**Tài liệu báo cáo đồ án chuyên ngành**

**Bản ghi thay đổi**

\*A - Added M - Modified D - Deleted

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Effective Date | Changed Items | A\* M, D | Change Description | New Version |
| Sep 10 |  | A | The first version | 1.0 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Nội dung

Bảng nội dung

* 1. Mục đích
  2. Định nghĩa, viết tắt, và mô tả tóm tắt
  3. Tài liệu

1. Các yêu cầu ở mức cao

2.1 Product Perspective

2.2 Miêu tả tác nhân, ca sử dụng

2.2.1 Biểu đồ

2.2.2 Miêu tả tác nhân

2.2.3 Miêu tả các ca sử dụng

2.2.4 Biểu đò tác nhân và UC

2.3 Môi trường hệ thống

2.4 Giả thuyết và các điều kiện lệ thuộc

3 Yêu cầu về chức năng

3.1 Module 1

4 Thiết kế

4.1 CSDL

4.2 Thuật toán

6 Lập trình

6.1 Thành viên

7.Kiểm thử

8 Bảo trì

**Mở đầu**

Hệ thống gợi ý (Recommender System - RS) được ứng dụng khá thành công trong thực tiễn giúp người dùng giải quyết vấn đề quá tải thông tin. Hiện nay, hệ thống gợi ý đang được nghiên cứu và ứng dụng ở nhiều lĩnh vực khác nhau đặc biệt là thương mại điện tử. Trên thế giới, đã có nhiều công ty, tổ chức áp dụng thành công hệ thống gợi ý như là một dịch vụ thương mại của mình nhằm gợi ý các dịch vụ, sản phẩm và các thông tin cần thiết đến người dùng như: website mua sắm trực tuyến Amazon (www.amazon.com) cung cấp cho khách hàng những sản phẩm mà họ có thể quan tâm, cổng video clip YouTube (www.youtube.com), giới thiệu phim của MovieLens (www.movielens.org),... Việc gợi ý sản phẩm phù hợp góp phần làm tăng doanh số bán hàng hoặc số lượng truy cập, download của hệ thống. Đồng thời giúp cho khách hàng có thể tìm kiếm được những thông tin thú vị hoặc những sản phẩm mà họ muốn tìm dễ dàng hơn. Hệ thống gợi ý giúp người dùng chọn lựa được thông tin phù hợp nhất cho mình dựa trên những hành vi/phản hồi (feedbacks) mà người dùng đã thực hiện trong quá khứ. Các phản hồi có thể được xác định một cách tường minh (explicit feedback) như thông qua việc đánh giá/xếp hạng (ví dụ, rating từ 1 đến 5; hay like (1) và dislike (0),…) mà người dùng đã bình chọn cho trên sản phẩm – trong trường hợp này gọi là dự đoán xếp hạng (rating prediction) [4] hoặc các phản hồi có thể được xác định một cách không tường minh hay còn gọi là tiềm ẩn (implicit feedback) như số lần click chuột, số lần chọn mua sản phẩm, thời gian mà người dùng đã duyệt/xem sản phẩm,… Rất nhiều hệ thống lớn thu thập thông tin phản hồi từ khách hàng một cách tường minh, như Ebay, Amazon, LastFM, NetFlix,.. ở đó người dùng bình chọn sản phẩm từ đến (rất thích); hay Youtube thu thập thông tin qua like()/ disklike(), và các hệ thống khác [3]. Thông qua việc thu thập phản hồi tường minh, hệ thống dễ dàng xác định rõ mức độ yêu thích của người dùng trên sản phẩm, từ đó dự đoán các sản phẩm tiếp theo mà người dùng có thể thích để gợi ý cho họ. Tuy nhiên, điều này có thể gây bất lợi do không phải người dùng lúc nào cũng sẳn sàng/vui lòng để lại các phản hồi của họ, vì vậy hệ thống phải nên tự xác định người dùng cần gì thông qua phản hồi tiềm ẩn. Trong bài viết này, chúng tôi đề xuất một giải pháp xây dựng hệ thống gợi ý cho bán hàng trực tuyến, sử dụng phản hồi tiềm ẩn từ người dùng (như số lần duyệt/xem sản phẩm, số lần mua sản phẩm). Trước hết chúng tôi đề xuất phương pháp thu thập và khai thác thông tin phản hồi tiềm ẩn từ người dùng, sau đó lựa chọn và đề xuất kết hợp các mô hình sử dụng thông tin phản hồi tiềm ẩn. Kế đến là việc xây dựng hệ thống và tích hợp các giải thuật gợi ý vào hệ thống. Sau khi có hệ thống hoàn chỉnh, chúng tôi thu thập dữ liệu từ người dùng thực nhằm đánh giá hiệu quả của hệ thống gợi ý. Kết quả cho thấy khả năng mà hệ thống gợi ý phù hợp với sở thích của từng người dùng là khá tốt.

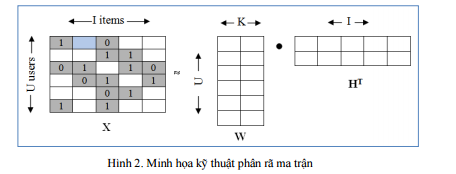
II. HỆ THỐNG GỢI Ý (Recommender Systems - RS) A. Hệ thống gợi ý Mục đích của hệ thống gợi ý (RS) là dựa vào sở thích, thói quen, nhu cầu,... trong quá khứ của người sử dụng để dự đoán sở thích trong tương lai của họ. Trong hệ thống gợi ý người ta quan tâm đến 3 đối tượng: người dùng (user), sản phẩm (item - item gọi chung là mục tin nhưng trong bài viết này liên quan đến gợi ý sản phẩm nên từ đây về sau chúng tôi tạm gọi item là sản phẩm) và các đánh giá sở thích của người dùng trên sản phẩm gọi là rating. Thông thường người ta gọi U là tập tất cả người dùng (users) và u là một người dùng cụ thể nào đó (u∈U). I là tập tất cả các sản phẩm (items) sẽ được gợi ý như máy tính, sách, phim ảnh,.. và i là một sản phẩm cụ thể nào đó (i∈I). I là tập các sản phẩm có thể lên đến hàng trăm, hàng nghìn hoặc thậm chí là hàng triệu sản phẩm trong một số ứng dụng, như việc gợi ý về sách, phim ảnh, âm nhạc. Tương tự như vậy, tập người dùng U cũng có thể rất lớn, lên đến hàng triệu trường hợp. R là một tập hợp các giá trị dùng để ước lượng ‘sở thích’ (preference) của người dùng, và rui∈R (R⊂ℜ) là xếp hạng của người dùng u trên sản phẩm i. Giá trị rui có thể được xác định một cách tường minh (explicit feedback) như thông qua việc đánh giá/xếp hạng (ví dụ, rating từ 1 đến 5; hay like (1)/ dislike (0),…) mà u đã bình chọn cho i – trong trường hợp này gọi là dự đoán xếp hạng (rating prediction) hoặc rui có thể được xác định một cách không tường minh hay còn gọi là tiềm ẩn (implicit feedback) như số lần click chuột, số lần chọn mua sản phẩm, thời gian mà u đã duyệt/xem i,… [2][7] Nội dung bài viết này, chỉ quan tâm nhiều đến cách xác định rui không tường minh. Các thông tin này được biểu diễn thông qua một ma trận như trong Hình 1. Trong đó mỗi dòng là một người dùng u, mỗi cột là một sản phẩm i, và giao giữa dòng và cột là các đánh giá rating của người dùng như số lần click chuột hay chọn mua sản phẩm,…. Các ô có giá trị là những item mà các user đã xem đến hoặc chọn mua trong quá khứ. Những ô trống là những item chưa được xem đến (điều đáng lưu ý là mỗi user chỉ click xem hoặc chọn mua cho một vài item trong quá khứ, do vậy có rất nhiều ô trống trong ma trận này – còn gọi là ma trận thưa – sparse matrix). Hình 1. Ma trận biểu diễn xếp hạng của người dùng trên sản phẩm (user-item-rating matrix) Nhiệm vụ chính của RS là dựa vào các ô đã có giá trị trong ma trận này (dữ liệu thu được từ quá khứ), để dự đoán các ô còn trống (của user hiện hành), sau đó sắp xếp kết quả dự đoán (ví dụ, từ cao xuống thấp) và chọn ra Top-N items theo thứ tự, từ đó gợi ý chúng đến người dùng. Một cách hình thức, nếu gọi Dtrain ⊆ U × I × R là tập dữ liệu huấn luyện, Dtest ⊆ U × I × R là tập dữ liệu kiểm thử, và một ánh xạ r: U × I→ R (u, i) ↦ rui Mục tiêu của RS là tìm một hàm ̂: U × I → ℜ sao cho ξ(r, ̂) thỏa mãn một điều kiện nào đó. Ví dụ, nếu ξ là một hàm ước lượng lỗi như Mean Absolute Error (MAE) hay Root Mean Squared Error (RMSE) thì nó cần phải được tối tiểu. ∑ ( ) ∈ = − test r)i,(u, D ui ui test r ˆ D| | 1 MAE r (1) RMSE = ∑( ) ∈ − test Driu test ui iu r r D ),,( 2 ),( ˆ | | 1 (2) Hiện nay, có rất nhiều giải thuật được đề xuất cho hệ thống gợi ý, chúng có thể được gom lại theo 3 nhóm [1][2][7]: - Gợi ý dựa trên cộng tác: người dùng sẽ nhận gợi ý những sản phẩm được ưa thích xuất phát từ những người có cùng thị hiếu và sở thích với mình. Nhóm này dựa vào các phương pháp chủ yếu: o Phương pháp láng giềng (Neighborhood-based, còn gọi là Memory-based), trong đó hoặc là dựa trên dữ liệu quá khứ của người dùng “tương tự - similarity” (user-based approach), hoặc là dựa trên dữ liệu quá khứ của những item “tương tự” (item-based approach). o Dựa trên mô hình (Model-based): Nhóm này liên quan đến việc xây dựng các mô hình dự đoán dựa trên dữ liệu thu thập được trong quá khứ. Như mô hình Bayesian, các mô hình nhân tố tiềm ẩn (latent factor models): trong đó kỹ thuật phân rã ma trận (matrix factorization) là một điển hình. - Gợi ý dựa trên nội dung: người dùng sẽ được gợi ý những sản phẩm tương tự với những sản phẩm đã được người dùng đó ưa thích trước đây. - Gợi ý dựa trên cách tiếp cận kết hợp: kết hợp hai phương pháp tiếp cận dựa trên nội dung và cộng tác. Sau đây chúng tôi tóm lược lại một trong những kỹ thuật trong nhóm lọc cộng tác của hệ thống gợi ý và kỹ thuật sử dụng phản hồi tiềm ẩn, từ đó làm cơ sở cho việc đề xuất mô hình cho hệ thống.

**1.1 Mục đích**

Mục đích của hệ thống gợi ý (RS) là dựa vào sở thích, thói quen, nhu cầu,... trong quá khứ của người sử dụng để dự đoán sở thích trong tương lai của họ. Trong hệ thống gợi ý người ta quan tâm đến 3 đối tượng: người dùng (user), sản phẩm (item) và các đánh giá sở thích của người dùng trên sản phẩm gọi là rating.Tài liệu này chỉ định được dung cho các thành viên trong dự án thực hiện và xác định đúng các chức năng của hệ thống.

**1.2 Các kỹ thuật gợi ý**

B. Kỹ thuật phân rã ma trận (matrix factorization – MF) Kỹ thuật phân rã ma trận (MF) là một trong những phương pháp dựa trên mô hình thành công nhất hiện nay (stateof-the-art) trong RS [1][2]. MF là việc chia một ma trận lớn X thành hai ma trận có kích thước nhỏ hơn W và H, sao cho ta có thể xây dựng lại X từ hai ma trận nhỏ hơn này càng chính xác càng tốt [5]. X ~ WHT như minh họa như trong Hình 2. Trong đó, W ∈ℜ|U|×K là một ma trận mà mỗi dòng u là một véc tơ bao gồm K nhân tố tiềm ẩn (latent factors) mô tả người dùng u; và H ∈ℜ|I|×K là một ma trận mà mỗi dòng i là một véc tơ bao gồm K nhân tố tiềm ẩn mô tả cho item i (lưu ý: K<<

****

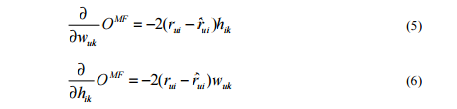
Gọi wuk và hik là các phần tử tương ứng của hai ma trận W và H, khi đó xếp hạng của người dùng u trên mục tin i được dự đoán bởi công thức:

****

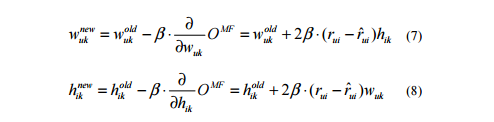
Như vậy, vấn đề chủ chốt của kỹ thuật MF là làm thế nào để tìm được giá trị của hai tham số W và H. Hai tham số này có được bằng cách tối ưu hóa hàm mục tiêu (objective function) như RMSE ở công thức (2). Ta có thể viết lại hàm mục tiêu của MF như sau:

****

Một trong những kỹ thuật có thể dùng để tối ưu hóa hàm mục tiêu là dùng SGD (Stochastic Gradient Descent) [5]. Để tối ưu hóa hàm mục tiêu (4), trước tiên ta khởi tạo các giá trị ngẫu nhiên cho W và H, sau đó từng bước cập nhật giá trị của chúng cho đến khi hàm mục tiêu hội tụ về giá trị nhỏ nhất (convergence). Để làm được điều đó, ta cần phải xác định là nên tăng hay nên giảm các giá trị của W và H qua mỗi lần cập nhật, do vậy cần phải tìm đạo hàm tương ứng của chúng:

****

Sau khi tìm đạo hàm, các phần tử của W và H sẽ được cập nhật ngược hướng với giá trị của đạo hàm, qua công thức:

****

Trong đó β là tốc độ học (learning rate **)**Quá trình cập nhật sẽ được thực hiện đến khi nào hàm mục tiêu đạt được giá trị nhỏ nhất.

Chính tắc hóa (Regularization): Để ngăn ngừa sự quá khớp hay còn gọi là học vẹt (overfitting – xảy ra khi mô hình dự đoán cho kết quả tốt trên dữ liệu huấn luyện, nhưng cho kết quả kém trên dữ liệu thử nghiệm) người ta thay đổi hàm mục tiêu (4) bằng cách thêm vào một đại lượng gọi là chính tắc hóa (regularization) để điều khiển độ lớn của các giá trị trong W và H. Hàm mục tiêu (4) bây giờ trở thành:

**1.2 Các định nghĩa, viết tắt và mô tả**

|  |  |  |
| --- | --- | --- |
| **#** | **Abbreviations/Terms** | **Explanation** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Table 1: Abbreviations & Terminologies

1.3 Tài liệu

|  |  |  |
| --- | --- | --- |
| # | Tên | Mô tả |
| 1 | PHƯƠNG PHÁP XÂY DỰNG HỆ THỐNG GỢI Ý SẢN PHẨM SỬ DỤNG PHẢN HỒI TIỀM ẨN | Tác giả : Lưu Nguyễn Anh Thư, Nguyễn Thái Nghe  Thuộc: Khoa Công Nghệ Thông Tin và Truyền Thông, Trường Đại Học Cần Thơ |

Bảng 2: Tài liệu

**2 Các yêu cầu ở mức cao**

## 2.1 Product Perspective

Ứng dụng được viết nhằm mục đích xây dựng để tạo ra một bảng kết quả sau đó sẽ đẩy lên web. Do đó, chúng ta có thể dung bất kì hệ điều hành nào hay bất kì ngôn ngữ lập trình nào để xây dựng. Trong tài liệu này chúng tôi xin đề xuất về việc làm trên hệ điều hành window với ngôn ngữ lập trình C# sử dụng nền tảng .Net . Còn môi trường web, trong tài liệu chúng tôi xấy dựng trên mã nguồn mở PHP và sử dụng framework joomla.

2.2 Miêu tả tác nhân và các ca sử dụng

2.2.1 biểu đồ

2.2.2 Miêu tả tác nhân

|  |  |  |
| --- | --- | --- |
| **#** | **Actor Name** | **Definition & Interests** |
| 1 | Admin |  |
| 2 | Division Leader |  |
| 3 | Staff |  |
| 4 | BOD |  |
| 5 | System |  |

2.2.3 Miêu tả UC

2.2.4 Biểu đồ UC và Actor

2.3 Môi trường hệ điều hành

Trong tài liệu này chúng tôi đê xuất thông tin cần thiết sau:

OS: yêu cầu HDH window XP trở lên. Đề xuất window 7

Kĩ thuật: RAM 1Gb trở lên, HDD: 10Gb trở lên, chip : Pentium trở lên, Xung: 1Ghz trở lên

Công cụ: Visual studio, Microsoft SQL server, xmind, MS project, MS visio, Rational Rose. Đề xuất: visual studio 2012 , MS SQL server 2012,

2.4 Giả thiết và các điều kiện lệ thuộc

3. Yêu cầu chức năng

3.1 Module 1: Lấy dữ liệu động

3.2 Module 2: Phân cụm người dung

3.3 Module 3: Gợi ý sản phẩm

3.4 Module 4: Gợi ý quảng cáo