HỆ THỐNG TƯ VẤN THÔNG MINH

Lê Tuấn Anh

14th December 2017

LỜI CAM ĐOAN

Tôi cam đoan toàn bộ nội dung của báo cáo đều do tôi nghiên cứu thực hiện và biên soạn, không sao chép từ bất kì tài liệu nào khác. Các thông tin tham khảo trong báo cáo đều được nêu rõ nguồn gốc. Tôi sẽ chịu trách nhiệm nếu có bất cứ sai phạm nào so với lời cam kết.

Tp. Hồ Chí Minh, tháng 12 năm 2017

Lê Tuấn Anh

LỜI CẢM ƠN

Trong suốt thời gian làm việc và hoàn thành đề tài thực tập tốt nghiệp, Tôi xin gửi lời cảm ơn chân thành đến PGS. TS. Quản Thành Thơ, thầy đã giải đáp tận tình những thắc mắc của tôi và đồng thời giúp tôi định hướng cho đề tài của mình. Tôi cũng xin cảm ơn thầy Th.S Mai Đức Trung đã hỗ trợ tạo điều kiện cho tôi những buổi gặp mặt với thầy Thơ thuận tiện. Bên cạnh đó, Tôi xin cám ơn cô Đỗ Thị Minh Phụng, giảng viên trường đại học Công Nghệ Thông Tin, vì sự giúp đỡ nhiệt tình và những lời khuyên rất bổ ích đối với đề tài của tôi.

Tôi xin gửi lời cảm ơn tới tất cả các thầy cô trong khoa Khoa học và Kỹ thuật Máy tính đã hết lòng truyền dạy những kiến thức học thuật cũng như kinh nghiệm thực tế trong suốt quãng thời gian tôi học tập tại trường.

Tôi xin cảm ơn gia đình và bạn bè đã hỗ trợ, quan tâm và đồng hành cùng tôi trong suốt khoản thời gian 4 năm vừa qua.

Một lần nữa, tôi xin chân thành cảm ơn!

Tp. Hồ Chí Minh, tháng 12 năm 2017

Lê Tuấn Anh

Mục lục

1 Giới thiệu **7**

1.1 Giới thiệu đề tài 7

1.2 Mục tiêu, giới hạn và các giai đoạn của đề tài 8

1.2.1 Mục tiêu và giới hạn của đề tài 8

1.2.2 Giai đoạn Thực Tập Tốt Nghiệp 8

1.2.3 Giai đoạn Luận Văn Tốt Nghiệp 8

1.3 Phạm vi đề tài 9

1.4 Cấu trúc báo cáo 9

2 Phân tích vấn đề **10**

2.1 Ví dụ minh họa và phân tích 10

2.2 Giải pháp đề nghị 11

2.2.1 Hệ thống tư vấn 11

2.2.2 Cơ chế chọn môn học phù hợp với sinh viên 11

2.3 Kiến trúc hệ thống 12

3 Kiến thức nền tảng **13**

I Kiến thức nền tảng **14**

3.1 Lọc dựa trên nội dung (CBF: Content-based Filtering) 15

3.1.1 Ưu điểm và Nhược điểm 15

3.1.2 Cốt lõi sử dụng trong Lọc dựa trên nội dung 15

3.1.3 Ưu điểm và nhược điểm 17

3.1.4 Các phương pháp 18

II Công nghệ sử dụng **21**

4 KẾT QUẢ **23**

4.1 Kết quả đạt được trong giai đoạn Thực Tập Tốt Nghiệp 23

4.1.1 Tóm tắt kết quả 23

4.1.2 Kế hoạch cho giai đoạn Luận Văn Tốt Nghiệp 23

Danh sách hình vẽ

2.1 Hình 2.1.1: Quần jeans dành phái nam [1] và nữ [2] 10
2.2 Hình 2.1.2: Hiển thị thông tin sản phẩm [3] 10
2.3 Hình 2.3: Sơ đồ kiến trúc hệ thống 12
2.4 Bảng 2.4: Thiết kế hệ thống tư vấn 12

3.1 3.1: Dữ liệu của IoT and Analytics. 16
3.2 Hình 3.2: Công thức tính TF 16
3.3 Hình 3.3: Kết quả TF 16
3.4 Hình 3.4: Kết quả IDF 17
3.5 Hình 3.5: Sau khi chuẩn hóa 17
3.6 Hình 3.6: Kết quả độ tương tự 18
3.7 Hình 3.7: hai thuộc tính vector của một mặt hàng 18
3.8 Hình 3.8: Minh họa phân rã ma trận 19
3.9 Hình 3.9: Ví dụ cho những giá trị rời rạc thực của vector đặc trưng x 20

Danh sách bảng

4.1 Trình bày kế hoạch của tôi cho giai đoạn Luận Văn Tốt Nghiệp. . . . 23

Chương 1

Giới thiệu

Trong chương này, tôi sẽ giới thiệu tổng quan về đề tài, mục tiêu đặt ra của đề tài ở giai đoạn Thực Tập Tốt Nghiệp và Luận Văn Tốt Nghiệp cũng như giới hạn của đề tài. Cuối cùng là phần giới thiệu cấu trúc tổng quát của báo cáo.

1.1 Giới thiệu đề tài

Từ những năm 90 của thế kỷ 20, hệ thống tư vấn đã trở thành đề tài hấp dẫn đối với nhiều công ty. Hiện nay, việc tiếp cận thông tin là việc rất dễ dàng cho mọi người khi họ sử dụng Internet. Từ đó, nó cũng tạo ra một cuộc bùng nổ thông tin toàn cầu. Và đặc biệt cùng với sự phát triển của Thương Mại Điện Tử (E-Commerce), số lượng thông tin được trao đổi tăng theo cấp số nhân. Vì vậy, thông tin sẽ được cung cấp nhanh chóng. Tuy nhiên, sở thích của mỗi người là khác nhau. Vì thế, sẽ mất nhiều thời gian về việc tìm kiếm thông tin hoặc sản phẩm nào phù hợp với mục đích của người sử dụng. Việc sử dụng các công cụ tìm kiếm như Google, Bing, ... có thể giúp chúng ta cải thiện việc tìm những thông tin cần thiết, nhưng với lượng thông tin ngày càng tăng, các công cụ tìm kiếm không thể đáp ứng hoàn toàn nhu cầu của con người, bởi vì thời gian bỏ ra vẫn khá lớn khi thông tin cần phải được lọc để nhận được thông tin chính xác cho vấn đề chúng ta muốn tìm kiếm. Hoặc do chúng ta có thể sẽ chưa xác định rõ được thông tin chúng ta muốn tìm kiếm là gì. Hệ Thống Tư Vấn (Recommendation Systems) ra đời. Hệ thống này sẽ tư vấn cho người dùng những thông tin cần thiết cho họ thông qua tính toán và dự đoán sở thích, mong muốn của họ. Từ đó sẽ tư vấn cho người dùng những thông tin phù hợp nhất cho họ. Với hệ thống tư vấn này sẽ làm tăng những trải nghiệm thú vị, chất lượng cho khách hàng, thu hút và tạo ra những khách hàng tìm năng, tăng khả năng tương tác với khách hàng. Từ đó, sẽ giúp sản phẩm của công ty đến gần hơn với khách hàng, chất lượng phục vụ từ đó sẽ được cải thiện.

Ví dụ: khi người dùng truy cập vào một trang web thương mại điện tử nào đó (Ví dụ: Zalora), giả sử khách hàng là nam và đang cần tìm một chiếc quần bò. Khi họ truy cập vào thông tin chi tiết của sản phẩm họ cần tìm, hệ thống tư vấn có nhiệm vụ gợi ý những sản phẩm khác hoặc đi kèm, hoặc phù hợp với sở thích của khách hàng, chẳng hạn áo, đồng hồ, mắt kính, thắt lưng...

Ví dụ điển hình khác có thể là FaceBook. Khi một người mới đã có tài khoản, lần

đầu tiên khi người dùng này truy cập vào trang chủ của mình, họ sẽ được hệ thống tư vấn của FaceBook gợi ý kết bạn với những người bạn mới. Cách làm của hệ thống này có thể nó sẽ dựa vào những thông tin mà khi người dùng đã cung cấp khi đăng ký tài khoản, và thông qua mail của người dùng.

1.2 Mục tiêu, giới hạn và các giai đoạn của đề tài

1.2.1 Mục tiêu và giới hạn của đề tài

Mục tiêu của đề tài là xây dưng một hệ thống tư vấn môn học cho sinh viên, dựa trên năng lực, kết quả học tập, mức độ khó của môn học được đánh gía thông qua những sinh viên đã học... Để xây dựng hệ thống tư vấn, tôi sẽ áp dụng các phương pháp sẽ được đề cập trong bài báo cáo này.

Khi sinh viên đăng ký môn học, hệ thống sẽ tự động hiển thị những môn học phù hợp với năng lực của sinh viên.

Các thành phần của hệ thống cũng là sản phẩm của đề tài này như sau:

1.2.2 Giai đoạn Thực Tập Tốt Nghiệp

Mục tiêu của giai đoạn Thực Tập Tốt Nghiệp như sau:

- Tìm hiểu về Hệ Thống Tư Vấn.
- Làm thế nào để xây dựng Hệ Thống Tư Vấn.
- Các phương pháp tính toán khả năng môn học phù hợp với sinh viên như thế nào.
- So sánh ưu, nhược điểm của 2 phương pháp chính: lọc dựa trên nội dung (content-based filtering) và lọc cộng tác (Collaborative Filtering).
- Hiện thực hai phương pháp Matrix Factorization và Fatorization Machines bằng Python với bộ dữ liệu(dataset) MoviesLen, đây là tập dữ liệu cơ bản đánh giá về các bộ phim do người dùng đã đánh giá.

1.2.3 Giai đoạn Luận Văn Tốt Nghiệp

Mục tiêu của giai đoạn Luận Văn Tốt Nghiệp như sau:

- Xây dựng một hệ thống tư vấn với dữ liệu thật lấy từ trường đại học Công Nghệ Thông Tin và một số đặc điểm(feature) đặc trưng được lấy từ trang web: http://www.spoj.com/. Thông tin của dữ liệu đã được mã hóa để bảo đảm việc an toàn thông tin của trường.
- Áp dụng những phương pháp đã tìm hiểu trong giai đoạn Thực Tập Tốt Nghiệp với dữ liệu thật và so sánh để tìm được phương pháp có kết quả tốt nhất.

- Từ phương pháp trả về kết quả tốt nhất, tập trung phân tích sâu hơn về phương pháp này.

1.3 Phạm vi đề tài

Tập trung vào hai kỹ thuật để tư vấn cho người, Matrix Fatorization và Fatorization Machines.

Hệ thống chỉ áp dụng trong hệ thống của trường đại học Công Nghệ Thông Tin.

Các thông tin và sản phẩm tất cả đều từ dữ liệu của trường đại học Công Nghệ Thông Tin và trang web Spoj.

1.4 Cấu trúc báo cáo

Chương 2

Phân tích vấn đề

Trong chương này, Tôi sẽ trình bày các vấn đề đi kèm với các ví dụ cụ thể và phân tích, đưa ra giải pháp đề nghị phù hợp với giải pháp này.

2.1 Ví dụ minh họa và phân tích

Khi sinh viên truy cập vào trang đăng ký môn học của trường, những vấn đề gây khó khăn cho sinh viên khi trong việc tìm kiếm môn học:

Thông tin về môn học không nhiều, điều này không thể giúp sinh viên tự chọn những môn học phù hợp. Vì thế sẽ khiến sinh viên gặp khó khăn khi xác định được môn học sinh viên mong muốn. Ví dụ: môn học tên là Mật mã an ninh mạng, sinh viên có thể chỉ biết được khái quát môn học dựa trên summary hoặc các khóa học tương tự, nhưng sinh viên có thể không biết được môn này có vừa sức hoặc phù hợp với mình hay không.

Hình 2.1: Hình 2.1.1: Quần jeans dành phái nam [1] và nữ [2]

Hình 2.2: Hình 2.1.2: Hiển thị thông tin sản phẩm [3]

2.2 Giải pháp đề nghị

2.2.1 Hệ thống tư vấn

Có hai phương pháp phổ biến và được sử dụng rộng rãi, những phương pháp này sẽ được trình bày cụ thể hơn ở Chương 3: Lọc dựa trên nội dung (Content-based Filtering), Lọc cộng tác (Collaborative Filtering).

Hệ thống cần phải xây dựng hồ sơ sinh viên(user profile), cần có những thông tin về sinh viên như: điểm thi đại học(nếu là sinh viên năm nhất), điểm trung bình năm. Về thông tin môn học: đánh giá về môn học của những sinh viên đã học, mức độ khó của môn học trên mặt bằng chung, có thể dựa trên xác suất để xét môn học nằm trong ngưỡng nào đối với năng lực của nhiều sinh viên.

2.2.2 Cơ chế chọn môn học phù hợp với sinh viên

Ở giai đoạn đầu của quá trình hiện thực hệ thống, tôi sẽ sử dụng các thông tin sau:

- Thông tin về điểm thi đại học, điểm trung bình năm của sinh viên.
- Tôi sẽ đặt và chứng minh ngưỡng như thế nào để chia mức độ khó cho môn học.
- Tỉ lệ sinh viên đậu(≥ 5) hoặc điểm trung bình toàn bộ sinh viên so với toàn bộ sinh viên.

Tất cả các thông tin này đều do dữ liệu của website cung cấp trong cơ sở dữ liệu và được hệ thống xử lý, khai phá. Từ đó hiển thị môn học phù hợp với sinh viên.

Ở giai đoạn tiếp theo, tôi sẽ cân nhắc sử dụng thêm một số thông tin khác như: Thông tin cá nhân (giới tính, tuổi, nghề nghiệp...) từ mạng xã hội của sinh viên(có thể sẽ sử dụng thêm phương pháp Social Recommendation Systems).

Lưu ý: là những thông tin này sẽ được tôi cân nhắc kỹ và chưa chắc sẽ được sử dụng nếu nguồn lực và thời gian không cho phép.

2.3 Kiến trúc hệ thống

Dưới đây là mô hình tổng quan về kiến trúc hệ thống

Hình 2.3: Hình 2.3: Sơ đồ kiến trúc hệ thống

Front-end: sinh viên sẽ tương tác với trang web thông qua việc tìm kiếm hoặc click chọn một môn học mà sinh viên yêu thích, dữ liệu này sẽ được chuyển đến khối Adapter để chuẩn hóa dữ liệu thành một dạng dữ liệu đầu vào chung cho mọi định dạng.

Adapter: kết quả trả về sẽ là thông tin dữ liệu đã được chuẩn hóa theo định dạng đầu vào của Back-end.

Backend: nhiệm vụ chính là thông qua hệ thống tư vấn sẽ trả về kết quả danh sách những môn học được tư vấn thông qua giải thuật được sử dụng trong hệ thống tư vấn. Danh sách những môn học này sẽ được chuẩn hóa trở lại thành định dạng ban đầu của nó. Cuối cùng sau khi chuẩn hóa, phía Front-end có nhiệm vụ hiển thị danh sách những môn học đã được tư vấn.

Dưới đây là mô hình kiến trúc hệ thống tư vấn:

Hình 2.4: Bảng 2.4: Thiết kế hệ thống tư vấn

Adapter Pattern: do dữ liệu đầu vào có thể là nhiều file extension khác nhau như: .csv, .sql...

Khi dữ liệu qua Adapter Pattern, mọi dữ liệu đều được xử lý theo 1 định dạng chung cho mọi dữ liệu đầu vào.

Chương 3

Kiến thức nền tảng

Trong chương này, Tôi sẽ giới thiệu chi tiết về hai phương pháp đã nêu ở trên, nền tảng trong hệ thống của tôi.

Part I

Kiến thức nền tảng

3.1 Lọc dựa trên nội dung (CBF: Content-based Filtering)

Dữ liệu do người dùng cung cấp, có thể rõ ràng (được đánh giá) hoặc không rõ ràng. Dựa trên dữ liệu người dùng, hồ sơ người dùng được thiết lập, hồ sơ sẽ tư vấn cho người dùng. Sự tương tác, lịch sử tìm kiếm của người dùng càng nhiều với hệ thống, thì hệ thống sẽ tư vấn hiệu quả, chính xác và liên tục. Trong hệ thống lọc dựa trên nội dung, cần phải xây dựng cho mỗi mặt hàng một hồ sơ, hồ sơ này là một hoặc một tập hồ sơ đặc tả những tính đặc trưng quan trọng của mặt hàng. Trong một số trường hợp cụ thể, hồ sơ bao gồm các đặc trưng của mặt hàng mà nó dễ dàng được nhận thấy. Ví dụ, xem xét các đặc điểm của một bộ phim có thể liên quan đến hệ thống tư vấn:

- Tập các diễn viên trong bộ phim: sở thích của một số người dùng khi chọn xem một bộ phim nào đó có thể chỉ vì có diễn viên nổi tiếng và/hoặc người dùng yêu thích.
- Đạo diễn: một số đạo diễn được yêu thích do cốt truyện rành mạch, các yếu tố bất ngờ của bộ phim khiến người dùng.
- Năm của bộ phim được sản xuất: những bộ phim cũ (thông thường những bộ phim phiên bản đầu tiên thường lột tả được chính xác nội dung bộ phim), hoặc những bộ phim mới nhất tại thời điểm của họ vì hiệu ứng và công nghệ sẽ làm bộ phim thu hút người dùng.
- Thể loại của bộ phim: khi người dùng có quá nhiều bộ phim, họ thường sẽ dựa trên những thể loại họ yêu thích (hài hước, hành động, kinh dị....).

3.1.1 Ưu điểm và Nhược điểm

Ưu điểm: [1] Hệ thống có thể đưa ra tư vấn tốt nhất cho mỗi người dùng độc lập. Yêu cầu tính phân loại thấp, bởi vì mô hình người dùng có thể được tạo tự động. Nhược điểm: [1] Yêu cầu tính toán cao. Ví dụ: mỗi mặt hàng phải được phân tích bởi tính đặc trưng của nó, mô hình người dùng được xây dựng. Mô tả nội dung: Việc mô tả nội dung khó khăn. Ví dụ: Video, Music... Phụ thuộc tính năng của mặt hàng. Bỏ qua chất lượng và tính phổ biến của mặt hàng.

3.1.2 Cốt lõi sử dụng trong Lọc dựa trên nội dung

Sử dụng cơ chế TF-IDF [3] và Vector Space Model [4], ta có thể xác định được độ tương tự giữa hai document (tất cả những thông tin liên quan đến mặt hàng) được xét. [2]:

Sử dụng cơ chế Term Frequency (TF)and Inverse Document Frequency(IDF)

Thông qua các đặc điểm của document, để quyết định được quan hệ của một document, sử dụng cơ chế TF-IDF. Nói cách khác, độ tương tự (tương đồng) được định

nghĩa là khoảng cách giữa các điểm, hoặc là góc giữa những vector trong không gian n-chiều:Làm thế nào để tính TF-IDF? Thông qua ví dụ dưới đây: Giả sử tập của tất cả documents là 10^6. Mỗi DF, tổng các từ trong mỗi document. Ví dụ: DF của Analytics: 5000, Data: 50000...

Hình 3.1: 3.1: Dữ liệu của IoT and Analytics.

Hình 3.2: Hình 3.2: Công thức tính TF

Dựa vào công thức hình 3.2, ta có bảng sau:

Hình 3.3: Hình 3.3: Kết quả TF

Ví dụ: TF của (Article 1 – Analytics):

$1 + \lg tf_{Article1 \lor Analytics} = 1 + \lg 21_{Article1 \lor Analytics}$

Giá trị của (Article 1 - Clound) = 0 do giá trị tại vị trí này nhỏ hơn hoặc bằng 0. Để tính độ lớn của vector (Length of Vector), ta tính root-mean-squared-error: Ví dụ: giá trị của hàng Article 1 với tất cả các cột (Analytics, Data, Cloud, Smart, Insight):

$\sqrt{2.322219295^2 + 2.380211242^2 + 0^2 + 1.301029996^2 + 1.301029996^2} = 3.800456039$

Để tính IDF, ta tính $\lg \frac{N}{DF}$ với N: total docs là 1 triệu (10^6). Ta có: Ví dụ: IDF của (Article 1 – Analytics):

$\lg \frac{10^6}{DF_{Article1 \lor Analytics}} = \lg \frac{10^6}{5000} = 2.301029996$

Để chuẩn hóa vector, độ lớn document vector / term vector.

Ví dụ: chuẩn hóa giá trị (Article 1 – Analytics):

$\frac{TF_{Article1 \lor Analytics}}{Lengthofvector_{Article}} = \frac{2.322219295}{3.800456039} = 0.61103701$

Để xác định độ tương tự giữa hai article, tính cosine của 2 article đó, hay còn gọi là sum of dot product:

Ví dụ: cos (Article 1, Article 2) = 0.611*0.594 + 0.626*0.692 + 0*0.325 = 0.796 = $\cos(A_1, A_2)$

Kết luận: ta sẽ chọn giá trị cosin nào gần 1 nhất vì như thế thì góc được tạo bởi 2 Article đó càng về 0. Từ đó ta có thể nói rằng 2 Article đó càng tương tự nhau.

Sử dụng model Vector Space Model

Mô hình không gian vector (Vector Space Model) [4]: là một mô hình đại số (algebraic model) thể hiện thông tin văn bản như một vector, các phần tử của vector này thể hiện mức độ quan trọng của một từ và cả sự xuất hiện hay không xuất hiện (Bag of words) của nó trong một document. Mỗi mặt hàng được xem như là một vector, trong đó gồm n thuộc tính của vector (thuộc tính cũng được xem là vector) n-chiều.

Hình 3.4: Hình 3.4: Kết quả IDF

Hình 3.5: Hình 3.5: Sau khi chuẩn hóa

Như vậy góc giữa các vector được tính toán để xác định độ tương tự giữa các vector. Vector hồ sơ người dùng sẽ được tạo dựa trên hành động trên các mặt hàng đã tìm kiếm ở quá khứ

Hình 3.6 biểu diễn 2 thuộc tính trong không gian 2-D (2 chiều), Cloud và Analytics. M_1, M_2 là thuộc tính. U_1, U_2 là người dùng. Khi giá trị của Analytics tăng thì thuộc tính M_2 tăng nhanh hơn so với M_1, và ngược lại, Khi giá trị của Cloud tăng thì thuộc tính M_1 tăng nhanh hơn so với M_2. U1 thích thuộc tính Cloud hơn so với Analytics, và U_2 ngược với U_1. Từ đó ta có thể xét cosin góc giữa vector hồ sơ người dùng $U_i$và vector thuộc tính.

Lọc cộng tác (CF: Collaborative Filtering).

Thay vì sử dụng các đặc điểm của các mặt hàng để đưa ra quyết định về độ tương tự của mặt hàng, ta tập trung vào độ tương tự của 2 mặt hàng nào đó được người dùng xếp hạng. Lọc cộng tác là một giải thuật phổ biến trong hệ thống, dựa vào dự đoán và tư vấn trên đánh giá hoặc thao tác của người dùng trong hệ thống. Phân tích mối liên hệ giữa người dùng và sự phụ thuộc lẫn nhau giữa sản phẩm và người dùng mới. Giả sử nếu người dùng ưa thích những mặt hàng thông qua chất lượng, các thuộc tính đặc trưng. . . , người dùng có khả năng sẽ thích những mặt hàng khác. Ví dụ: một nhóm người cùng thích một mặt hàng nào đó, nếu John là thành viên trong nhóm đó, có khả năng John cũng sẽ thích mặt hàng đó. Lọc cộng tác gồm neighborhood methods và mô hình yếu tố ngầm (latent factor):

- Neighborhood methods: được đánh giá theo trọng số trung bình tổng thể của rating trên các mặt hàng.
- Latent factor: giải thích rating bằng đặc tả cả các mặt hàng lẫn người dùng.

3.1.3 Ưu điểm và nhược điểm

Ưu điểm [5]: Lọc cộng tác có nội dung độc lập, không giới hạn tính toán trong khi xử lý. Bởi vì rating do người dùng bình chọn, dữ liệu được đánh giá thật, đảm bảo chất lượng. Thời gian tính toán của lọc cộng tác nhanh hơn lọc dựa trên nội dung. Nhược điểm [5]: vấn đề "cold-start": Nếu người dùng mới không đánh giá hoặc không có mặt hàng, hệ thống không thể tìm thấy những người dùng có cùng sở thích. Nếu mặt hàng là mới trong hệ thống và chưa được đánh giá. Trong quan hệ mới, không có người dùng nào đánh giá mặt hàng. Dữ liệu thưa thớt => không thể tư vấn.

Hình 3.6: Hình 3.6: Kết quả độ tương tự

Hình 3.7: Hình 3.7: hai thuộc tính vector của một mặt hàng

3.1.4 Các phương pháp

Matrix Factorization (MF) [6]

Mô hình yếu tố ngầm (latent factor model) sử dụng Matrix Factorization. Đặc tả các mặt hàng và người dùng thông qua những vector yếu tố, được nội suy từ các mẫu xếp hạng mặt hàng. Ưu điểm: Khả năng mở rộng tốt với dự đoán chính xác. Linh hoạt với nhiều mô hình. Khi phản hồi không rõ ràng, hệ thống tư vấn có thể nội suy sở thích người dùng thông qua việc quan sát thao tác của người dùng: Lịch sử thanh toán, lịch sử duyệt web, mẫu tìm kiếm, những di chuyển của chuột. Nhược điểm: đối với dữ liệu nhỏ (thông thường dữ liệu của các công ty nhỏ hoặc dữ liệu được sử dụng rộng rãi (public data)) sẽ trả về kết quả tốt và nhanh chóng. Tuy nhiên, với dữ liệu lớn, như của các công ty thương mại điện tử, kết quả trả về không hiệu quả, thậm chí thời gian đưa ra kết quả không đảm bảo với yêu cầu phi chức năng của người dùng thường là tối đa 5 giây. Biểu thức toán học của Matrix Factorization: Để miêu tả sự tương tác giữa người dùng u và mặt hàng i – sở thích chung của người dùng với tính đặc trưng của mặt hàng, qua công thức:

$\hat{r} = q_i^T p_u$ (1)

Với q_i: mặt hàng i đại diện cho sự liên kết giữa mặt hàng và những đặc tính của mặt hàng đó, p_u: người dùng u đại diện cho sự liên kết giữa người dùng u và những đặc tính của mặt hàng đó. Mục tiêu chính của cách tiếp cận này chính là tìm mối liên hệ giữa mỗi cặp mặt hàng và người dùng, từ đó sẽ tạo thành vector yếu tố q_i, $p_u \in R^f$. Khi đã có những vector yếu tố này, công việc đánh giá xếp hạng của người dùng dành cho item đó thông qua công thức (1). Mô hình tiêu biểu trong lọc công tác thường được sử dụng đó là SVD (Singular Value Decomposition). Mô hình này yêu cầu ma trận, trong đó mỗi giá trị của ma trận là giá trị rating của người dùng cho mặt hàng đó. Nhưng vấn đề khó khăn khi sử dụng mô hình này là khi khá nhiều giá trị trong ma trận missing (nghĩa là những mặt hàng đó không được người dùng đánh giá). Để tìm các vector yếu tố (p_u, q_i), hệ thống tối thiểu cost function:

$min_{q^*,p^*} \sum_{(u,i \in K)} (r_{ui} - q_i^T {p_u}^2) + \lambda(\| q_i \|^2 + \| p_u \|^2)$ (2)

λ cross-validation: chia dữ liệu thành k tập con có cùng kích thước. Tại mỗi tập, ta xét 1 tập con trong tập đó là tập test, các tập còn lại là tập training. Có 2 cách tiếp cận để là tối thiểu hóa giá trị của công thức (2), đó là Stochastic Gradient Descent (SGD) và Alternating Least Squares (ALS):

- SGD: Với mỗi trường hợp trong tập training, hệ thống sẽ dự đoán rui và kiểm tra giá trị dự đoán chênh lệch bao nhiêu so với giá trị chính xác được tính ra từ trường hợp đó trong tập training:

  $e_{ui} = \hat{r} = q_i^T p_u$

  Sau đó, nhiệm vụ tiếp theo là thay đổi các giá trị của (q_i, p_u) dựa trên các

giá trị kiểm tra. Công việc này sẽ dùng vòng lặp cho tới khi giá trị kiểm tra nhỏ nhất và giá trị của (q_i, p_u) phù hợp nhất.

- ALS: một cách tiếp cận khác là sử dụng Alternating Least Squares. Trong kỹ thuật ALS, khi ta cố định tất cả các giá trị (ví dụ p_u), hệ thống sẽ tính q_i thông qua xử lý vấn đề least-squares, hoặc ngược lại.

  Trong khi SGD dễ dàng sử dụng và hiện thực và thời gian xử lý nhanh hơn so với ALS, nhưng ALS lại được ưu chuộng hơn khi sử dụng vì 2 lý do: bởi vì 1 trong 2 giá trị (q_i, p_u) đã được cố định, vì thế hệ thống sẽ chỉ cần tính hoặc q_i hoặc p_u độc lập với nhau. Thứ hai là hiệu quả trong khi tập dữ liệu không rõ ràng, cụ thể.

Thêm Bias: ta có thể hiểu đơn giản bias khi được thêm vào, nó sẽ ảnh hưởng đến mối liên hệ giữa user và item. Ví dụ: trong lọc cộng tác, dữ liệu trong hệ thống lớn có khuynh hướng là một số user đánh giá các item cao hơn so với người khác.

$$q_i \leftarrow q_i + \gamma(e_{ui}p_u - \lambda q_i)$$

$$p_u \leftarrow p_u + \gamma(e_{ui}q_i - \lambda p_u)$$

Trong đó bui là bias ảnh hưởng đến giá trị xếp hạng rui, bu, bi: độ quan sát của người dùng và mặt hàng từ μ với μ: xếp hạng trung bình tổng thể. Từ đó, ta có thể mở rộng cách tính tối thiểu cost function – squared error function:

$$\hat{r_{ui}}(t) = \mu + b_i(t) + b_u(t) + q_i^T p_u(t)$$

Để làm giảm trọng số quan sát, ta có thể thêm mức độ tin tưởng cui vào công thức (4):

$$min_{p*,q*,b*} \sum_{(u,i)\in K} c_{ui}(r_{ui} - \mu - b_u - b_i - q_i^T p_u)^2 + \lambda(\| p_u \|^2 + \| q_i \|^2 + b_u^2 + b_i^2)$$

Hình 3.8: Hình 3.8: Minh họa phân rã ma trận

Để dự đoán xếp hạng của người dùng hàng 2, cột 2, ta áp dụng biểu thức (1), ta có:

R_{ui} = 0.5*0.8 + 0.6*0.1 = 0.46

Factorization Machines [8]

Với những tập dữ liệu nhỏ, việc sử dụng matrix factorization để xử lý những vấn đề missing rating sẽ nhận được những kết quả tích cực. Nhưng đối với tập dữ liệu lớn (Ví dụ: tập dữ liệu của các công ty thương mại điện tử), lượng dữ liệu cần được xử lý là rất lớn, kết quả khi sử dụng matrix factorization không đạt được hiệu quả như mong đợi. Vì vậy một cách tiếp cận khác để xử lý vấn đề dữ liệu thưa thớt lớn (missing value). Đó là Factorization Machines, nó sẽ kết hợp những ưu điểm của Support Vector Machine (SVM) và mô hình phân rã ma trận (factorization model).

Điểm giống với SVM đó là đều tính toán trên những giá trị thực của các vector yếu tố. Nhưng trái với SVM, FMs mô hình hóa tất cả sự tương tác giữa các biến được sử dụng trong các tham số được phân rã => có khả năng đánh giá sự tương tác lẫn nhau, thậm chí trong vấn đề dữ liệu thưa thớt lớn mà SVM không giải quyết được.

Và ưu điểm khác của FMs đó là thời gian tính toán là tuyến tính, như vậy độ phức tạp chỉ là O(n) => Tối ưu hóa vấn đề.

Ưu điểm: Cho phép đánh giá tham số khi dữ liệu thưa thớt. Độ phức tạp là tuyến tính, được tối ưu. Làm việc với các giá trị thực của vector đặc trưng.

Hình 3.9: Hình 3.9: Ví dụ cho những giá trị rời rạc thực của vector đặc trưng x

Mỗi hàng biểu diễn vector đặc trưng $x^(i)$ với mục tiêu trả về là $y^(i)$. Theo như hình 3.7, hầu hết các giá trị của vector đặc trưng x đều bằng 0. Đặt m(x) là số lượng những phần tử của vector đặc trưng x bằng 0 và mD là trung bình số lượng những phần tử còn lại khác 0 trong vector đặc trưng x.

Biểu thức toán học của Factorization Machines:

- Trường hợp 2 chiều, biểu thức được biểu diễn như sau:

$$\hat{y}(x) := w_0 + \sum_{i=1}^{n} w_i x_i + \sum_{i=1}^{n} \sum_{i=i+1}^{n} \langle v_i, v_j \rangle x_i x_j \ (1)$$

Với: $\langle v_i, v_j \rangle := sum_{f=1}^{k} v_{i,f} v_{j,f}$

w_0: bias toàn cục, w_i: trọng số của biến thứ i. $<v_i, v_j>$ mô hình hóa sự tương tác giữa biến thứ i và j để phân rã.

Với dữ liệu thưa thớt lớn sẽ không đủ dữ liệu để đánh giá sự tương tác giữa các biến một cách trực tiếp và độc lập. FMs có thể xử lý được vấn đề này bằng cách loại bỏ các tham số độc lập bằng cách phân rã chúng. Có nghĩa là một dữ liệu không missing có thể giúp ta đánh giá những dữ liệu missing khi những dữ liệu này có liên quan đến dữ liệu tương tác với chúng.

$$\sum_{i=1}^{n} \sum_{j=i+1}^{n} \langle v_i, v_j \rangle x_i x_j = \frac{1}{2} \sum_{f=1}^{k} ((\sum_{i=1}^{n} v_{i,f} x_i x_j)^2 - \sum_{i=1}^{n} v_{i,f}^2 x_i^2)$$

Từ biểu thức trên, ta có thể thấy thời gian tính toán sẽ là O(kn).

Với tác vụ dự đoán, FMs có đủ khả năng iđược sử dụng trong 3 nhóm sau đây:

Regression, Binary Classification, Ranking. Trong những trường hợp này, biểu thức chính quy sẽ được thêm vào để tránh vấn đề overfitting.

Để tối ưu hóa thời gian tính toán theo tuyến tính, các tham số chính w_0, w, V sẽ được tối ưu hóa thông qua phương thức Stochastic Gradient Descent (SGD), bằng cách đạo hàm biểu thức (1) theo các tham số nêu trên. Từ đó ta có biểu thức sau:

Theo biểu thức trên, mỗi giá trị góc sẽ có thời gian tính toán là O (1) => toàn bộ thời gian tính toán là O(kn). Điều này đã chứng minh được FMs có thời gian tính toán là tuyến tính.

Trường hợp d-chiều, biểu thức sẽ được mở rộng dựa trên biểu thức (1) như sau:

$$\hat{y}(x) := w_0 + \sum_{i=1}^{n} w_i x_i + \sum_{l=2}^{d} \sum_{i_1}^{n} \dots \sum_{i_l=i_{l-1}+1}^{n} (\Pi_{j=1}^{l} x_{i_j}) (\sum_{f=1}^{k_l} \Pi_{j=1}^{l} v_{i_{i,f}}^{l}) \qquad (2)$$

Nếu theo biểu thức (2), thời gian tính toán sẽ trở thành O (k_d, n^d).

Part II

Công nghệ sử dụng

Hiện nay có rất nhiều ngôn ngữ hỗ trợ rất tốt trong lĩnh vực Data Science, nhưng phổ biến và được ưu chuộng nhất hiện nay có thể là Python, R, và C++ (ngoài ra còn có Matlab vì đây cũng là một công cụ khá mạnh và hiệu quả khi ta cần xử lý ảnh hoặc cần tìm những điểm nổi bật khi thống kê số liệu để đưa ra những kết luận đặc trưng). Những ngôn ngữ này hỗ trợ rất nhiều thư viện để xử lý vấn đề trong Machine Learning, DataMining, Image Processing, Recommendation Systems...

Nhưng trong bài báo cáo này, tôi tập trung sử dụng ngôn ngữ Python vì: dễ dàng sử dụng, cú pháp đơn giản, ngắn gọn, cụ thể, thư viện hỗ trợ khá phong phú. Python được ra đời lần đầy tiên vào năm 1991 bởi Guido van Rossom. Python được sử dụng rộng rãi trên toàn thế giới và là ngôn ngữ được các trường đại học tại Mỹ ưu dùng để giảng dạy. Python được hỗ trợ tốt trên cả ba hệ điều hành phổ biến nhất hiện nay là Windows, Linux, MacOS, dễ dàng cài đặt và sử dụng.Ngoài ra JetBrain còn đưa ra IDE được dùng cho Python, đó là PyCharm. Python còn hỗ trợ lập trình hướng đối tượng (OOP-Oriented Object Programming).

Các thư viện hỗ trợ trong Machine Learning, đó là: Numpy, Scipy, Pandas, Matplotlib, Scikit-learn... Và trong báo cáo này, tôi đều sử dụng tất cả các thư viện vừa nêu trên.

Ngoài ra, đối với Matrix Fatorization, ta có thể cài đặt gói package của Python tên là PyMF,đối với Factorization Machines, fastFM là thư viện khá tốt để hiện thực bằng Python.

Chương 4

KẾT QUẢ

Trong chương này, tôi sẽ tóm tắt kết quả mà tôi đạt được trong giai đoạn Thực Tập Tốt Nghiệp. Cuối cùng là phần kế hoạch của tôi cho giai đoạn Luận Văn Tốt Nghiệp.

4.1 Kết quả đạt được trong giai đoạn Thực Tập Tốt Nghiệp

4.1.1 Tóm tắt kết quả

Trong giai đoạn Thực Tập Tốt Nghiệp, tôi đã đạt được các kết quả sau:

- Phân tích tính thực tiễn, xác định hướng đi tốt nhất cho hệ thống.
- Tìm hiểu về Hệ thống tư vấn, những vấn đề cần giải quyết trong hệ thống tư vấn.
- Các phương pháp sử dụng trong hệ thống.
- Hiện thực trên dữ liệu thật tìm trên mạng.

Bên cạnh đó, quá trình làm việc của tôi từ lúc bắt đầu đề tài đến lúc kết thúc giai đoạn Thực Tập Tốt Nghiệp cũng được đính kèm ở Phụ Lục A.

4.1.2 Kế hoạch cho giai đoạn Luận Văn Tốt Nghiệp

Bảng 4.1: Trình bày kế hoạch của tôi cho giai đoạn Luận Văn Tốt Nghiệp.

Thời gian	Công việc dự kiến
06/2017 - 09/2017	Xin dữ liệu thật từ công ty Zalora, trụ sở chính ở Singa Sau khi nhận được dữ liệu thật, tiến hành hiện thực các phương pháp trên và