ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

NGUYỄN THỊ TƯƠI

ỨNG DỤNG CÁC MÔ HÌNH CHỦ ĐỀ ẨN VÀO MÔ HÌNH PHÂN HẠNG LẠI DÒNG CẬP NHẬT TRÊN MẠNG XÃ HỘI TWITTER

LUẬN VĂN THẠC SĨ NGÀNH HỆ THỐNG THÔNG TIN

HÀ NỘI - 2016

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

NGUYỄN THỊ TƯƠI

ỨNG DỤNG CÁC MÔ HÌNH CHỦ ĐỀ ẨN VÀO MÔ HÌNH PHÂN HẠNG LẠI DÒNG CẬP NHẬT TRÊN MẠNG XÃ HỘI TWITTER

Ngành: Hệ Thống Thông Tin

Chuyên ngành: Hệ Thống Thông Tin

Mã số: 60480104

LUẬN VĂN THẠC SĨ NGÀNH HỆ THỐNG THÔNG TIN

NGƯỜI HƯỚNG DẪN KHOA HỌC: PGS.TS. HÀ QUANG THỤY

HÀ NỘI - 2016

LỜI CẢM ƠN

Lời đầu tiên, tôi xin gửi lời cảm ơn và lòng biết ơn sâu sắc nhất tới PGS.TS Hà Quang Thụy, đã tận tình hướng dẫn và chỉ bảo tôi trong suốt quá trình thực hiện luận văn tốt nghiệp.

Tôi xin chân thành cảm ơn các thầy, cô trong trường đại học Công Nghệ - đại học Quốc gia Hà Nội đã cho tôi nền tảng kiến thức tốt và tạo mọi điều kiện thuận lợi cho tôi học tập và nghiên cứu.

Tôi cũng xin gửi lời cảm ơn đến các thầy cô, các anh chị và các bạn trong phòng thí nghiệm DS&KTLab và đề tài QG.15.22 đã hỗ trợ tôi rất nhiều về kiến thức chuyên môn trong quá trình thực hiện luận văn. Tôi xin cảm ơn tất cả mọi người đã ủng hộ và khuyến khích tôi trong suốt quá trình học tập tại trường.

Cuối cùng, tôi xin được gửi lời cám ơn vô hạn tới gia đình và bạn bè, những người đã luôn bên cạnh, giúp đỡ và động viên tôi trong quá trình học tập cũng như trong suốt quá trình thực hiện luận văn.

Tôi xin chân thành cảm ơn!

Hà Nội, ngày tháng năm 2016

Học viên

Nguyễn Thị Tươi

ỨNG DỤNG CÁC MÔ HÌNH CHỦ ĐỀ ẨN VÀO MÔ HÌNH PHÂN HẠNG LẠI DÒNG CẬP NHẬT TRÊN MẠNG XÃ HỘI TWITTER

Nguyễn Thị Tươi

Khóa K20, chuyên ngành Hệ Thống Thông Tin

Tóm tắt Luận văn:

Twitter là một trong những mạng xã hội phát triển mạnh với đông đảo thành viên. Khái niệm "vòng kết nối" của mỗi người dùng được định nghĩa là tập hợp các bạn bè của người dùng đó. Vòng kết nối càng lớn, lượng tin (dòng cập nhật) gửi tới trang nhà (timelines) của người dùng sẽ càng nhiều. Theo nghiên cứu của Liangjie và cộng sự (2012), người dùng có thể mất khá nhiều thời gian với các dòng cập nhật vô ích. Nhằm tư vấn và giảm thiểu thời gian lãng phí cho người dùng, giải pháp xếp hạng dòng cập nhật trên mỗi trang của người dùng là một chủ đề nghiên cứu được quan tâm. Nói cách khác, bài toán Xếp hạng dòng cập nhật được chú trọng. Đây chính là bài toán trọng tâm của luận văn.

Theo Chunjing Xiao và cộng sự (2015), độ ảnh hưởng người dùng (user influence) được đánh giá là rất hữu ích trong hệ tư vấn. Với mục đích tiếp tục phát triển nghiên cứu năm 2013 về mô hình xếp hạng dòng cập nhật, luận văn đề xuất phương pháp nâng cao hiệu quả tính hạng cho mô hình bằng cách áp dụng độ ảnh hưởng người dùng vào làm giàu đặc trưng. Độ ảnh hưởng của người dùng được tìm thông qua luật kết hợp dựa trên cơ sở nghiên cứu của Fredrik Erlandsson và cộng sự (2016). Thuật toán Apriori là một trong những thuật toán tìm luật kết hợp phổ biến nhất, được sử dụng cho mô hình này. Bổ sung đặc trưng độ ảnh hưởng người dùng qua luật kết hợp vào mô hình tính hạng là điểm mới so với các công trình trước đó. Phương pháp học xếp hạng CRR (Combined Regression and Ranking), một phương pháp học xếp hạng kết hợp SVM-rank và hồi quy; và phân phối xác suất chủ đề ẩn LDA (Latent Dirichlet Allocation) làm giàu đặc trưng nội dung tiếp tục được sử dụng trong mô hình. Thực nghiệm đối với dữ liệu Twitter của người dùng Jon Bowzer Bauman cho kết quả khả quan.

Từ khóa: dòng cập nhật, CRR, LDA, Apriori

LỜI CAM ĐOAN

Tôi xin cam đoan mô hình xếp hạng các dòng cập nhật trên mạng xã hội Twitter và thực nghiệm được trình bày trong luận văn là do tôi đề ra và thực hiện dưới sự hướng dẫn của PGS.TS Hà Quang Thụy.

Tất cả các tài liệu tham khảo từ các nghiên cứu liên quan đều có nguồn gốc rõ ràng từ danh mục tài liệu tham khảo trong luận văn. Trong luận văn, không có việc sao chép tài liệu, công trình nghiên cứu của người khác mà không chỉ rõ về tài liệu tham khảo.

Hà Nội, ngày tháng năm 2016

Học viên

Nguyễn Thị Tươi

Mục lục

Danh sách hình vẽ

Danh sách bảng biểu

Danh sách các từ viết tắt

STT	Tên viết tắt	Cụm từ đầy đủ
2	CRR	Combined Regression and Ranking
3	LDA	Latent Dirichlet Allocation
4	pLSA	Probabilistic Latent Semantic Analysis
5	P@K	Precision@K
6	MAP	Mean Average Precision
7	AR	Association Rule

MỞ ĐẦU

Ngày nay, mạng xã hội phát triển mạnh mẽ mang những nhận xét, đánh giá, những thông tin phản ánh xã hội thực tới mỗi người, và ngày càng đi sâu vào cuộc sống của mỗi chúng ta. Chúng cung cấp nhiều thông tin cập nhật có tính thời gian thực có được từ kết nối trực tuyến của mọi người. Dòng các tin mới đến trang cá nhân của mỗi người dùng được gọi là dòng cập nhật của người dùng đó. Mặc dù dòng cập nhật đưa đến những thông tin mới, nhưng tồn tại một hạn chế là không ít người dùng đã phải dành khá nhiều thời gian với dòng cập nhật, bởi có không ít tin mới trong dòng cập nhật mang lại thông tin không cần thiết cho họ. Nhiều người dùng rơi vào tình cảnh bị ngập trong dòng cập nhật mà không thể xử lý chúng một cách đầy đủ. Với mục đích giải quyết vấn đề này, giải pháp được quan tâm là sắp xếp các tin trong dòng cập nhật sao cho hợp lý nhất với mỗi người dùng. L. Hong và cộng sự (2012) nêu bật vấn đề *xếp hạng dòng cập nhật* (gọi tắt là Xếp hạng dòng).

Bài toán xếp hạng dòng trong mạng xã hội được đặt ra để giải quyết vấn đề cập nhật tin cho mỗi người dùng, đưa ra danh sách các tin trong dòng cập nhật theo một thứ tự (theo "hạng") quan tâm của người dùng, như là một hình thức tư vấn cho người dùng đó. Dù không nhận được sự phản hồi của người dùng như hệ thống tư vấn, nhưng lọc nội dung vẫn có thể được áp dụng trong mô hình giải quyết bài toán. Bài toán xếp hạng này khác biệt với bài toán xếp hạng kết quả tìm kiếm ở điểm là bài toán xếp hạng dòng không có câu truy vấn. Do đó, không thể dựa theo đặc trưng đối tượng xếp hạng có chứa nhiều thông tin liên quan tới câu truy vấn để tiến hành sắp xếp. Với bài toán này, việc xếp hạng các tin trong dòng cập nhật cần căn cứ vào lịch sử hành vi của người dùng để tìm ra mối quan hệ giữa cá nhân người dùng đó với đối tượng xếp hạng, thậm chí cả quan hệ với người dùng khác.

Tương tự như các mạng xã hội khác, người dùng trên Twitter cũng đối mặt với lượng lớn các dòng cập nhật liên tục từ những người bạn của mình. Như đã đề cập trong [1], chúng tôi tập trung vào bài toán xếp hạng dòng trên mạng xã hội Twitter, và tiếp tục phát triển mô hình xếp hạng dòng của mình**.** Phương pháp xếp hạng đang được quan tâm nhiều trong thời gian gần đây – phương pháp học tính hạng [2, 3, 4] được áp dụng trong mô hình này. Cụ thể, đó là phương pháp học tính hạng CRR [5] (Combined Regression and Ranking).

Mô hình xếp hạng dòng sử dụng thuật toán học tính hạng – thuật toán dựa trên nền tảng học máy, nên việc xây dựng các tập dữ liệu huấn luyện là cần thiết. Chúng tôi đi

tìm các yếu tố đặc trưng của tweet. Như đã phát biểu trong [1], yếu tố nội dung của tweet - một yếu tố cơ sở tất yếu cho quá trình học, được tìm ra dựa vào phương pháp phân cụm không giám sát, đó là mô hình chủ đề ẩn [6, 7]. Yếu tố nội dung được biểu diễn dưới hình thức một tập các phân phối tweet theo chủ đề. Trong mô hình xếp hạng dòng, mô hình chủ đề ẩn LDA được sử dụng. Ngoài yếu tố nội dung, độ ảnh hưởng người dùng được nhận diện là một yếu tố quan trọng. Theo C. Xiao và cộng sự (2015), F. Riquelme và P. G. Cantergiani (2016) [8, 9], các cập nhật của người dùng có độ ảnh hưởng lớn thường được nhiều người theo dõi hơn. Dựa trên quan điểm này, chúng tôi nhận thấy các dòng cập nhật từ những người bạn có ảnh hưởng tới người dùng đang xét nên được tư vấn cho người dùng đó. Hay nói cách khác, độ ảnh hưởng người dùng nên được tham gia vào quá trình học tính hạng. Do vậy, chúng tôi quyết định cải thiện mô hình tính hạng [1] với sự tham gia của đặc trưng độ ảnh hưởng người dùng. F. Erlandsson và cộng sự (2016) [10] đã thực hiện tìm các người dùng có độ ảnh hưởng lớn trên mạng xã hội dựa vào khai phá luật kết hợp. Theo hướng tiếp cận này, chúng tôi công thức hóa độ ảnh hưởng của người dùng qua số lượng luật kết hợp tìm được trên tập các tweet. Thuật toán khai phá luật kết hợp được sử dụng là thuật toán Apriori [11].

Khái quát lại, luận văn đề xuất phương pháp cải thiện mô hình tính hạng mà chúng tôi đã đề xuất trong [1] thành mô hình với cốt lõi là phương pháp học tính hạng, xây dựng đặc trưng nội dung dựa trên mô hình LDA, và xây dựng đặc trưng người dùng dựa trên luật kết hợp. Nội dung của luận văn chia thành các chương như sau:

Chương 1: Luận văn trình bày về các dòng cập nhật của mỗi người dùng trên mạng xã hội Twitter và phát biểu bài toán xếp hạng các dòng cập nhật đó. Đồng thời nêu lên hướng giải quyết và ý nghĩa của bài toán này.

Chương 2: Luận văn trình bày về các phương pháp mà mô hình đề xuất sẽ sử dụng: phương pháp học tính hạng, mô hình chủ đề ẩn và luật kết hợp.

Chương 3: Luận văn trình bày mô hình xếp hạng dòng và cách hoạt động của mô hình đó.

Chương 4: Luận văn trình bày thực nghiệm cho việc áp dụng mô hình xếp hạng trong chương 3 vào việc tính hạng tập các tweet của người dùng trên Twitter.

Chương 1.

DÒNG CẬP NHẬT TRÊN MẠNG XÃ HỘI TWITTER VÀ BÀI TOÁN XẾP HẠNG DÒNG

Trong chương này, chúng tôi trình bày một cách chi tiết về mạng xã hội Twitter, các dòng cập nhật cũng như bài toán xếp hạng dòng và ý nghĩa của bài toán.

1.1. Mạng xã hội Twitter và dòng cập nhật trên Twitter

Twitter là dịch vụ mạng xã hội ra đời năm 2006, một trang micro-blog được phát triển bởi Twitter Inc, cung cấp một dịch vụ mạng miễn phí cho phép người dùng sử dụng gửi và nhận các tin nhắn (tweet), và đã trở thành một hiện tượng phổ biến toàn cầu. Số lượng thành viên của Twitter lên tới gần 500 triệu người dùng vào tháng 12 năm 2012 [12], và hiện thời, số lượng thành viên tích cực tháng là khoảng 316 triệu người[1].

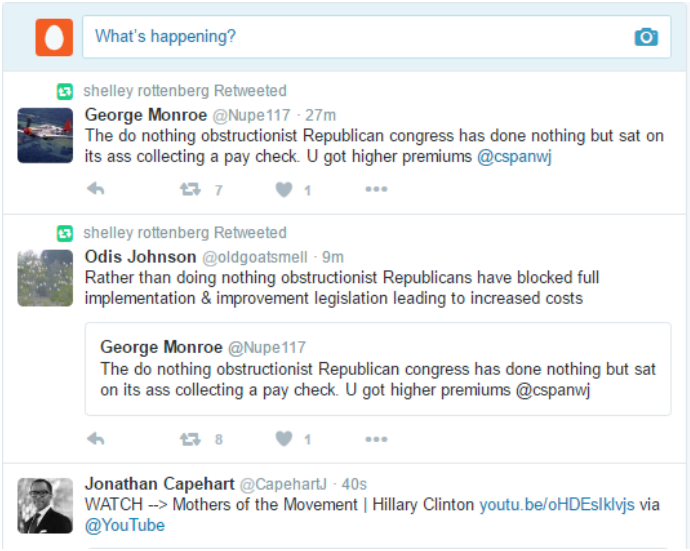
Twitter là lưới kết nối các thành viên với nhau một cách dễ dàng. Khi muốn theo dõi thông tin của thành viên khác, mỗi thành viên chỉ cần thực hiện *following* thành viên đó. Twitter có các chỉ số *follower* – số người theo dõi mình, *following* - số người mình theo dõi, retweet, *tweets* – số tweet mình đã viết... Các tweet có độ dài 144 kí tự, như tin nhắn SMS, hiển thị trên trang cá nhân của mỗi người. Số lượng kí tự hạn hẹp là một trong các yếu tố làm cho tweet lan nhanh hơn trên mạng xã hội. Một số hoạt động trên Twitter như *reply, retweet, favorite…* Hơn nữa, Twitter hỗ trợ giao diện chương trình ứng dụng (API) đầy đủ, cho phép mọi thành viên có thể sử dụng để lập trình ứng dụng. API giúp người sử dụng có thể lấy được các thông tin về các người dùng trong mạng xã hội như tên truy cập, ID, số lượng bạn bè, số lượng tweet mỗi ngày…

Dòng cập nhật trên mạng xã hội Twitter được hiểu là dòng cập nhật của mỗi người dùng. Người dùng A following B, thì A được gọi là follower của B, và B được gọi là followee của A. Khi các followee đăng các thông điệp (tweet), các tweet này sẽ được

[1] https://www.socialbakers.com/statistics/twitter/, truy cập ngày 28 tháng 10 năm 2016.

hiển thị trên timelines của follower [13]. Khi số lượng followee là lớn thì lượng dòng cập nhật đến trang của follower có thể lên tới hàng trăm tweet. C. Li và cộng sự [13] cũng chỉ ra rằng một khi số lượng dòng cập nhật là lớn, các cập nhật mới sẽ hiển thị trên đầu, thay thế các cập nhật cũ. Như vậy bất kì người dùng nào cũng có thể rơi vào tình cảnh bị tràn ngập thông tin và dễ bỏ qua những tin cần thiết với bản thân họ. Giải pháp xếp hạng dòng cập nhật của mỗi người dùng được đưa ra để giải quyết vấn đề này.

Hình 1.1. Minh họa dòng cập nhật trên Twitter minh họa cách hiển thị các dòng cập nhật trên trang cá nhân của người dùng trên mạng xã hội Twitter. Các tweet mới được hiển thị lần lượt theo thời gian: tweet đến sau cùng sẽ được hiển thị ở đầu danh sách.



Hình 1.1. Minh họa dòng cập nhật trên Twitter

Mỗi khi người dùng đăng nhập vào trang Twitter, hệ thống sinh ra danh sách các dòng cập nhật từ nhiều nơi, dựa vào kết nối following của người đó. Danh sách cập nhật có thể nhỏ hoặc lớn, phụ thuộc vào việc cập nhật tin của các bạn bè. Nếu số lượng các cập nhật trong danh sách vượt quá ngưỡng cho phép, các cập nhật này không thể hiển thị trên một màn hình và người dùng cần phải thực hiện thao tác cuộn xuống. Một danh sách các cập nhật mà người dùng nhìn thấy trên một màn hình được gọi là một *trang*. Khi lượng cập nhật càng nhiều thì số lượng trang sẽ càng lớn, người dùng sẽ phải thực hiện cuộn xuống nhiều lần nếu muốn xem hết các cập nhật đó. Trong trường hợp các cập nhật mà người dùng quan tâm ở các trang cuối, người dùng rất dễ bỏ qua. Nếu các cập nhật mà người dùng này quan tâm được đưa lên các trang đầu thì sẽ giảm thiểu được số lần thực hiện thao tác cuộn đó, tiết kiệm được thời gian và tăng sự yêu thích mạng xã hội này của người dùng.

L. Hong và cộng sự (2012) [14] đã khẳng định phương pháp xếp hạng là phù hợp để giải quyết vấn đề này của người dùng. Xếp hạng lại các dòng cập nhật cho ra một danh sách theo mức độ giảm dần sự quan tâm và khi hiển thị các cập nhật đó theo danh sách này, người dùng sẽ rút ngắn được thời gian của mình. Như vậy, danh sách đó chính

là một sự tư vấn cho người dùng nên xem các cập nhật ở đầu danh sách trước các cập nhật ở phía cuối danh sách. Hay nói cách khác là tư vấn cho người dùng bằng cách đưa các cập nhật có độ quan tâm cao lên đầu danh sách. Để thực hiện được việc sắp xếp này, công việc cần thiết là đi tìm lời giải của *bài toán xếp hạng dòng cập nhật.* Mục tiếp theo của luận văn sẽ phát biểu tường minh bài toán đó.

1.2. Bài toán xếp hạng dòng cập nhật

Bài toán xếp hạng dòng cập nhật là bài toán sắp xếp các cập nhật đến trang của mỗi người dùng. Trước khi phát biểu về bài toán này trên mạng xã hội Twitter, chúng tôi đưa ra một số định nghĩa để tường minh hơn về bài toán.

1.2.1. Một số định nghĩa

Trong bài toán xếp hạng dòng cập nhật trên mạng xã hội Twitter, chúng tôi đưa ra một số định nghĩa như sau:

- Dòng trên mạng xã hội Twitter được hiểu là dòng cập nhật của người dùng. Mỗi người dùng có các tweet mới (các cập nhật) đăng bởi các bạn bè trên trang của họ, đó là dòng cập nhật của họ.
- Xếp hạng dòng trên mạng xã hội Twitter cơ bản là xếp hạng các tweet mới của mỗi người dùng trên mạng xã hội này.
- Trang là tập hợp các tweet của mỗi người dùng hiển thị trên kích thước một màn hình máy tính mà không cần thao tác di chuyển xuống, kéo thanh cuộn.
- Quy ước các tweet được hiển thị theo thứ tự giảm dần và chia thành các trang trên màn hình. Di chuyển giữa các trang bằng thao tác cuộn xuống hay cuộn lên.
- Để phân biệt các tweet được người dùng u_i quan tâm với các tweet không được quan tâm, ta đưa ra định nghĩa interesting Tweet (InT) và not interesting tweet (NinT). InT là các tweet được người dùng quan tâm, sự quan tâm thể hiện ở việc u_i đó thực hiện các hành động retweet, reply và favorite. Ngược lại, NinT là các tweet không được người dùng thực hiện retweet, reply hay favorite.

1.2.2. Bài toán xếp hạng dòng cập nhật

Bài toán xếp hạng dòng trên mạng xã hội Twitter là bài toán sắp xếp các tweet xuất hiện trong mỗi trang người dùng theo mức độ quan tâm của người dùng đó.

Ta có:

- Tập các người dùng trên mạng xã hội Twitter là $U = \{u_i\}, i = 1, N$
- Tập các người dùng mà u_i following là $U_i = \{u_{i'}\}, i' = 1, n\ (i \neq i')$

- Tập các tweet hiển thị trên timelines của u_i là $T_{u_i} = \{t_{u_i j}\}$. Đây là tập hợp các tweet do các người dùng trong tập U_i đăng lên Twitter.

Nhiệm vụ của bài toán là sắp thứ tự các tweet t_k theo mức độ quan tâm của người dùng u_i. Bài toán được phát biểu như sau:

Input: Các tweet mới đưa lên trên trang của người dùng u_i.

Output: Danh sách các tweet đó theo thứ tự giảm dần mức độ quan tâm của người dùng u_i.

1.3. Hướng tiếp cận giải quyết bài toán

Để giải quyết một bài toán xếp hạng các dòng cập nhật hay các tweet mới đến của mỗi người dùng, hoàn toàn có thể áp dụng phương pháp xếp hạng đã được nghiên cứu trước đó dù bài toán này không có câu truy vấn. Một trong các hướng giải quyết gần đây là kĩ thuật học máy để học hàm xếp hạng tự động như học xếp hạng [4]. L. Hong và cộng sự [14] đã đề cập tới một mô hình giải bài toán xếp hạng cập nhật trên mạng xã hội LinkedIn, có liên quan tới phương pháp học tính hạng. D. P. Rout (2015) [15] cũng đưa ra vấn đề xếp hạng các dòng cập nhật trên Twitter timelines theo hướng học xếp hạng. Trong [1], chúng tôi nghiên cứu và áp dụng phương pháp học tính hạng cùng mô hình chủ đề ẩn được sử dụng để làm giàu đặc trưng dữ liệu vào bài toán trên. Trong luận văn, chúng tôi nâng cao mô hình xếp hạng của mình bằng cách áp dụng độ ảnh hưởng của người dùng vào làm giàu đặc trưng vì độ ảnh hưởng của người dùng được đánh giá là rất hữu ích trong hệ tư vấn… [8, 9]. Do vậy, đây sẽ là một đặc trưng quan trọng góp phần vào nâng cao mô hình xếp hạng. Đặc trưng này được tìm ra dựa vào luật kết hợp [10].

Học tính hạng là một trong các phương pháp xếp hạng đang được nghiên cứu mạnh trong những năm gần đây [3, 4, 12] và có thể sử dụng vào bài toán này nếu coi các dòng thông tin của mỗi người dùng tương ứng với các kết quả của truy vấn. Vì vậy, chúng tôi sử dụng phương pháp học xếp hạng để thực hiện sắp xếp các tweet mới đến của mỗi người dùng. Danh sách kết quả cũng được coi như là một tư vấn cho người dùng, chúng tôi sử dụng lọc nội dung trong các hệ thống tư vấn cho bài toán này với chủ định đi tìm các yếu tố liên quan tới nội dung tweet mà người dùng quan tâm. Mục đích để tư vấn tweet tương tự cho người dùng xem trước bằng cách gán vị trí đầu vào tweet đó trong danh sách và ngược lại. Nội dung của tweet qua sự phân bố xác suất của nội dung theo các chủ đề để tính hạng đã được chứng minh là có hiệu quả trong bài toán xếp hạng [1]. Ngoài nội dung, sự ảnh hưởng của bạn bè người dùng cũng có thể ảnh hưởng đến thứ hạng của các tweet. Lí giải đơn giản rằng, khi một người bạn thân của người dùng này đăng một tweet thì sự quan tâm của người dùng này sẽ cao hơn so với một người lạ đăng

tweet. Việc sử dụng độ ảnh hưởng của người dùng thể hiện qua các luật kết hợp vào bài toán xếp hạng hoàn toàn chưa được nghiên cứu trước đây.

Một cách tổng quan, với mục đích nâng cao mô hình xếp hạng dòng mà chúng tôi đã đề xuất trong [1], chúng tôi thực hiện áp dụng độ ảnh hưởng của người dùng dựa trên luật kết hợp vào làm giàu đặc trưng. Đây là một hướng mở rộng mà hoàn toàn do luận văn đưa ra và thực hiện, vận dụng từ những kiến thức về học xếp hạng, mô hình chủ đề ẩn và luật kết hợp.

Như giới thiệu ở trên, học xếp hạng là cách sắp xếp dựa trên học máy, là một trong những hướng nghiên cứu được quan tâm nhiều. Đến nay có khá nhiều phương pháp học xếp hạng như SVM-rank, RankRLS, SoftRank, CRR... [4]. Trong luận văn, chúng tôi tiếp tục áp dụng phương pháp học xếp hạng CRR [5] cho bài toán xếp hạng dòng với giả thiết rằng với mỗi người dùng, tập tweet mới được coi là kết quả của một khóa truy vấn.

Về mô hình chủ đề ẩn, hiện tại đã có nhiều mô hình cải biên LDA áp dụng vào bài toán tương tự như Author-Topic Model [16], Twitter-User model [17]. Tuy nhiên, để tập trung vào việc xác minh tính hiệu quả của việc áp dụng độ ảnh hưởng của người dùng qua luật kết hợp, chúng tôi tiếp tục sử dụng LDA để chỉ số hóa nội dung của tweet. Mỗi tweet được coi là một tài liệu, mô hình này sẽ tìm ra các chủ đề ẩn theo xác suất phân phối các chủ đề trên tweet đó. Sự phân phối xác suất này thể hiện được nội dung mà tweet nói đến.

Luật kết hợp (Association Rule - AR) là khái niệm khá phổ biến trong lĩnh vực khai phá dữ liệu [11]. Mục đích của luật kết hợp là tìm ra các mối quan hệ trong khối lượng lớn dữ liệu dựa trên khai phá tập phổ biến (frequent itemset) [18]. Trong [10], F. Erlandsson và các cộng sự đã coi mỗi người dùng là một đối tượng trong tập các người dùng trên mạng xã hội Twitter. Từ đó tìm các luật kết hợp giữa các người dùng, để tìm ra những người dùng có độ ảnh hướng lớn. Dựa trên kết quả này, chúng tôi thực hiện tìm các luật kết hợp dựa vào thuật toán Apriori [11] và xem xét mỗi tweet có bao nhiêu luật kết hợp liên quan tới. Sau đó chúng tôi sử dụng số lượng luật kết hợp liên quan đại diện cho độ ảnh hưởng của người dùng. Như đã nói, độ ảnh hưởng của người dùng được sử dụng để làm giàu đặc trưng người dùng của tweet tham gia vào việc tính hạng.

1.4. Ý nghĩa của bài toán xếp hạng dòng

Các cập nhật xã hội trên trang của mỗi người dùng cung cấp cơ hội cho chúng ta truy cập thông tin nhanh chóng, nhưng khi số lượng bạn bè của chúng ta là một con số khá lớn, lượng cập nhật sẽ trở nên khổng lồ. Do đó, để giải quyết vấn đề tràn ngập thông tin cho người dùng, bài toán xếp hạng các dòng cập nhật được đặt ra trên mạng xã hội Twitter. Đây là một bài toán sắp xếp các tweet mới đến trên trang của mỗi người dùng. Kết quả của bài toán là sự tư vấn cho người dùng, giúp họ nhanh chóng hơn trong việc

nắm bắt các thông tin mình quan tâm và tiết kiệm thời gian cho bản thân. Mặt khác, sự tư vấn cho người dùng có kết quả tốt sẽ mang lại sự yêu thích của người dùng với mạng xã hội và số lượng người tham gia mạng sẽ tăng lên đáng kể.

1.5. Tóm tắt chương 1

Trong chương 1, luận văn đã trình bày tổng quan về mạng xã hội Twitter và nội dung liên quan tới dòng cập nhật. Luận văn cũng đã nêu lên được vấn đề bất lợi cho người dùng khi bị tràn ngập thông tin và phát biểu được bài toán xếp hạng các dòng cập nhật cùng hướng tiếp cận để giải quyết bài toán. Ngoài ra, luận văn cũng đã nêu lên ý nghĩa của bài toán này.

Chương tiếp theo, chúng tôi thực hiện chi tiết hóa nền tảng kiến thức liên quan về học xếp hạng, mô hình chủ đề ẩn và luật kết hợp. Đồng thời, chúng tôi trình bày thuật toán học xếp hạng, phương pháp mô hình chủ đề ẩn cũng như thuật toán tìm luật kết hợp được lựa chọn để xây dựng mô hình xếp hạng dòng.

Chương 2.

CÁC PHƯƠNG PHÁP HỌC XẾP HẠNG, MÔ HÌNH CHỦ ĐỀ ẨN VÀ LUẬT KẾT HỢP

Chương này trình bày các nội dung nền tảng liên quan tới mô hình giải quyết bài toán. Mục đầu tiên trình bày nội dung cơ bản về xếp hạng dòng, các phương pháp học xếp hạng và các phương pháp đánh giá xếp hạng. Mục tiếp theo giới thiệu phương pháp làm giàu đặc trưng dựa trên mô hình chủ đề ẩn. Mục sau đó trình bày luật kết hợp và thuật toán sinh luật kết hợp. Mục cuối cùng trình bày nội dung ý tưởng khai thác đặc trưng chủ đề ẩn và đặc trưng độ ảnh hưởng của người dùng dựa trên luật kết hợp trong học xếp hạng dòng của mô hình xếp hạng do luận văn đề xuất.

2.1. Một số nội dung cơ bản về Xếp hạng dòng

2.1.1. Giới thiệu

Xếp hạng nói chung được hiểu là sự sắp xếp. Nhiều ứng dụng, phần mềm có sự sắp xếp, đơn giản như MS Excel, MS Dos, sự sắp xếp theo chiều tăng hay giảm của các dữ liệu... hay phức tạp hơn, trong các máy tìm kiếm, sắp xếp các kết quả trả về sao cho phù hợp... Đặc biệt, sắp xếp các *dòng thông tin mới* (tweet mới) trên mạng xã hội Twitter trên timelines của mỗi người dùng mang tính cá nhân và tư vấn cao. Đây chính là Xếp hạng dòng và cũng được coi là Xếp hạng đối tượng (với đối tượng là Tweet). Công việc thiết yếu là sắp xếp các đối tượng tweet của mỗi người dùng theo sự giảm dần mức độ quan tâm của mỗi người dùng đó. Mỗi đối tượng tweet cần xác định giá trị thứ hạng thể hiện mức độ quan tâm của người dùng với nó. Do vậy, để xếp hạng các đối tượng, ta cần xác định hàm tính giá trị thứ hạng, gọi là *hàm tính hạng*. Mỗi đối tượng gồm có các đặc trưng là những chi tiết của bản thân đối tượng đó. Hàm tính hạng là sự kết hợp của các đặc trưng này.

2.1.2. Học xếp hạng

Học xếp hạng là một loại học máy giám sát hoặc bán giám sát, trong đó mục tiêu là để tự động xây dựng một mô hình xếp hạng từ dữ liệu huấn luyện là tập dữ liệu đã có xếp hạng đúng. Học xếp hạng là một trong các phương pháp điển hình trong việc xếp hạng đối tượng đang nhận được khá nhiều sự quan tâm của các nhà nghiên cứu. Như đã giới thiệu, chúng tôi sử dụng học xếp hạng cho bài toán đặc biệt Xếp hạng dòng (không có câu truy vấn) với giả thiết tất cả các tweet mới tương ứng với tập kết quả trả về với một câu truy vấn.

Như đã đề cập trong [1], các thuật toán học xếp hạng đều có hai nhiệm vụ chính: (1) xây dựng hàm tính hạng, (2) tính toán thứ hạng của đối tượng mới. Các nhiệm vụ có đầu vào và đầu ra khác nhau, cụ thể như sau:

- *Xây dựng hàm tính hạng*
  - Đầu vào: Tập các đối tượng có sẵn thứ tự đúng và các đặc trưng
  - Đầu ra: Hàm tính hạng
- *Tính toán thứ hạng đối tượng mới*
  - Đầu vào: Tập đối tượng mới và hàm tính hạng
  - Đầu ra: Thứ hạng của mỗi đối tượng

Hàm tính hạng thu được từ các thuật toán học được sử dụng để tính hạng cho các tài liệu mới: cho một tập các đối tượng mới cần được sắp xếp thứ tự, hàm tính hạng thu được sẽ tính toán ra thứ hạng của mỗi đối tượng trong danh sách đó. Để biết được độ chính xác của hàm tính hạng này, tập dữ liệu kiểm tra được sử dụng. Các độ chính xác thu được nhờ việc áp dụng các phương pháp đánh giá xếp hạng.

Một số hướng tiếp cận của học xếp hạng.

TY. Liu [4] đã phân tích các thuật toán học xếp hạng và chỉ ra sự phân chia các thuật toán đó theo các hướng tiếp cận như sau:

- **Hướng tiếp cận Pointwise**
  Theo hướng này, các đối tượng x_i trong dữ liệu học có một điểm số hay thứ tự y_i. Tiếp đó, học xếp hạng có thể được xấp xỉ bởi hồi quy (hồi quy có thứ tự). Với D = {(x_i, y_i)}, hàm tính hạng h(x_i) thỏa mãn, r(x_i) = y_i. Một số thuật toán học xếp hạng như: OPRF [4], SLR [19]…
- **Hướng tiếp cận Pairwise**
  Có D = {(x_i, x_j)} là tập các cặp đối tượng được sắp thứ tự, với mỗi cặp (x_i, x_j) có thứ hạng của x_i cao hơn thứ hạng của x_j, hay x_i phù hợp hơn x_j: $x_i > x_j$). Tìm r(x):

$$\forall (x_i, x_j) \in S \text{ có } x_i > x_j \text{ thì } r(x_i) > r(x_j)$$

  Một số thuật toán học xếp hạng như SVM-rank, RankRLS …

- **Hướng tiếp cận Listwise**
  Các thuật toán theo hướng này cố gắng trực tiếp sắp xếp tất cả các đối tượng trong dữ liệu học. Điều này thực sự khó khăn. Khi thứ hạng của K đối tượng đầu tiên được xác định thì tất cả các đối tượng khác đều có hạng thấp hơn. Với D = {x_1, x_2…, x_m} có sắp thứ tự: $x_1 > x_2 > \ldots > x_m$, tìm hàm tính hạng r(x) sao cho $r(x_1) > r(x_2) > \ldots > r(x_m)$. Một số thuật toán học xếp hạng như ListMLE, PermuRank …

Sử dụng phương pháp học xếp hạng để xây dựng mô hình tính hạng, cần xây dựng tập dữ liệu huấn luyện là đầu vào của quá trình học. Việc xây dựng cũng như định dạng của dữ liệu huấn luyện, luận văn sẽ đề cập trong phần sau. Ngay sau đây, chúng tôi sẽ nói về các thuật toán học xếp hạng cụ thể như SVM-rank và CRR. Thuật toán SVM-rank là một thuật toán khá phổ biến và thuật toán CRR là kết quả của ý tưởng kết hợp thuật toán xếp hạng (SVM-rank) với hồi quy tuyến tính. Để hiểu hơn về sự kết hợp trong CRR, chúng tôi nghiên cứu và áp dụng thuật toán này vào mô hình đề xuất của mình để xây dựng mô hình tính hạng cho mỗi người dùng.

2.1.3. Các phương pháp học xếp hạng điển hình

2.1.3.1. Phương pháp SVM-rank

Xếp hạng SVM (SVM-rank) [20] là một ứng dụng của máy véc-tơ hỗ trợ (Support vector machine) được sử dụng để giải quyết bài toán xếp hạng bằng việc sử dụng thuật toán học giám sát SVM. SVM-rank được Joachims công bố năm 2002 với mục đích cải thiện hiệu suất của các công cụ tìm kiếm trên Internet. SVM-rank là thuật toán học xếp hạng theo hướng tiếp cận pairwise. Chẳng hạn, ta có tập sắp thứ tự D = {(d_1, 3), (d_2, 1), (d_3, 1)}, khi đó có các cặp so sánh thứ tự (d_2, d_1) và (d_3, d_1), cặp (d_2, d_3) không xác định thứ tự so sánh.

Giải quyết bài toán theo hướng tiếp cận Pairwise, xếp hạng được đưa về *bài toán phân lớp cho từng cặp đối tượng*. Với X là tập các đặc trưng của từng đối tượng và R là tập các thứ hạng, ta có ánh xạ thể hiện hàm tính hạng: $X \to R$, $x_i > x_j \leftrightarrow r(x_i) > r(x_j)$

$$r(x) = w^T x \tag{2.1}$$

Tư tưởng chính của SVM [21] là xác định biên (siêu phẳng) chia không gian các đối tượng cần xếp hàng thành hai nửa và tìm siêu phẳng tốt nhất (tối ưu) mà khoảng cách từ siêu phẳng tới đối tượng gần nhất trong cả 2 tập phân chia là lớn nhất.

Với dữ liệu có thể phân tách tuyến tính, siêu phẳng có dạng: $w^T x + b = 0$. Từ đây, có thể thấy mối quan hệ giữa hàm tính hạng $r(x)$ và siêu phẳng. Do đó, dựa vào phương pháp SVM, tìm được siêu phẳng sẽ suy ra hàm tính hạng $r(x)$. Đây chính là tư tưởng chính của SVM-rank.

Các công cụ SVM^{light}, SVM^{rank} do T. Joachims cung cấp[2] cho người dụng lựa chọn học xếp hạng đối tượng dựa vào phương pháp này.

Nhiều phương pháp dựa vào tối ưu SVM, chẳng hạn [5, 22]…Trong [5], sự kết hợp xếp hạng dựa trên SVM-rank với hồi quy, Sculley đưa ra thuật toán CRR sẽ được trình bày trong phần tiếp theo.

2.1.3.2. Phương pháp CRR

Trong [5], D.Sculley đưa ra nhận định rằng mô hình hồi quy tốt sẽ cho xếp hạng tốt, nhưng mô hình hồi quy chưa thực sự hoàn hảo có thể dẫn tới hiệu quả của xếp hạng là không tốt. Tương tự với mô hình xếp hạng, trong trường hợp không tốt, mô hình xếp hạng có thể cho kết quả không cao. Tác giả tìm ra phương pháp kết hợp cho hiệu quả tốt ở cả hồi quy và xếp hạng. Tư tưởng chính của phương pháp này là xây dựng mô hình tính hạng dựa trên mô hình hồi quy tuyến tính và mô hình tính hạng pairwise (sử dụng SVM-rank):

❖ ***Phương thức hồi quy***

Mục tiêu của hồi quy có giám sát là học mô hình w để dự đoán giá trị mục tiêu thực $y' \in R$ cho véc-tơ đặc trưng x, sử dụng hàm dự đoán f(w,x), có sai số nhỏ và hàm loss function l(y,y') (loss function là hàm tính độ sai lệch giữa y và y').

Mục tiêu để rủi ro cho mô hình là thấp nhất là làm cho sai số nhỏ, với loss function được cho bởi công thức:

$$L(w,D) = \frac{1}{|D|} \sum_{(x,y,q)\in D)} l(y, f(w,x)) \qquad (2.2)$$

Ở đây, l(y,y') là hàm sai số cho từng đối tượng và được tính theo hàm logistic loss [5, 23], với y' = f(w,x) và y là giá trị đúng của x.

Công thức thể hiện sai số nhỏ nhất với mô hình w như sau:

$$min_{w\in R^m} L(w,D) + \frac{\lambda}{2}||w||_2^2 \qquad (2.3)$$

Logistic loss [5, 23] thường được sử dụng trong hồi quy tuyến tính, phương thức này thường sử dụng trong phân lớp, nhưng nó cũng có thể là phương thức cho hồi quy trong việc dự đoán giá trị thực. Logistic loss như sau:

$$y \in [0,1], y' \in [0,1], l(y,y') = ylogy' + (1-y)\log(1-y'). \qquad (2.4)$$

[2] https://www.cs.cornell.edu/people/tj/svm_light/svm_rank.html

Đây cũng là hàm lồi và Hàm dự đoán $(w, x) = \frac{1}{1+e-(w,x)}$. Hàm biến đổi khi tính theo hướng Pairwise là $t(y) = \frac{1+y}{2}$. Giá trị của t (y – y') luôn nằm trong [0, 1] khi y, y' cũng thuộc đoạn đó.

❖ ***Phương thức xếp hạng.***

Mục tiêu của phương thức này là học mô hình w với mức độ sai số nhỏ trong tập dữ liệu D, sử dụng hàm dự đoán f(w,x) cho mỗi véc-tơ đặc trưng trong tập đó, đối với loss function trên cơ sở xếp hạng. Học xếp hạng theo hướng tiếp cận Pairwise được tác giả lựa chọn và sử dụng SVM-rank.

Có tập dữ liệu D, tập huấn luyện D, ta mở rộng tới tập P là các tập các cặp ứng viên. Quá trình học sẽ thực hiện với tập P các cặp véc-tơ đó.

Các cặp ứng viên có dạng (a, y_a, q_a), (b, y_b, q_b) với $y_a \neq y_b$ và $q_a \neq q_b$. Nếu $y_a > y_b$, thì a phù hợp hơn b, tương đương với thứ hạng của a trước thứ hạng của b. Với tập dữ liệu cho trước Dlearn, $|P| = O(|D|^2)$, nhưng kết quả có thể là $|P| << O(|D|^2)$.

Với tập P như trên, cần tìm mô hình w tối ưu hàm mục tiêu Pairwise, thể hiện trong công thức:

$$min_{w \in R^m} L(w, P) + \frac{\lambda}{2}||w||_2^2 \qquad (2.5)$$

Ở đây, loss function L (w, P) được định nghĩa qua các cặp véc-tơ khác nhau từ P:

$$L(w, P) = \frac{1}{|P|} \sum_{(((a,y_a,q_a),(b,y_b,q_b)) \in P)} l(t(y_a - y_b), f(w, a - b)) \qquad (2.6)$$

Hàm biến đổi t(y) sẽ trả ra một giá trị chênh lệch y, hàm l được tính theo square loss [5].

Square loss thể hiện sự chênh lệch giữa giá trị đúng y và giá trị dự đoán y' theo hàm bình phương: $l(y, y') = (y - y')^2$. Đây là một hàm lồi và hàm biến đổi t(y) được xác định: $t(y) = y$. Hàm dự đoán $f(x) = (w, x)$

❖ ***Sự kết hợp xếp hạng và hồi quy.***

Phương pháp kết hợp xếp hạng và hồi quy nhằm tối ưu kết quả với sai số hồi quy L (w, D) và sai số xếp hạng Pairwise L (w, P). Sự kết hợp thể hiện trong biểu thức:

$$\min_{w \in R^m} \quad \alpha L(w, D) + (1 - \alpha) L(w, P) + \frac{\lambda}{2}||w||_2^2$$

Tham số $\alpha \in [0, 1]$ thể hiện sự điều chỉnh qua lại giữa sai số hồi quy và sai số xếp hạng pairwise. Nếu lấy $\alpha = 1$ sẽ trở về hồi quy chuẩn, lấy $\alpha = 0$ sẽ trờ về vấn đề xếp hạng pairwise thông thường và không có sự kết hợp nào giữa hồi quy và xếp hạng trong các trường hợp này. Thuật toán D. Sculley đưa ra (thuật toán CRR) được trình bày như Hình 2.1:

Cho trước: α, λ, dữ liệu huấn luyện D và số lần lặp t.

$w_o \leftarrow \emptyset$
$for\ i = 1\ to\ t$
    $lấy\ ngẫu\ nhiên\ số\ z\ từ\ [0,1]$
    $if\ z < \alpha\ then$
        $(x, y, q) \leftarrow RandomExample(D)$
    $else$
        $\big((a, y_a, q), (b, y_b, q)\big) \leftarrow RandomCandidatePair(P)$
        $x \leftarrow (a - b)$
        $y \leftarrow t(y_a - y_b)$
    $end\ if$
    $\eta_i \leftarrow \frac{1}{i\lambda}$
    $w_i \leftarrow StochasticGradientStep\big(w_{i-1}, x, y, \lambda, \eta_i\big)$
$end\ for$
$return\ w_t$

Hình 2.1. Thuật toán CRR [5]

Thuật toán thuần cho việc tối ưu sự kết hợp sẽ liệt kê đầy đủ tập các cặp ứng viên P. Số thành phần thuộc P là bình phương số thành phần thuộc D hay $|P|=|D|^2$ nên khó thực hiện ở tập dữ liệu lớn. T. Joachims [22] đã đưa ra phương thức cho độ phức tạp O(|D|log|D|).

Thuật toán đưa ra phương thức tối ưu sự kết hợp hồi quy và xếp hạng sử dụng phương pháp *Stochastic gradient descent* [5]. Phương pháp này giúp tối thiểu hàm mục tiêu, vấn đề xuất hiện trong học mô hình.

Phương thức StochasticGradientStep trả ra kết quả khác nhau với các hàm sai số khác nhau. Chẳng hạn, với square loss, y ∈ R, phương thức này trả ra $\big(1 - \eta_i\lambda\big)w_{i-1} + \eta_i x(y - (w_{i-1}, x))$. Với logistic loss, giả sử y∈{0,1}, phương thức trả ra $\big(1 - \eta_i\lambda\big)w_{i-1} + \eta_i x\left(y - \frac{1}{1+e^{-(w_{i-1},x)}}\right)$.

Như vậy, mô hình w được trả ra là mô hình học tính hạng.

2.1.4. Phương pháp đánh giá xếp hạng dòng

Để đánh giá chất lượng một xếp hạng, không chỉ các độ đo thông dụng trong học máy như độ chính xác (precision), độ hồi tưởng (recall), độ đo F không sử dụng mà độ đo Rooted Mean Squared Error (RMSE) – được dùng để đo độ chính xác của giá trị dự đoán - cũng vậy. Các bài về lọc cộng tác như [24, 25, 26], cũng thảo luận về các mặt hạn chế của các loại thước đo này. L. Hong và cộng sự [14] đã phân tích và lựa chọn các thước đo phổ biến dựa trên xếp hạng trong thu hồi thông tin (Information Retrieval). Đó là độ chính xác mức k (Precision@K – P@K) và độ chính xác trung bình (Mean Average Precision – MAP). Trong luận văn, chúng tôi cũng sử dụng các độ đo này để đánh giá mô hình xếp hạng.

Trước khi trình bày các thước đo, chúng tôi đưa ra một ví dụ; sau đó vừa trình bày vừa thực hiện đánh giá với các thước đo đó.

Ví dụ: Giả sử 6 đối tượng được xếp hạng tương ứng là: c, a, e, b, d

Một xếp hạng của các đối tượng cần đánh giá là: c, b, e, a, d.

❖ **Độ chính xác mức K:** *P@K*

Độ chính xác xếp hạng ở mức K - Precision@K (P @K): độ chính xác của K đối tượng đầu bảng xếp hạng. Xác định số đối tượng đúng ở K vị trí đầu tiên của xếp hạng và gọi là Match@K, và độ chính xác mức K:

$$\mathrm{P@K} = \frac{\mathrm{Match@K}}{\mathrm{K}} \tag{2.7}$$

Với ví dụ trên ta có: P@1 = 1/1, P@2 = 1/2, P@3 = 2/3; P@4 = 2/4; P@5 = 3/5;

❖ **Độ chính xác trung bình: MAP**

Độ chính xác trung bình là giá trị trung bình của các P@K tại các mức K có đối tượng đúng. Gọi I(K) là hàm xác định đối tượng ở vị trí hạng K nếu đúng I(K) =1 và ngược lại I(K) = 0. Độ chính xác trung bình:

$$AP = \frac{\sum_{K=1}^{n} P@K \times I(K)}{\sum_{j=1}^{n} I(j)} \tag{2.8}$$

Với n là số đối tượng được xét.

MAP là độ chính xác trung bình trên N xếp hạng. (N truy vấn, mỗi truy vấn có một thứ tự xếp hạng kết quả tương ứng). MAP được tính như sau:

$$MAP = \frac{\sum_{i=1}^{N} AP_i}{N} \quad (2.9)$$

Với ví dụ trên (N = 1), ta có $MAP = AP_1 = \frac{1}{3}\left(\frac{1}{1} + \frac{2}{3} + \frac{3}{5}\right) \approx 0.67$

2.2. Mô hình chủ đề ẩn

2.2.1. Giới thiệu

Mô hình chủ đề ẩn [6] là mô hình xác suất phân phối các chủ đề ẩn trên mỗi tài liệu. Chúng được xây dựng dựa trên ý tưởng rằng mỗi tài liệu có một xác suất phân phối vào các chủ đề, và mỗi chủ đề là sự phân phối kết hợp giữa các từ khóa. Hay nói cách khác, ý tưởng cơ bản là dựa trên việc coi tài liệu là sự pha trộn của các chủ đề. Biểu diễn các từ và tài liệu dưới dạng phân phối xác suất có lợi ích rất lớn so với không gian vector thông thường.

Ý tưởng của các mô hình chủ đề ẩn là xây dựng những tài liệu mới dựa theo phân phối xác suất. Trước hết, để tạo ra một tài liệu mới, cần chọn ra một phân phối những chủ đề cho tài liệu đó, điều này có nghĩa tài liệu được tạo nên từ những chủ đề khác nhau, với những phân phối khác nhau. Tiếp đó, để sinh các từ cho tài liệu ta có thể lựa chọn ngẫu nhiên các từ dựa vào phân phối xác suất của các từ trên các chủ đề. Một cách hoàn toàn ngược lại, cho một tập các tài liệu, có thể xác định một tập các chủ đề ẩn cho mỗi tài liệu và phân phối xác suất của các từ trên từng chủ đề. Nhận thấy sự phù hợp của khía cạnh này, luận văn sử dụng chúng cho mô hình đề xuất.

Sử dụng mô hình chủ đề ẩn để biết được xác suất các chủ đề ẩn trong tweet đang xét. Xác suất đó được biểu diễn theo vectorr thể hiện sự phân bố nội dung của tweet trên các chủ đề theo xác suất. Từ đó, sử dụng vector này làm đặc trưng nội dung cho tweet để xếp hạng.

2.2.2. Phương pháp mô hình chủ đề ẩn

Hai phân tích chủ đề sử dụng mô hình ẩn là Probabilistic Latent Semantic Analysis (pLSA) và Latent Dirichlet Allocation (LDA):

- pLSA là một kỹ thuật thống kê nhằm phân tích những dữ liệu xuất hiện đồng thời [7]. Phương pháp này được phát triển dựa trên LSA [6], mặc dù pLSA là một bước quan trọng trong việc mô hình hóa dữ liệu văn bản, tuy nhiên nó vẫn còn chưa hoàn thiện ở chỗ chưa xây dựng được một mô hình xác suất tốt ở mức độ tài liệu. Điều đó dẫn đến vấn đề gặp phải khi phân phối xác suất cho một tài liệu nằm ngoài tập dữ liệu học, ngoài ra số lượng các tham số có thể tăng lên một cách tuyến tính khi kích thước của tập dữ liệu tăng.

- LDA là một mô hình sinh xác suất cho tập dữ liệu rời rạc dựa trên phân phối Dirichlet, được D. M. Blei và cộng sự phát triển vào năm 2003 [6, 27]. LDA được xây dựng dựa trên ý tưởng: mỗi tài liệu là sự trộn lẫn của nhiều chủ đề (topic).

LDA là một mô hình hoàn thiện hơn so với PLSA và có thể khắc phục được những nhược điểm đã nêu trên. Do đó, chúng tôi chọn loại mô hình chủ đề ẩn này để sử dụng trong việc xây dựng mô hình tính hạng dòng của luận văn.

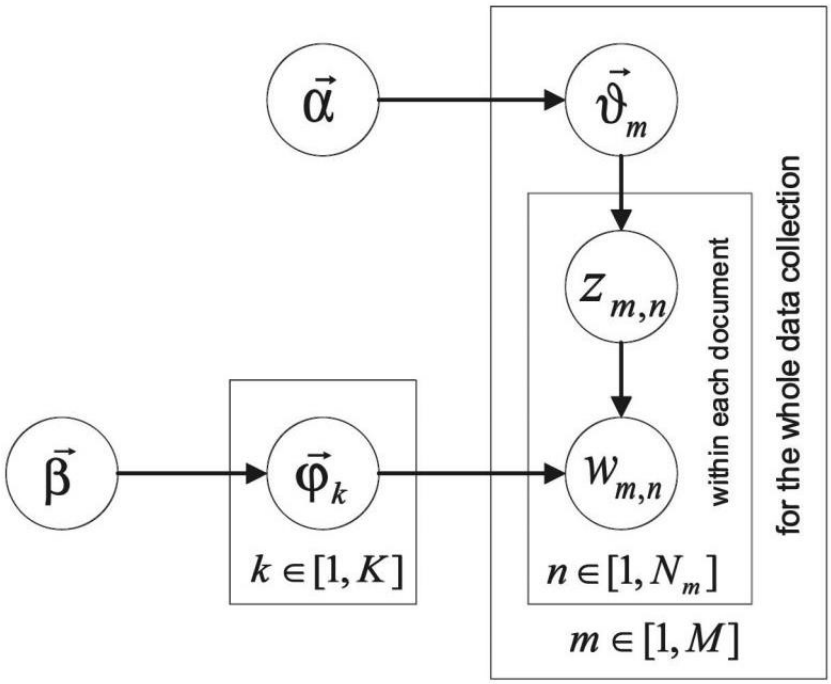
Về bản chất, LDA là một mô hình Bayes phân cấp 3 mức (mức kho ngữ liệu, mức tài liệu và mức từ ngữ). Mỗi tài liệu trong tập hợp được coi là một hỗn hợp xác định trên tập cơ bản các chủ đề. Mỗi chủ đề là một hỗn hợp không xác định trên tập cơ bản các xác suất chủ đề. Về khía cạnh mô hình hóa văn bản, các xác suất chủ đề là một biểu diễn cụ thể, rõ ràng cho một tài liệu. Dưới đây, luận văn sẽ trình bày những nét cơ bản về mô hình sinh trong LDA.

❖ ***Mô hình sinh trong LDA***

Cho trước tập M tài liệu $D = \{d_1, d_2 \ldots d_M\}$, trong đó tài liệu thứ m gồm N_m từ, từ w_i được rút ra từ tập các thuật ngữ $\{t_1, t_2 \ldots t_V\}$, V là số các thuật ngữ.

Quá trình sinh trong mô hình LDA diễn ra như sau:

- Mô hình LDA sinh các từ $w_{m,n}$ có thể quan sát, các từ này được phân chia về các tài liệu.
- Với mỗi tài liệu, một tỉ lệ chủ đề $\vec{\vartheta}_m$ được chọn từ phân bố Dirichlet *(Dir($\vec{\alpha}$))*, từ đó, xác định các từ thuộc chủ đề cụ thể.
- Sau đó, với mỗi từ thuộc tài liệu, chủ đề của từ đó được xác định là một chủ đề cụ thể bằng cách lấy mẫu từ phân bố đa thức *(Mult($\vec{\vartheta}_m$))*.
- Cuối cùng, từ phân bố đa thức (*Mult($\vec{\varphi}z_{m,n}$)*), một từ cụ thể $w_{m,n}$ được sinh ra dựa trên chủ đề đã được xác định. Các chủ đề $\vec{\varphi}z_{m,n}$ được lấy mẫu một lần trong toàn kho ngữ liệu.



Hình 2.2. Mô hình biểu diễn của LDA [27]

Các khối vuông trong hình trên biểu diễn các quá trình lặp.

Các tham số đầu vào bao gồm:

- α và β: tham số mức tập hợp kho ngữ liệu
- $\vec{\vartheta}_m$: phân bố chủ đề trên tài liệu m (tham số mức tài liệu)
- Và $\Theta = \{\vec{\vartheta}_m\}_{m=1}^{M}$: ma trận M x K
- $z_{m,n}$: chỉ số chủ đề của từ thứ n trong tài liệu m (biến mức từ ngữ)
- $\vec{\varphi} z_{m,n}$: phân bố thuật ngữ trên chủ đề cụ thể $z_{m,n}$
- Và $\Phi = \{\vec{\varphi}_k\}_{k=1}^{K}$: ma trận K x V
- $w_{m,n}$: từ thứ n của văn bản n (biến mức từ ngữ)
- M: số lượng các tài liệu
- N_m: số lượng từ trong tài liệu m (độ dài của văn bản sau khi đã loại bỏ stop word)
- K: số lượng các chủ đề ẩn
- $DirvàMult$: phân bố Dirichlet và phân bố đa thức

Vì $w_{m,n}$ phụ thuộc điều kiện vào phân bố $\vec{\varphi}_k$ và $z_{m,n}$ phụ thuộc vào phân bố $\vec{\vartheta}_m$, xác suất để một chỉ mục chủ đề $w_{m,n}$ là một từ t nằm trong phân bố chủ đề trên tài liệu $\vec{\vartheta}_m$ và phân bố từ trên chủ đề (Φ) là:

$$p(w_{m,n} = t|\vec{\vartheta}_m, \Phi) = \sum p(w_{m,n} = t|\vec{\varphi}_k)\, p(z_{m,n} = k|\vec{\vartheta}_m) \quad (2.10)$$

Với xác suất của mỗi thuật ngữ, ta có thể xác định được xác suất chung của tất cả các biến đã biết và biến ẩn với các tham số Dirichlet cho trước:

$$p(\vec{d}_{\mathrm{m}}, \vec{z}_{\mathrm{m}}, \vec{\vartheta}_{\mathrm{m}}, \Phi|\vec{\alpha}, \vec{\beta}) = p(\Phi|\vec{\beta}) \prod_{n=1}^{N\_m} p(w_{\mathrm{m,n}}|\vec{\varphi} z_{m,n}) p(z_{\mathrm{m,n}}|\vec{\vartheta}_{\mathrm{m}})\, p(\vec{\vartheta}_{\mathrm{m}}|\vec{\alpha}) \quad (2.11)$$

Tính tích phân trên $\vec{\vartheta}_{\mathrm{m}}$, Φ và tổng trên $\vec{z}_{\mathrm{m}}$, ta xác định được xác suất của tài liệu $\vec{d}_{\mathrm{m}}$. Khi đã có xác suất của mỗi tài liệu $p(\vec{d}_{\mathrm{m}}|\vec{\alpha}, \vec{\beta})$, xác suất của cả kho ngữ liệu D = {d_1, d_2, …, d_M} là tích của tất cả các xác suất của tất cả các tài liệu nằm trong đó:

$$p(D|\vec{\alpha}, \vec{\beta}) = \prod_{n=1}^{M} p(\vec{d}_m/\vec{\alpha}, \vec{\beta}) \quad (2.12)$$

❖ ***Ước lượng tham số và suy luận thông qua Gibbs Sampling cho mô hình LDA***

Ước lượng tham số cho mô hình LDA bằng tối ưu hóa một cách trực tiếp và chính xác suất của toàn bộ tập dữ liệu là khó có thể thực hiện. Một giải pháp đã được đề ra là sử dụng phương pháp ước lượng xấp xỉ như phương pháp biến phân [6] và lấy mẫu Gibbs [28]. Lấy mẫu Gibbs được xem là một thuật toán nhanh, đơn giản và hiệu quả để huấn luyện LDA.

Một chủ đề được gán cho một từ cụ thể được lấy mẫu theo phân bố đa thức sau:

$$p(z_i = k|\vec{z}_{\neg i}, \vec{w}) = \frac{n_{k,\neg i}^{(t)} + \beta_t}{\left[\sum_{v=1}^{K} n_k^{(v)} + \beta_v\right] - 1} \frac{n_{m,\neg i}^{(k)} + \alpha_k}{\left[\sum_{j=1}^{K} n_m^{(j)} + \alpha_j\right] - 1} \quad (2.13)$$

Trong đó:

- $n_{k,\neg i}^{(t)}$ là số lần từ *t* được gán cho chủ đề *k*, không tính đến lần gán hiện thời;
- $\left[\sum_{v=1}^{V} n_k^{(v)} - 1\right]$ là số từ được gán cho chủ đề k, không tính lần gán hiện thời;
- $n_{m,\neg i}^{(k)}$ là số từ trong tài liệu m được gán cho chủ đề k, không tính lần gán hiện thời;
- $\left[\sum_{j=1}^{K} n_m^{(j)} - 1\right]$ là số từ trong tài liệu m, không kể từ *t*.

Sau khi lấy mẫu Gibbs, giá trị các tham số được xác định, các phân phối ẩn được tính như sau:

$$\varphi_{k,t} = \frac{n_k^{(t)} + \beta_t}{\sum_{v=1}^{V} n_k^{(v)} + \beta_v} \quad (2.14)$$

$$\vartheta_{m,k} = \frac{n_m^{(k)} + \alpha_k}{\sum_{j=1}^{K} n_m^{(j)} + \alpha_j} \quad (2.15)$$

Với mô hình chủ đề ẩn LDA, cho trước một tập các văn bản, LDA tìm xem topic model nào đã sinh ra tập các văn bản trên. Bao gồm:

- Tìm phân phối xác suất trên tập từ đối với mỗi topic

- Tìm phân phối topic của mỗi tài liệu

Trong luận văn, chúng tôi sử dụng phân phối topic của mỗi tài liệu được tìm ra từ LDA để làm đặc trưng nội dung cho việc xây dựng tập huấn luyện cho quá trình học của phương pháp học xếp hạng.

2.3. Luật kết hợp

2.3.1. Giới thiệu

Luật kếp hợp (Association Rule - AR) là lớp các quy tắc quan trọng trong khai phá dữ liệu, được R. Agarwal và cộng sự giới thiệu năm 1993 [29]. Khai phá luật kết hợp được xem là một nhiệm vụ khai phá dữ liệu cơ bản. Mục đích của khai phá luật kết hợp là tìm ra các mối quan hệ đồng xảy ra trong khối lượng lớn dữ liệu. Luật kết hợp không chỉ ứng dụng rộng rãi trong phân tích dữ liệu thị trường [11], mà còn được ứng dụng trong tìm những người dùng có độ ảnh hưởng lớn tới các người dùng khác trên mạng xã hội [10].

Các khái niệm cơ bản của luật kết hợp được tóm tắt như dưới đây.

Cho tập các giao dịch (transactiones) $T = \{t_1, t_2, \dots, t_n\}$, và tập các đối tượng (item) $I = \{i_1, i_2, \dots, i_m\}$. Mỗi giao dịch t_i là tập các item với $t_i \subseteq I$.

Những luật kết hợp này có dạng $X \rightarrow Y, với\ X \subseteq I, Y \subseteq I, và\ X \cap Y = \emptyset$

X (hoặc Y) là một nhóm các item và được gọi là itemset. Một itemset gồm k itemes gọi là k-itemset.

Để đo lường luật kết hợp, độ hỗ trợ (support) và độ tin cây (confidence) là 2 tham số được sử dụng.

Độ hỗ trợ (Support) của luật kết hợp X→Y là tần suất của giao dịch chứa tất cả các item trong cả hai tập X và Y.

$$support = \frac{(X \cup Y).count}{n} \tag{2.16}$$

Trong đó:

n là tổng số giao dịch.

$(X \cup Y).count$ là số giao dịch có X và Y

Độ tin cậy (*confidence)* của luật kết hợp là xác suất xảy ra Y khi đã biết X.

$$confidence = \frac{(X \cup Y).count}{X.count} \tag{2.17}$$

Trong đó:

$(X \cup Y).count$ là số giao dịch có X và Y

$X.count$ là số giao dịch có X

Mục tiêu: Với cơ sở dữ liệu giao dịch T, khai phá luật kết hợp là tìm các luật kết hợp trong T thỏa mãn 2 tiêu chí *minimum support (minsup)* và *minimum confidence (minconf)*. Nói cách khác, cần tìm các luật kết hợp AR sao cho $support(AR) \geq minsup$ và $confidence(AR) \geq minconf$.

Minimum support và *Minimum confidence* gọi là các giá trị ngưỡng (threshold) và phải xác định trước khi sinh các luật kết hợp.

Tập phổ biến là itemset mà tần suất xuất hiện của nó >= minsup.

2.3.2. Thuật toán Apriori

Thuật toán Apriori là một thuật toán phổ biến trong việc tìm tất cả các luật kết hợp thỏa mãn minsup và minconf trong một cơ sở dữ liệu giao dịch. Thuật toán bao gồm 2 bước chính:

Bước 1. **Tạo tất cả các tập phổ biến**: Một tập phổ biến là một tập các item có độ support lớn hơn hoặc bằng minsup

Bước 2. **Tạo tất cả các luật kết hợp mạnh từ các tập phổ biến trong bước 1**: Một luật kết hợp mạnh là luật có độ confidence lớn hơn hoặc bằng minconf

Một tập phổ biến có k phần tử gọi là frequent k-itemset. Trong bước 1, thực hiện tìm frequent k+1-itemset từ frequent k-itemset. Trong bước 2, thực hiện tìm các luật kết hợp trong các frequent k-itemset với $k \geq 2$

Chi tiết các bước được thể hiện ở các mục dưới đây.

2.3.2.1. Tạo các tập phổ biến

Thuật toán Apriori tìm tất cả tập phổ biến bằng cách sử dụng frequent k-itemset để tìm frequent (k+1)-itemset, cho đến khi không có frequent (k+n)-itemset được tìm thấy. Các bước chính như sau:

1. Duyệt toàn bộ cơ sở dữ liệu giao dịch để có được độ support S của 1-itemset, so sánh S với minsup, để có được 1-itemset (L_1)
2. Sử dụng L_{k-1} nối (join) L_{k-1} … để sinh ra các ứng viên (candidate) k-itemset. Loại bỏ các ứng viên itemset không thể là tập phổ biến thu được các candidate k-itemset
3. Duyệt toàn bộ cơ sở dữ liệu giao dịch để có được độ support của mỗi candidate k-itemset, so sánh S với minsup để thu được frequent k –itemset (L_k)
4. Lặp lại từ bước 2 cho đến khi tập candidate itemset rỗng.

Mã giả tạo các tập phổ biến của thuật toán thể hiện trong Hình 2.3 và Hình 2.4.

```
Algorithm Apriori(T)
    1  C1 ← init − pass(T);                          //the first pass over T
2  F1 ← {f | f ∈ C1, f.count/n ≥ minsup};            //n is the no of transactions in T
3  for (k = 2; Fk−1 ≠ ∅; k++) do                     //subsequent passes over T
4      Ck ← candidate − gen(Fk−1);
5      for each transaction t ∈ T do                 //scan the data once
6          for each candidate c ∈ Ck do
7          if c is contained in t then
8          c.count++;
9          endfor
10       endfor
11 Fk ← {c ∈ Ck | c.count/n ≥ minsup}
12 endfor
13 return F ← UkFk;
```

Hình 2.3. Thuật toán Apriori tạo các tập phổ biến [11]

```
Function candidate − gen(Fk−1)
1  C1 ← ∅;                                           //initialize the set of candidates
2  forall f1, f2 ∈ Fk−1                              //find all pairs of frequent itemsets
3      with f1 = {i1, …, ik−2, ik−1}                 //that differ only in the last item
4      and f2 = {i1, …, ik−2, i'k−1}
5      and ik−1 < i'k−1 do                           //according to the lexicographic
       order
6          c ← {i1, …, ik−2, i'k−1}                  //join the two itemsets f1 and f2
7          Ck ← Ck ∪ {c};                            //add the new itemset c to the
           candidate
8          for each (k − 1)-subset s of c do
9              if(s ∉ Fk−1) then
10                 delete c from Ck;                 //delete c from the candidate
11         endfor
12 endfor
13 return Ck;                                        //return the generated candidates
```

Hình 2.4. Hàm candidate-gen [11]

2.3.2.2. Tạo luật kết hợp

Sử dụng các tập phổ biến để tạo tất cả các luật kết hợp theo các bước chính sau:

1. Với mỗi tập phổ biến f, chúng ta tìm tất cả tập con không rỗng của f.
2. Với mỗi tập con α, chúng ta có rule $(f-\alpha)\rightarrow\alpha$ nếu $confidence = (f.count)/((f-\alpha).count) \geq minconf$. (Ở đây, $f.count$ hoặc $(f-\alpha).count$ là độ support của f hoặc $(f-\alpha)$).

Mã giả tạo các luật kết hợp thể hiện trong Hình 2.5.

```
ALgorithm genRules(F)                          //F is the set of all frequent itemset
1 for each frequent k − itemset f_k in F, k ≥ 2 do
2   output every 1 − item consequent rule of f_k with confidence ≥
    minconf and support ← f_k.count/n //n is the total number of transactions in T
3   H_1 ← {consequents of all 1-item consequent rules derived from f_k above};
4   ap − genRules{f_k, H_1};
5   endfor

Procedure ap − genRules{f_1, H_m}     //H_m is the set of m-item consequents
1   if(k > m + 1) AND (H_m ≠ ∅) then
2     H_{m+1} ← candidate − gen(H_m);
3     for each h_{m+1} in H_{m+1} do
4       conf ← f_k.count/(f_k − h_{m+1}).count;
5       if(conf ≥ minconf) then
6           output the rule (f_k − h_{m+1}) → h_{m+1} with confidence = conf and
            support = f_k.count/n;      //n is the total number of transactiones in T
        else
8           delete h_{m+1} from H_{m+1};
9     endfor
10    ap − genRules(f_k, H_{m+1});
11  endif
```

Hình 2.5. Thuật toán sinh luật kết hợp [11]

2.4. Nhận xét và ý tưởng

Như đã trình bày ở trên, học xếp hạng đang là giải pháp xếp hạng phổ biến trong những năm gần đây. Đến nay đã có rất nhiều công trình nghiên cứu về học xếp hạng và áp dụng với nhiều dữ liệu khác nhau từ kết quả tìm kiếm đến các văn bản… Với giả thiết cho bài toán Xếp hạng dòng: các dòng cập nhật trên trang người dùng tương đương

với tập các kết quả tìm kiếm; chúng tôi đưa ra ý tưởng sử dụng thuật toán học xếp hạng để giải quyết bài toán này. Theo phương pháp học xếp hạng, mô hình do chúng tôi đề xuất cần xây dựng một mô hình tính hạng. Mô hình này thể hiện sự kết hợp giữa các đặc trưng của tài liệu (tweet). Thuật toán CRR được sử dụng để sinh mô hình tính hạng.

Mô hình chủ đề ẩn LDA khá mạnh mẽ trong việc tìm ra phân phối xác suất của các tài liệu trên các chủ đề ẩn, liên quan mật thiết tới nội dung của các tài liệu đó. Hiểu về mô hình này, chúng tôi tận dụng nó vào việc tìm ra đặc trưng nội dung của tài liệu (tweet) để phục vụ cho việc xây dựng mô hình tính hạng.

Độ ảnh hưởng người dùng đã được chỉ ra là một yếu tố đặc biệt ảnh hưởng tới hành vi người dùng. Nắm được điều này, chúng tôi cũng tận dụng nó vào việc làm giàu đặc trưng người dùng phục vụ cho xây dựng mô hình tính hạng. Đặc trưng này được tính dựa trên khai phá luật kết hợp giữa các người dùng trên tập các tweet.

Như vậy, ý tưởng cốt lõi của mô hình xếp hạng là sử dụng phương pháp *học tính hạng* để *xây dựng mô hình tính hạng* cho các dòng cập nhật của mỗi người dùng trên mạng xã hội Twitter. Ở giai đoạn xác định các đặc trưng xây dựng mô hình tính hạng, *mô hình chủ đề ẩn* được sử dụng trong hệ thống để *bổ sung các đặc trưng* liên quan đến nội dung và *khai phá luật kết hợp* giữa các người dùng để bổ sung *đặc trưng độ ảnh hưởng người dùng* cho các tweet.

2.5. Tóm tắt chương 2

Trong chương 2, luận văn đã trình bày cơ sở nền tảng về học tính hạng, phương pháp xếp hạng CRR, mô hình chủ đề ẩn LDA và thuật toán Apriori khai phá luật kết hợp. Chúng tôi cũng trình bày được ý tưởng của mình qua việc nêu rõ vai trò của phương pháp học tính hạng, mô hình chủ đề ẩn LDA, và khai phá luật kết hợp trong mô hình đề xuất. Chúng tôi vận dụng phương pháp học tính hạng để tìm ra mô hình tính hạng tương ứng với mỗi người dùng. Vận dụng LDA tìm ra phân phối chủ đề cho mỗi tài liệu để bổ sung đặc trưng nội dung, và dùng Apriori để tìm luật kết hợp, phản ánh đặc trưng ảnh hưởng người dùng cho các dòng cập nhật.

Chương tiếp theo, chúng tôi sẽ trình bày tổng quan về mô hình đề xuất dựa trên các ý tưởng của mình và mô hình hóa hệ thống dưới dạng sơ đồ.

Chương 3.

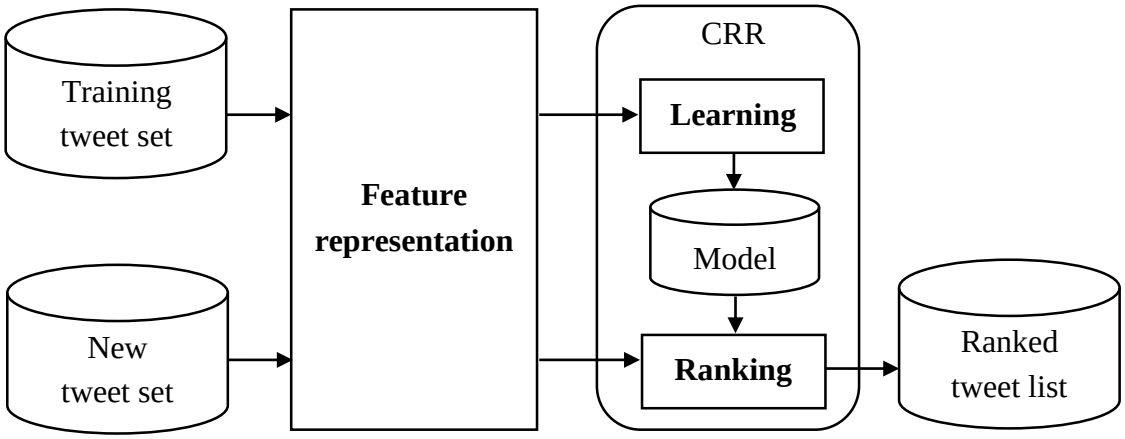
MÔ HÌNH XẾP HẠNG DÒNG CẬP NHẬT TRÊN TWITTER

Trong chương này, luận văn trình bày chi tiết về mô hình xếp hạng dòng xã hội cho mỗi người dùng và cách hoạt động của mô hình đó. Mô hình này phát triển từ mô hình của chúng tôi [1] với bổ sung ý tưởng khai thác độ ảnh hưởng người dùng [8, 9] tìm được qua phương pháp khai phá luật kết hợp [10].

3.1. Phương pháp đề xuất

Như đã được đề cập trong [1], mô hình xếp hạng dòng cập nhật bao gồm hai pha chính: học tính hạng (learning) và xếp hạng (ranking)

- Learning: Tìm ra ***mô hình tính hạng*** theo sự quan tâm của người dùng dựa vào nội dung tweet và độ ảnh hưởng của người gửi.
- Ranking: Sử dụng các kết quả của pha learning để tính hạng cho các tweet mới. Từ đó, thực hiện ***xếp hạng các tweet mới***



Hình 3.1. Mô hình xếp hạng dòng [1]

Theo C. Xiao và cộng sự, F. Riquelme và P. G. Cantergiani [8, 9], độ ảnh hưởng của người dùng được đánh giá là rất hữu ích trong hệ tư vấn, tuyên truyền thông tin…Vì

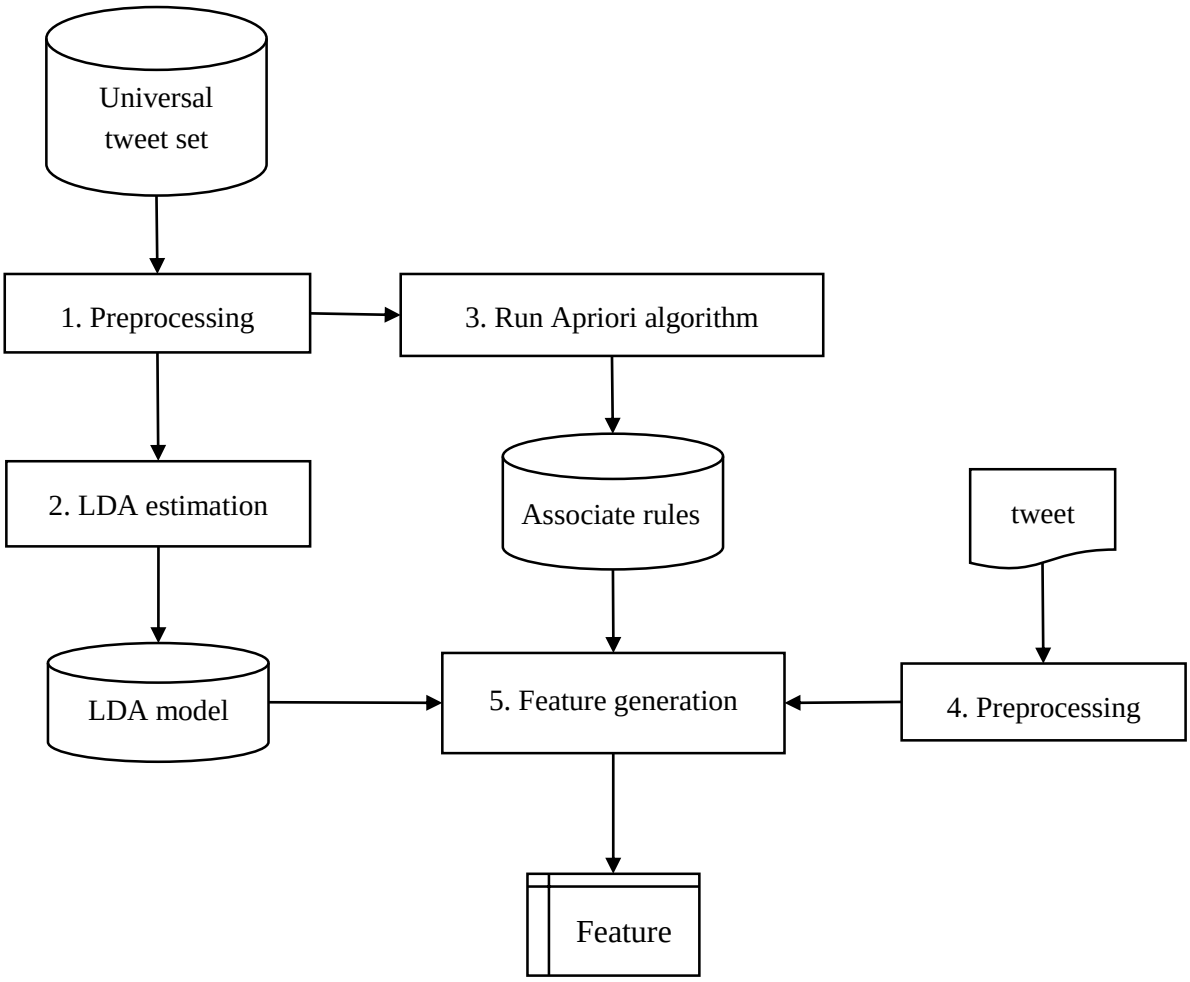
vậy, độ ảnh hưởng của người dùng rất có thể nâng cao hiệu quả cho mô hình xếp hạng dòng cập nhật [1]. Luận văn tập trung nâng cao mô hình này ở bước biểu diễn đặc trưng (feature representation). Ngoài việc sử dụng các đặc trưng cho tweet như cũ, chúng tôi sử dụng **độ ảnh hưởng của người dùng** vào làm giàu đặc trưng cho hệ thống phân hạng.

Để đo lường độ ảnh hưởng của người dùng, F. Erlandsson và cộng sự [10] đã coi danh sách các tweet trên Facebook trong một miền xác định như một cơ sở dữ liệu giao dịch, và tập item là tập các người dùng. Sau đó thực hiện tìm ra các luật kết hợp giữa các người dùng. Từ đó đưa ra danh sách những người dùng có độ ảnh hưởng lớn tới những người dùng khác. Dựa theo quan điểm này, chúng tôi thực hiện đo lường độ ảnh hưởng của người dùng dựa vào luật kết hợp giữa các người dùng trên mạng xã hội Twitter. Cơ sở dữ liệu giao dịch là tập các tweet $T = \{t_1, t_2 \ldots t_m\}$, và tập item là tập các người dùng Twitter $U = \{u_1, u_2 \ldots u_n\}$.

Giả sử có luật kết hợp $\{u_1, u_2\} \rightarrow \{u_3\}$ với độ support = 50% và confidence = 80%.

Luật kết hợp này chỉ ra rằng người dùng u_1 và u_2 có ảnh hưởng tới người dùng u_3. Nếu tweet nào được người dùng u_1 và u_2 thích thì xác suất người dùng u_3 cũng thích là 80%, và 50% các bài tweet là cả ba người dùng này đều thích.

- Thuật toán Apriori [11] được sử dụng để tìm các luật kết hợp cho tập người dùng liên quan. Hình 3.2 thể hiện bước biểu diễn đặc trưng sau khi đã cải tiến mô hình.



Hình 3.2 Bước biểu diễn đặc trưng (Feature representation)

PBước tiền xử lý dữ liệu (1. Preprocessing) cho tweet bao gồm các nhiệm vụ sau:

- Tách từ (word segmentation): xử lý loại bỏ các dấu cách nếu thừa, tách các từ ghép như "won't" thành "will not"…
- Loại bỏ tên người dùng vì nó không bổ sung nghĩa cho nội dung của tweet (bắt đầu bằng kí tự @)
- Loại bỏ từ dừng[3] – những từ không có ý nghĩa.
- Loại bỏ các kí tự đặc biệt, như là kí tự "#" – kí tự được sử dụng để đánh dấu hash tag (cách thức cho phép người dùng đánh dấu các từ khóa mà mình quan tâm để dễ dàng truy cập sau này)
- Thực hiện tạo đầu vào cho ước lượng LDA và thuật toán Apriori.

[3] Những từ phổ biến và không có nghĩa. Danh sách các stop word lấy tại đây: http://jmlr.org/papers/volume5/lewis04a/a11-smart-stop-list/english.stop

3.2. Đặc trưng và điểm số quan tâm của tweet

Dữ liệu huấn luyện và dữ liệu kiểm tra được xây dựng dựa trên các đặc trưng và điểm số quan tâm (nhãn) của tweet.

Mỗi tweet t được đăng lên trên trang người dùng u có các thông tin về:

- Thời gian tweet đến (thời gian u_t đăng tweet)
- Người đăng u_t
- Đặc điểm của tweet (retweet, reply, hash tag…)
- Nội dung của tweet.

Các ý tưởng sử dụng thông tin của tweet.

- Chúng tôi sử dụng yếu tố thời gian để phân chia các tweet cũ và mới. Các thông tin về người đăng (hay người gửi) tweet và nội dung của tweet được khai thác để tìm ra các đặc trưng cần thiết cho việc xây dựng mô hình tính hạng.
- Thông tin liên quan tới người đăng tweet được sử dụng để tính các đặc trưng tác giả gửi tweet, độ ảnh hưởng người dùng. Đặc điểm của tweet cũng được sử dụng tính các đặc trưng retweet, reply... Đặc trưng nội dung tweet thu được thông qua việc sử dụng mô hình chủ để ẩn. Luận văn sử dụng tập phân phối xác suất của các chủ đề trên mỗi tweet là tập đặc trưng nội dung của tweet đó.
- Để tính toán mức độ quan tâm (hay thứ hạng) của các tweet với người dùng u, chúng tôi khảo sát các hành động retweet, reply, favorite có được người nhận thực hiện trên mỗi tweet hay không.

3.2.1. Điểm số quan tâm của tweet

Như đã được đề cập trong [1], một số nội dung chính về điểm số của tweet được trình bày như sau.

Xét $T_{u_i} = \{t_{u_i j}\}, j = [1, \dots]$ là tập các dòng cập nhật - tweet của người dùng u_i. Trong đó gồm có tập các tweet mà u_i quan tâm (interesting tweet) ($T_{u_i 1}$) và tập các tweet mà u_i không quan tâm ($T_{u_i 2}$).

Để tìm nhãn cho mỗi tweet hay nói cách khác là tính điểm số quan tâm của tweet trên trang của người dùng u_i, chúng tôi thực hiện tìm danh sách các người dùng được u_i retweet hay reply. Gọi Urw_{u_i} là tập người bạn được u_i retweet và Ure_{u_i} là tập người bạn được u_i reply. Với mỗi tweet $t_{u_i j}$, (j là số thứ tự của tweet trong tập các tweet đang xét của người dùng u_i), thực hiện tính các điểm sau:

$$Score_{rw}(t_{u_ij}) = \begin{cases} 1, & t_{u_{ij}} \in Urw_{u_j} \\ 0, & trường\ hợp\ còn\ lại \end{cases} \quad (3.1)$$

$$Score_{re}\left(t_{u_{ij}}\right) = \begin{cases} 1, & t_{uij} \in Ure_{u_i} \\ 0, & trường\ hợp\ còn\ lại \end{cases} \quad (3.2)$$

$$Score_{fv}(t_{u_ij}) = \begin{cases} 1, & t_{u_{ij}} là\ favourite \\ 0, & trường\ hợp\ còn\ lại \end{cases} \quad (3.3)$$

Điểm của tweet t_{u_ij} là tổng điểm của 3 điểm trên. Nếu điểm của tweet lớn hơn 0 thì đó là interesting tweet.

$$l(t) = \begin{cases} 1,2\ or\ 3, & t \in T_{u_i1} \\ 0, & t \in T_{u_i2} \end{cases} \quad (3.4)$$

3.2.2. Đặc trưng của tweet

Theo phương pháp học xếp hạng, thứ hạng của đối tượng được học thông qua các giá trị của các đặc trưng của đối tượng đó. Thực hiện việc học xếp hạng với đối tượng là tweet, chúng tôi xác định các đặc trưng sử dụng trong việc học xếp hạng tweet. Sau khi xác định các đặc trưng, mỗi tweet sẽ được biểu diễn dưới dạng vector các đặc trưng. Có 6 đặc trưng đơn lẻ và 1 đặc trưng nội dung. Nếu đặc trưng nội dung thể hiện qua k chủ đề ẩn, thì tweet được biểu diễn dưới dạng vector có kích thước là k+6. Dưới đây là tóm lược cách tính các đặc trưng cũ – đã được đề cập trong [1] và trình bày cách tính đặc trưng độ ảnh hưởng người dùng.

1. Đặc trưng tác giả gửi tweet

Điểm của tác giả đăng tweet được tính theo số following và follower của tác giả đó:

$$author(u) = \frac{i(u)}{i(u) + o(u)} \quad (3.5)$$

Trong đó, i(u) là số người theo dõi của u (follower) và o(u) là số người u theo dõi (following).

2. Đặc trưng nội dung

Trên cơ sở [1], luận văn sử dụng tập phân phối xác suất của các chủ đề trên mỗi tài liệu là thành phần của tập đặc trưng nội dung.

Giả sử chúng ta xác định được K topic từ tập dữ liệu học. Với mỗi tweet *t*, luận văn tính các xác suất để tài liệu d thuộc vào topic *i* là *pt(i)*, với *i=1,…,k.*

Từ đó xác định được tập đặc trưng nội dung từ mô hình chủ đề ẩn LDA là:

$$T = [pt_1, pt_2 \dots pt_k] \quad (3.6)$$

3. Đặc trưng Retweet

Retweet là một trong các chỉ số đặc biệt của tweet. Khi tweet được sự tán thành của nhiều người dùng, tweet đó có thể được retweet và lan nhanh trên mạng xã hội. Hệ thống nên tư vấn cho người dùng đọc các tweet này. Do đó, đặc trưng retweet cũng được xét để xây dựng tập huấn luyện. Đặc trưng này được tính điểm như sau:

$$Rw(t_{uj}) = \begin{cases} 1, & t_{uj} \text{ được } retweet \\ 0, & trường\ hợp\ còn\ lại \end{cases} \quad (3.7)$$

4. Đặc trưng reply

Tương tự với đặc trưng retweet, đặc trưng reply cũng được tính dựa theo công thức như sau:

$$Re(t_{uj}) = \begin{cases} 1, & t_{uj} \text{ là } tweet\ reply \\ 0, & trường\ hợp\ còn\ lại \end{cases} \quad (3.8)$$

5. Đặc trưng hash tag

Hash tag là đặc trưng liên quan tới nội dung của tweet. Đặc trưng này có giá trị nhị phân, tweet đang xét có điểm được tính như sau:

$$htag(t) = \begin{cases} 1, & t\ chứa\ hashtag \\ 0, & trường\ hợp\ còn\ lại \end{cases} \quad (3.9)$$

6. Đặc trưng url

Url cũng là một đặc trưng liên quan tới nội dung của tweet. Đặc trưng này có giá trị nhị phân, tweet đang xét có điểm được tính như sau:

$$url(t) = \begin{cases} 1, & t\ chứa\ URL \\ 0, & trường\ hợp\ còn\ lại \end{cases} \quad (3.10)$$

7. Đặc trưng độ ảnh hưởng người dùng

Như đã giới thiệu ở chương 2, xét cơ sở dữ liệu giao dịch là tập các tweet T = {t_1, t_2…t_n}. Tập các item là tập các người dùng: U = {u_1, u_2… u_m}.

Minh họa cơ sở dữ liệu giao dịch như Bảng 3.1

Bảng 3.1 Minh họa cơ sở giao dịch tìm luật kết hợp giữa các người dùng

Giao dịch (Tweet)	**Item (User)**
t1	User 1, User 2, User 3

t2	User 1, User 4
t3	User 4, User 5
t4	User 1, User 2, User 4
t5	User 1,User 2,User 6, User 4, User 3
…	…
tm	User 2, User 3, User 6

Để thực hiện tính độ ảnh hưởng của người dùng cho tweet t của user u_k, ta thực hiện các bước sau:

Bước 1: Tìm tập luật kết hợp.

Với user đang xét u_k, thực hiện tìm các luật kết hợp có dạng

$\{u_i, \ldots u_j\} \rightarrow \{u_k\}$ thỏa mãn độ support >= minsup và conf >= minconf với minsup, minconf cho trước. Như vậy, ta sẽ có tập các luật kết hợp thỏa mãn: A = $\{a_1, a_2 \ldots\}$

Ví dụ ta xét User 3, với minconf = 0.7. Một số luật kết hợp như dưới đây:

Luật kết hợp a_1 = {User 2, User 6} → {User 3} có conf = 2/2 = 1 (thỏa mãn)

Luật kết hợp a_2 = {User 1, User 2} → {User 3} có conf = 2/3 = 0.67 (không thỏa mãn)

Luật kết hợp a_3 = {User 2} → {User 3} có conf = 3/4 = 0.75 (thỏa mãn).

Luật kết hợp a_4 = {User 6} → {User 3} có conf = 2/2 = 1 (thỏa mãn)

Như vậy ta có A = $\{a_1, a_3, a_4\}$

Bước 2: Tìm tập user tham gia vào tweet t

Với tweet t, ta tìm tập các user tham gia vào tweet này qua các hoạt động thích, retweet, reply: U(t) = $\{u_1, u_2 \ldots u_t\}$ với $u_t \neq u_k$

Bước 3: Xác định độ ảnh hưởng qua số lượng luật kết hợp phù hợp

Gọi n(t) là số lượng các luật kết hợp trong A thỏa mãn có sự tham gia của các user trong U(t). Độ ảnh hưởng người dùng tới user u_k được tính như sau:

$$influ(t) = n(t) \tag{3.11}$$

3.3. Tóm tắt chương 3

Trong chương 3, luận văn đã cụ thể hóa mô hình xếp hạng với các công việc cần làm trong mỗi giai đoạn. Ngoài ra, chương này cũng trình bày cách tính điểm cho tweet (nhãn tweet) và các đặc trưng để xây dựng tập dữ liệu huấn luyện.

Chương tiếp theo, chúng tôi hiện thực hóa các công việc phải làm trong thực nghiệm với người dùng trên Twitter. Do tính tương tự giữa các người dùng, chúng tôi lựa chọn thực nghiệm với một người dùng ngẫu nhiên trên mạng xã hội Twitter.

Chương 4.

THỰC NGHIỆM VÀ ĐÁNH GIÁ

Trong chương này, chúng tôi trình bày thực nghiệm dựa trên mô hình đề xuất ở chương ba. Chúng tôi tiến hành thực nghiệm với dữ liệu tweet là các dòng cập nhật của một người dùng trên Twitter. Việc lựa chọn người dùng là hoàn toàn ngẫu nhiên. Bắt đầu với việc xây dựng tập dữ liệu huấn luyện và tập dữ liệu kiểm tra dựa trên công cụ JGibbLDA (cài đặt của mô hình chủ đề ẩn LDA) và các chương trình tự xây dựng. Sau đó, thực hiện quá trình học xếp hạng với chương trình mã nguồn mở chạy thuật toán CRR.

Chúng tôi thực hiện hai thí nghiệm: (1) sử dụng mô hình LDA và sử dụng đặc trưng độ ảnh hưởng người dùng dựa trên luật kết hợp, (2) sử dụng mô hình LDA nhưng không sử dụng đặc trưng độ ảnh hưởng người dùng dựa trên luật kết hợp. Dựa vào kết quả thực nghiệm, chúng tôi tiến hành đánh giá, so sánh, nhận xét, rút ra kết luận và đưa ra định hướng tiếp theo.

4.1. Môi trường thực nghiệm

4.1.1. Cấu hình phần cứng

Bảng 4.1 Cấu hình máy tính thực nghiệm

Thành phần	Chỉ số
CPU	Intel Core i3-2330M 2.2Ghz
RAM	4GB
HDD	500GB
OS	Ubuntu 11.10 (32bit) Window 8 (hỗ trợ tính dữ liệu)

4.1.2. Công cụ phần mềm

Trong quá trình thực nghiệm, chúng tôi sử dụng một số công cụ và phần mềm mã nguồn mở được liệt kê trong Bảng 4.2:

Bảng 4.2 Danh sách phần mềm sử dụng trong thực nghiệm

STT	Tên phầm mềm	Tác giả	Nguồn
1	Eclipse-SDK-3.7.0		http://www.eclipse.org/dowloads
2	Mã nguồn mở thuật toán CRR: sofia-ml	D.Sculley	http://code.google.com/p/sofia-ml
3	JGibbLDA	Xuan-Hieu Phan và Cam-Tu Nguyen	http://jgibblda.sourceforge.net/
4	MS-Excel trong bộ MS-Office 2013	Microsoft	http://www.microsoft.com
5	Stopword	Nguyễn Thị Tươi	Tự xây dựng với ngôn ngữ java
6	Apriori	Nguyễn Thị Tươi	Tự xây dựng với ngôn ngữ java

Dưới đây là mô tả về các phần mềm chính trong thực nghiệm.

❖ **Phần mềm mã nguồn mở sofia-ml**

Đây là phần mềm viết bằng ngôn ngữ C++ trên môi trường Linux, chạy thuật toán CRR, do tác giả của thuật toán này xây dựng. Trong thực nghiệm, chúng tôi sử dụng hai chức năng của phần mềm này:

(1) Xây dựng hàm tính hạng – model từ tập tweet huấn luyện.

Thực hiện chức năng này, phần mềm có đầu vào đầu ra như sau:

- *Đầu vào: file dữ liệu train*
- *Đầu ra: file chứa mô hình tính hạng*

(2) Dùng model thu được, dự đoán hạng cho tập tweet test

- *Đầu vào: file dữ liệu test và mô hình tính hạng thu được ở (1)*
- *Đầu ra: file chứa các hạng dự đoán của mỗi tweet.*

Các file dữ liệu có định dạng như sau:

```
<class-label> <feature-id>:<feature-value> ... <feature-id>:<feature-value>\n
<class-label> qid:<optional-query-id> <feature-id>:<feature-value> ... <feature-id>:<feature-value>\n
<class-label> <feature-id>:<feature-value> ... <feature-id>:<feature-value># Optional comment or extra data, following the optional "#" symbol.\n
```

❖ **Phần mềm mã nguồn mở JGibbLDA**

Đây là phần mềm viết bằng ngôn ngữ Java, là cài đặt của mô hình chủ đề ẩn LDA sử dụng phương pháp lấy mẫu Gibb. Trong thực nghiệm, chúng tôi sử dụng phần mềm này để tìm phân phối xác suất các chủ đề nội dung trên mỗi tweet.

Các chức năng mà chúng tôi sử dụng bao gồm ước lượng mô hình LDA từ tập tweet huấn luyện (tham số -est) và suy ra phân phối xác suất cho tập tweet test từ mô hình LDA ước lượng được (tham số -inf).

(1) Ước lượng mô hình LDA:

```
$ lda -est [-alpha <double>] [-beta <double>] [-ntopics <int>] [-niters
<int>] [-savestep <int>] [-twords <int>] -dfile <string>
```

Trong đó, các tham số trong [] là không bắt buộc.

- `-est`: Ước lượng mô hình LDA từ tập dữ liệu hỗn tạp
- `-alpha <double>`: giá trị của `alpha` , tham số của LDA. Mặc định alpha là 50 / K (K là số lượng chủ đề)
- `-beta <double>`: giá trị của `beta` cũng là tham số của LDA. Mặc định là 0.1
- `-ntopics <int>`: Số lượng chủ đề. Mặc định là 100
- `-niters <int>`: Số lần lặp lấy mẫu Gibbs. Giá trị mặc định là 2000.
- `-savestep <int>`: Số bước mô hình LDA được lưu vào ổ cứng (đếm bởi số lần lặp lấy mẫu Gibbs). Giá trị mặc định là 200.
- `-twords <int>`: Số lượng các từ xuất hiện nhiều nhất trong mỗi chủ đề. Mặc định là 0.
- `-dfile <string>`: File dữ liệu huấn luyện.

(2) Suy ra phân phối xác suất cho dữ liệu mới:

```
$ lda -inf -dir <string> -model <string> [-niters <int>] [-twords <int>]
-dfile <string>
```

- `-inf`: Suy ra phân phối xác suất cho dữ liệu mới sử dụng mô hình LDA đã có trước.
- `-dir <string>`: Thư mục chứa mô hình
- `-model <string>`: Tên của mô hình đã có trước.
- `-niters <int>`: Số lần lặp lấy mẫu Gibbs. Giá trị mặc định là 20.
- `-twords <int>`: Số lượng từ xuất hiện nhiều nhất trong mỗi chủ đề. Mặc định là 0.
- `-dfile <string>`: File chứa dữ liệu mới

(3) Cả dữ liệu huấn luyện mô hình và dữ liệu mới đều có định dạng đầu vào như sau:

```
[M]
```

[$document_1$]

[$document_2$]

...

[$document_M$]

Trong đó, dòng đầu tiên là tổng số tài liệu. Mỗi dòng sau đó là một tài liệu. [$document_i$] là tài liệu thứ i của tập dữ liệu bao gồm N_itừ:

[$document_i$] = [$word_{i1}$] [$word_{i2}$] ... [$word_{iNi}$]

Trong đó, [$word_{ij}$] (i=1..M, j=1..N_i) là các chuỗi kí tự và chúng phân tách nhau bởi dấu cách.

(4) Đầu ra: là tập các file

<model_name>.others,
<model_name>.phi,
<model_name>.theta,
<model_name>.tassign,
<model_name>.word.

Trong đó:

<model_name>.others: file chứa các tham số sử dụng như anfa, beta,…

<model_name>.phi: File chứa phân phối từ - chủ đề. Mỗi dòng là một chủ đề, mỗi cột là một từ trong file wordmap.txt.

<model_name>.theta: File chứa phân phối chủ đề - tài liệu. Mỗi dòng là một tài liệu, và mỗi cột là một chủ đề.

<model_name>.tassign: File chứa chủ đề chính của từ trong dữ liệu huấn luyện. Mối dòng là một tài liệu chứa danh sách <wordij>:<chủ đề của wordij>

<model_file>.twords: File chứa tập các từ phổ biến nhất trong mỗi chủ đề.

❖ **Chương trình Stopword**

Chương trình này có nhiệm vụ sau:

- Loại bỏ tên người dùng (bắt đầu bằng @) trong mỗi tweet
- Loại bỏ các dấu # (sau là hash tag) trong mỗi tweet
- Loại bỏ các URL
- Loại bỏ các kí tự đặc biệt.

- Loại bỏ các stop word trong tweet

Chương trình có đầu vào là tập nội dung các tweet và đầu ra là file chứa nội dung tweet đã thực hiện bước loại bỏ trên.

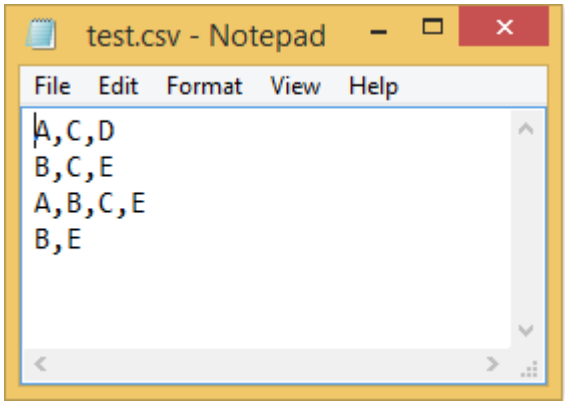
❖ **Chương trình Apriori**

Chương trình này có nhiệm vụ tìm các luật kết hợp giữa các người dùng. Input và output của chương trình như sau:

Input: Cơ sở dữ liệu giao dịch (.csv file), độ support nhỏ nhất (minsup) và độ tin cậy nhỏ nhất (minconf).

Output: txt file chứa thông tin các tập phổ biến thỏa mãn minsup và các luật kết hợp thỏa mãn minconf.

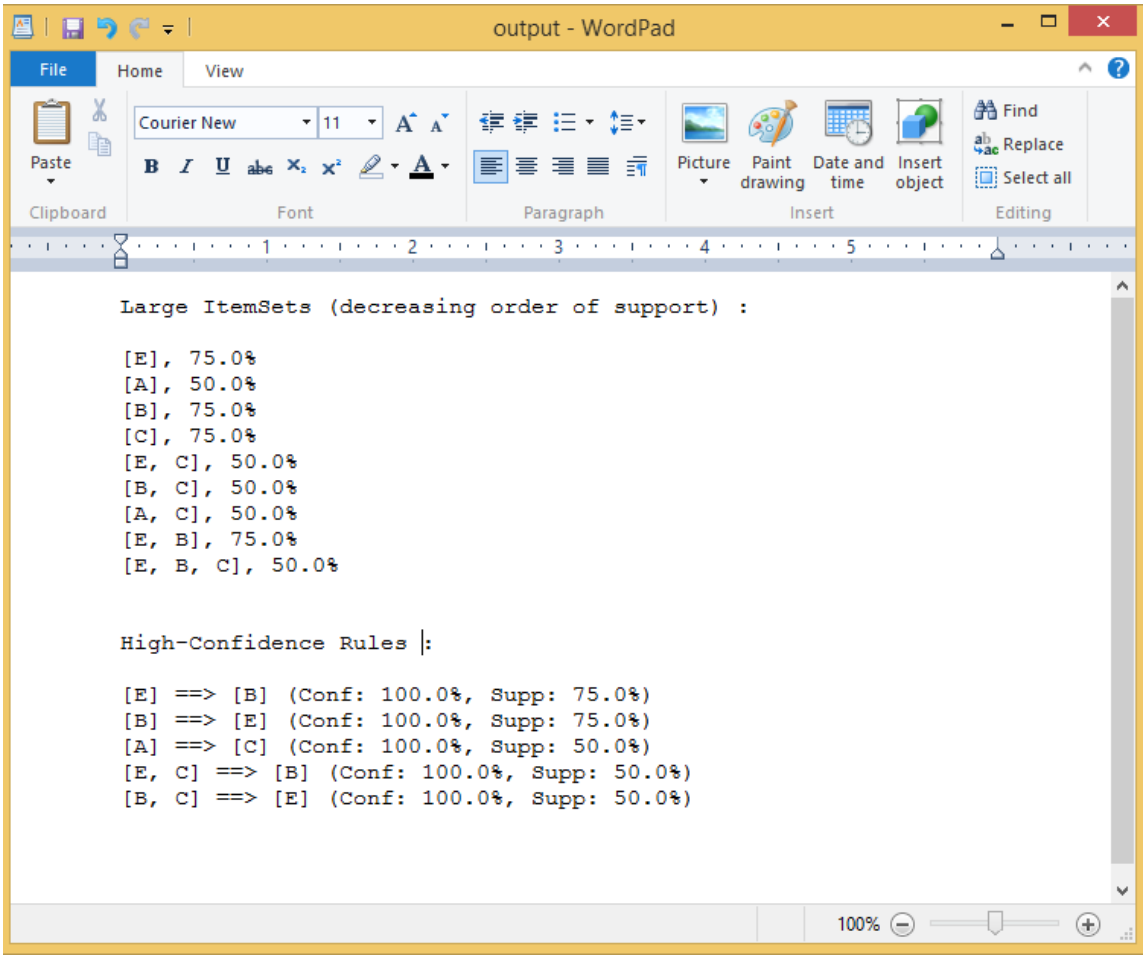
Định dạng input file (cơ sở dữ liệu giao dịch (.csv)) như sau:



Hình 4.1. Định dạng input file chương trình Apriori

Mỗi dòng là một giao dịch. Các item ngăn cách nhau bằng dấu “,”.

Định dạng file output như sau:



Hình 4.2. Định dạng file output chương trình Apriori

4.2. Dữ liệu thực nghiệm

Trong thực nghiệm, chúng tôi sử dụng các dòng tweet của người dùng có tên Jon Bowzer Bauman (@JonBowzerBauman). Hình 4.3 minh họa về người dùng này trên Twitter.

Dữ liệu thực nghiệm được stream trong thời gian tháng 10 năm 2016, bao gồm hơn 6400 dòng cập nhật đến trang của người dùng này.

Hình 4.3. Minh họa người dùng được sử dụng trong thực nghiệm

4.3. Thực nghiệm

Chúng tôi thực hiện hai thí nghiệm sau với mục đích làm rõ vai trò của việc sử dụng luật kết hợp bổ sung đặc trưng độ ảnh hưởng của người dùng cho tweet trong xếp hạng dòng:

- Thí nghiệm 1 (TN1): Thực hiện xây dựng mô hình tính hạng có sử dụng mô hình LDA và **sử dụng đặc trưng độ ảnh hưởng người dùng** dựa trên luật kết hợp
- Thí nghiệm 2 (TN2): Thực hiện xây dựng mô hình tính hạng có sử dụng mô hình LDA **nhưng không sử dụng đặc trưng độ ảnh hưởng người dùng** dựa trên luật kết hợp

Với thí nghiệm 1, chúng tôi tiến hành các công việc sau:

(1) Thu thập và tiền xử lý dữ liệu.
(2) Xây dựng mô hình chủ để ẩn và đặc trưng nội dung
(3) Tìm tập luật kết hợp và xây dựng đặc trưng độ ảnh hưởng người dùng
(4) Tính các giá trị cho các đặc trưng còn lại của tweet.
(5) Xây dựng dữ liệu huấn luyện và dữ liệu kiểm tra.
(6) Học tính hạng từ dữ liệu huấn luyện
(7) Sử dụng mô hình tính hạng cho dữ liệu kiểm tra và đánh giá.

Với thí nghiệm 2, chúng tôi không thực hiện công việc (3)

Các công việc trên được thực hiện như sau.

Công việc 1: Lấy dữ liệu, tiền xử lý dữ liệu

Sử dụng streaming API của Twitter, chúng tôi có được dữ liệu tweet liên quan đến người dùng Jon Bowzer Bauman là hơn 6400 tweet.

Do sử dụng streaming API, nên tất cả các cập nhật đều được lấy về, bao gồm:

(1) Tweet do chính người dùng đang xét đăng lên
(2) Tweet do bạn (trong danh sách following) đăng lên
(3) Retweet tweet mà người retweet là người dùng đang xét
(4) Retweet tweet mà người viết tweet và người retweet đều là bạn
(5) Retweet tweet mà người viết tweet không phải bạn, nhưng người retweet là bạn.
(6) Reply tweet mà người reply là người dùng đang xét
(7) Reply tweet mà người reply là bạn (following)

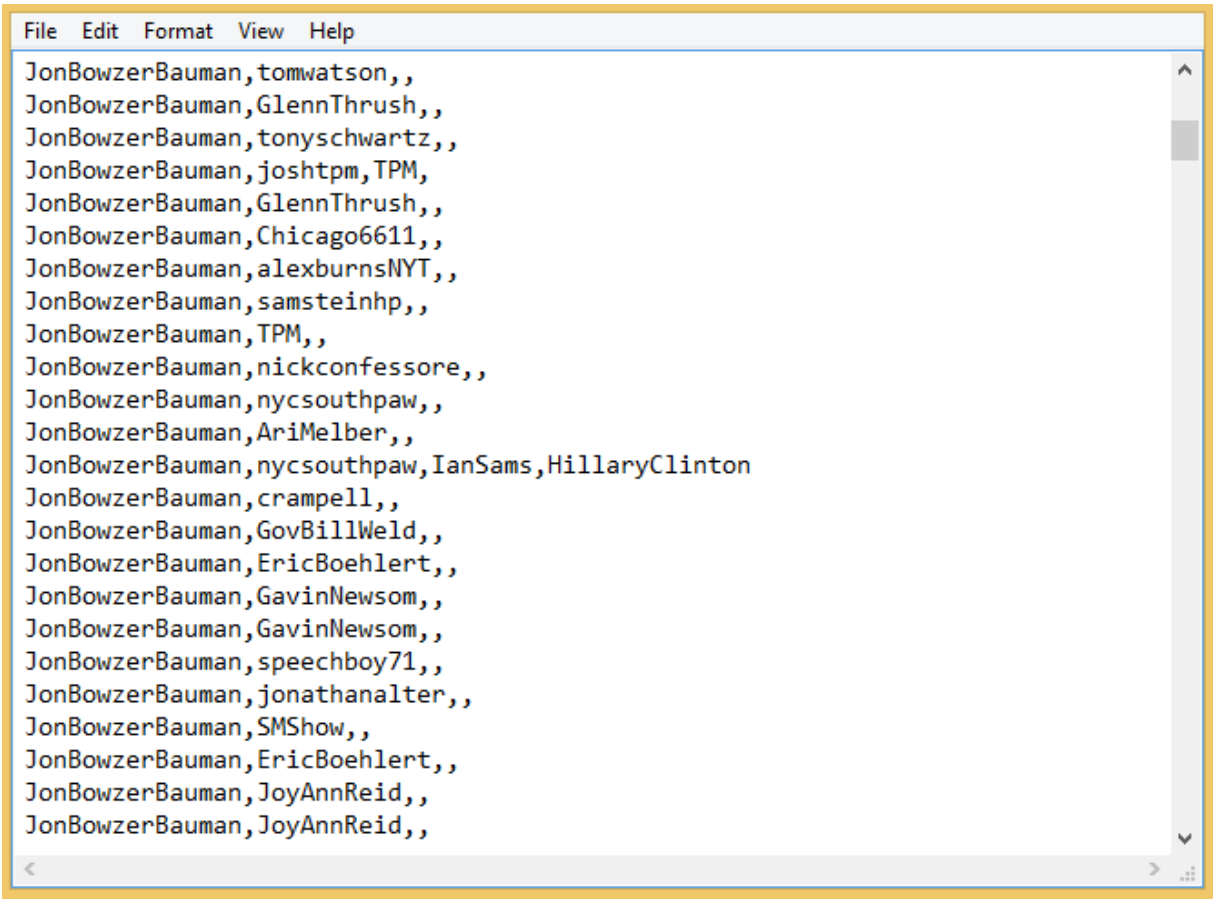
Tuy nhiên, qua khảo sát trên mạng xã hội Twitter, chúng tôi nhận thấy, trong các loại trên, (1), (3), (6) không được hiển thị trên timelines của người dùng.

Do vậy, chúng tôi lọc các tweet này ra, thu được 5854 tweet. Chia tập tweet làm tập huấn luyện (5254 tweet) và tập kiểm tra (600 tweet).

Tiếp theo, chúng tôi sử dụng chương trình Stopword để xây dựng tập dữ liệu chủ đề ẩn huấn luyện và tập dữ liệu chủ đề ẩn kiểm tra với định dạng đầu vào của phần mềm JGibbLDA.

Sau cùng, chúng tôi thực hiện tạo cơ sở dữ liệu giao dịch người dùng dựa trên các tweet kiểu (3), (6) (những tweet liên quan trực tiếp từ người dùng) làm đầu vào cho thuật toán Apriori.

Minh họa cơ sở dữ liệu giao dịch người dùng như Hình 4.4



Hình 4.4. Minh họa cơ sở dữ liệu người dùng

Công việc 2: Xây dựng mô hình chủ đề ẩn và đặc trưng nội dung

Đặc trưng nội dung được xây dựng dựa trên mô hình chủ đề ẩn LDA như sau

- Sử dụng tập dữ liệu chủ đề ẩn huấn luyện và tập dữ liệu chủ đề ẩn kiểm tra đã xây dựng trong công việc 1.
- Chạy phần mềm **JGibbLDA** với câu lệnh:

```
$lda -est -ntopics 30 -twords 20 -dfile models/tweet/tweethoc.dat
```

Chúng tôi chủ định tìm sự phân phối của mỗi tweet trên 30 chủ đề. Đầu ra bao gồm các file model-final.others, model-final.phi, model-final.theta, model-final.tassign,

model-final.twords (là mô hình model-final) . *Chúng tôi sử dụng file model-final.theta để làm đặc trưng nội dung cho dữ liệu huấn luyện.* Đặc trưng nội dung của dữ liệu huấn luyện được minh họa như hình dưới đây:

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5	Topic 6	Topic 7	Topic 8	Topic 9	Topic 10	Top
0.0679	0.0309	0.0309	0.0309	0.0309	0.0309	0.0309	0.0309	0.0494	0.0309	
0.0265	0.0265	0.0265	0.0265	0.0265	0.0423	0.0265	0.0582	0.0265	0.0265	
0.0242	0.0386	0.0242	0.0242	0.0242	0.0242	0.0242	0.0242	0.0676	0.0531	
0.0303	0.0303	0.0485	0.0303	0.0303	0.0303	0.0303	0.0303	0.0303	0.0303	
0.0430	0.0269	0.0269	0.0269	0.0430	0.0269	0.0269	0.0269	0.0430	0.0269	
0.0269	0.0269	0.0269	0.0269	0.0591	0.0269	0.0269	0.0269	0.0269	0.0430	
0.0253	0.0404	0.0707	0.0253	0.0707	0.0253	0.0556	0.0404	0.0253	0.0404	
0.0278	0.0278	0.0444	0.0611	0.0278	0.0278	0.0278	0.0278	0.0278	0.0278	
0.0655	0.0298	0.0298	0.0298	0.0298	0.0298	0.0298	0.0298	0.0476	0.0298	
0.0292	0.0292	0.0292	0.0292	0.0468	0.0292	0.0292	0.0468	0.0292	0.0292	
0.0314	0.0314	0.0314	0.0314	0.0314	0.0314	0.0314	0.0314	0.0314	0.0503	
0.0303	0.0303	0.0485	0.0303	0.0303	0.0303	0.0303	0.0485	0.0303	0.0485	
0.0444	0.0278	0.0278	0.0278	0.0278	0.0611	0.0278	0.0278	0.0278	0.0278	
0.0314	0.0503	0.0314	0.0314	0.0314	0.0314	0.0314	0.0314	0.0314	0.0314	
0.0282	0.0282	0.0282	0.0452	0.0452	0.0452	0.0282	0.0282	0.0282	0.0282	
0.0309	0.0309	0.0309	0.0309	0.0494	0.0309	0.0309	0.0309	0.0309	0.0679	
0.0298	0.0298	0.0298	0.0298	0.0298	0.0298	0.0298	0.0476	0.0476	0.0298	
0.0278	0.0278	0.0278	0.0278	0.0278	0.0278	0.0278	0.0444	0.0278	0.0278	
0.0437	0.0273	0.0273	0.0601	0.0273	0.0437	0.0273	0.0273	0.0273	0.0273	
0.0298	0.0476	0.0298	0.0298	0.0298	0.0298	0.0298	0.0298	0.0476	0.0298	

Hình 4.5. Minh họa đặc trưng nội dung (dữ liệu huấn luyện)

- Sử dụng mô hình thu được, chạy phần mềm **JGibbLDA** với câu lênh:

```
$lda -inf -dir models/tweet -model model-final -twords 20 -dfile tweettest.dat
```

Đầu ra là các file tweettest.dat.model-final.others, tweettest.dat.model-final.phi, tweettest.dat.model-final.theta, tweettest.dat.model-final.tassign, tweettest.dat.model-final.twords. *Chúng tôi sử dụng file tweettest.dat.model-final.theta để làm đặc trưng nội dung cho dữ liệu kiểm tra.* Đặc trưng nội dung của dữ liệu kiểm tra được minh họa như hình dưới đây:

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5	Topic 6	Topic 7	Topic 8	Topic 9	Topic 10	Top
0.0298	0.0298	0.0298	0.0298	0.0298	0.0298	0.0298	0.0298	0.0476	0.0476	
0.0292	0.0292	0.0292	0.0292	0.0292	0.0292	0.0292	0.0292	0.0292	0.0468	
0.0292	0.0292	0.0292	0.0292	0.0292	0.0292	0.0292	0.0292	0.0468	0.0292	
0.0417	0.0260	0.0417	0.0417	0.0573	0.0260	0.0260	0.0260	0.0260	0.0260	
0.0298	0.0476	0.0298	0.0298	0.0298	0.0298	0.0298	0.0298	0.0298	0.0298	
0.0321	0.0321	0.0321	0.0321	0.0321	0.0321	0.0321	0.0321	0.0321	0.0321	
0.0256	0.0410	0.0256	0.0256	0.0256	0.0256	0.0256	0.0564	0.0256	0.0256	
0.0292	0.0292	0.0292	0.0292	0.0292	0.0292	0.0468	0.0292	0.0292	0.0468	
0.0298	0.0476	0.0298	0.0298	0.0298	0.0298	0.0298	0.0476	0.0298	0.0298	
0.0278	0.0278	0.0278	0.0278	0.0278	0.0611	0.0444	0.0278	0.0278	0.0444	
0.0265	0.0423	0.0582	0.0265	0.0265	0.0265	0.0265	0.0265	0.0265	0.0265	
0.0398	0.0249	0.0249	0.0249	0.0249	0.0547	0.0249	0.0249	0.0398	0.0547	
0.0444	0.0444	0.0444	0.0278	0.0278	0.0278	0.0278	0.0278	0.0278	0.0278	
0.0273	0.0273	0.0437	0.0273	0.0601	0.0273	0.0273	0.0273	0.0273	0.0273	
0.0260	0.0260	0.0260	0.0260	0.0260	0.0260	0.0260	0.0260	0.0260	0.0417	
0.0417	0.0260	0.0260	0.0260	0.0260	0.0573	0.0729	0.0260	0.0260	0.0260	
0.0410	0.0256	0.0410	0.0256	0.0410	0.0410	0.0256	0.0256	0.0256	0.0256	
0.0282	0.0282	0.0452	0.0621	0.0282	0.0282	0.0452	0.0282	0.0282	0.0282	
0.0327	0.0327	0.0327	0.0327	0.0327	0.0327	0.0523	0.0327	0.0327	0.0327	

Hình 4.6. Minh họa đặc trưng nội dung (dữ liệu kiểm tra)

Công việc 3: Tìm tập luật kết hợp và xây dựng đặc trưng độ ảnh hưởng người dùng

Vì tweet có số lượng lớn, nhưng có nhiều người dùng tham gia, nên độ hỗ trợ sẽ thấp. Ngoài ra, việc sử dụng các tweet do chính người dùng reply hay retweet, nên các luật tìm được sẽ có độ tin cậy là 100%. Trong thực nghiệm, sử dụng cơ sở dữ liệu giao dịch người dùng cùng độ hỗ trợ 2.5% và độ tin cậy 100%, chạy chương trình Apriori để tìm ra các luật kết hợp giữa các người dùng với người dùng đang xét. Minh họa tập luật kết hợp như Hình 4.7.

```
Large ItemSets (decreasing order of support) :

[KatyTurNBC], 3.0023094688221708%
[joshtpm], 3.233256351039261%
[aseitzwald], 3.233256351039261%
[realDonaldTrump], 9.006928406466514%
[JonBowzerBauman], 100.0%
[JoyAnnReid], 3.9260969976905313%
[kurteichenwald], 2.7713625866050 81%
[JonBowzerBauman, aseitzwald], 3.233256351039261%
[JonBowzerBauman, JoyAnnReid], 3.9260969976905313%
[JonBowzerBauman, KatyTurNBC], 3.0023094688221708%
[JonBowzerBauman, joshtpm], 3.233256351039261%
[JonBowzerBauman, realDonaldTrump], 9.006928406466514%
[JonBowzerBauman, kurteichenwald], 2.7713625866050 81%


High-Confidence Rules (decreasing order of confidence) with
support(LHS union RHS) :

[kurteichenwald] ==> [JonBowzerBauman] (Conf: 100.0%, Supp:
2.7713625866050 81%)
[realDonaldTrump] ==> [JonBowzerBauman] (Conf: 100.0%, Supp:
9.006928406466514%)
[joshtpm] ==> [JonBowzerBauman] (Conf: 100.0%, Supp:
3.233256351039261%)
[KatyTurNBC] ==> [JonBowzerBauman] (Conf: 100.0%, Supp:
3.0023094688221708%)
[JoyAnnReid] ==> [JonBowzerBauman] (Conf: 100.0%, Supp:
3.9260969976905313%)
[aseitzwald] ==> [JonBowzerBauman] (Conf: 100.0%, Supp:
3.233256351039261%)
```

Hình 4.7. Minh họa luật kết hợp

Với mỗi tweet trong tập huấn luyện và tập kiểm tra, đếm số luật kết hợp mà có sự tham gia của tác giả tweet, và sử dụng nó làm đặc trưng độ ảnh hưởng người dùng.

Công việc 4: Tính các giá trị cho các đặc trưng còn lại và điểm số của tweet.

Dựa theo công thức đã nêu trong chương 3, sử dụng MS-Excel, chúng tôi thực hiện tính giá trị cho các đặc trưng tác giả, retweet, reply, URL và hash tag cho tập dữ liệu huấn luyện và tập dữ liệu kiểm tra. Chúng tôi cũng thực hiện tính điểm số của mỗi tweet, và sử dụng điểm số làm nhãn cho tweet.

Công việc 5: Xây dựng dữ liệu huấn luyện và dữ liệu kiểm tra

Với giá trị các đặc trưng đã tính trước, chúng tôi sử dụng MS-Excel để xây dựng file huấn luyện và file kiểm tra theo định dạng đầu vào của phần mềm sofia-ml.

Minh họa dữ liệu cho file huấn luyện và file kiểm tra như các hình dưới đây:

0	1:	0.99	2:	1	3:	0	4:	0	5:	1	6:	0	7:	0.0292	8:	0.0292	9:	0.0292	10:	0.0292	11:
0	1:	0.73	2:	1	3:	0	4:	0	5:	1	6:	0	7:	0.0260	8:	0.0260	9:	0.0260	10:	0.0260	11:
0	1:	0.99	2:	0	3:	0	4:	0	5:	1	6:	0	7:	0.0273	8:	0.0437	9:	0.0273	10:	0.0273	11:
1	1:	1	2:	0	3:	0	4:	0	5:	1	6:	0	7:	0.0321	8:	0.0321	9:	0.0321	10:	0.0321	11:
0	1:	0.98	2:	0	3:	0	4:	0	5:	1	6:	0	7:	0.0452	8:	0.0282	9:	0.0282	10:	0.0282	11:
0	1:	1	2:	0	3:	0	4:	0	5:	1	6:	0	7:	0.0278	8:	0.0278	9:	0.0278	10:	0.0278	11:
0	1:	0.98	2:	1	3:	0	4:	0	5:	1	6:	0	7:	0.0485	8:	0.0303	9:	0.0303	10:	0.0303	11:
0	1:	0.98	2:	0	3:	0	4:	0	5:	1	6:	0	7:	0.0260	8:	0.0260	9:	0.0260	10:	0.0260	11:
0	1:	0.99	2:	0	3:	0	4:	0	5:	0	6:	0	7:	0.0238	8:	0.0238	9:	0.0524	10:	0.0238	11:
0	1:	0.93	2:	0	3:	0	4:	0	5:	1	6:	0	7:	0.0278	8:	0.0278	9:	0.0778	10:	0.0278	11:
0	1:	0.98	2:	1	3:	0	4:	0	5:	0	6:	0	7:	0.0282	8:	0.0452	9:	0.0282	10:	0.0282	11:
0	1:	1	2:	0	3:	0	4:	0	5:	1	6:	0	7:	0.0468	8:	0.0292	9:	0.0643	10:	0.0292	11:
0	1:	0.99	2:	0	3:	1	4:	0	5:	0	6:	0	7:	0.0314	8:	0.0314	9:	0.0314	10:	0.0314	11:
0	1:	0.96	2:	0	3:	0	4:	0	5:	1	6:	0	7:	0.0249	8:	0.0398	9:	0.0547	10:	0.0398	11:
0	1:	1	2:	1	3:	0	4:	0	5:	0	6:	0	7:	0.0253	8:	0.0253	9:	0.0404	10:	0.0253	11:
0	1:	1	2:	0	3:	0	4:	0	5:	0	6:	0	7:	0.0245	8:	0.0245	9:	0.0245	10:	0.0392	11:

Hình 4.8. Minh họa dữ liệu file huấn luyện (TN1)

0	1:	0.97	2:	1	3:	0	4:	0	5:	1	6:	0	7:	0.0327	8:	0.0327	9:	0.0327	10:	0.0327	11:
0	1:	0.97	2:	1	3:	0	4:	0	5:	0	6:	0	7:	0.0256	8:	0.0410	9:	0.0256	10:	0.0564	11:
0	1:	0.88	2:	0	3:	0	4:	0	5:	0	6:	0	7:	0.0260	8:	0.0573	9:	0.0417	10:	0.0260	11:
0	1:	0.37	2:	0	3:	0	4:	1	5:	0	6:	0	7:	0.0309	8:	0.0494	9:	0.0309	10:	0.0494	11:
0	1:	0.81	2:	1	3:	0	4:	0	5:	0	6:	0	7:	0.0282	8:	0.0621	9:	0.0282	10:	0.0282	11:
0	1:	0.97	2:	1	3:	0	4:	0	5:	1	6:	0	7:	0.0321	8:	0.0321	9:	0.0513	10:	0.0321	11:
0	1:	0.96	2:	0	3:	0	4:	0	5:	0	6:	0	7:	0.0260	8:	0.0260	9:	0.0260	10:	0.0260	11:
0	1:	0.88	2:	0	3:	0	4:	0	5:	1	6:	0	7:	0.0298	8:	0.0298	9:	0.0298	10:	0.0298	11:
0	1:	0.99	2:	1	3:	0	4:	0	5:	1	6:	2	7:	0.0321	8:	0.0321	9:	0.0321	10:	0.0321	11:
1	1:	0.81	2:	1	3:	0	4:	0	5:	1	6:	0	7:	0.0423	8:	0.0265	9:	0.0265	10:	0.0265	11:
0	1:	1	2:	0	3:	0	4:	0	5:	1	6:	0	7:	0.0287	8:	0.0287	9:	0.0287	10:	0.0287	11:
0	1:	0.99	2:	1	3:	0	4:	0	5:	1	6:	2	7:	0.0249	8:	0.0249	9:	0.0398	10:	0.0398	11:
0	1:	0.99	2:	1	3:	0	4:	1	5:	1	6:	2	7:	0.0333	8:	0.0333	9:	0.0333	10:	0.0333	11:

Hình 4.9. Minh họa dữ liệu file kiểm tra (TN1)

0	1:	0.99	2:	0	3:	0	4:	0	5:	1	6:	0.0327	7:	0.0327	8:	0.0327	9:	0.0327	10:	0.0327	11:	0.0327	12:
0	1:	0.99	2:	1	3:	0	4:	0	5:	1	6:	0.0468	7:	0.0292	8:	0.0292	9:	0.0292	10:	0.0292	11:	0.0292	12:
0	1:	0.81	2:	0	3:	0	4:	0	5:	0	6:	0.0245	7:	0.0539	8:	0.0539	9:	0.0392	10:	0.0245	11:	0.0392	12:
0	1:	0.99	2:	0	3:	0	4:	0	5:	1	6:	0.0249	7:	0.0398	8:	0.0249	9:	0.0398	10:	0.0249	11:	0.0398	12:
0	1:	0.95	2:	1	3:	0	4:	0	5:	1	6:	0.0309	7:	0.0309	8:	0.0309	9:	0.0309	10:	0.0309	11:	0.0494	12:
0	1:	0.96	2:	0	3:	0	4:	0	5:	1	6:	0.0260	7:	0.0573	8:	0.0260	9:	0.0260	10:	0.0260	11:	0.0417	12:
0	1:	0.87	2:	1	3:	0	4:	1	5:	0	6:	0.0273	7:	0.0273	8:	0.0273	9:	0.0601	10:	0.0273	11:	0.0437	12:
0	1:	0.97	2:	1	3:	0	4:	0	5:	1	6:	0.0386	7:	0.0242	8:	0.0386	9:	0.0242	10:	0.0676	11:	0.0386	12:
0	1:	0.99	2:	1	3:	0	4:	1	5:	1	6:	0.0298	7:	0.0298	8:	0.0298	9:	0.0298	10:	0.0298	11:	0.0476	12:
0	1:	0.81	2:	1	3:	0	4:	1	5:	0	6:	0.0278	7:	0.0278	8:	0.0611	9:	0.0278	10:	0.0444	11:	0.0278	12:
0	1:	0.88	2:	0	3:	0	4:	1	5:	0	6:	0.0278	7:	0.0444	8:	0.0444	9:	0.0278	10:	0.0611	11:	0.0278	12:
0	1:	0.18	2:	1	3:	0	4:	0	5:	1	6:	0.0303	7:	0.0485	8:	0.0303	9:	0.0303	10:	0.0303	11:	0.0303	12:
0	1:	0.84	2:	0	3:	0	4:	0	5:	1	6:	0.0314	7:	0.0314	8:	0.0314	9:	0.0314	10:	0.0314	11:	0.0314	12:

Hình 4.10. Minh họa dữ liệu file huấn luyện (TN2)

0	1:	0.99	2:	0	3:	0	4:	0	5:	1	6:	0.0667	7:	0.0303	8:	0.0303	9:	0.0485	10:	0.0303	11:	0.0485	12:
0	1:	0.99	2:	1	3:	0	4:	0	5:	0	6:	0.0303	7:	0.0303	8:	0.0303	9:	0.0303	10:	0.0303	11:	0.0485	12:
0	1:	0.99	2:	0	3:	0	4:	0	5:	1	6:	0.0292	7:	0.0292	8:	0.0292	9:	0.0292	10:	0.0292	11:	0.0292	12:
0	1:	0.96	2:	1	3:	0	4:	1	5:	1	6:	0.0327	7:	0.0327	8:	0.0327	9:	0.0327	10:	0.0327	11:	0.0523	12:
0	1:	0.99	2:	1	3:	0	4:	0	5:	1	6:	0.0314	7:	0.0314	8:	0.0314	9:	0.0314	10:	0.0314	11:	0.0314	12:
0	1:	0.63	2:	1	3:	0	4:	0	5:	1	6:	0.0292	7:	0.0292	8:	0.0292	9:	0.0292	10:	0.0292	11:	0.0292	12:
0	1:	0.99	2:	0	3:	0	4:	0	5:	1	6:	0.0287	7:	0.0632	8:	0.0460	9:	0.0287	10:	0.0287	11:	0.0460	12:
0	1:	0.81	2:	0	3:	0	4:	1	5:	0	6:	0.0298	7:	0.0298	8:	0.0476	9:	0.0298	10:	0.0298	11:	0.0298	12:
0	1:	0.99	2:	1	3:	0	4:	0	5:	1	6:	0.0292	7:	0.0292	8:	0.0292	9:	0.0292	10:	0.0468	11:	0.0292	12:
0	1:	0.63	2:	1	3:	0	4:	0	5:	1	6:	0.0423	7:	0.0582	8:	0.0265	9:	0.0265	10:	0.0423	11:	0.0265	12:
0	1:	0.96	2:	1	3:	0	4:	1	5:	0	6:	0.0269	7:	0.0591	8:	0.0269	9:	0.0269	10:	0.0269	11:	0.0430	12:
0	1:	0.81	2:	0	3:	0	4:	1	5:	0	6:	0.0287	7:	0.0287	8:	0.0460	9:	0.0287	10:	0.0287	11:	0.0287	12:
0	1:	0.99	2:	0	3:	0	4:	0	5:	1	6:	0.0287	7:	0.0287	8:	0.0287	9:	0.0287	10:	0.0287	11:	0.0460	12:
0	1:	0.96	2:	0	3:	0	4:	0	5:	0	6:	0.0410	7:	0.0256	8:	0.0256	9:	0.0256	10:	0.0564	11:	0.0410	12:

Hình 4.11. Minh họa dữ liệu file kiểm tra (TN2)

Công việc 6: Sinh mô hình tính hạng

Ở cả hai thí nghiệm, chúng tôi chạy phần mềm sofia-ml với chức *năng sinh mô hình*, đầu vào là file huấn luyện, thu được hàm tính hạng. Sau đó, chúng tôi chạy sofia-ml với chức năng *dự đoán hạng* (để kiểm tra mô hình), đầu vào là file kiểm tra và hàm tính hạng, chúng tôi thu được file kết quả chứa nhãn dự đoán của các tweet trong dữ liệu kiểm tra.

Công việc 7: Đánh giá mô hình tính hạng

Để đánh giá hàm tính hạng, chúng tôi sử dụng MS-Excel để tính các độ chính xác P@K, Map đã trình bày trong 2.1.4. với hai file kết quả thu được từ hai thí nghiệm.

4.4. Kết quả và Đánh giá

Sau khi thực nghiệm với hai thí nghiệm (1) và (2), chúng tôi thu được 2 hàm tính hạng. Sử dụng MS-Excel, chúng tôi đánh giá các mô hình của thí nghiệm trên, thể hiện trong các hình sau:

TN1				
Dự đoán	Dự đoán	Đánh giá	P@k	P@k * I(k)
1.60704	2	0	0.421849	0
0.926122	1	0	0.421141	0
1.59794	2	0	0.420436	0
0.082801	0	1	0.421405	0.4214047
0.922971	1	0	0.420701	0
1.45476	1	0	0.42	0

P@50	0.64
P@100	0.55
P@600	0.42
Map	0.7634

TN2				
Dự đoán	Dự đoán	Đánh giá	P@k	P@k * I(k)
1.87917	2	0	0.415126	0
1.13125	1	0	0.41443	0
1.87133	2	0	0.413735	0
0.017484	0	1	0.414716	0.4147157
1.1273	1	0	0.414023	0
1.74291	2	0	0.413333	0

P@50	0.62
P@100	0.51
P@600	0.41
Map	0.701

Hình 4.12. Đánh giá hai mô hình

Bảng 4.3 thể hiện sự so sánh giữa hai mô hình thu được:

Bảng 4.3 Bảng so sánh hai mô hình thu được

Mô hình	MAP
Mô hình 1	76,34%
Mô hình 2	70,1%

Mô hình 1 thu được ở thí nghiệm 1 và mô hình 2 thu được ở thí nghiệm 2.

Các mô hình với độ chính xác mức K và độ chính xác trung bình Map được thể hiện trong bảng trên cho thấy mô hình 1 có độ chính xác cao hơn. Vì vậy, việc bổ sung them phần khai phá khoản mục thường xuyên trong luật kết hợp làm tăng chất lượng của các đặc trưng người dùng cho tweet, góp phần tăng độ chính xác của xếp hạng dòng trên mạng xã hội Twitter.

Kết luận và định hướng nghiên cứu tiếp theo

Qua tìm hiểu về luật kết hợp và dựa trên các kiến thức về học xếp hạng, mô hình chủ đề ẩn, luận văn đã thực hiện bổ sung phần khai phá khoản mục thường xuyên trong luật kết hợp nhằm tăng chất lượng của các đặc trưng cho mô hình xếp hạng dòng cập nhật trên mạng xã hội.

Luận văn đạt được các kết quả sau đây:

- Đề nghị mô hình xếp hạng dòng cập nhật cải tiến từ mô hình của chúng tôi [1] với bổ sung độ ảnh hướng người dùng được tính theo thuật toán Apriori.
- Xây dựng phần mềm thực nghiệm và kết quả thực nghiệm đối với hai phương án đạt MAP trên 0.70.

Tuy nhiên, do hạn chế về thời gian nên luận văn vẫn tồn tại những hạn chế như: dữ liệu và các đặc trưng sử dụng cho xếp hạng chưa được phong phú.

Trong thời gian tới, chúng tôi sẽ thực hiện với dữ liệu tốt hơn, các đặc trưng phong phú hơn, để nâng cao kết quả thực nghiệm.

Tài liệu tham khảo

[1] Thi-Tuoi Nguyen, Tri-Thanh Nguyen và Quang-Thuy Ha, *Applying Hidden Topics in Ranking Social Update Streams on Twitter,* RIVF 2013: 180-185.

[2] Rinkesh Nagmoti, Ankur Teredesai và Martine De Cock, *Ranking Approaches for Microblog Search,* Web Intelligence 2010: 153-157.

[3] Yajuan Duan, Long Jiang, Tao Qin, Ming Zhou và Heung, *An Empirical Study on Learning to Rank of Tweets,* COLING 2010: 295-303.

[4] Tie-Yan Liu, *Learning to Rank for Information Retrieval,* Foundations and Trends in Information Retrieval 3(3): 225-331, 2009.

[5] D.Sculley, *Combined Regression and Ranking,* KDD 2010: 979-988.

[6] D. Blei, A., Ng, and M. Jordan, *Latent Dirichlet Allocation In Journal of Machine Learning Research,* January/2003: 993-1022.

[7] Thomas Hofmann, *Probabilistic Latent Semantic Analysis,* UAI 1999: 289-196.

[8] Chunjing Xiao, Yuxia Xue, Zheng Li, Xucheng Luo và Zhiguang Qin, *Measuring User Influence Based on Multiple Metrics on YouTube,* PAAP 2015: 177-182.

[9] Fabián Riquelme và Pablo Gonzalez Cantergiani, *Measuring user influence on Twitter: A survey* , Inf. Process. Manage. 52(5): 949-975. 2016.

[10] Fredrik Erlandsson, Piotr Bródka và Anton Borg, *Finding Influential Users in Social Media Using Association Rule Learning,* Entropy 18(5). 2016.

[11] Bing Liu, "Chapter 2. Association Rules and Sequential Patterns," trong *Web Data Mining, 2nd Edition: Exploring Hyperlinks, Contents, and Usage Data*, Springer, 2011.

[12] Shea Bennet, *Twitter On Track For 500 Million Total Users By March, 250 Million Active Users By End Of 2012, http://www.mediabistro.com/alltwitter/twitter-active-total-users_b17655,* 2012.

[13] Cheng Li, Yue Lu, Qiaozhu Mei, Dong Wang và Sandeep Pandey, *Click-through Prediction for Advertising in Twitter Timeline,* KDD 2015: 1959-1968.

[14] Liangjie Hong, Ron Bekkerman, Joseph Adler và Brian Davison, *Learning to rank social update streams,* SIGIR'12: 651-660, 2012.

[15] Dominic Paul Rout, *A Ranking Approach to Summarising Twitter Home Timelines.,* PhD Thesis, The University of Sheffield, 2015.

[16] M. Rosen-Zvi, T. Griffiths, M. Steyvers và P. Sm, *The Author-Topic Model for Authors and Documents,* In Proc. of the 20th Conference on Uncertainty in Artificial Intelligence. 2004.

[17] Zhiheng Xu, Rong Lu, Liang Xiang và Qing Yang, *Discovering User Interest on Twitter with a Modified Author-Topic Model,* Web Intelligence and Intelligent Agent Technology (WI-IAT), 2011 IEEE/WIC/ACM International Conference on 2011.

[18] Charu C, Aggarwal và Jiawei Han, *Frequent Pattern Mining,* Springer. 2014.

[19] Norbert Fuhr, *Optimum polynomial retrieval functions based on the probability ranking principle,* ACM Transactions on Information Systems 7(3): 183–204, 1989.

[20] Joachims Thorsten, *Optimizing Search Engines using Clickthrough Data,* KDD'02: 133-142, 2002.

[21] Joachims Thorsten, *Making large-scale support vector machine learning practical,* Advances in kernel methods 1999, 169–184.

[22] Joachims Thorsten, *A support vector method for multivariate performance measures,* ICML 2005: 377–384.

[23] T. M. Mitchell, *Generative and discriminative classifiers: Naive bayes and logistic regression,* Machine Learning (Chapter 1), http://www.cs.cmu.edu/~tom/mlbook/NBayesLogReg.pdf, 2005.

[24] S. Rendle, C. Freudenthaler, Z. Gantner và S.-T. Lars, *BPR: Bayesian personalized ranking from implicit feedback,* CoR abs/1205.2618, 2012.

[25] Yehuda Koren và Joe Sill, *OrdRec: an ordinal model for predicting personalized item rating distributions,* RecSys 2011: 117–124.

[26] S.-H. Yang, B. Long, A. J. Smola, H. Zha và Z. Zheng, *Collaborative competitive filtering: learning recommender using context of user choice,* SIGIR 2011: 295–304.

[27] Xuan-Hieu Phan, Cam-Tu Nguyen, Dieu-Thu Le, Le-Minh Nguyen, Susumu Horiguchi, Senior Member, IEEE và Quang-Thuy Ha, *A Hidden Topic-Based Framework toward Building Applications with Short Web Documents*, tập 23 NO. 7, July 2011.

[28] Gregor Heinrich, *Parameter Estimation for Text Analysis*, Technical report, University of Leipzig, 2005.

[29] Rakesh Agrawal, Tomasz Imielinski và Arun N. Swami, *Mining Association Rules between Sets of Items in Large Databases*, SIGMOD Conference 1993: 207-216.