảm bớt thời gian và chi phí tính toán cho quá trình máy học.
* Hiện tại, những nghiên cứu của luận văn được thực hiện với dữ liệu có dấu hiệu cạnh là (+1/1). Hướng nghiên cứu tiếp theo có thể là dự đoán loại liên kết có trọng số, với trọng số thể hiện mức độ liên kết.
* Nghiên cứu và tận dụng thêm những thông tin khác của mạng xã hội vào mô hình dự đoán và có thể mở rộng bài toán sang hướng dự đoán ngữ nghĩa của liên kết.

# DANH MỤC CÔNG BỐ KHOA HỌC CỦA TÁC GIẢ

1. Anh-Thu Nguyen-Thi, Phuc Quang Nguyen, Thanh Duc Ngo, Tu-Anh Nguyen-Hoang (2015), “Transfer AdaBoost SVM for Link Prediction in Newly Signed Social Networks using Explicit and PNR Features”, Paper has been accepted for publication in *Proceedings of* *The Nineteenth International Conference on Knowledge-Based and Intelligent Information & Engineering Systems*, 7-9/9/2015, Singapore.

# TÀI LIỆU THAM KHẢO

**Tiếng Anh**

1. [Amin Javari](http://dl.acm.org/author_page.cfm?id=87958775357&CFID=523576734&CFTOKEN=94719529), [Mahdi Jalili](http://dl.acm.org/author_page.cfm?id=81337489888&CFID=523576734&CFTOKEN=94719529) (2014), “Cluster-Based Collaborative Filtering for Sign Prediction in Social Networks with Positive and Negative Links”, *ACM Transactions on Intelligent Systems and Technology (TIST)*.

2. Ben-Hur A., Weston J. (2010), “A User’s Guide to Support Vector Machines”, *Methods in Molecular Biology*, 609: 223-239.

3. Cartwright D., Harary F. (1956), “Structure balance: A generalization of Heider’s theory”, Psych. Rev. 63.

4. Chiang K., Nagarajan N., Tewari A., Dhillon I.S. (2011), “Exploiting longer cycles for link prediction in signed networks”, In *Proceedings of the 20th ACM Conference on Information and Knowledge Management*, 1157-1162.

5. Dai W., Yang Q., Xue G., Yu Y. (2007), “Boosting for Transfer Learning”, *Proc. 24th Int’l Conf. Machine Learning*, pp. 193-200.

6. Davis J., Goadrich M. (2006), “The relationship between precision-recall and roc curves”, *Technical report* 1551, University of Wisconsin Madison.

7. Dietterich T.G. (2000), “An experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting, and randomization”, *Machine Learning* 40 (2), 139–157.

8. DuBois T., Golbeck J., Srinivasan A. (2011), “Predicting trust and distrust in social networks”, In *Proceedings of the third IEEE International Conference on Social Computing*, SocialCom.

9. Freund Y., Schapire R.E. (1997), “A decision-theoretic generalization of online learning and an application to boosting”, *Journal of Computer and System Sciences 55* (1), 119–139.

10. Guan-Nan Wang, Hui Gao, Lian Chen, Dennis N. A. Mensah,Yan Fu (2015), “Predicting Positive and Negative Relationships in Large Social Networks”, *Journal of Computer and System Sciences*.

11. Guha R.V., Kumar R., Raghavan P., Tomkins A. (2004), “Propagation of trust and distrust”, In *Proc. 13th WWW*.

12. Kunegis J., Lommatzsch A., Bauckhage C. (2009), “The Slashdot Zoo: Mining a social network with negative edges”, In *Proc. 18th WWW*, pages 741–750.

13. Leskovec J., Huttenlocher D., Kleinberg J. (2010), “Predicting positive and negative links in online social networks”, In *WWW*, pages 641–650.

14. Leskovec J., Huttenlocher D., Kleinberg J. (2010), “Signed networks in social media”, In *Proc. 28th CHI*.

15. Li X., Wang L., Sung E. (2008), “Adaboost with SVM-based component classifiers”, *Engineering Applications of Artificial Intelligence*.

16. Liben-Nowell D., Kleinberg J. (2003), “The link prediction problem for social networks”, In *CIKM*, pages 556–559.

17. Meyer D. (2006), “Support vector machines: The interface to libsvm in package e1071”, In *Technical Univ*., Wien, Austria.

18. Pan S.J., Yang Q. (2010), “A survey on transfer learning”, *IEEE Transactions on Knowledge and Data Engineering*, 22:1345–1359.

19. Raina R., Battle A., Lee H., Packer B. (2007), “Self-taught learning: Transfer learning from unlabeled data”, In *ICML*, pages 759–766.

20. Schwenk H., Bengio Y. (2000), “Boosting neural networks”, *Neural Computation 12*, 1869–1887.

21. Tuyen H., Thanh V., Bac L. (2013), “A Decision-making based Feature for Link Prediction in Signed Social Networks”, In *IEEE RIVF*, pp. 169-174.

22. Yang B., Da-You L. (2007), “A heuristic clustering algorithm for mining Communities in Signed Networks”, *Journal of Computer Science and Technology*, v.22 n.2, p.320-328.

23. Ye J., Cheng H., Zhu Z., Chen M. (2013), “Predicting positive and negative links in signed social networks by transfer learning”, In *Proceedings of the 22nd international conference on World Wide Web*, International World Wide Web Conferences Steering Committee, pp. 1477–1488.

24. Zhuang F., Luo P., Xiong H. , He Q., Xiong Y., Shi Z. (2010), “Exploiting associations between word clusters and document classes for cross-domain text categorization”, In *SDM*.

# PHỤ LỤC

## Phụ lục 1: Kết quả thực nghiệm chi tiết

Như đã giới thiệu trong chương 4, phương pháp thực nghiệm của luận văn là ***4-fold cross validation***. Nghĩa là, dữ liệu thực nghiệm sẽ được chia thành 4 phần bằng nhau và lần lượt tiến hành thực nghiệm trên từng phần. Sau đó, luận văn tính kết quả thực nghiệm trung bình.

Trong chương 4, luận văn đã trình bày các kết quả thực nghiệm trung bình của 5 phương pháp (*Target, Combine + Latent, TAS + Latent, Combine + PNR và TAS + PNR*) trên 6 cặp đồ thị khác nhau (*Epinions – Slashdot, Epinions – Wiki, Slashdot – Epinions, Slashdot – Wiki, Wiki – Epinions và Wiki – Slashdot*). Sau đây, luận văn sẽ trình bày kết quả thực nghiệm chi tiết trên 4 tập dữ liệu thực nghiệm khác nhau.

* ***Bảng PL 1.1***, ***Bảng PL 1.2***, ***Bảng PL 1.3*** và ***Bảng PL 1.4*** thể hiện độ chính xác dự đoán của các phương pháp *Target*, *Combine + Latent*, *TAS + Latent*, *Combine + PNR*, *TAS + PNR* trên 6 cặp đồ thị với 4 tập dữ liệu kiểm tra khác nhau.

Bảng PL 1.1. Độ đo accuracy (%) của các phương pháp thực nghiệm trên tập dữ liệu kiểm tra thứ nhất (E – Epinions, S – Slashdot, W – Wiki)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Phương pháp** | **E – S** | **E – W** | **S – E** | **S – W** | **W – E** | **W – S** |
| Target | 50.4 | 50.62 | 54.93 | 50.19 | 56.21 | 51.94 |
| Combine + Latent | 57.56 | 53.88 | 58.7 | 54.02 | 63.25 | 55.7 |
| TAS + Latent | 61.33 | 54.9 | 64.03 | 54.86 | 65.15 | 56.33 |
| Combine + PNR | 83.83 | 80.21 | 91.37 | 81.54 | 88.96 | 86.97 |
| TAS + PNR | **89.28** | **83.17** | **93.21** | **87.78** | **92.76** | **89.15** |

Bảng PL 1.2. Độ đo accuracy (%) của các phương pháp thực nghiệm trên tập dữ liệu kiểm tra thứ hai (E – Epinions, S – Slashdot, W – Wiki)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Phương pháp** | **E – S** | **E – W** | **S – E** | **S – W** | **W – E** | **W – S** |
| Target | 50.37 | 51.04 | 57.74 | 49.9 | 58.03 | 50.34 |
| Combine + Latent | 52.08 | 52.58 | 62.79 | 50.43 | 62.04 | 54.13 |
| TAS + Latent | 61.75 | 52.72 | 64.34 | 51.18 | 64.68 | 56.45 |
| Combine + PNR | 83.86 | 81.1 | 91.97 | 86.57 | 89.29 | 87.23 |
| TAS + PNR | **87.4** | **85.82** | **94.11** | **86.9** | **93.2** | **90.2** |

Bảng PL 1.3. Độ đo accuracy (%) của các phương pháp thực nghiệm trên tập dữ liệu kiểm tra thứ ba (E – Epinions, S – Slashdot, W – Wiki)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Phương pháp** | **E – S** | **E – W** | **S – E** | **S – W** | **W – E** | **W – S** |
| Target | 50.37 | 50.69 | 58.12 | 49.94 | 57.17 | 52.23 |
| Combine + Latent | 52.79 | 51.9 | 65.39 | 50.06 | 61.75 | 54.54 |
| TAS + Latent | 61.32 | 52.49 | 67.58 | 51.2 | 64.73 | 57.22 |
| Combine + PNR | 83.55 | 80.52 | 91.84 | 85.83 | 89.06 | 86.7 |
| TAS + PNR | **86.97** | **86.31** | **93.15** | **87.09** | **93.37** | **90.25** |

Bảng PL 1.4. Độ đo accuracy (%) của các phương pháp thực nghiệm trên tập dữ liệu kiểm tra thứ tư (E – Epinions, S – Slashdot, W – Wiki)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Phương pháp** | **E – S** | **E – W** | **S – E** | **S – W** | **W – E** | **W – S** |
| Target | 50.4 | 50.73 | 58.16 | 50 | 57.55 | 52.06 |
| Combine + Latent | 51.1 | 52.75 | 63.48 | 50.72 | 62.58 | 54.67 |
| TAS + Latent | 61.23 | 52.89 | 65.32 | 51.74 | 64.39 | 56.51 |
| Combine + PNR | 84.29 | 78.84 | 91.93 | 84.07 | 89.13 | 87 |
| TAS + PNR | **87.75** | **86.05** | **93.5** | **86.45** | **93.34** | **90.43** |

* ***Hình PL 1.1***, ***Hình PL 1.2***, ***Hình PL 1.3*** và ***Hình PL 1.4*** biểu diễn độ chính xác dự đoán của các phương pháp *Target*, *Combine + Latent*, *TAS + Latent*, *Combine + PNR*, *TAS + PNR* trong các bảng trên bằng biểu đồ cột.



**Hình PL 1.**1**.** Độ đo *accuracy* trên tập dữ liệu kiểm tra thứ nhất



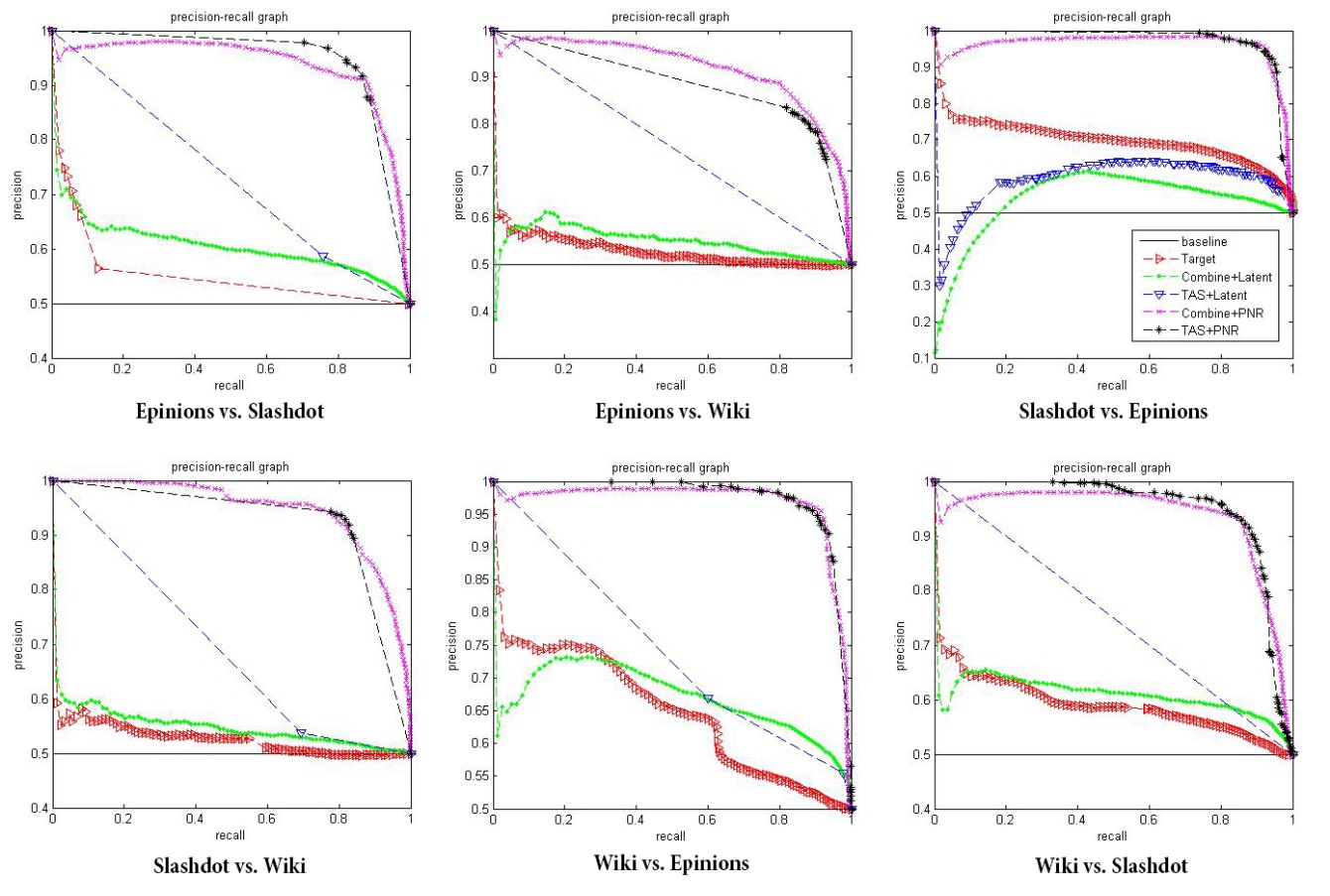
**Hình PL 1.**2**.** Độ đo *accuracy* trên tập dữ liệu kiểm tra thứ hai



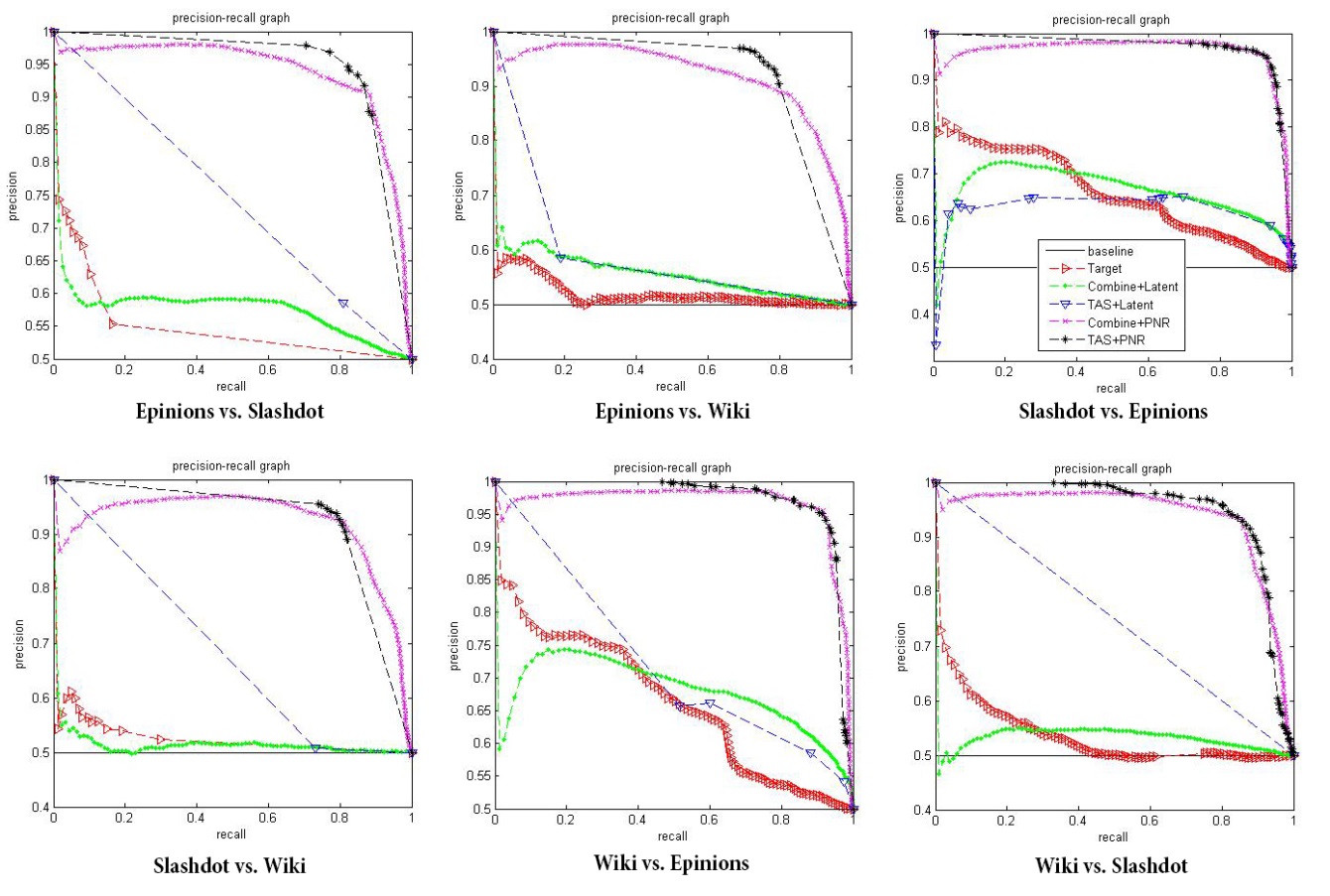
**Hình PL 1.**3**.** Độ đo *accuracy* trên tập dữ liệu kiểm tra thứ ba

**Hình PL 1.**4**.** Độ đo *accuracy* trên tập dữ liệu kiểm tra thứ tư

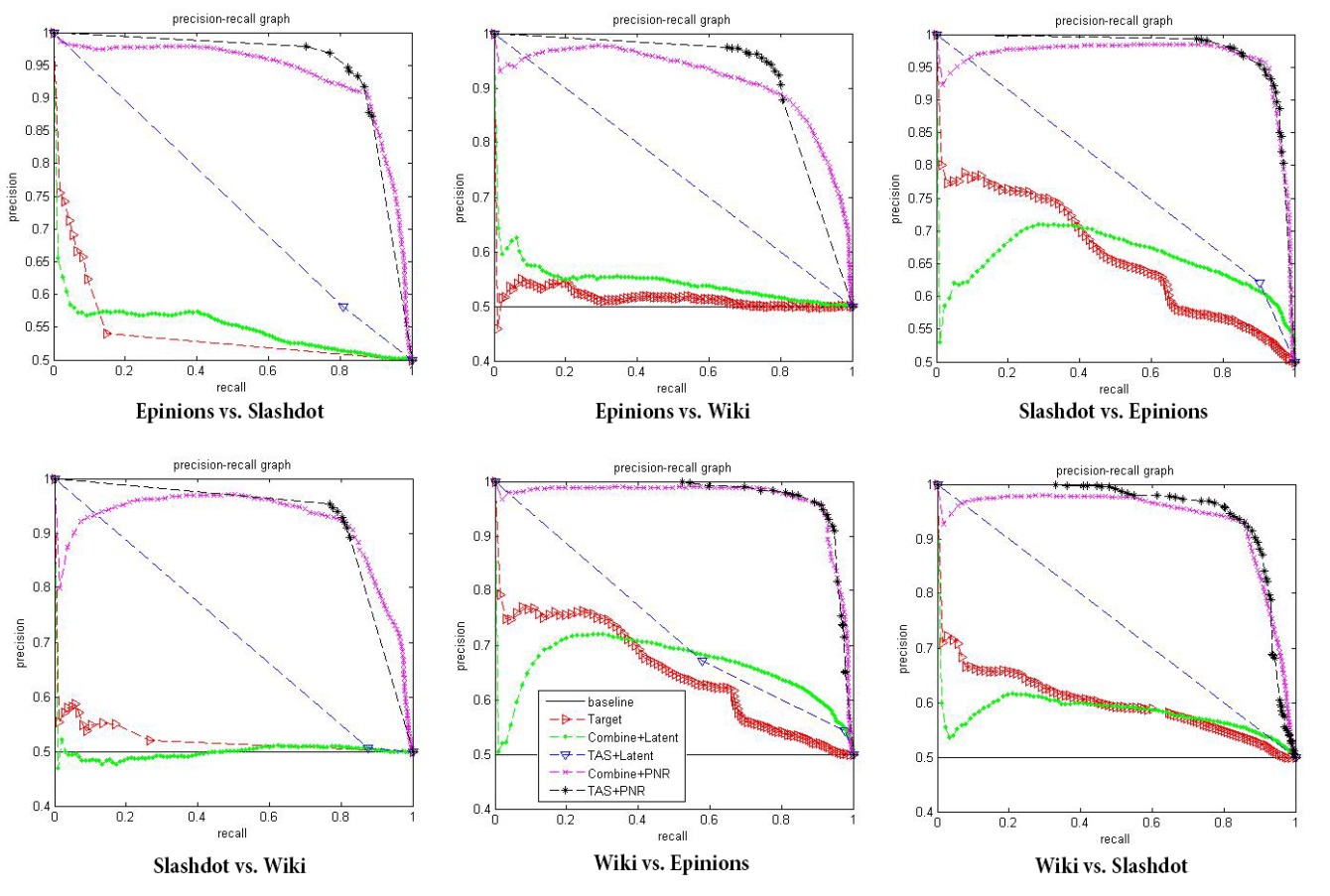
* *Đường cong PR* của các phương pháp thực nghiệm *Target*, *Combine + Latent*, *TAS + Latent*, *Combine + PNR*, *TAS + PNR* trên 6 cặp đồ thị với 4 tập dữ liệu kiểm tra khác nhau được biểu diễn trong ***Hình PL 1.5***, ***Hình PL 1.6***, ***Hình PL 1.7*** và ***Hình PL 1.8***.



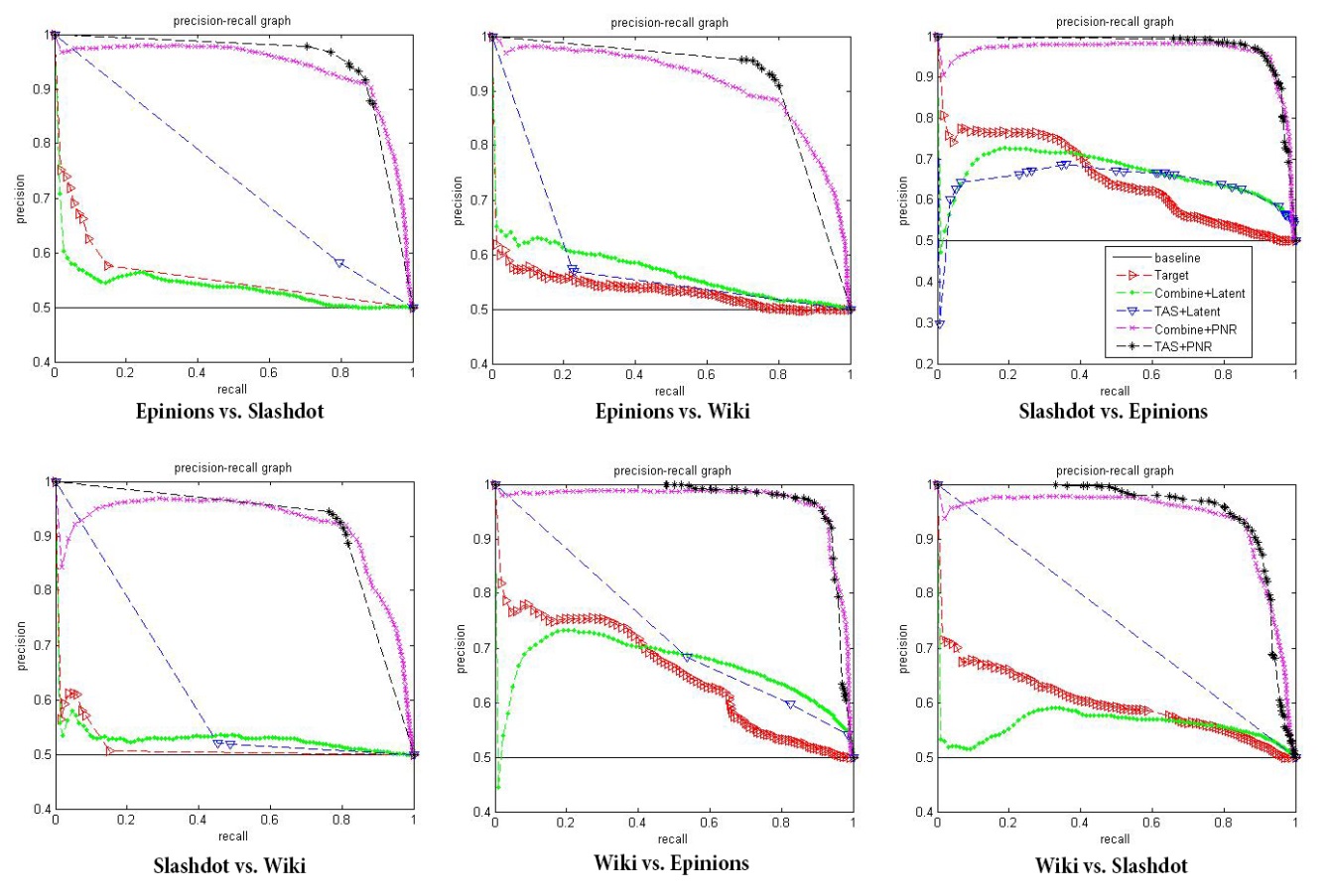
**Hình PL 1.**5**.** Đường cong PR trên tập dữ liệu kiểm tra thứ nhất



**Hình PL 1.**6**.** Đường cong PR trên tập dữ liệu kiểm tra thứ hai



**Hình PL 1.**7**.** Đường cong PR trên tập dữ liệu kiểm tra thứ ba



**Hình PL 1.**8**.** Đường cong PR trên tập dữ liệu kiểm tra thứ tư

* ***Bảng PL 1.5***, ***Bảng PL 1.6***, ***Bảng PL 1.7*** và ***Bảng PL 1.8*** thể hiện thời gian rút trích đặc trưng trên 4 tập dữ liệu kiểm tra khác nhau theo 2 cách: đặc trưng *explicit* kết hợp với đặc trưng *latent*, đặc trưng *explicit* kết hợp với đặc trưng *PNR*.

Bảng PL 1.5. Thời gian thực hiện (phút) của quá trình rút trích đặc trưng (E – Epinions, S – Slashdot, W – Wiki) trên tập dữ liệu kiểm tra thứ nhất

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **E – S** | **E – W** | **S – E** | **S – W** | **W – E** | **W – S** |
| **Explicit + Latent** | 190.10 | 144.46 | 83.66 | 54.46 | 38.73 | 53.53 |
| **Explicit + PNR** | **143.19** | **115.96** | **50.14** | **34.14** | **24.21** | **35.44** |

Bảng PL 1.6. Thời gian thực hiện (phút) của quá trình rút trích đặc trưng (E – Epinions, S – Slashdot, W – Wiki) trên tập dữ liệu kiểm tra thứ hai

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **E – S** | **E – W** | **S – E** | **S – W** | **W – E** | **W – S** |
| **Explicit + Latent** | 215.31 | 151.74 | 88.28 | 70.54 | 46.98 | 75.72 |
| **Explicit + PNR** | **152.29** | **116.11** | **51.06** | **34.29** | **25.13** | **44.53** |

Bảng PL 1.7. Thời gian thực hiện (phút) của quá trình rút trích đặc trưng (E – Epinions, S – Slashdot, W – Wiki) trên tập dữ liệu kiểm tra thứ ba

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **E – S** | **E – W** | **S – E** | **S – W** | **W – E** | **W – S** |
| **Explicit + Latent** | 185.08 | 146.16 | 93.54 | 57.95 | 41.01 | 67.21 |
| **Explicit + PNR** | **136.84** | **115.98** | **46.96** | **34.16** | **21.02** | **29.08** |

Bảng PL 1.8. Thời gian thực hiện (phút) của quá trình rút trích đặc trưng (E – Epinions, S – Slashdot, W – Wiki) trên tập dữ liệu kiểm tra thứ tư

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **E – S** | **E – W** | **S – E** | **S – W** | **W – E** | **W – S** |
| **Explicit + Latent** | 196.34 | 148.42 | 94.54 | 59.61 | 47.20 | 53.56 |
| **Explicit + PNR** | **135.18** | **116.12** | **47.04** | **34.30** | **21.10** | **27.42** |

* ***Bảng PL 1.9***, ***Bảng PL 1.10***, ***Bảng PL 1.11*** và ***Bảng PL 1.12*** biểu diễn thời gian huấn luyện và kiểm tra dữ liệu với SVM trên 4 tập dữ liệu kiểm tra khác nhau.

Bảng PL 1.9. Thời gian thực hiện (phút) của quá trình huấn luyện và kiểm tra dữ liệu với SVM (E – Epinions, S – Slashdot, W – Wiki) trên tập dữ liệu kiểm tra thứ nhất

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **E – S** | **E – W** | **S – E** | **S – W** | **W – E** | **W – S** |
| **Combine + Latent** | 600.16 | 1595.10 | 4390.40 | 251.36 | 955.97 | 75.73 |
| **Combine + PNR** | **8.84** | **7.22** | **18.19** | **30.16** | **2.17** | **2.37** |

Bảng PL 1.10. Thời gian thực hiện (phút) của quá trình huấn luyện và kiểm tra dữ liệu với SVM (E – Epinions, S – Slashdot, W – Wiki) trên tập dữ liệu kiểm tra thứ hai

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **E – S** | **E – W** | **S – E** | **S – W** | **W – E** | **W – S** |
| **Combine + Latent** | 691.90 | 677.00 | 3657.90 | 110.17 | 1551.80 | 1016.00 |
| **Combine + PNR** | **7.19** | **10.28** | **20.26** | **13.13** | **3.39** | **2.87** |

Bảng PL 1.11. Thời gian thực hiện (phút) của quá trình huấn luyện và kiểm tra dữ liệu với SVM (E – Epinions, S – Slashdot, W – Wiki) trên tập dữ liệu kiểm tra thứ ba

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **E – S** | **E – W** | **S – E** | **S – W** | **W – E** | **W – S** |
| **Combine + Latent** | 543.36 | 612.35 | 3016.80 | 301.06 | 1644.70 | 1131.80 |
| **Combine + PNR** | **13.62** | **6.78** | **15.03** | **13.22** | **2.99** | **2.96** |

Bảng PL 1.12. Thời gian thực hiện (phút) của quá trình huấn luyện và kiểm tra dữ liệu với SVM (E – Epinions, S – Slashdot, W – Wiki) trên tập dữ liệu kiểm tra thứ tư

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **E – S** | **E – W** | **S – E** | **S – W** | **W – E** | **W – S** |
| **Combine + Latent** | 705.87 | 586.43 | 3280.00 | 132.36 | 1940.50 | 1161.80 |
| **Combine + PNR** | **12.66** | **11.04** | **18.26** | **22.69** | **1.99** | **2.02** |

* Tương tự, ***Bảng PL 1.13***, ***Bảng PL 1.14***, ***Bảng PL 1.15*** và ***Bảng PL 1.16*** biểu diễn thời gian huấn luyện và kiểm tra dữ liệu với TAS.

Bảng PL 1.13. Thời gian thực hiện (phút) của quá trình huấn luyện và kiểm tra dữ liệu với TAS (E – Epinions, S – Slashdot, W – Wiki) trên tập dữ liệu kiểm tra thứ nhất

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **E – S** | **E – W** | **S – E** | **S – W** | **W – E** | **W – S** |
| **TAS + Latent** | 4273.20 | 4865.50 | 8185.40 | 5016.60 | 852.42 | 978.84 |
| **TAS + PNR** | **1070.40** | **1073.70** | **1772.00** | **1055.20** | **214.71** | **302.63** |

Bảng PL 1.14. Thời gian thực hiện (phút) của quá trình huấn luyện và kiểm tra dữ liệu với TAS (E – Epinions, S – Slashdot, W – Wiki) trên tập dữ liệu kiểm tra thứ hai

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **E – S** | **E – W** | **S – E** | **S – W** | **W – E** | **W – S** |
| **TAS + Latent** | 7826.10 | 3557.90 | 8799.10 | 4340.30 | 840.51 | 963.50 |
| **TAS + PNR** | **1157.60** | **737.85** | **1666.40** | **1222.70** | **241.80** | **195.84** |

Bảng PL 1.15. Thời gian thực hiện (phút) của quá trình huấn luyện và kiểm tra dữ liệu với TAS (E – Epinions, S – Slashdot, W – Wiki) trên tập dữ liệu kiểm tra thứ ba

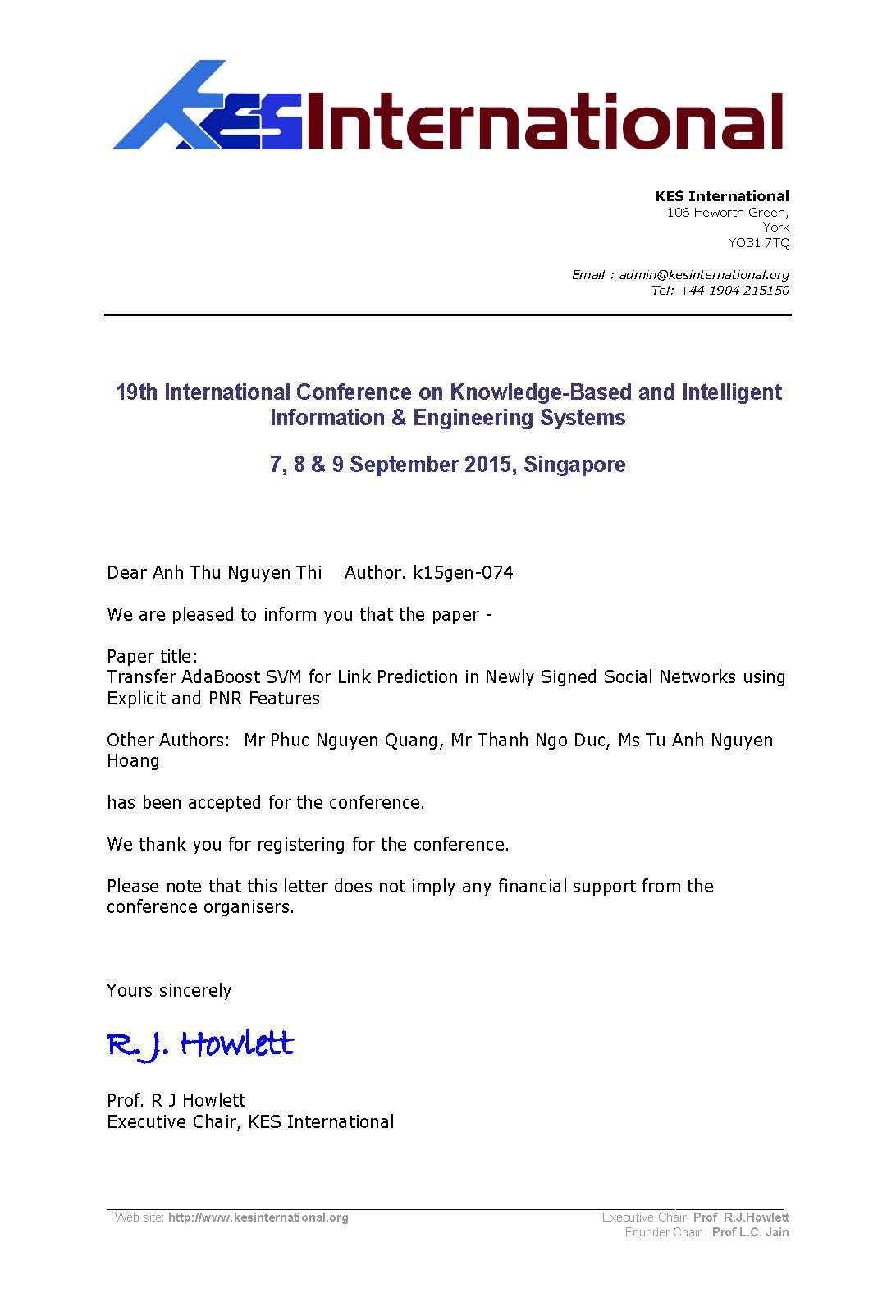
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **E – S** | **E – W** | **S – E** | **S – W** | **W – E** | **W – S** |
| **TAS + Latent** | 4950.60 | 3062.20 | 5965.50 | 5590.90 | 1126.20 | 1207.60 |
| **TAS + PNR** | **1267.00** | **526.77** | **1555.40** | **957.18** | **387.45** | **271.12** |

Bảng PL 1.16. Thời gian thực hiện (phút) của quá trình huấn luyện và kiểm tra dữ liệu với TAS (E – Epinions, S – Slashdot, W – Wiki) trên tập dữ liệu kiểm tra thứ tư

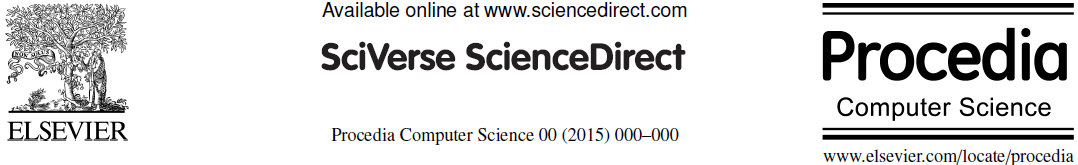
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **E – S** | **E – W** | **S – E** | **S – W** | **W – E** | **W – S** |
| **TAS + Latent** | 6278.20 | 2714.20 | 6135.50 | 6522.30 | 939.40 | 1252.40 |
| **TAS + PNR** | **1479.90** | **713.40** | **2468.70** | **1124.70** | **230.92** | **198.86** |

## Phụ lục 2: Nội dung bài báo được chấp nhận đăng trong hội nghị KES

1. **Thư chấp nhận đăng bài từ hội nghị**



1. **Nội dung bài báo**



19th International Conference on Knowledge Based and Intelligent Information and Engineering Systems

Transfer AdaBoost SVM for Link Prediction in Newly Signed Social Networks using Explicit and PNR Features

Anh-Thu Nguyen-Thi\*, Phuc Quang Nguyen, Thanh Duc Ngo, Tu-Anh Nguyen-Hoang

University of Information Technology, Vietnam National University HCMC, Quarter 6, Linh Trung Ward, Thu Duc Dist, Ho Chi Minh city 700000, Vietnam

**Abstract**

In signed social network, the user-generated content and interactions have overtaken the web. Questions of whom and what to trust has become increasingly important. We must have methods which predict the signs of links in the social network to solve this problem. We study signed social networks with positive links (friendship, fan, like, etc) and negative links (opposition, anti-fan, dislike, etc). Specifically, we focus how to effectively predict positive and negative links in newly signed social networks. With SVM model, the small amount of edge sign information in newly signed network is not adequate to train a good classifier. In this paper, we introduce an effective solution to this problem. We present a novel transfer learning framework is called Transfer AdaBoost with SVM (TAS) which extends boosting-based learning algorithms and incorporates properly designed RBFSVM (SVM with the RBF kernel) component classifiers. With our framework, we use explicit topological features and Positive Negative Ratio (PNR) features which are based on decision-making theory. Experimental results on three networks (Epinions, Slashdot and Wiki) demonstrate our method that can improve the prediction accuracy by 40% over baseline methods. Additionally, our method has faster performance time.

 2015 The Authors. Published by Elsevier B.V.

Peer-review under responsibility of KES International.

*Keywords*: Link Prediction; Signed Social Network; AdaBoost Algorithm

1. **Introduction**

As well as the development of online social network, user-generated content is created and consumed at impressive rates. With so much user interactions and contents are created, the question of whom and what to trust has become an increasingly important challenge. Fortunately, online social networks have allowed people to indicate whom they trust (*positive links*) and distrust (*negative links*). However, this does not solve the problem, we need a signed link prediction system which predicts the signs of links in online social network. Then, we can algorithmically use that positive and negative information to make suggestions to other users about whom they in turn should trust and help a user make decisions, sort and filter information, receive recommendations.

\* Corresponding author. Tel.: +84-090-737-9067

*E-mail address*: thunta@uit.edu.vn, nguyenthianhthu6789@gmail.com

1877-0509  2015 The Authors. Published by Elsevier B.V.

Peer-review under responsibility of KES International.

Examples include Epinions[[7]](#footnote-8) whose users can express trust or distrust of others [15], Slashdot[[8]](#footnote-9) whose participants can declare others to be either “friends” or “foes” [10] and Wiki[[9]](#footnote-10) whose users can vote for or against the promotion of others to adminship [2]. On Epinions, the trust and distrust information is used to determine the reviews shown, using an undisclosed algorithm. On Slashdot, the posts of users tagged as foes are given a lower score, and may thus be hidden. On Wiki, the voting information is used to automatically search for likely future administrators.

A signed link prediction system works through two phases: the offline and the online phases [19]. The overview of the whole framework is given in the Fig. 1. The purpose of offline phase is to learn a prediction model from the training data with three steps: preprocessing, feature extraction and training. The online phase begins with the step of preprocessing and feature extraction similar to the offline phase. Then, the prediction model (trained in the offline phase) is used to assign an edge to positive or negative.

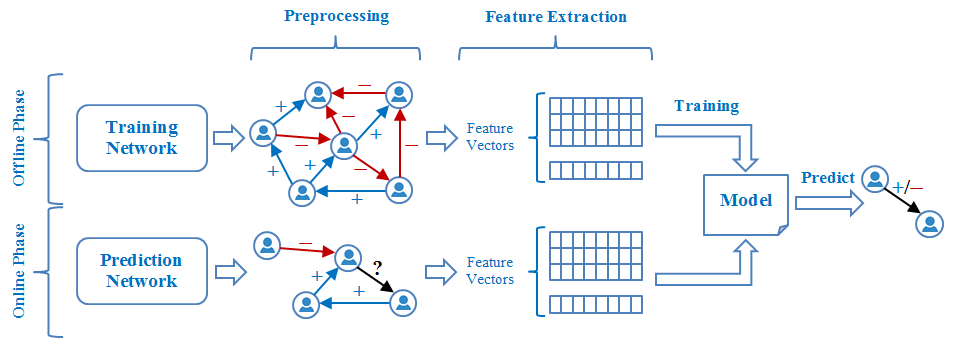


Fig. 1. Framework of a signed link prediction system.

Previous research of signed social networks [14] has shown that the prediction model makes a very strong assumption on the input network: *the signs of all links except the one to be predicted are known in advance*. Thus, we study the edge sign prediction problem with a more realistic setting. Given a newly signed social network, the paucity of available signs makes it difficult to train a good classifier to predict unknown link signs. To solve this problem, we consider leveraging another more mature signed social network, which has the abundant edge sign information. This approach is known as *transfer learning* [16,17].

In this paper, we present a novel transfer learning framework called *TAS* which extends boosting-based learning algorithms and incorporates properly designed *RBFSVM* (SVM with the RBF kernel) component classifiers. With our framework, we use *explicit topological features* [20]. Besides, we propose to use *PNR feature* [19] which is based on the strong theory of decision-making. PNR is a generalizable feature and outperforms most state-of-the-art features in three criteria: accuracy, generalization and speed. Our experimental results on three real signed social networks demonstrate that our method can improve the prediction accuracy and reduces time to extract features, train and test data over baseline methods.

The rest of the paper is organized as follows. We discuss related work in Section 2. Section 3 presents our proposed method which is called TAS. Additionally, we present explicit features and PNR features which are used in our approach in this section. Section 4 shows the experimental results with discussions. Finally, we conclude the paper in Section 5.

1. **Related Word**

For the edge sign prediction problem, existing studies can be categorized into two major approaches: a matrix kernel approach [9,10] and a machine learning approach [3,11]. Guha et al [9] proposed their leading work on trust propagation in signed social network. Kunegis et al [10] extended the method of graphical spectrum analysis by using kernels taken from signed Laplacian matrices of graphs. Leskovec et al [11] proposed *16-dimensional features* corresponding to 16 types of triads in the balance theory and used a logistic regression model to predict the sign of links. Kai-Yang et al [3] tried to improve the quality of feature representation. Instead of using triads, the authors took all cycles of m vertices containing predicted links. The major contribution of their studies is the connections to theories of balance and status in social psychology. However, in their studies, the prediction model makes a very strong assumption on the input network: the signs of all links except the one to be predicted are known in advance, which is not very practical in reality.

Thus, we study the edge sign prediction problem for a newly signed social network whose the edge sign information is very scarce. We try to develop a general framework for transfer learning based on *TrAdaBoost* [4]. We use *RBFSVM* (SVM with the RBF kernel) as component classifier in *TrAdaBoost*. Our method is called *TAS*. The overview of the whole our method is given in the Fig. 2. Our method considers to leverage another more mature signed social network (*Source Graph*) to construct a high-quality classification model for a newly signed social network (*Target Graph*). Thus, we need use generalizable features which can apply to many social networks. In our method, we use *explicit topological features* [20] (node degree, betweenness centrality, triad count and edge embeddedness) which express manifest properties of the edge instances. Instead of using explicit features, we use *PNR features* which are based on decision-making theory.

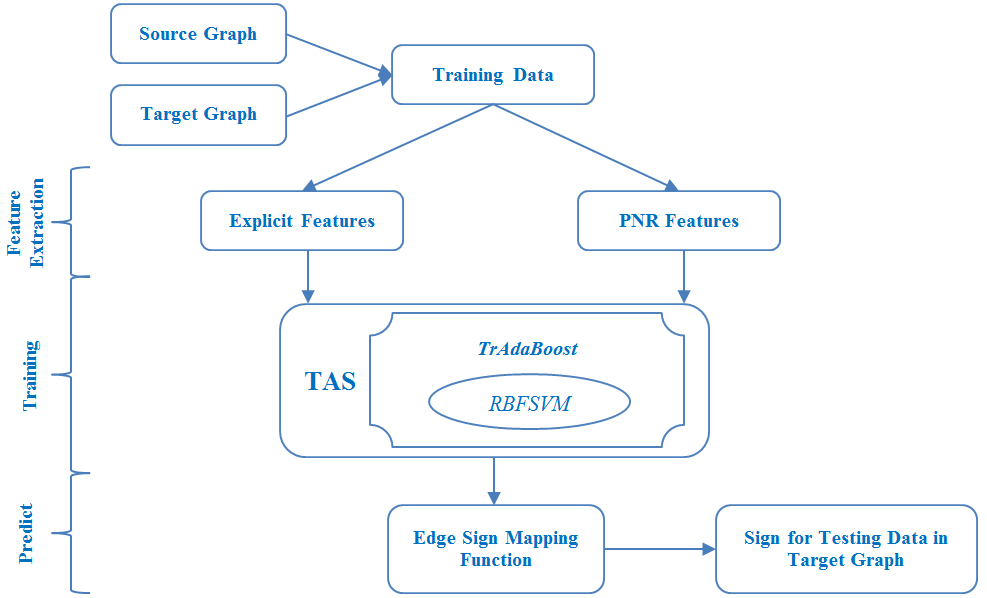


Fig. 2. Our method for signed link prediction in newly signed social networks.

1. **Proposed Method**

This section describes our proposed method for edge sign prediction. Firstly, we present our formal definitions for the edge sign prediction problem. Secondly, we propose a TAS which likes learning algorithm in the transfer learning framework. Finally, we describe proposed features for the edge sign prediction problem.

* 1. *Problem Formulation*

A newly signed social network is called *Target Graph* for edge sign prediction. It is a directed graph . We let denotes the set of vertices,  denotes the set of edges with sign labels,  denotes the set of edges whose signs are unknown, and  is a mapping function which denotes the signs of edges (*positive or negative*).

Because the paucity of available signs in newly signed social network makes it difficult to train a good classifier, we need to leverage another more mature signed social network. Thus, we have another directed graph which is called *Source Graph*. We let denotes the set of vertices, denotes the set of edges with sign labels.

Formally, we let  denotes the training data.  and . Let denotes the testing data. For an edge instance , we encode the essential information of edge into feature vector for training step. Each feature vector is labeled as positive or negative. Feature vectors which are extracted from training data and their labels are used to train a prediction model.

* 1. *Transfer Learning Through TAS*

Source graph may have a different joint distribution of the edge instances and the class labels from the target graph. Besides good knowledge, source graph also contains noisy data. The useful knowledge from source graph is advantageous to the process of classification, while the noisy part of the data does not affect the classifier too much. Thus, training data are abundant, but the basic classifiers learn from these data can not classify the testing data well due to different data distributions. We need a prediction model that leverages the labeled instances in both the source and target graphs. Therefore, we construct *Transfer AdaBoost with SVM*. To construct this prediction model, we borrow the AdaBoost idea from Dai et al [4] and AdaBoost with SVM-based component classifiers from Li et al [13].

AdaBoost (*Freund and Schapire,1997*) [8] is a learning framework which aims to boost the accuracy of a weak learner (*component classifier*) by carefully adjusting the weights of training instances and learns a classifier accordingly. But, for source graph training instances, when they are wrongly predicted due to distribution changes by the prediction model, these instances could be those that are the most dissimilar to the target graph instances. Therefore, this learning algorithm doesn't train a good classifier. To solve this problem, we borrow the *Transfer AdaBoost* from Dai et al, which extends AdaBoost (*Freund and Schapire, 1997*) for transfer learning. We should give edge instances in , that are less similar to the target edge instances in , smaller weights to weaken their impacts; conversely, for edge instances in that are more similar to the target edge instances in , we should give larger weights to attach more importance to them.

Previous researches of transfer learning that use Decision Trees [6] or Neural Networks [18] as component classifiers in AdaBoost have been reported. Still, some difficulties remain. What should be the suitable tree size when Decision Trees are used as component classifiers? How could the complexity be controlled to avoid overfitting when Neural Networks are used as component classifiers? To solve this problem, we borrow the *AdaBoostSVM* from Li et al. The AdaBoostSVM is AdaBoost incorporating properly designed RBFSVM (*SVM with the RBF kernel*) component classifiers. From Li et al [13], the distributions of accuracy and diversity over RBFSVM component classifiers by designing parameter adjusting strategies have promising results. Li et al demonstrate AdaBoost approach that uses RBFSVM component classifiers outperforms other AdaBoost approaches using component classifiers such as Decision Trees and Neural Networks. Thus, we incorporate properly designed *RBFSVM* (SVM with the RBF kernel) component classifiers in *Transfer AdaBoost*. RBFSVM uses two regularization parameters:  controls its model complexity and training error;  is the free parameter of the Gaussian radial basis function. We select proper values of and following Li et al [13].

A formal description of Transfer AdaBoost with SVM (*TAS*) is given in Algorithm 1. In TAS, we use to denote the weights of edges in , and  to denote the weights of edges in . For an edge is the predicted edge sign for , and is the true edge sign. Because the use of source graph is leverage for target graph, the source graph edges will never have a larger influence than the target graph edges. Therefore, the weights of source graph edges would never increase and are always less than those of target graph edges. For any target graph edge , its weight will always get increased by a factor of . For any source graph edge , its weight will always get decreased by a factor of .

* 1. *Proposed Features*

Previous study of edge sign prediction problem for a newly signed social network [20] uses *explicit topological features* (node degree, betweenness centrality, triad count and edge embeddedness) which express manifest properties of the edge instances in the source or target graph and *latent topological features* which can capture the common patterns between source graph and target graph. However, with *latent feature*, when the distributional differences between the source and target graphs become larger, the transfer learning performance becomes worse. Thus, we need a feature which is more generalizable than *latent feature*. In this paper, we propose to use *PNR feature* [19] which is based on the strong theory of decision-making.

We present how to construct PNR feature for edge sign prediction. For a directed edge , we use  to denote the number of positive outgoing edges at  and  to denote the number of negative outgoing edges at . Similarly, and are the number of positive and negative incoming edges at . The term of is an extremely small value to ensure that the denominators are nonzero. We are interested in the ratios between four terms at and :

(1)

(2)

**Algorithm 1:** Transfer AdaBoost with SVM (TAS)

**Input**: The two labeled data and , the unlabeled data , a SVM-based component classifiers

RBFSVM, and the maximum number of iterations .

**Output**: Edge sign classifier .

**begin**

Let ,

The initial weight vector, that . We allow the users to specify the initial values for .

for do

1. Set
2. Call RBFSVM component classifier that is provided the combined training set with the distribution over and the unlabeled data , .
3. Calculate the error of on :
4. If , select new appropriate parameters ( and ) following [13] for an RBF kernel. Then, go to (2).
5. Set ,
6. Update weight vector :

where is the proportion between positive and negative outgoing edges at and is the proportion between positive and negative incoming edges at . Additionally, since and may reach positive infinity, their wide values ranges cause difficulties in the learning step. To overcome this, a threshold is used to cut the ranges down.

(3)

(4)

The PNR feature of the edge is given by concatenating two limited ratios:

(5)

The strength of this feature is that it has a close connection with the decision-making theory in terms of past experience; benefit maximization; herd behaviour; anchoring and adjustment heuristic.

* *Past experience*: Past decisions influence on future decisions. An example is that a person usually gives more positive votes than negative votes. It becomes the habit that this person tends to give positive votes. For an edge , the first element indicates the voting history of the voter .
* *Benefit maximization*: The quality of the object affects directly the decisions of the voters and then, it should be considered for prediction. We can evaluate its quality indirectly by taking its incoming edges. The large value of is an evidence for its good quality and otherwise. Generally, a good object tends to receive more positive votes from new people and otherwise.
* *Herd behaviour*: Herd behaviour in human societies is defined as a phenomenon in which independent people observe and mimic the actions of others, even mistaken. An example is that a person with many fans (large value of ) may usually receive more friend requests than hostile relationships.
* *Anchoring and Adjustment heuristic*: Heuristics are general strategies which can help us make right decisions quickly. In fact, when a person wants to make friend with a stranger, such initial information (likes/dislikes, friends/enemies, etc) will be useful anchors for this person to reach better estimate. In the PNR feature, these anchors are encoded into the term of to enrich prediction with necessary information.

While represents the past experience of voters, implies the principles of benefit maximization, herd behaviour, anchoring and adjustment heuristic. This theoretical foundation helps *PNR feature* significantly outperform the *latent feature* in all aspects: the accuracy, the generalization and the speed. Experimental results demonstrate that *PNR feature* is fitter than *latent feature* when we use them in the transfer learning performance.

We have constructed both explicit and PNR features for edge sign prediction. For an edge instance with label , we have 11 features, including node degrees and , betweenness centrality and , triad counts , , , , edge embeddedness , PNR features  and . Similarly, we can define features for edge instances in .

1. **Experiments**

In this section, we present our experiments to evaluate the our method. Firstly, we describe data preparation and evaluation methods. Secondly, we briefly introduce four baseline methods which are compared with our method and present some detailed settings of five methods. Finally, we present experimental results and discussions.

1. *Data Preparation and Evaluation Methods*

We use three online social networks Epinions, Slashdot and Wiki. All networks are downloaded from Stanford Large Network Dataset Collection[[10]](#footnote-11). Because the original graphs are too large and sparse, we select 19,987 nodes from Epinions, 15,999 nodes from Slashdot and 6,998 nodes from Wiki with the highest degrees [20]. Table 1 shows the statistics of the three extracted networks.

Table 1. Statistics of three extracted networks.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Galleries** | **Nodes** | **Edges** | **Positive Edges** | **Negative Edges** | **(%) Positive Edges** |
| **Epinions** | 19,987 | 634,209 | 555,601 | 78,608 | 87.6 |
| **Slashdot** | 15,999 | 371,122 | 283,993 | 87,129 | 76.5 |
| **Wiki** | 6,998 | 113,844 | 83,832 | 30,012 | 73.6 |

As the edge signs in all these networks are overwhelmingly positive, we overcome this bias by following the methodology of Guha et al. [9] to generate balanced databases. We consider each pair of networks out of the three. We use one network as the source graph and the other as the target graph. There are totally 6 pairs to test. In each target graph, we partition the edge instances into four parts. We use one part as the testing data and randomly sample 10 percentage of edge instances in the remaining three parts to form the labeled edge set . This and in the source graph form the training data.

To evaluate the our method, we use three degrees: *accuracy*, *precision* and *recall* [5]. Accuracy, precision and recall are the basic measures used in evaluating a classification model. Besides, we compare performance time of the methods.

1. *Experimental Settings*

For evaluation purpose, we use MATLAB to set up following five methods:

* *Target*: using labeled edge instances in the target graph for training with SVM model (the RBF kernel).
* *Combine+Latent*: using all edge instances in the source graph and labeled edge instances in the target graph for training with SVM model (the RBF kernel).
* *TAS+Latent*: using all edge instances in the source graph and labeled edge instances in the target graph for training with TAS.
* *Combine+PNR*: using both source graph edges and labeled target graph edges for training with SVM model (the RBF kernel).
* *TAS+PNR* (our method): using both source graph edges and labeled target graph edges for training with TAS.

*Target*, *Combine+Latent* and *TAS+Latent* use the explicit and latent topological features that are proposed in [20]. *Combine+PNR* and *TAS+PNR* use our proposed features (explicit features and PNR features). Besides TAS algorithm, we use SVM model with a RBF kernel because this kernel usually outperforms the different kernels in both accuracy and convergence time [1].

We construct latent features following [20]. However, Ye et al don't publish value of parameters such as the trade-off regularization parameter and the convergence threshold for an iterative update algorithm. Moreover, they don't also introduce classifier methods initializing latent feature matrices. Thus, after some preliminary test, we set trade-off parameter is 10, convergence threshold is 10-1. To initialize latent feature matrices, we implement Naive Bayes classifier (*a widely used classifier method*).

1. *Experiments Results And Discussions*

We evaluate our method based on four criteria: accuracy, precision, recall and speed (performance time). We have 6 pairs (source - target) from three networks Epinions, Slashdot and Wiki. Fig 3 shows the accuracies of *Target*, *Combine+Latent*, *TAS+Latent*, *Combine+PNR* and *TAS+PNR* on 6 pairs. This shows that *TAS+PNR* has the best edge sign prediction result. In Fig 4, we present five Precision Recall Curve (PR curve) for methods on 6 pairs. The precision recall area under curve (PR AUC) is just the area under the PR curve. The higher it is, the better the method is. Therefore, our method has the best result on most pairs.

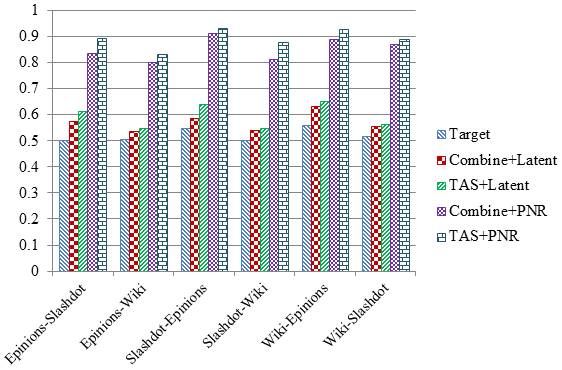


Fig. 3. Prediction Accuracy with 10% of Labeled Target Edge Instances.

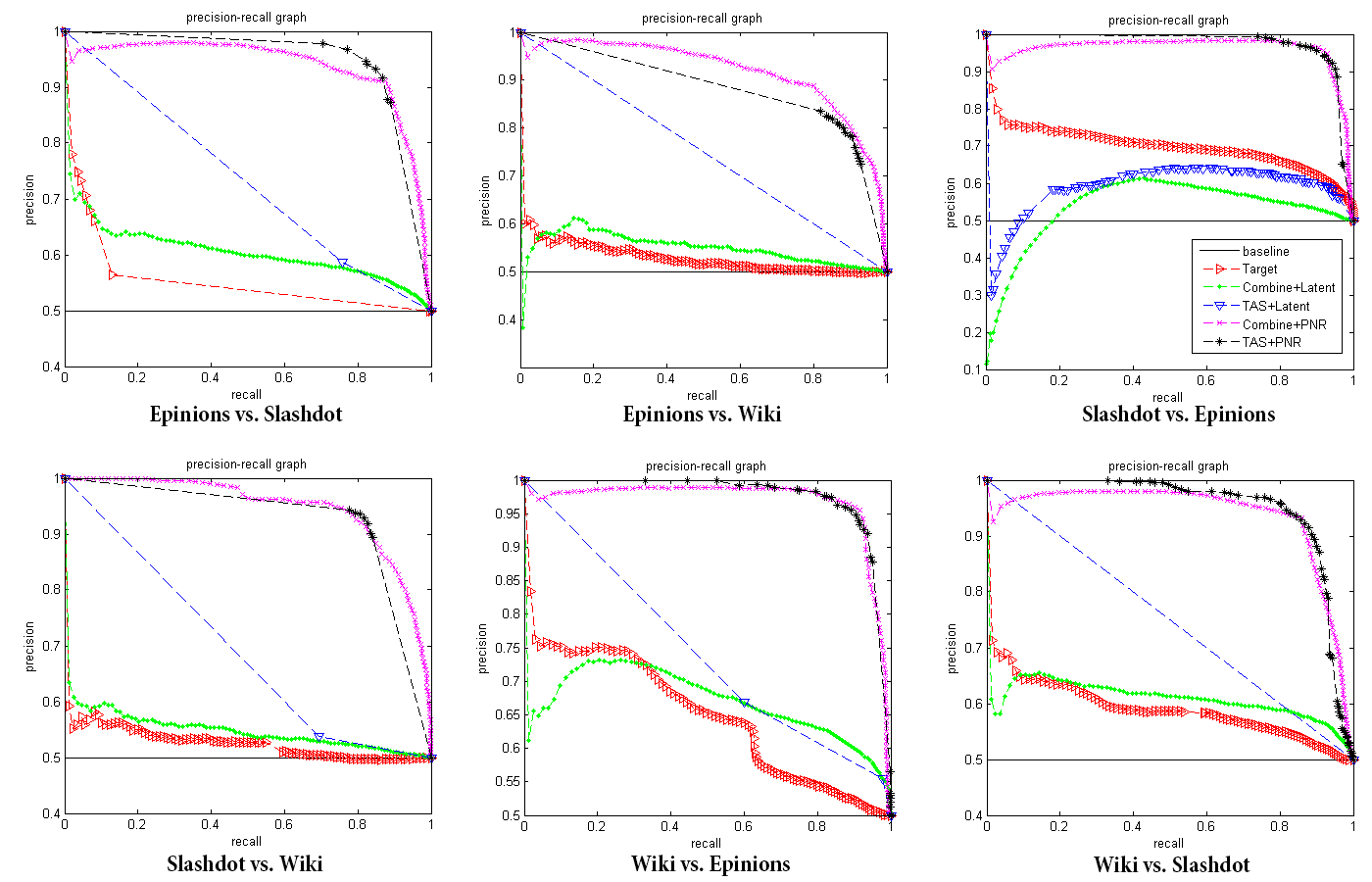


Fig. 4. PR Curve with 10% of Labeled Target Edge Instances..

From Fig 3 and Fig 4, we can see that our method (*TAS+PNR*) can improve the prediction accuracy by 40% over baseline methods. In the first group of experiment, we use Epinions as the source graph and Slashdot as the target graph. *Target* has the worst result because these method uses only small amount of edge sign information in target graph. *Combine+Latent* can improve the accuracy over *Target* but the noise in the source edge instances may become more obvious. Thus, *TAS+Latent* has better than *Combine+Latent*. However, the latent feature becomes worse when the distributional differences between the source and target graph become large. The PNR feature that is based on the strong theory of decision-making has high generalization. Therefore, *Combine+PNR* and *TAS+PNR* have high accuracy. With TAS algorithm, we replace latent feature with PNR feature. Our method has the best result. We can observe similar trends in residual pairs.

In the end, we turn to the speed evaluation of methods. We measure the performance time of feature extraction, training and prediction. All our experiments are conducted on the same PCs with 2.90 GHz CPU and 12G RAM. First, we compare the speed of feature extraction. Then, we measure the performance time of training and prediction. Table 2 shows the speed of two way feature extraction: explicit features combine latent features, explicit features combine PNR features. In table 3 and table 4, we present the speed of training and prediction with SVM and TAS.

Table 2. Performance time (in minute) of feature extraction (E - Epinions, S - Slashdot, W - Wiki).

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **E – S** | **E – W** | **S – E** | **S – W** | **W – E** | **W – S** |
| **Explicit+Latent** | 190.10 | 144.46 | 83.66 | 54.46 | 38.73 | 53.53 |
| **Explicit+PNR** | **143.19** | **115.96** | **50.14** | **34.14** | **24.21** | **35.44** |

Table 3. Performance time (in minute) of training and prediction with SVM (E - Epinions, S - Slashdot, W - Wiki).

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **E – S** | **E – W** | **S – E** | **S – W** | **W – E** | **W – S** |
| **Combine+Latent** | 600.16 | 1595.10 | 4390.40 | 251.36 | 955.97 | 75.73 |
| **Combine+PNR** | **8.84** | **7.22** | **18.19** | **30.16** | **2.17** | **2.37** |

Table 4. Performance time (in minute) of training and prediction with TAS (E - Epinions, S - Slashdot, W - Wiki).

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **E – S** | **E – W** | **S – E** | **S – W** | **W – E** | **W – S** |
| **TAS+Latent** | 4273.20 | 4865.50 | 8185.40 | 5016.60 | 852.42 | 978.84 |
| **TAS+PNR** | **1070.40** | **1073.70** | **1772.00** | **1055.20** | **214.71** | **302.63** |

From table 2, table 3 and table 4, when we replace latent feature with PNR feature, the speed of feature extraction, training and prediction is faster. The performance time decrease because PNR has two benefits: low cost feature, simple implementation.

1. **Conclusion**

In this paper, we studied the problem of signed link prediction in social networks that have both positive and negative links. We focus how to effectively predict sign links in newly signed social network whose the edge sign information is very scarce. We propose a novel transfer learning framework called *TAS* that extends boosting-based learning algorithms and incorporates properly designed RBFSVM component classifiers. *TAS* can select the most useful source graph instances as additional training data for predicting the labels of target graph techniques when the noise in the source graph instances cause the model to predict wrongly on the test edges from the target graph. Besides, we replace *latent feature* with *PNR feature* that is low cost feature and has a close connection with the decision-making theory in terms of past experience; benefit maximization; herd behaviour; anchoring and adjustment heuristic. The results of experiments on three networks Epinions, Slashdot and Wiki show that our method really improves on previous methods in two criteria accuracy and speed significantly.

**References**

1. Ben-Hur A, Weston J. A User’s Guide to Support Vector Machines. *Methods in Molecular Biology*; 2010. p. 223-239.

2. Burke M, Kraut R. Mopping up: Modeling wikipedia promotion decisions. *In Proc. CSCW*; 2008.

3. Chiang K, Nagarajan N, Tewari A, Dhillon IS. Exploiting longer cycles for link prediction in signed networks. *In Proceedings of the 20th ACM Conference on Information and Knowledge Management*;2011. p. 1157-1162.

4. Dai W, Yang Q, Xue G, Yu Y. Boosting for Transfer Learning. Proc. 24th Int’l Conf. *Machine Learning*; 2007. p. 193-200.

5. Davis J, Goadrich M. The relationship between precision-recall and roc curves. *Technical report 1551, University of Wisconsin Madison*; 2006.

6. Dietterich TG. An experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting, and randomization. *Machine Learning 40 (2)*; 2000. p. 139-157.

7. DuBois T, Golbeck J, Srinivasan A. Predicting trust and distrust in social networks. *In Proceedings of the third IEEE International Conference on Social Computing, SocialCom*; 2011.

8. Freund Y, Schapire RE. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences 55 (1)*; 1997. p. 119-139.

9. Guha RV, Kumar R, Raghavan P, Tomkins A. Propagation of trust and distrust. *In Proc. 13th WWW*; 2004.

10. Kunegis J, Lommatzsch A, Bauckhage C. The Slashdot Zoo: Mining a social network with negative edges. *In Proc. 18th WWW*; 2009. p. 741-750.

11. Leskovec J, Huttenlocher D, Kleinberg J. Predicting positive and negative links in online social networks. *In WWW*; 2010. p. 641-650.

12. Leskovec J, Huttenlocher D, Kleinberg J. Signed networks in social media. *In Proc. 28th CHI*; 2010.

13. Li X, Wang L, Sung E. Adaboost with SVM-based component classifiers. *Engineering Applications of Artificial Intelligence, 21*; 2008.

14. Liben-Nowell D, Kleinberg J. The link prediction problem for social networks. *In CIKM*; 2003. p. 556-559.

15. Massa P, Avesani P. Controversial users demand local trust metrics: an experimental study on epinions.com community. *In AAAI ’05, AAAI Press*; 2005. p. 121-126.

16. Pan SJ, Yang Q. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering, 22*; 2010. p. 1345-1359.

17. Raina R, Battle A, Lee H, Packer B, Ng AY. Self-taught learning: Transfer learning from unlabeled data. *In ICML*; 2007. p. 759-766.

18. Schwenk H, Bengio Y. Boosting neural networks. *Neural Computation 12*; 2000. p. 1869-1887.

19. Tuyen H, Thanh V, Bac L. A Decision-making based Feature for Link Prediction in Signed Social Networks. *In IEEE RIVF*; 2013. p. 169-174.

20. Ye J, Cheng H, Zhu Z, Chen M. Predicting positive and negative links in signed social networks by transfer learning. *In Proceedings of the 22nd international conference on World Wide Web, International World Wide Web Conferences Steering Committee*; 2013. p. 1477-1488.

21. Zhuang F, Luo P, Xiong H, He Q, Xiong Y, Shi Z. Exploiting associations between word clusters and document classes for cross-domain text categorization. *In SDM*; 2010.

1. http://www.fmsasg.com/SocialNetworkAnalysis/ [↑](#footnote-ref-2)
2. http://www.epinions.com/ [↑](#footnote-ref-3)
3. http://slashdot.org/ [↑](#footnote-ref-4)
4. http://people.kyb.tuebingen.mpg.de/pgehler/code/index.html [↑](#footnote-ref-5)
5. http://snap.stanford.edu/data/index.html [↑](#footnote-ref-6)
6. https://en.wikipedia.org/wiki/Accuracy\_and\_precision [↑](#footnote-ref-7)
7. www.epinions.com [↑](#footnote-ref-8)
8. slashdot.org [↑](#footnote-ref-9)
9. [www.wikipedia.org](http://www.wikipedia.org/) [↑](#footnote-ref-10)
10. http://snap.stanford.edu/data/index.html [↑](#footnote-ref-11)