**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA CÔNG NGHỆ PHẦN MỀM**

**TRẦN MINH LUẬN**

**NGUYỄN THANH ANH TUYÊN**

**KHÓA LUẬN TỐT NGHIỆP**

**XÂY DỰNG FRAMEWORK ĐÁNH GIÁ**

**CÁC THUẬT TOÁN KHUYẾN NGHỊ VIỆC LÀM**

**KỸ SƯ NGÀNH KỸ THUẬT PHẦN MỀM**

**TP. HỒ CHÍ MINH, 2016**

**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA CÔNG NGHỆ PHẦN MỀM**

**TRẦN MINH LUẬN – 12520248**

**NGUYỄN THANH ANH TUYÊN - 12520492**

**KHÓA LUẬN TỐT NGHIỆP**

**XÂY DỰNG FRAMEWORK ĐÁNH GIÁ**

**CÁC THUẬT TOÁN KHUYẾN NGHỊ VIỆC LÀM**

**KỸ SƯ NGÀNH KỸ THUẬT PHẦN MỀM**

**GIẢNG VIÊN HƯỚNG DẪN**

**TS. HUỲNH NGỌC TÍN**

**TP. HỒ CHÍ MINH, 2016**

DANH SÁCH HỘI ĐỒNG BẢO VỆ KHÓA LUẬN

Hội đồng chấm khóa luận tốt nghiệp, thành lập theo Quyết định số …………………… ngày ………………….. của Hiệu trưởng Trường Đại học Công nghệ Thông tin.

* 1. …………………………………………. – Chủ tịch.
  2. …………………………………………. – Thư ký.
  3. …………………………………………. – Ủy viên.
  4. …………………………………………. – Ủy viên.

|  |  |
| --- | --- |
| **ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**  **TRƯỜNG ĐẠI HỌC**  **CÔNG NGHỆ THÔNG TIN** | **CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**  **Độc Lập - Tự Do - Hạnh Phúc** |
|  | ***TP. HCM, ngày…..tháng…..năm……..*** |

NHẬN XÉT KHÓA LUẬN TỐT NGHIỆP

(CỦA CÁN BỘ HƯỚNG DẪN)

|  |  |  |
| --- | --- | --- |
| **Tên khóa luận:** | | |
| **XÂY DỰNG FRAMEWORK ĐÁNH GIÁ CÁC THUẬT TOÁN KHUYẾN NGHỊ VIỆC LÀM** | | |
| **Nhóm SV thực hiện:** | | **Cán bộ hướng dẫn:** |
| Trần Minh Luận | 12520248 | TS. Huỳnh Ngọc Tín |
| Nguyễn Thanh Anh Tuyên | 12520492 |  |

**Đánh giá Khóa luận**

1. Về cuốn báo cáo:

Số trang Số chương

Số bảng số liệu Số hình vẽ

Số tài liệu tham khảo Sản phẩm

Một số nhận xét về hình thức cuốn báo cáo:

1. Về nội dung nghiên cứu:

1. Về chương trình ứng dụng:

1. Về thái độ làm việc của sinh viên:

**Đánh giá chung:**

**Điểm từng sinh viên:**

Trần Minh Luận**:………../10**

Nguyễn Thanh Anh Tuyên**:………../10**

|  |  |  |
| --- | --- | --- |
|  | **Người nhận xét**  (Ký tên và ghi rõ họ tên) | |
| **ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**  **TRƯỜNG ĐẠI HỌC**  **CÔNG NGHỆ THÔNG TIN** | | **CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**  **Độc Lập - Tự Do - Hạnh Phúc** | |
|  | | ***TP. HCM, ngày…..tháng…..năm……..*** | |

NHẬN XÉT KHÓA LUẬN TỐT NGHIỆP

(CỦA CÁN BỘ PHẢN BIỆN)

|  |  |  |
| --- | --- | --- |
| **Tên khóa luận:** | | |
| **XÂY DỰNG FRAMEWORK ĐÁNH GIÁ CÁC THUẬT TOÁN KHUYẾN NGHỊ VIỆC LÀM** | | |
| **Nhóm SV thực hiện:** | | **Cán bộ phản biện:** |
| Trần Minh Luận | 12520248 |  |
| Nguyễn Thanh Anh Tuyên | 12520492 |  |

**Đánh giá Khóa luận**

1. Về cuốn báo cáo:

Số trang Số chương

Số bảng số liệu Số hình vẽ

Số tài liệu tham khảo Sản phẩm

Một số nhận xét về hình thức cuốn báo cáo:

1. Về nội dung nghiên cứu:

1. Về chương trình ứng dụng:

1. Về thái độ làm việc của sinh viên:

**Đánh giá chung:**

**Điểm từng sinh viên:**

Trần Minh Luận**:………../10**

Nguyễn Thanh Anh Tuyên**:………../10**

|  |  |
| --- | --- |
|  | **Người nhận xét**  (Ký tên và ghi rõ họ tên) |

LỜI CẢM ƠN

|  |  |
| --- | --- |
| **ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**  **TRƯỜNG ĐẠI HỌC**  **CÔNG NGHỆ THÔNG TIN** | **CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**  **Độc Lập - Tự Do - Hạnh Phúc** |

ĐỀ CƯƠNG CHI TIẾT

|  |  |
| --- | --- |
| **TÊN ĐỀ TÀI: XÂY DỰNG FRAMEWORK ĐÁNH GIÁ CÁC PHƯƠNG PHÁP KHUYẾN NGHỊ VIỆC LÀM.** | |
| **Cán bộ hướng dẫn:** TS. Huỳnh Ngọc Tín | |
| **Thời gian thực hiện:**Từ ngày 10/06/2016 đến ngày 16/01/2017 | |
| **Sinh viên thực hiện:**  Trần Minh Luận - 12520248  Nguyễn Thanh Anh Tuyên - 12520492 | |
| **Nội dung đề tài:***(Mô tả chi tiết mục tiêu, phạm vi, đối tượng, phương pháp thực hiện, kết quả mong đợi của đề tài)*   * **Mục tiêu đề tài:** * Xây dựng một framework sử dụng các độ đo đánh giá phổ biến để đánh giá độ chính xác của các phương pháp khuyến nghị việc làm. * **Phạm vi đề tài:** * Hệ thống sử dụng thư viện Mahout của Apache để xây dựng các thuật toán khuyến nghị gồm: lọc cộng tác (Collaborative filtering), lọc nội dung (Content-based) và phương pháp lai tuyến tính giữa lọc cộng tác và lọc nội dung (Simple hybrid). * Nguồn dữ liệu việc làm được thu thập tự động từ các website đăng tuyển trực tuyến: vietnamworks.com, careerlink.vn, itviec.com * Nguồn dữ liệu gán nhãn phục vụ cho việc đánh giá các thuật toán khuyến nghị được thu thập từ việc khảo sát người dùng. * Hệ thống sẽ cài đặt các độ đo đánh giá chuẩn phổ biến như: Precision, Recall, F-Measure, NDCG, RMSE, MAE, MRR. * **Đối tượng nghiên cứu:** * Các thuật toán khuyến nghị: lọc cộng tác, lọc nội dung, phương pháp lai. * Các độ đo đánh giá thuật toán: Precision, Recall, F-Measure, NDCG, RMSE, MAE, MRR. * **Phương pháp thực hiện gồm các bước chính:** * Thu thập dữ liệu việc làm và tiến hành khảo sát mức độ phù hợp của việc làm đối với người dùng * Xây dựng thuật toán khuyến nghị việc làm * Xây dựng các độ đo đánh giá các thuật toán khuyến nghị đã cài đặt * Tiến hành thực nghiệm, phân tích, giải thích các kết quả thực nghiệm. * **Kết quả mong đợi của đề tài:** * Hoàn thành framework phát triển trên nền web cho phép người dùng sử dụng chạy thực nghiệm đánh giá các thuật toán khuyến nghị việc làm. * Có bộ dữ liệu gán nhãn để chạy thực nghiệm * Báo cáo đề tài   **Kế hoạch thực hiện:***(Mô tả kế hoạch làm việc và phân công công việc cho từng sinh viên tham gia)*   * **Kế hoạch làm việc:** * Viết thuyết minh đề tài (13/10/2016 – 14/10/2016) * Khảo sát hiện trạng các nghiên cứu, các hệ thống liên quan. (6/2015 – 7/2015) * Xây dựng tập dữ liệu gán nhãn, phân tích đánh giá dữ liệu gán nhãn. (7/2015 – 6/2016) * Xây dựng kiến trúc framework. (6/2016 – 6/2016) * Tìm hiểu và cài đặt các thuật toán khuyến nghị. (6/2016 – 10/2016) * Tìm hiểu và cài đặt các độ đo đánh giá. (6/2016 – 10/2016) * Chạy thực nghiệm, phân tích, giải thích kết quả thực nghiệm. (10/2016 – 11/2016) * Viết báo cáo. (10/2016 – 12/2016) | |
| * **Phân công công việc:**  |  |  | | --- | --- | | Luận | Tuyên | | Viết thuyết minh đề tài | | | Khảo sát hiện trạng các nghiên cứu, các hệ thống liên quan |  | | Xây dựng tập dữ liệu gán nhãn, phân tích đánh giá dữ liệu gán nhãn | | | Xây dựng kiến trúc framework | | | Tìm hiểu và cài đặt thuật toán Content-based, Hybrid | Tìm hiểu và cài đặt thuật toán Collaborative filtering | |  | Tìm hiểu và cài đặt các độ đo đánh giá | | Chạy thực nghiệm, phân tích, giải thích kết quả thực nghiệm | | | Viết báo cáo | | | |
| **Xác nhận của CBHD**  (Ký tên và ghi rõ họ tên) | **TP. HCM, ngày….tháng …..năm…..**  **Sinh viên**  (Ký tên và ghi rõ họ tên) |

MỤC LỤC

[Chương 1. GIỚI THIỆU 3](#_Toc465370405)

[1.1. Dẫn nhập 3](#_Toc465370406)

[1.2. Mục tiêu đề tài 3](#_Toc465370407)

[1.3. Đối tượng và phạm vi nghiên cứu 4](#_Toc465370408)

[1.4. Bố cục của luận văn 4](#_Toc465370409)

[Chương 2. TỔNG QUAN VỀ HỆ KHUYẾN NGHỊ VÀ CÁC PHƯƠNG PHÁP TIẾP CẬN 5](#_Toc465370410)

[2.1. Dẫn nhập 5](#_Toc465370411)

[2.2. Khái niệm hệ khuyến nghị 5](#_Toc465370412)

[2.3. Phát biểu bài toán khuyến nghị 5](#_Toc465370413)

[2.4. Các phương pháp tiếp cận cơ bản 7](#_Toc465370414)

[2.4.1. Tiếp cận lọc cộng tác (Collaborative filtering approach) 7](#_Toc465370415)

[2.4.1.1. Lọc dựa trên người dùng 8](#_Toc465370416)

[2.4.1.2. Lọc dựa trên đối tượng khuyến nghị 12](#_Toc465370417)

[2.4.2. Tiếp cận nội dung (Content based approach) 13](#_Toc465370418)

[2.4.2.1. Giới thiệu 13](#_Toc465370419)

[2.4.2.2. Phương pháp tính độ tương tự 13](#_Toc465370420)

[2.4.2.3. Phương pháp xây dựng vector đặc trưng 14](#_Toc465370421)

[2.4.2.4. Phương pháp khuyến nghị 15](#_Toc465370422)

[2.4.3. Simple Hybrid approach (Tiếp cận lai đơn giản) 15](#_Toc465370423)

[2.4.4. So sánh các phương pháp khuyến nghị 16](#_Toc465370424)

[2.5. Các phương pháp đánh giá hệ khuyến nghị 16](#_Toc465370425)

[2.5.1. Các phương pháp thiết lập thực nghiệm 17](#_Toc465370426)

[2.5.2. Các độ đo đánh giá 19](#_Toc465370427)

[2.5.2.1. Độ đo Precision (P) 19](#_Toc465370428)

[2.5.2.2. Độ đo Recall (R) 20](#_Toc465370429)

[2.5.2.3. Độ đo F-Measure 21](#_Toc465370430)

[2.5.2.4. Độ đo Root Mean Square Error (RMSE) 21](#_Toc465370431)

[2.5.2.5. Độ đo Normalized Discounted Cumulative Gain (NDCG) 22](#_Toc465370432)

[2.5.2.6. Độ đo Mean Average Precision (MAP) 23](#_Toc465370433)

[2.5.2.7. Độ đo Mean Reciprocal Rank (MRR) 24](#_Toc465370434)

[2.6. Khó khăn và thách thức 24](#_Toc465370435)

[2.7. Kết chương 24](#_Toc465370436)

[Chương 3. PHÂN TÍCH THIẾT KẾ HỆ THỐNG 25](#_Toc465370437)

[3.1. Dẫn nhập 25](#_Toc465370438)

[3.2. Xác định yêu cầu 25](#_Toc465370439)

[3.2.1. Danh sách các yêu cầu: 25](#_Toc465370440)

[3.2.2. Đặc tả yêu cầu upload dataset 25](#_Toc465370441)

[3.2.3. Đặc tả yêu cầu chạy thực nghiệm thuật toán khuyến nghị 26](#_Toc465370442)

[3.2.4. Đặc tả chức năng chạy đánh giá thuật toán khuyến nghị 26](#_Toc465370443)

[3.3. Phân tích yêu cầu 26](#_Toc465370444)

[3.3.1. Biểu đồ phân rã chức năng hệ thống 26](#_Toc465370445)

[3.3.2. Sơ đồ tổng thể hệ thống 27](#_Toc465370446)

[3.3.3. Mô tả xử lý upload dataset 27](#_Toc465370447)

[3.3.4. Mô tả xử lý chạy các thuật toán khuyến nghị 28](#_Toc465370448)

[3.3.5. Mô tả xử lý chạy đánh giá các thuật toán khuyến nghị 28](#_Toc465370449)

[3.4. Kiến trúc hệ thống 30](#_Toc465370450)

[3.4.1. Kiến trúc tổng quát 30](#_Toc465370451)

[3.4.2. Các thành phần trong kiến trúc 30](#_Toc465370452)

[3.5. Thiết kế dữ liệu 31](#_Toc465370453)

[3.5.1. System database 31](#_Toc465370454)

[3.5.1.1. Bảng USER 31](#_Toc465370455)

[3.5.1.2. Bảng Task 32](#_Toc465370456)

[3.5.2. User Dataset 33](#_Toc465370457)

[3.6. Thiết kế giao diện 33](#_Toc465370458)

[3.6.1. Giao diện đăng nhập 33](#_Toc465370459)

[3.6.2. Giao diện upload dataset 35](#_Toc465370460)

[3.6.3. Giao diện đăng ký tài khoản 38](#_Toc465370461)

[3.6.4. Giao diện chạy thực thực nghiệm 39](#_Toc465370462)

[3.6.5. Giao diện xem kết quả thực nghiệm 41](#_Toc465370463)

[3.7. Kết chương 43](#_Toc465370464)

[Chương 4. THIẾT LẬP THỰC NGHIỆM 44](#_Toc465370465)

[4.1. Dẫn nhập 44](#_Toc465370466)

[4.2. Môi trường thực nghiệm 44](#_Toc465370467)

[4.3. Dữ liệu thực nghiệm 44](#_Toc465370468)

[4.3.1. Thống kê về bộ dữ liệu (dataset) 44](#_Toc465370469)

[4.3.2. Nhận xét về bộ dữ liệu 48](#_Toc465370470)

[4.4. Cách tiến hành thực nghiệm 50](#_Toc465370471)

[4.5. Kết quả thực nghiệm 51](#_Toc465370472)

[4.6. Nhận định về kết quả thực nghiệm 52](#_Toc465370473)

[4.7. Kết chương 52](#_Toc465370474)

[Chương 5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 53](#_Toc465370475)

[5.1. Các kết quả đạt được 53](#_Toc465370476)

[5.2. Giá trị thực tiễn của khóa luận 53](#_Toc465370477)

[5.3. Hướng phát triển 53](#_Toc465370478)

DANH MỤC HÌNH VẼ

[Hình 1.1: Tên hình 1 3](#_Toc367742554)

DANH MỤC BẢNG

[Bảng 1.1: Tên bảng 1 3](#_Toc367742567)

[Bảng 2.1: Tên bảng 1 4](#_Toc367742568)

DANH MỤC TỪ VIẾT TẮT

TÓM TẮT KHÓA LUẬN

MỞ ĐẦU

# GIỚI THIỆU

## Dẫn nhập

Tìm kiếm việc làm là một nhu cầu của tất cả mọi người. Với sự trợ giúp của các công cụ đăng tuyển trực tuyến nên các ứng viên có thể tìm kiếm các thông tin ứng tuyển nhanh và dễ dàng hơn. Tuy vậy, các công cụ này thường chứa rất nhiều thông tin việc dẫn đến quá tải. Theo thống kê ngày 22/11/2016 trang tuyển dụng Vietnamworks hiện có 8,116 đăng tuyển [1], trang careerbuider hiện có 8,895 đăng tuyển [2],... với đăng tuyển nhiều như vậy các hệ thống tìm kiếm thông thường sẽ cho ra rất nhiều kết quả khác nhau và gây ra nhiễu cho người dùng. Vì vậy xây dựng một hệ thống để thay thế hoặc hỗ trợ các hệ thống tìm kiếm thông thường sẽ mang lại nhiều lợi ích hơn cho ứng viên và nhà tuyển dụng.

Xây dựng hệ thống khuyến nghị [3] việc làm có thể giúp cho người tìm việc giảm được thời gian tìm kiếm và đem lại hiệu quả cao hơn các hệ thống tìm kiếm thông thường. Đây là đề tài được nhiều khoa học gia trên thế giới quan tâm và có một số website đăng tuyển đã đưa hệ thống khuyến nghị vào sử dụng như Linkedin, Indeed,… và ở Việt Nam cũng đã có một số hệ thống khuyến nghị tương tự. Để có thể tìm ra phương pháp khuyến nghị tốt luận văn này hướng đến xây dựng framework đánh giá các thuật toán khuyến nghị khác nhau, phục vụ cho mục đích nghiên cứu khoa học về xây dựng hệ thống khuyến nghị việc làm.

## Mục tiêu đề tài

Mục tiêu của đề tài hướng đến tìm hiểu và xây dựng một framework dựa trên các phương pháp khuyến nghị và đánh giá có sẵn phục vụ cho việc nghiên cứu hệ thống khuyến nghị. Cụ thể gồm:

* Xây dựng dataset về việc làm, hồ sơ người dùng, dữ liệu gán nhãn.
* Tìm hiểu và cài đặt các thuật toán khuyến nghị sau:
  + Lọc cộng tác.
  + Lọc nội dung.
  + Phương pháp lai đơn giản.
* Tìm hiểu và cài đặt một số độ đo:
  + Precision and recall
  + NDCG
  + RMSE
  + F-Measure
  + MAP
  + MRR
* Xây dựng framework đánh giá các thuật toán khuyến nghị.

## Đối tượng và phạm vi nghiên cứu

* Đối tượng: các phương pháp khuyến nghị và các độ đo chuẩn.
* Phạm vi: lĩnh vực tìm kiếm việc làm.

## Bố cục của luận văn

Luận văn được bố cục theo những chương sau:

Chương 1: Giới thiệu về đề tài, nội dung nghiên cứu.

Chương 2: Trình bày lý thuyết về hệ khuyến nghị, các phương pháp khuyến nghị, các độ đo chuẩn. Các khó khăn và thách thức khi xây dựng hệ thống khuyến nghị.

Chương 3: Nội dung chương này sẽ trình bày về framework được xây dựng.

Chương 4: Chương này sẽ trình bày các thực nghiệm trên tập dữ liệu do nhóm tự xây dựng, cách thức tiến hành thực nghiệm và các giải thích cho kết quả thực nghiệm.

Chương 5: Ở chương cuối luận văn sẽ trình bày tóm gọn những kết quả của nhóm cũng như hướng phát triển tiếp theo.

# TỔNG QUAN VỀ HỆ KHUYẾN NGHỊ VÀ CÁC PHƯƠNG PHÁP TIẾP CẬN

## Dẫn nhập

Dựa trên mục tiêu, nội dung và đối tượng nghiên cứu, chương này sẽ trình bày một cách tổng quát về hệ khuyến nghị, các phưong pháp tiếp cận và ưu điểm, nhược điểm của từng phương pháp tiếp cận truyền thống, cùng với đó là các phương pháp để đánh giá một thuật toán khuyến nghị.

## Khái niệm hệ khuyến nghị

Hệ khuyến nghị tiếng Anh là Recommender System hay Recommendation System, là một hệ thống con của hệ thống lọc thông tin dùng để tiên đoán giá trị đánh giá hoặc yêu thích mà một người dùng dành cho một đối tượng [4].

Theo Ricci và cộng sự trong Recommender Systems Handbook [5], hệ khuyến nghị là những công cụ phần mềm và các kỹ thuật đưa ra các gợi ý về các đối tượng có thể được sử dụng bởi người dùng. Những gợi ý được đưa ra nhằm hỗ trợ người dùng của hệ thống đưa ra các quyết định chẳng hạn nên mua gì, nên nghe nhạc gì hay nên đọc tin tức gì. Hệ khuyến nghị đã chứng tỏ giá trị ý nghĩa đối với những người dùng online, giúp họ đương đầu với việc quá tải thông tin và trở thành một trong những công cụ mạnh mẽ và phổ biến nhất trong thương mại điện tử.

Theo quan điểm của luận văn, hệ khuyến nghị là một hệ thống thông minh với các thuật toán có khả năng tiên đoán được những đối tượng người dùng có thể quan tâm, nhằm mục đích giúp người dùng nhanh chóng tiếp cận được những gì mà người dùng cần.

## Phát biểu bài toán khuyến nghị

Hiện nay có nhiều nghiên cứu liên quan đã phát biểu về bài toán khuyến nghị và các khái niệm liên quan như Jannach và cộng sự [6], Adomavicius và Tuzhilin [7], Bobadilla và cộng sự [8]. này sẽ dựa và các nghiên cứu liên quan trên để trình bày một số khái niệm liên quan và phát biểu bài toán khuyến nghị.

1. Không gian người dùng [7] [9]

Không gian người dùng là tập tất cả các người dùng được quan sát bởi hệ thống để thực hiện khuyến nghị, ký hiệu:

1. Không gian đối tượng khuyến nghị [7] [9]

Không gian đối tượng khuyến nghị là tất cả các đối tượng có khả năng sẽ được hệ thống khuyến nghị cho người dùng. Tùy vào mục tiêu của mỗi hệ thống mà đối tượng khuyến nghị sẽ là gì, chẳng hạn như phim, các bài báo, nhạc… Với hệ thống mà khóa luận đang phát triển thì các đối tượng khuyến nghị chính là những việc làm đã đăng tuyển. Ký hiệu:

1. Hàm hữu ích [9]

Hàm hữu ích là một phép ánh xạ: , dùng để xác định giá trị hữu ích của đối tượng khuyến nghị với người dùng . Với là tập các giá trị hữu ích có giá trị trong một khoảng cụ thể và được sắp xếp theo thứ tự giảm dần.

**Phát biểu bài toán khuyến nghị**

Cho trước:

* Không gian người dùng
* Không gian đối tượng khuyến nghị

Tìm hàm hữu ích , xác định giá trị hữu ích của đối tượng khuyến nghị đối với người dùng . Với mỗi người dùng hệ khuyến nghị cần trả về một danh sách các đối tượng khuyến nghị có giá trị hữu ích cao nhất được sắp xếp theo thứ tự giảm dần.

Việc làm thế nào tìm được hàm hữu ích hiện nay đã có rất nhiều các phương pháp, hướng tiếp cận khác nhau. Phần tiếp theo sẽ trình bày về các phương pháp tiến cận truyền thống phổ biến nhất.

## Các phương pháp tiếp cận cơ bản

Theo Adomavicius và Tuzhilin [7], hệ khuyến nghị thường được chia thành 3 hướng tiếp cận sau dựa trên cách thức thực hiện khuyến nghị gồm: Collaborative filtering approach (tiếp cận lọc cộng tác), content based approach (tiếp cận nội dung) và hybrid (tiếp cận lai).

### Tiếp cận lọc cộng tác (Collaborative filtering approach)

Collaborative filtering approach (tiếp cận lọc cộng tác) hay viết tắt là CF, là phương pháp tiếp cận dựa trên dữ liệu hành vi, sở thích trong quá khứ của người dùng – ma trận đánh giá, ý tưởng cơ bản của CF là nếu người dùng có cùng sở thích ở quá khứ thì cũng sẽ có cùng sở thích trong tương lai.

1. Ma trận đánh giá [9]

Cho không gian người dùng và không gian các đối tượng khuyến nghị . Ma trận kích thước chứa các giá trị đánh giá với . Những giá trị đánh giá thể hiện mức độ hữu ích của đối tượng với người dùng (hay ). Giá trị có thể là nguyên hay thực trong một khoảng cố định nào đó tùy thuộc vào bài toán cụ thể. Thông thường giá trị đánh giá trong các hệ thống ứng dụng phổ biến thường từ 1 (không hữu ích) đến 5 (rất hữu ích). Nếu một người dùng chưa thể hiện đánh giá với một đối tượng thì và cần được tính toán (dấu ? trong hình 2.1)



Hình 2.1. Dấu ? là các giá trị cần tiên đoán trong ma trận đánh giá.

Tiếp cận lọc cộng tác được xem là hướng tiếp cận phổ biến nhất và thành công nhất để xây dựng hệ khuyến nghị cho các hệ thống thương mại điện tử [9] [6]. Với rất nhiều các nghiên cứu, cài đặt và thực nghiệm đã thực hiện trước đó, CF được chia thành 2 nhóm chính gồm: CF dựa trên bộ nhớ và CF dựa trên mô hình. Trong phạm vi của đề tài, khóa luận tiến hành tìm hiểu cài đặt và thực nghiệm với các thuật toán CF dựa trên bộ nhớ.

CF dựa trên bộ nhớ dùng các kỹ thuật của thống kê để tính toán tìm người dùng hoặc đối tượng khuyến nghị tương tự từ thông tin trong ma trận đánh giá. Tiếp cận CF dựa trên bộ nhớ tìm cách ước lượng giá trị của hàm hữu ích , thể hiện cho mức độ hữu ích của đối tượng khuyến nghị đối với người dùng . Dựa trên ma trận đánh giá, CF sẽ khuyến nghị cho người dùng các đối tượng tương tự với đối tượng mà người dùng đã đánh giá trước đó (Item based) hoặc khuyến nghị các đối tượng mà những người dùng có đồng sở thích với người dùng đã đánh giá (User based).

Tiếp theo sẽ trình bày chi tiết về 2 phương pháp lọc dựa trên người dùng (User based) và lọc dựa trên đối tượng khuyến nghị (Item based).

#### Lọc dựa trên người dùng

1. Những người dùng đồng sở thích

Những người dùng đồng sở thích là những người có sở thích hay những đánh giá trong quá khứ tương tự với người dùng trên cùng những đối tượng khuyến nghị từ ma trận đánh giá. Ký hiệu là những người dùng đồng sở thích với , trong đó , , là không gian người dùng của hệ thống.

**Ý tưởng chính:** ý tưởng thực hiện của phương pháp lọc dựa trên người dùng gồm 3 bước chính: đầu tiên xác định danh sách những người dùng có cùng sở thích với người dùng hay ; Bước 2 tiến hành ước lượng giá trị hàm hữu ích của đối tượng khuyến nghị với người dùng bằng cách tổng hợp giá trị đánh giá của đối với . Bước 3 là thực hiện khuyến nghị dựa trên giá trị hàm hữu ích ước lượng được [9] [6].

**Các bước thực hiện:** các tác giả Owen và cộng sự [10] đã tiến hành cài đặt thuật toán với các bước có thể tóm gọn theo mã giả như sau:

Trong đó các phương pháp để tính độ tương tự (), làm thế nào để xác định được neighborhood và tổng hợp có trọng số giá trị đánh giá sẽ được trình bày tiếp theo.

Tính độ tương tự ()

Để tính độ tương tự giữa các người dùng, Owen và cộng sự [10] đã tiến hành cài đặt và thực nghiệm với các công thức tính độ tương tự của:

*Pearson correlation coefficient* [11]

Trong đó:

* + - N là số đối tượng khuyến nghị p mà u và cùng đánh giá
    - với là giá trị đánh giá của u cho p
    - là tổng các giá trị đánh giá của u
    - là tổng các giá trị đánh giá của
    - là tổng của bình phương các giá trị đánh giá của u
    - là tổng của bình phương các giá trị đánh giá của

***Tanimoto coefficient*** [12]

Trong đó:

* a là số đối tượng khuyến nghị u đã đánh giá
* b là số đối tượng khuyến nghị đã đánh giá
* c là số đối tượng khuyến nghị cả u và đã đánh giá

***Euclidean distance*** [13][14]

Kết quả không thuộc một khoảng cố định nên sẽ được normalized thành:

Trong đó:

* , là giá trị đánh giá của u, trên cùng 1 đối tượng khuyến nghị
* n là số đối tượng khuyến nghị mà u và cùng đánh giá
* với là tổng số đối tượng khuyến nghị

***LogLikelihood similarity*** [15]

Trong đó:

* k11 là những đối tượng khuyến nghị được u và đánh giá
* k12 là những đối tượng khuyến nghị được u đánh giá mà không đánh giá
* k21 là những đối tượng khuyến nghị được đánh giá mà u không đánh giá
* k22 là những đối tượng khuyến nghị không được cả u và đánh giá

Cách xác định TopN người dùng đồng sở thích [6] [10]

Việc xác định danh sách những người dùng đồng sở thích sẽ ảnh hưởng trực tiếp đến kết quả của hàm hữu ích, vì vậy ta cần cân nhắc để chọn danh sách TopN tốt nhất. Nếu chúng ta xem xét tất cả các người dùng đồng sở thích sẽ ảnh hưởng đến hiệu suất của việc tính toán và cũng sẽ ảnh hưởng tiêu cực đến kết quả tính toán bởi những người dùng có mức độ tương tự thấp.

Trong thực tiễn có 2 phương pháp được sử dụng phổ biến đó là lấy danh sách tất cả người dùng có độ tương tự lớn hơn một ngưỡng (threshold) nhất định hoặc lấy danh sách TopN cố định n người dùng có độ tương tự cao nhất.

Với cách thứ nhất ta có phải xem xét ngưỡng (threshold) ở mức nào là hợp lý, vì nếu chọn quá cao thì danh sách người dùng sẽ rất hạn chế (thậm chí không có), còn nếu chọn thấp quá thì danh sách này dường như không bị cắt giảm là bao so với ban đầu.

Với cách thứ 2 kích thước của tập người dùng đồng sở thích sẽ không thể bao phủ, giá trị n được chọn nếu quá cao sẽ không giới hạn được danh sách người dùng (tương tự khi chọn threshold thấp) điều này sẽ dẫn đến nhiễu dữ liệu tiên đoán, nhưng nếu chọn n quá thấp sẽ ảnh hưởng tiêu cực đến khả năng tiên đoán của hệ thống.

Phương pháp tổng hợp đánh giá

Sau khi đã tìm được danh sách người dùng có cùng sở thích và mức độ tương tự (giá trị ) ta cần tổng hợp lại để được kết quả cuối cùng của hàm hữu ích để thực hiện khuyến nghị.

Owen và cộng sự [10] đã nghiên cứu và cài đặt với công thức sau cho thư viện Mahout [16]:

Trong đó:

* + N là số người dùng có sở thích tương tự với u
  + M là số người dùng có đánh giá cho đối tượng p
  + là giá trị đánh giá của u với p trong ma trận đánh giá

#### Lọc dựa trên đối tượng khuyến nghị

Tương tự như lọc dựa trên người dùng, lọc dựa trên đối tượng khuyến nghị cũng với ý tưởng và các bước thực hiện khá giống nhau gồm 3 bước chính: đầu tiên xác định danh sách những đối tượng khuyến nghị tương tự nhất với đối tượng khuyến nghị ; Bước 2 tiến hành ước lượng giá trị hàm hữu ích của đối tượng khuyến nghị với người dùng bằng cách tổng hợp giá trị đánh giá của cho đối với . Bước 3 là thực hiện khuyến nghị dựa trên giá trị hàm hữu ích ước lượng được [9] [6].

### Tiếp cận nội dung (Content based approach)

#### Giới thiệu

Hướng tiếp cận dựa trên nội dung (Content based approach) [17] là một dạng đặc biệt của bài toán truy vấn và lọc thông tin [18]. Hướng tiếp cận này tìm cách khuyến nghị cho người dùng những đối tượng tương tự với những đối tượng mà người dùng quan tâm trong quá khứ [18]. Hay nói cách khác tiếp cận theo hướng nội dung trả lời cho câu hỏi sau của người sử dụng: “Hãy cho tôi biết những đối tượng tương tự với những đối tượng mà tôi thích” [18]. Các đối tượng ở đây có thể là các video ca nhạc, game, bài báo, việc làm, sản phẩm,… Ví dụ, một người thường xem những đăng tuyển liên quan đến những từ khóa như java, mysql, lập trình,… thì hệ thống đưa ra cho người đó những việc làm chứa những từ khóa liên quan mà người dùng chưa xem.

Để xây dựng một hệ thống khuyến nghị tiếp cận theo hướng nội dung chúng ta cần cung cấp cho hệ thống tập các thông tin sẵn có về đối tượng được khuyến nghị như thể loại, mô tả, tính chất, đặc điểm… và một tập dữ liệu mô tả sở thích của người dùng. Nhiệm vụ của hệ thống là “học” sở thích của người dùng và tìm ra những đối tượng có đặc điểm tương tự với sở thích của người dùng.

#### Phương pháp tính độ tương tự

Độ tương tự của người dùng u và đối tượng i có thể tính bằng độ tương tự Cosine [19], khoảng cách Euclide [13], hoặc sử dụng Tanimoto [20] để tính toán. Thông thường các nghiên cứu sử dụng độ tương tự Cosine để tính toán. Công thức tính Cosine được biểu diễn như sau:

Trong đó A và B là các vector đặc trưng. Có nhiều cách để biểu diễn nội dung của đối tượng được khuyến nghị như biểu diễn ở dạng cây, biểu diễn ở dạng văn có cấu trúc và biểu diễn văn bản phi cấu trúc. Thông thường người ta chọn phương pháp biểu diễn nội dung của đối tượng dưới dạng một vector đặc trưng [18]. Phương pháp xây dựng vector đặc trưng sẽ được trình bày chi tiết ở mục tiếp theo.

#### Phương pháp xây dựng vector đặc trưng

Các vector đặc trưng có thể được xây dựng bằng nhiều cách khác nhau. Thông thường để xây dựng vector đặc trưng các văn bản phi cấu trúc sẽ được tách thành *n* cụm từ. Sau đó biểu diễn các cụm từ thành một vector với số chiều là *n.* [18] Giá trị của mỗi chiều sẽ bằng số lần mà cụm từ đó xuất hiện trong văn bản gốc. Tuy nhiên theo Dietmar Jannach thì việc biểu diễn bằng cách trên có một vài vấn đề cần lưu ý[18]:

* + Không phải tất cả các cụm từ có độ quan trọng giống như nhau.
  + Các văn bản dài dẫn đến tăng cơ hội trùng lắp với hồ sơ của người dùng.

Để khắc phục tình trạng trên, theo ông Dietmar Jannach [18] và một số tài liệu khác [21] [5] thì không nên dùng trực tiếp tần xuất mà cụm từ xuất hiện trong tài liệu được mô hình hóa mà nên dùng giá trị TF-IDF thay thế. Việc tính giá trị TF-IDF có thể sử dụng công thức sau:

**-**

Trong đó TF(i,j) chính là số lần cụm từ *i* xuất hiện trong văn bản *j,* IDF(i) chính là giá trị đo độ phổ biến của cụm từ *i* trong tất cả các văn bản. IDF(i) được tính theo công thức:

Trong đó *N* là số văn bản thô cần mô hình hóa. *n(i)* là số văn bản chứ cụm từ *i*.

#### Phương pháp khuyến nghị

Các bước để xây dựng hệ thống khuyến nghị dựa trên nội dung như sau:

* Bước 1: Biểu diễn nội dung của những đối tượng được khuyến nghị *i* dưới dạng một vector đặc trưng(feature vector) [22].
* Bước 2: Xây dựng và biểu diễn sở thích của người dùng *u* dưới dạng một vector đặc trưng.
* Bước 3: Tính toán độ tương tự cosine [19] của các đối tượng *i* thuộc tập đối đượng được khuyến nghị đối với người dùng *u*.
* Bước 4: Đưa ra *top N* những đối tượng có độ tương tự cao nhất.

Mã giả cho thuật toán:

Input: UserSet, JobSet, RatingSet

foreach(User in UserSet){

User = ModelingUserProfile(User)

}

foreach(Job in JobSet){

Job = ModelingItem(Job)

}

foreach(Job in JobSet){

JobVector = GetFeaturedVector(Job);

foreach(User in UserSet){

UserVector=GetFeaturedVector(User,GetUserRating(User, RatingSet));

Weight = Cosine(JobVector, UserVector);

If Weight is in TopN Then

AddToTopN(User,Job, Weight)

}

}

### Simple Hybrid approach (Tiếp cận lai đơn giản)

Có rất nhiều phương pháp khuyến nghị khác nhau. Mỗi phương pháp có một điểm mạnh và điểm yếu riêng. Phương pháp lai ra đời nhằm tận dụng các điểm mạnh của các phương pháp khuyến nghị khác nhau và hạn chế các điểm yếu [9]. Phương pháp lai kết hợp kết quả khuyến nghị của các phương pháp khuyến nghị khác theo một cách nào đó nhằm đưa ra một hệ thống khuyến nghị tốt hơn [18]. Có nhiều các lai khác nhau. Cách tiếp cận lai đơn giản nhất là sử dụng phương pháp lai có trọng số(Weighted) [18] [17]. Mỗi kết quả đầu ra của các phương pháp *i* sẽ được kết hợp tuyến tính với nhau theo một hệ số *βi*. Các hệ số *βi* này sẽ dần được hiệu chỉnh cho đến khi thu được kết quả tốt nhất [9] [23].

Công thức dự đoán cho phương pháp lai có trọng số [18]



Mã giả thuật toán khuyến nghị dựa trên phương pháp lai đơn giản

Input: UserSet, JobSet, RatingSet, Alpha

CFResult = RunCF(RatingSet);

CBResult = RunCB(UserSet,JobSet, RatingSet);

foreach(Job in JobSet)

{

foreach(User in UserSet)

{

CFScore = GetScore(CFResult, User, Job);

CBScore = GetScore(CBResult, User, Job);

HBScore = CFScore \* Alpha + (1 - Alpha) \* CBScore;

If(HBScore in topN) Then

AddToTopN(User,Job, HBScore)

}

}

### So sánh các phương pháp khuyến nghị

## Các phương pháp đánh giá hệ khuyến nghị

Ở các phần trước khóa luận đã trình bày về hệ khuyến nghị, các khái niệm, phát biểu bài toán và các hướng tiếp cận. Hệ khuyến nghị có khá nhiều hướng tiếp cận và bản thân mỗi hướng tiếp cận cũng có nhiều các thuật toán khác nhau để áp dụng vì vậy nảy sinh nhu cầu cần phải đánh giá được hệ khuyến nghị nào là tốt nhất hay thuật toán nào là tốt nhất. Về khái nệm về một hệ khuyến nghị tốt còn tùy thuộc vào mục tiêu của hệ khuyến nghị là nhằm tối ưu lợi nhuận (ví dụ: với hệ thống thương mại điện tử cần hệ khuyến nghị để tăng khả năng bán hàng – tăng doanh thu) hay hướng tối ưu cho người dùng (ví dụ: hệ thống chỉ khuyến nghị những đối tượng tốt nhất tới người dùng) [9]. Ở đây khóa luận quan niệm thuật toán tốt nhất là thuật toán có thể khuyến nghị những đối tượng tốt nhất đến người dùng.

Trong phần này khóa luận sẽ trình bày về các phương pháp thiết lập thực nghiệm và các độ đo đánh giá để giúp ta xác định được đâu là thuật toán tốt nhất để sử dụng hay hướng tiếp cận nào sẽ là phù hợp cho từng bài toán cụ thể.

### Các phương pháp thiết lập thực nghiệm

Theo một khảo sát về đánh giá độ chính xác của các thuật toán khuyến nghị, các tác giả Gunawardana và Shani đã chỉ ra rằng có 2 phương pháp thiết lập thực nghiệm để đánh giá các thuật toán khuyến nghị gồm có: thiết lập đánh giá online và thiết lập đánh giá offline (gọi tắt là đánh giá online và đánh giá offline) [24].

**Thiết lập đánh giá online:** với thiết lập đánh giá online hệ thống sẽ đo lường được sự thay đổi hành vi của người dùng khi họ tương tác với hệ thống khuyến nghị. Ưu điểm của đánh giá online là có thể phản ánh đúng được hiệu quả của hệ thống khuyến nghị. Tuy nhiên tồn tại song song với ưu điểm này là nhiều nhược điểm như: để đánh giá online phải có một hệ thống được thiết lập chạy thật sự; phải xem xét nhiều yếu tố để có thể thay đổi các thuật toán để đánh giá; với hệ thống chạy thật sự thì có khả năng sẽ mang lại cho người dùng những trải nghiệm tiêu cực (vì hệ thống chưa qua đánh giá có thể sẽ không phải là hệ thống tốt); và một điều quan trọng nữa là phải tốn kém nhiều chi phí để thiết lập cho một hệ thống đánh giá online. Chính vì những nhược điểm trên, trong thực tế thường sử dụng các phương pháp đánh giá offline và khóa luận cũng tập trung vào các lý thuyết và thực nghiệm theo phương pháp đánh giá offline.

**Thiết lập đánh giá offline:** mục tiêu của đánh giá offline là để chọn ra được những thuật toán tốt nhất cho một hệ thống khuyến nghị. Khác với đánh giá online, đánh giá offline không cần một hệ thống chạy thực, đánh giá offline đưa ra giả thuyết “kết quả đánh giá offline tương đương với kết quả khi triển khai thực tế” tức là chúng ta mong đợi nếu kết quả đánh giá offline tốt thì kết quả thực tế cũng sẽ tốt. Để làm được điều này cần phải mô phỏng tốt quá trình đánh giá online – hệ thống khuyến nghị và người dùng đánh giá kết quả khuyến nghị đó. Thông thường để làm được điều này ta dựa vào dữ liệu đã đánh giá trong quá khứ của người dùng (ma trận đánh giá), ẩn giấu đi một phần dữ liệu và phần còn lại được sử dụng để huấn luyện hệ thống để hệ thống tiên đoán ra những dữ liệu đã bị giấu. Có nhiều cách để chia tập dữ liệu phần nào sẽ dùng để đánh giá, phần nào sẽ dùng để huấn luyện, tuy nhiên phổ biến nhất là phương pháp K-fold Cross-validation (tạm dịch: đánh giá chéo với k lần lặp).

**K-fold Cross-validation** [25]**:** Cross-validation là phương pháp thiết lập đánh giá kết quả thực nghiệm của các bài toán phân tích thống kê. Mục tiêu của cross-validation là để xác định cách lấy tập dữ liệu kiểm tra (GroundTruth) và tập dữ liệu huấn luyện (TrainingSet) để tránh vấn đề về “Overfiting”, giúp người dùng có thể nhận thấy sự độc lập giữa dữ liệu và mô hình huấn luyện.

K-fold Cross-validation được thiết lập như sau: với một tập dữ liệu đầu vào, hệ thống sẽ chia điều ngẫu nhiên tập dữ liệu đầu vào thành **k** phần với kích thước bằng nhau, lấy một phần làm dữ liệu kiểm tra (tập GroundTruth) và **k-1** phần còn lại được sử dụng để huấn luyện mô hình. Quá trình này lặp lại **k** lần với các tập GroundTruth khác nhau.



Hình 2.2 K-fold Cross-validation với k=4

### Các độ đo đánh giá

Theo Bobadilla và cộng sự [8], các độ đo đánh giá thường được chia thành các nhóm chính sau: nhóm các độ đo về sự chính xác của tiên đoán như: MAE (Mean Absolute Error) hay RMSE (Root Mean Square Error); nhóm các độ đo về danh sách khuyến nghị: Precision, Recall, F-Measure; nhóm các độ đo về danh sách xếp hạng các kết quả khuyến nghị: NDCG (Normalize Discounted Cumulative Gain), MAE (Mean Average Precision) hay MRR (Mean Reciprocal Rank).

#### Độ đo Precision (P)

Precision (P) là một độ đo phổ biến thường dùng trong lĩnh vực truy vấn thông tin nhằm xác định tỉ lệ kết quả trả về liên quan tới câu truy vấn [26]. Trong các hệ khuyến nghị, độ đo Precision cũng thường được sử dụng và được phát biểu: Precision là xác xuất một tiên đoán là đúng (thật sự có xảy ra) [9].

Precision (P) được xác định với công thức [9] [26] [27]:

Trong đó:

* (true positive): là số việc làm được tiên đoán đúng và được gán nhãn đúng
* (false positive): là số việc làm được tiên đoán đúng nhưng được gán nhãn sai
* (true negative): là số việc làm được tiên đoán sai nhưng được gán nhãn đúng
* (false negative): là số việc làm được tiên đoán sai và được gán nhãn sai

Để dể hình dung hơn cách xác định *true-positive, false-positive…* chúng ta có thể dựa vào bảng confusion matrix sau [26]:

|  |  |  |  |
| --- | --- | --- | --- |
| Confusion matrix | | Predict Condition | |
| Predict Positive | Predict Negative |
| True Condition | Condition Positive | True Positive | False Negative |
| Condition Negative | False Positive | True Negative |

Precision sẽ xem xét toàn bộ kết quả trả về, nhưng thông thường trong các thực nghiệm người dùng thường thiết lập Precision với TopN kết quả tốt nhất gọi là Precision TopN – ký hiệu P@n [26].

#### Độ đo Recall (R)

Tương tự như Precision, độ đo Recall (R) cũng là một độ đo được dùng trong lĩnh vực truy vấn thông tin. Recall trong lĩnh vực truy vấn thông tin là số document liên quan đến câu truy vấn mà được trả về thành công [26]. Trong hệ khuyến nghị việc làm, khóa luận phát biểu Recall là độ bao phủ hay xác suất một việc làm đúng xảy ra trong kết quả tiên đoán.

Recall (R) được xác định bởi công thức [26]:

Trong đó: các thông số đã được định nghĩa ở phần 2.5.2.1

#### Độ đo F-Measure

Precision và Recall có mối liên hệ tỉ lệ nghịch với nhau, khi Precision tăng thì Recall giảm và ngược lại, điều này gây khó khăn cho việc đánh giá một thuật toán hay một hệ khuyến nghị dựa trên 2 độ đo này cùng lúc. Do đó F-Measure được đưa ra là một độ đo kết hợp giữa Precision và Recall theo một trọng số xác định nhằm có một kết quả cân bằng giữa Precision và Recall để đánh giá một hệ khuyến nghị [26].

F-Measure được xác định bởi công thức sau [26]:

Trong đó:

* : là độ đo Precision
* : là độ đo Recall
* : là trọng số

Thông thường các nghiên cứu thực nghiệm thường sử dụng F-Measure với trọng số hay còn gọi là với công thức :

#### Độ đo Root Mean Square Error (RMSE)

Root Mean Square Error (RMSE) hay Root Mean Square Deviation (RMSD) hay lỗi bình phương trung bình thường được dùng để đo sự khác biệt giữa các giá trị tiên đoán và giá trị đánh giá thực tế [28].

Công thức RMSE [9]:

Trong đó:

* : là giá trị đánh giá của người dùng trên việc làm hệ thống tiên đoán được.
* : là giá trị đánh giá thật sự (đúng) của người dùng trên việc làm .
* : tập các đánh giá của người dùng trên việc làm cần tiên đoán.
* : kích thước tập

RMSE có nhược điểm là chỉ nên sử dụng khi ta không cần phân biệt giữa các lỗi đánh giá. Chẳng hạn, giá trị đánh giá thật sự của người dùng trên việc làm là 2, nhưng qua 2 phương pháp khác nhau cho ra 2 kết quả là 1 và 3. Khi đó độ lệch RMSE sẽ giống nhau, nhưng thật ra ý nghĩa của khuyến nghị lại khác nhau. Vì phương pháp cho ra tiên đoán 3 sẽ được ưu tiên khuyến nghị hơn là phương pháp tiên đoán ra 1.

#### Độ đo Normalized Discounted Cumulative Gain (NDCG)

DCG (Discounted Cumulative Gain) [29] [27]: là độ đo chất lượng xếp hạng của danh sách khuyến nghị. DCG đo tính hữu ích của 1 việc làm được khuyến nghị dựa trên vị trí của nó trong danh sách khuyến nghị, giá trị hữu ích được tích lũy từ đầu đến cuối danh sách xếp hạng, giá trị hữu ích sẽ bị giảm đi với những việc làm có thứ hạng thấp.

Công thức DCG [29]:

Trong đó:

* = 1: nếu việc làm thứ được khuyến nghị là có liên quan
* = 0: nếu việc làm thứ được khuyến nghị là không liên quan
* : là các việc làm được khuyến nghị

Với công thức tính trên DCG không thuộc một khoảng cố định sẽ khó khăn cho việc so sánh giữa các kết quả thực nghiệm, vì vậy DCG được chuẩn hóa thành NDCG (Normalized DCG) với công thức [29]:

Trong đó: : là giá trị cho danh sách tiên đoán lý tưởng gồm các việc làm liên quan đã sắp xếp theo mức độ liên quan.

#### Độ đo Mean Average Precision (MAP)

MAP là trung bình của các giá trị Average Precision (AP) trên tất cả các người dùng được khuyến nghị.

AP là giá trị trung bình của Precision tại từng vị trí của các việc làm trong danh sách xếp hạng các việc làm được khuyến nghị.

Công thức AP [30]:

Trong đó:

* : là số items được khuyến nghị cho 1 user
* : nếu item thứ k là có liên quan
* : nếu item thứ k là không liên quan
* : là precision

Công thức MAP [30]:

Trong đó:

* : là tổng số user được khuyến nghị
* : là average precision cho user thứ .

#### Độ đo Mean Reciprocal Rank (MRR)

Reciprocal rank (RR) là độ đo xem xét vị trí xếp hạng của việc làm liên quan đầu tiên được trả về. MRR là trung bình của RR thông qua tất cả user trong danh sách khuyến nghị [9].

Công thức [9]:

Trong đó:

* : là tổng số user được khuyến nghị
* : là vị trí xuất hiện lần đầu tiên của job liên quan trong danh sách khuyến nghị trả về.

## Khó khăn và thách thức

## Kết chương

Trong chương này khóa luận đã trình tổng thể về các hướng tiếp cận cho hệ khuyến nghị cũng như chi tiết về các thuật toán khuyến nghị mà khóa luận đã tiến hành cài đặt cho thực nghiệm, đồng thời khóa luận cũng đã trình bày chi tiết về các phương pháp đánh giá hệ khuyến nghị và các độ đo cụ thể đã được cài đặt. Trong chương tiếp theo khóa luận sẽ đi chi tiết về quá trình phân tích thiết kế và cài đặt hệ thống dựa trên kiến thức đã trình bày ở chương này.

# PHÂN TÍCH THIẾT KẾ HỆ THỐNG

## Dẫn nhập

Trong chương trước, chúng em đã trình bày một số vấn đề về các phương pháp tiếp cận và đánh giá hệ thống khuyến nghị. Chương này, chúng em sẽ trình bày chi tiết về framework mà chúng em xây dựng bao gồm kiến trúc, công nghệ, nền tảng.

## Xác định yêu cầu

### Danh sách các yêu cầu:

Bảng 1: Danh sách yêu cầu

|  |  |  |
| --- | --- | --- |
| Mã Yêu cầu | Tên yêu cầu | Ghi chú |
| 1.1 | Upload dataset |  |
| 1.2 | Chạy các thuật toán khuyến nghị |  |
| 1.3 | Chạy đánh giá các thuật toán khuyến nghị |  |

### Đặc tả yêu cầu upload dataset

Để bắt đầu sử dụng hệ framework. Người dùng phải cung dataset cho hệ thống. Dataset được upload bao gồm 03 file.

* File Cv.txt: Chứa dữ liệu về người dùng bao gồm các thông tin về cv người dùng.
* File Job.txt: Chứa dữ liệu về việc làm sẽ được khuyến nghị.
* Score.txt: Chứa dữ liệu về sở thích của người dùng trong file Cv.txt đối với những việc làm trong file Job.txt

Sau khi chọn file người dùng chọn chức năng upload để bắt đầu upload dữ liệu. Hệ thống sẽ lưu trữ dataset vào thư mục có dạng: mã người dùng/tên dataset và thông báo kết quả thực hiện cho người dùng.

### Đặc tả yêu cầu chạy thực nghiệm thuật toán khuyến nghị

Người dùng tạo chọn dataset cần thực nghiệm, chọn thuật toán cần chạy. Nếu tập dữ liệu chưa có thì người dùng có thể upload lên hệ thống. Sau đó người dùng bấm chạy. Hệ thống sẽ thực hiện chạy thuật toán khuyến nghị. Sau khi chạy xong. Hệ thống sẽ thông báo cho người dùng kết quả thực nghiệm và lưu trữ lại vào file output.txt.

Cấu trúc file output gồm 3 cột: UserId, JobId, Score. Các cột ngăn cách nhau bằng dấu kí tự tab.

### Đặc tả chức năng chạy đánh giá thuật toán khuyến nghị

Người dùng sẽ nhập những file output.txt sinh ra khi chạy thực nghiệm vào hệ thống. Hệ thống sẽ thực hiện tính toán và trả về kết quả kết quả so sánh các thuật toán khuyến nghị cho người dùng.

## Phân tích yêu cầu

### Biểu đồ phân rã chức năng hệ thống

Recomender evaluation system

Upload dataset

Run algorithms

Evaluate algorithms

### Sơ đồ tổng thể hệ thống



Hình 1: sơ đồ DFD tổng thể hệ thống

### Mô tả xử lý upload dataset

* Sơ đồ DFD của xử lý



Hình 2: Sơ đồ DFD upload dataset

* Mô tả xử lý: người dùng đưa dataset gồm 3 file job.txt, Cv.txt và Score.txt vào hệ thống. Hệ thống kiểm tra và lưu dataset lại.
* Các biểu mẫu liên quan:
  + File Job.txt: Định dạng file text, lưu trữ được kí tự unicode, các thuộc tính khác nhau được ngăn cách bằng dấu tab. Gồm các thuộc tính JobId,JobName, Category, JobRequirement, JobDescription, Tag, JobBenefit, Salary.
  + File Cv.txt: Định dạng file text, lưu trữ được kí tự unicode, các thuộc tính khác nhau được ngăn cách bằng dấu tab. Gồm các thuộc tính CvId, UserId, UserName, Education, Interested, ForeignLangue, ExpectedSalary.
  + File Score.txt: Định dạng file text, các thuộc tính khác nhau được ngăn cách bằng dấu tab. Gồm ba thuộc tính JobId, UserId, Score

### Mô tả xử lý chạy các thuật toán khuyến nghị

* Sơ đồ DFD của xử lý:



Hình 3: Sơ đồ DFD chạy thuật toán khuyến nghị

* Mô tả xử lý: Người dùng chọn thuật toán cần chạy thực nghiệm và tên của dataset. Hệ thống sẽ lấy dataset tương ứng từ kho User Dataset và chạy thực nghiệm dựa thuật toán mà người dùng chọn.
* Các biểu mẫu liên quan: Không có.

### Mô tả xử lý chạy đánh giá các thuật toán khuyến nghị

* Sơ đồ DFD của xử lý



Hình 4: DFD chạy đánh giá thuật toán khuyến nghị.

* Mô tả xử lý: Người dùng đưa vào thuật toán cần chạy và tên dataset thực nghiệm, sau đó chọn cách thức chia tỉ lệ train/test. Cuối cùng người dùng bấm xử lý để bắt đầu chạy thực nghiệm. Sau khi chạy xong, cập nhật kết quả vào database.
* Các biểu mẫu liên quan: Không có.

## Kiến trúc hệ thống

### Kiến trúc tổng quát

**SYSTEM DB**

**User DataSet**

**Spring MVC**

**Apache Mahout**

**Apache Lucene**

**RecAlg****EvaluationSystem**

**Dataset Analyzation**

**Runing & evaluation**

**System Interfaces**

**System core modules**

**System DB**

**Libraries**

### Các thành phần trong kiến trúc

* + System interfaces
* RecAlgComparisonSystem: Module này cung cấp cho người dùng chức năng quản lý dataset, xem thống kê dataset, quản lý các tiến trình thực thi các thuật toán khuyến nghị, xem kết quả và các đánh giá kết quả khuyến nghị.
  + System core modules
* Dataset analyzation: Đây là mô-đun phân tích dataset của người dùng.
* Experimental conduction & evaluation: chạy thực nghiệm các thuật toán khuyến nghị và đánh giá kết quả thực hiện.
  + Database
* System DB: database dùng để tổ chức lưu trữ thông tin người dùng, các tác vụ thực hiện của người dùng và các thông tin thiết lập chạy thuật toán.
* User dataset: hệ thống lưu trữ files dataset do người dùng upload cùng với đó là các file kết quả khuyến nghị, kết quả so sánh và kết quả thống kê.
  + Libraries
* Spring MVC: dùng hỗ trợ xây dựng website tương tác người dùng.
* Mahout: sử dụng các thư viện recommend có sẵn trong Mahout để thực hiện khuyến nghị.
* LUCENE: Dùng để tính toán tf-idf.

## Thiết kế dữ liệu

### System database

#### Bảng USER

* + USER (UserId, UserName, Email, Password)

Bảng 2: Mô tả table user

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên cột | Mô tả | Kiểu dữ liệu | Miền giá trị | Ghi chú |
| 1 | UserId | Mã người dùng | Integer | 11 | Not null, primary key |
| 2 | UserName | Tên người dùng | Nvarchar | 50 | Not null |
| 3 | Email | Email đăng nhập người dùng | Varchar | 50 | Not null |
| 4 | password | Mật khẩu đăng nhập | Varchar | 50 | Not null |

#### Bảng Task

* + Task (TaskId, UserId, TaskName, TimeCreate, Status, Algorithm, InputFolder, OutputFile)

Bảng 3: Mô tả table task

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên cột | Mô tả | Kiểu dữ liệu | Miền giá trị | Ghi chú |
| 1 | TaskId | Mã của task | Integer | 11 | Not null, primary key |
| 2 | UserId | Mã người dùng | Integer | 11 | Not null, foreign key đến User |
| 3 | TaskName | Tên task | Nvarchar | 100 | Not null |
| 4 | TimeCreate | Thời gian tạo task | Datetime |  | Not null |
| 5 | Status | Trạng thái task | Nvarchar | 10 | Not null |
| 6 | Algorithm | Tên thuật toán | Varchar | 20 | Not null |
| 7 | InputFolder | Đường dẫn đến thư mục chứa file input | Varchar | 100 | Not null |
| 8 | OutputFile | Đường dẫn file output | Varchar | 100 | Not null |

### User Dataset

* + Dữ liệu đầu vào để chạy các thuật toán gồm 3 file Job.txt, CV.txt và Score.txt. Quy ước trong các file .txt mỗi cột cách nhau bằng một ký tự tab.
    - File Job.txt gồm các cột sau: JobId, JobName, Location, Salary, Category, Requirement, Tag, Description.
    - File CV.txt gồm các cột sau: UserId, CVId, UserName, CVName, UserAddress, ExpectedSalary, Category, Language, Education, Skill, CareerObjective.
    - File Score.txt gồm các cột sau: UserId, JobId, Score.
  + Dữ liệu đầu ra
    - File output.txt có cấu trúc gồm 3 cột: UserId, JobId, Score.

## Thiết kế giao diện

### Giao diện đăng nhập

* + Giao diện:



Hình 5: Giao diện đăng nhập hệ thống

* + Mô tả giao diện:

Bảng 4: Bảng mô tả giao diện đăng nhập

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Chức năng |
| 1 | Email | HTML Form Input(text) | Chứa địa chỉ email đăng nhập. |
| 2 | Mật khẩu | HTML form Input(password) | Chứa password của người dùng. |
| 3 | Đăng nhập | Html form Input(Đăng nhập) | Gửi dữ liệu lên trang xử lý |
| 4 | Lấy lại mật khẩu | HTML link | Chuyển sang trang lấy lại mật khẩu. |
| 5 | Đăng ký | HTML link | Chuyển sang trang đăng ký |

* + Mô tả biến cố và xử lý:

Bảng 5: Bảng mô tả biến cố và xử lý trên giao diện đăng nhập

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Biến cố | Xử lý | Mã số xử lý |
| 1 | Người dùng Bấm chọn nút đăng nhập | Kiểm tra email và mật khẩu đã được đăng ký hay chưa. Nếu chưa tồn tại trong hệ thống thì thông báo và yêu cầu nhập lại. Nếu đã tồn tại thì chuyển sang màn hình trang chủ. | XL1 |
| 2 | Người dùng bấm link lấy lại mật khẩu | Chuyển sang trang lấy lại mật khẩu | XL2 |
| 3 | Người dùng bấm link đăng ký | Chuyển sang trang đăng ký | XL3 |
| 4 | Load trang | Kiểm tra trạng thái đăng nhập của user. Nếu user đã đăng nhập thì chuyển về trang chủ. Ngược lại thì hiển thị trang đăng nhập | XL4 |

### Giao diện upload dataset

* Giao diện mẫu



Hình 6: Giao diện upload dataser

* Mô tả giao diện

Bảng 6: Bảng mô tả giao diện upload dataset

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Chức năng |
| 1 | Danh sách dataset | HTML TABLE | Thông tin danh sách các dataset đã upload |
| 2 | Xem thống kê | HTML link | Hiển thị đường dẫn tới trang thống kê dataset |
| 3 | Tên dataset | HTML form input(text) | Chứa tên dataset do người dùng nhập |
| 4 | Chọn File score.txt | HTML form upload field | Chứa đường dẫn tới file score.txt do người dùng chọn |
| 5 | Chọn file Cv.txt | HTML form upload field | Chứa đường dẫn đến file cv.txt do người dùng chọn |
| 6 | Chọn file Job.txt | HTML form upload | Chứa đường dẫn đến file job.txt do người dùng chọn |
| 7 | Upload | HTML form input(submit) | Nút dùng để kích hoạt chức năng upload. |

* Mô tả biến cố

Bảng 7: Bảng mô tả biến cố và xử lý cho giao diện upload dataset

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Biến cố | Xử lý | Mã xử lý |
| 1 | Load trang | * Lấy danh sách dataset mà người dùng đã upload hiển thị lên bảng 1. * Kiểm tra email và mật khẩu đã được đăng ký hay chưa. Nếu chưa tồn tại trong hệ thống thì thông báo và yêu cầu nhập lại. Nếu đã tồn tại thì chuyển sang màn hình trang chủ. | XL5 |
| 2 | Người dùng bấm xem thống kê | Chuyển đến trang thống kê dataset của dataset tương ứng | XL6 |
| 3 | Người dùng bấm upload | Kiểm tra thông tin người dùng đã nhập. Lưu trữ lại dataset và thông báo kết quả thực hiện cho người dùng. | XL7 |

### Giao diện đăng ký tài khoản

* Giao diện mẫu



Hình 7: Giao diện đăng ký tài khoản

* Mô tả giao diện

Bảng 8: Mô tả giao diện đăng ký tài khoản

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Mô tả |
| 1 | Tên tài khoản | HTML form input(text) | Chứa tên tài khoản do người dùng nhập |
| 2 | Email | HTML form input(text) | Chứa email do người dùng nhập |
| 3 | Mật khẩu | HTML form input(password) | Chứa mật khẩu do người dùng nhập |
| 4 | Nhập lại mật khẩu | HTML form input(password) | Chứa mật khẩu do người dùng nhập |
| 5 | Đăng ký | HTML form input(submit) | Kích hoạt chức năng đăng ký tài khoản |
| 6 | Đăng nhập | HTML link | Quay về trang đăng nhập |

* Mô tả biến cố

Bảng 9: Bảng mô tả biến cố và xử lý trên giao diện đăng nhập

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Biến cố | Xử lý | Mã xử lý |
| 1 | Load trang | Kiểm tra cookie trên máy người dùng. Nếu người dùng đã đăng nhập thì yêu cầu đăng xuất trước khi tiếp tục thực hiện | XL8 |
| 2 | Người dùng chọn đăng ký tài khoản | Kiểm tra các thông tin mà người dùng nhập vào hệ thống. Nếu còn thiếu hoặc sai thì yêu cầu người dùng nhập lại. Nếu thông tin hợp lệ thì chuyển thông tin về trang xử lý. | XL9 |
| 3 | Người dùng chọn đăng nhập | Quay trở về trang đăng nhập hệ thống | XL10 |

### Giao diện chạy thực thực nghiệm

* Giao diện mẫu:



Hình 8: Giao diện chạy thực nghiệm

* Mô tả giao diện:

Bảng 10: Bảng mô tả giao diện chạy thực nghiệm

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Mô tả |
| 1 | Danh sách task | HTML table | Hiển thị danh sách các tác vụ đã được tạo và trạng thái. |
| 2 | Task | HTML link | Hiển thị liên kết đến trang kết quả thực nghiệm. |
| 3 | Tên task | HTML form input(text) | Chứa tên task người dùng nhập |
| 4 | Chọn thuật toán | HTML select | Dùng để lựa chọn thuật toán cần thực nghiệm |
| 5 | Chọn dataset | HTML select | Dùng để lựa chọn dataset cần thực nghiệm. |

* Các biến cố và xử lý:

Bảng 11: Biến cố và xử lý trên giao diện chạy thực nghiệm

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Biến cố | Xử lý | Mã xử lý |
| 1 | Load trang | * Kiểm tra trạng thái đăng nhập của user. Nếu chưa đăng nhập thì yêu cầu đăng nhập. * Load danh sách các tác vụ mà người dùng đang đăng nhập hệ thống đã tạo. Tô màu xanh với các tác vụ đã chạy thành công. Tô màu vàng với các tác vụ đang chạy. Tô màu đỏ với các tác vụ bị lỗi. | XL11 |
| 2 | Người dùng bấm vào tên tác vụ. | Nếu tác vụ thực hiện thành công. Chuyển người dùng đến trang xem kết quả. Ngược lại không làm gì cả. | XL12 |
| 3 | Người dùng chọn tạo tác vụ | Kiểm tra tính hợp lệ dữ liệu mà người dùng nhập. Nếu có sai sót hoặc thiếu thì yêu cầu người dùng nhập lại. Nếu hợp lệ thì lưu trữ thông tin và bảng task và tiến hành thực nghiệm. | XL13 |

### Giao diện xem kết quả thực nghiệm

* Giao diện mẫu



Hình 9: Giao diện xem kết quả thực nghiệm

* Mô tả giao diện

Bảng 12: Mô tả giao diện xem kết quả

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Mô tả |
| 1 | Resulte | HTML link | Hiển thị link download file result |
| 2 | Bảng kết quả | HTML table | Hiển thị kết quả thực nghiệm |

* Mô tả các biến cố và xử lý

Bảng 13: Bảng sự kiện và xử lý của giao diện xem kết quả

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Biến cố | Xử lý | Mã xử lý |
| 1 | Load trang | * Kiểm tra trạng thái đăng nhập của người dùng. Nếu chưa đăng nhập thì về trang đăng nhập * Lấy mã tác vụ cần xem và load kết quả thực hiện vào bảng 1 | XL14 |
| 2 | Người dùng chọn result | * Chuyển đến trang download dữ liệu | XL15 |

## Kết chương

Trong chương này chúng em đã trình bày một số chức năng mà framework đánh giá các thuật toán khuyến nghị sẽ hỗ trợ cũng như các công nghệ dùng để xây dựng hệ thống. Trong chương tiếp theo chúng em sẽ trình về việc thực nghiệm và đánh giá các các thuật toán khuyến nghị dựa vào dataset do chúng em xây dựng.

# THIẾT LẬP THỰC NGHIỆM

## Dẫn nhập

Ở các chương trước khóa luận đã trình bày về lý thuyết và cài đặt hệ thống, tiếp theo ở chương này khóa luận sẽ mô tả về cách chúng em tiến hành thực nghiệm và các kết quả đạt được cũng như những nhận định về kết quả thực nghiệm.

## Môi trường thực nghiệm

Hệ thống được cài đặt trên môi trường như sau:

* + Ngôn ngữ lập trình: Java, HTML, JavaScript, CSS
  + Thư viện, framework hỗ trợ:
    - Mahout: hỗ trợ các thuật toán khuyến nghị
    - Spring framework: hỗ trợ lập trình web
    - Bootstrap: hỗ trợ thiết kế giao diện web
  + Hệ điều hành: Windows server 2012, 8GB RAM, 8 Cores

## Dữ liệu thực nghiệm

Hiện nay có rất ít hoặc không có bộ dữ liệu chuẩn được công bố cho lĩnh vực việc làm, vì vậy nhóm đã quyết định tiến hành thu thập dữ liệu cho nghiên cứu thực nghiệm. Dữ liệu được nhóm thu thập tự động từ 3 website đăng tuyển trực tuyến lớn tại Việt Nam gồm: vietnamworks.com, itviec.com và careerlink.vn.

### Thống kê về bộ dữ liệu (dataset)

Thời gian tạo dataset (gồm thời gian tiến hành thu thập dữ liệu việc làm và thời gian khảo sát người dùng) gồm 2 đợt: đợt 1 từ 7/2015 đến 9/2015; đợt 2 từ 5/2016 đến 6/2016.

Bộ dữ liệu thu thập gồm:

* + Thông tin dữ liệu về việc làm: tổng số việc làm là 7623 trên tổng cộng 29 ngành nghề khác nhau, dữ liệu này được thu thập tự động từ nguồn là các website tuyển dụng: vietnamworks.com, careerlink.vn, itviec.com. Dữ liệu việc làm được trình bày bằng tiếng Việt hoặc tiếng Anh.

Biểu đồ 4.1 Biều đồ thống kê số việc làm trên mỗi ngành nghề

Biểu đồ 4.2 Biểu đồ phân bổ số người dùng đã đánh giá việc làm trên mỗi ngành

* + Thông tin người dùng: Tổng số người dùng trong hệ thống là 59 người trên nhiều ngành nghề khác nhau. Mỗi người dùng có 01 hồ sơ việc làm (CV), các CV được viết bằng tiếng Việt.
  + Tập dữ liệu gán nhãn: là tập dữ liệu đánh giá của người dùng đối với các việc làm trong hệ thống. Người dùng sẽ đánh giá các việc làm theo thang điểm từ 1 đến 5 theo mức độ liên quan của việc làm đối với nhu cầu người đánh giá. Tổng cộng có 17337 bộ dữ liệu gán nhãn.

Biểu đồ 4.3 Biểu đồ mức độ tích cực đánh giá dựa trên số lượng đánh giá

Cách thức thu thập tập dữ liệu gán nhãn:

* + Tập dữ liệu gán nhãn được thu thập bằng cách tiến hành khảo sát trên các người dùng là những bạn sinh viên sắp ra trường, trên nhiều lĩnh vực (phần lớn là sinh viên CNTT).
  + Người dùng tham gia khảo sát sẽ tạo 1 hồ sơ việc làm (CV) và tham gia đánh giá tối thiểu 200 việc làm do hệ thống chọn ngẫu nhiên với tỉ lệ 50% việc làm đúng chuyên ngành của người dùng và 50% việc làm không đúng chuyên ngành.
  + Người dùng thực hiện đánh giá việc làm theo mức độ phù hợp của việc làm đó với nhu cầu của họ. Mức độ phù hợp của việc làm được đánh giá theo 5 cấp độ từ 1 - 5:
    - 1 (Không liên quan): việc làm này hoàn toàn không liên quan đến nhu cầu của người đánh giá. Mức độ phù hợp từ 0 – 20%
    - 2 (Không phù hợp): việc làm này có thể liên quan tới lĩnh vực ngành nghề của người đánh giá, nhưng nó không phù hợp với nhu cầu của họ. Mức độ phù hợp từ 20 – 40%
    - 3 (Có thể chấp nhận): việc làm này tương đối phù hợp với nhu cầu của người đánh giá, mặc dù có thể nó không đáp ứng được một vài tiêu chí. Mức độ phù hợp từ 40 – 60%
    - 4 (Khá phù hợp): việc làm này có mức độ phù hợp với người đánh giá khá cao, người đánh giá có khả năng cao sẽ apply vào vị trí việc làm này. Mức độ phù hợp từ 60 – 80%
    - 5 (Rất phù hợp): việc làm này phù hợp hoàn toàn với người đánh giá về mọi tiêu chí. Mức độ phù hợp từ 80 – 100%

Biểu đồ 4.4 Biểu đồ phân bố số lượng việc làm được đánh giá theo số điểm

### Nhận xét về bộ dữ liệu

Từ những kết quả thống kê trên chúng em có những nhận định về tập dữ liệu thực nghiệm như sau:

* Tập dữ liệu việc làm tồn tại trên 2 ngôn ngữ tiếng Việt và tiếng Anh, điều này sẽ ảnh hưởng đến kết quả khuyến nghị của thuật toán lọc theo nội dung. Vì vậy khi tiến hành thực nghiệm, chúng em đã tiến hành loại bỏ những việc làm bằng tiếng Anh.
* Đa số người dùng điều thuộc lĩnh vực IT-Phần mềm (biểu đồ 4.1 và 4.2) và các việc làm được đánh giá nhiều nhất cũng thuộc về lĩnh vực này. Đây là một kết quả hợp lý cho tập dữ liệu.
* Số lượng việc làm được đánh giá bởi một người dùng quá thấp so với tổng số việc làm. Việc này tạo nên một ma trận đánh giá quá thưa ảnh hưởng xấu đến kết quả của thuật toán.
* Số việc làm được đánh giá 1 và 2 điểm chiếm tỉ lệ quá lớn (65% tổng số đánh giá) có nghĩa là 65% việc làm trong tập dữ liệu thực nghiệm là không liên quan tới người dùng (Biểu đồ 4.4).
* Dựa vào biểu đồ sau ta có thể thấy tỉ lệ việc làm được đánh giá liên quan (điểm đánh giá >= 3) là quá thấp so với tổng số việc làm. Tỉ lệ giao nhau (mối liên hệ) giữa các việc làm được đánh giá bởi từng người dùng là rất thấp.

Biểu đồ 4.5 Biểu đồ tỉ lệ việc làm được đánh giá liên quan, không liên quan và việc làm chưa được đánh giá

## Cách tiến hành thực nghiệm

Từ những thống kê và nhận định trên phần nào cho thấy kết quả thực nghiệm sẽ bị ảnh hưởng rất nhiều vì bộ dữ liệu thu thập chưa hoàn chỉnh. Do đó chúng em đưa ra giải pháp thực nghiệm với một tập dữ liệu giới hạn hơn bằng cách loại bỏ những việc làm bằng tiếng Anh, bộ dữ liệu chính thức được đưa vào thực nghiệm gồm: 4475 việc làm tiếng Việt, 59 người dùng và 8576 dữ liệu gán nhãn.

Thực nghiệm được tiến hành trên tất cả các thuật toán với CF (CF trên nhiều độ đo độ tương tự), CB và Simple Linear Hybrid.

Thực nghiệm được tiến hành theo phương pháp thiết lập đánh giá offline sử dụng K-fold Cross-validation với số fold k=10. Mỗi thuật toán khuyến nghị được tiến hành thực nghiệm với tất cả các độ đo đánh giá khác nhau đã trình bày ở mục 2.5.2.

## Kết quả thực nghiệm

Với các thiết lập và bộ dữ liệu như đã mô tả, kết quả thực nghiệm tốt nhất cho từng thuật toán như sau:

* Thuật toán CF đạt kết quả tốt nhất với phương pháp User-based với độ tương tự Euclidean Distance. Với kết quả tốt nhất F1@10 = 0.018
* Thuật toán CB cho kết quả tốt nhất với F1@5 = 0.01
* Thuật toán Hybrid cho kết quả tốt nhất với F1@5 = 0.013

Bảng 4.1 Bảng kết quả thực nghiệm với độ đo Precision, Recall và F-Measure

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Độ  Đo  Thuật  toán | Precision | | | Recall | | | F-Measure | | |
| Top5 | Top10 | Top15 | Top5 | Top10 | Top15 | Top5 | Top10 | Top15 |
| CF - User based - Euclidean Distance | 0.013 | 0.013 | 0.008 | 0.021 | 0.044 | 0.047 | 0.014 | 0.018 | 0.012 |
| CB | 0.008 | 0.007 | 0.005 | 0.017 | 0.03 | 0.034 | 0.01 | 0.01 | 0.008 |
| Hybrid | 0.011 | 0.007 | 0.005 | 0.026 | 0.038 | 0.037 | 0.013 | 0.011 | 0.009 |

Bảng 4.2 Bảng kết quả thực nghiệm với độ đo RMSE

|  |  |  |  |
| --- | --- | --- | --- |
| Độ đo  Thuật toán | RMSE | | |
| Top5 | Top10 | Top15 |
| CF - User based - Euclidean Distance | 0.031 | 0.047 | 0.067 |
| CB | 0.025 | 0.04 | 0.046 |
| Hybrid | 0.189 | 0.248 | 0.262 |

Bảng 4.3 Bảng kết quả thực nghiệm với độ đo NDCG, MRR và MAP

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Đô đo  Thuật toán | NDCG | | | MRR | | | MAP | | |
|  | Top5 | Top10 | Top15 | Top5 | Top10 | Top15 | Top5 | Top10 | Top15 |
| CF - User based - Euclidean Distance | 0.014 | 0.013 | 0.008 | 0.033 | 0.036 | 0.022 | 0.032 | 0.036 | 0.022 |
| CB | 0.008 | 0.007 | 0.005 | 0.015 | 0.019 | 0.017 | 0.016 | 0.019 | 0.016 |
| Hybrid | 0.011 | 0.007 | 0.006 | 0.026 | 0.020 | 0.023 | 0.026 | 0.020 | 0.023 |

## Nhận định về kết quả thực nghiệm

Với kết quả thực nghiệm ở trên có thể dễ thấy rằng kết quả thực nghiệm đạt kết quả quá thấp. Các nguyên nhân dẫn đến kết quả này có thể lý giải như sau:

* Bộ dữ liệu gán nhãn quá thưa (biểu đồ 4.5), với bộ dữ liệu thưa như vậy không thể sử dụng làm tập dữ liệu GroundTruth để đánh giá các thuật toán.

## Kết chương

Chương này chúng em đã trình bày về các thiết lập thực nghiệm và các kết quả đạt được cũng như những nhận định về kết quả đạt được. Chương tiếp theo sẽ là kết luận rút ra và hướng phát triển tiếp theo cho bài toán.

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## Các kết quả đạt được

## Giá trị thực tiễn của khóa luận

## Hướng phát triển

TÀI LIỆU THAM KHẢO

Theo chuẩn IEEE