BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC CẦN THƠ**

**KHOA CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG**

**🙞 🕮 🙜**



**LUẬN VĂN NGÀNH**

**NGÀNH KHOA HỌC MÁY TÍNH**

**Đề tài**

**HỆ THỐNG GỢI Ý**

**CÂU HỎI TƯƠNG TỰ**

**Sinh viên thực hiện : Nguyễn Tấn Pil**

**Mã số : B1812295**

**Khóa : 44**

Cần Thơ, 5/2022

BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC CẦN THƠ**

**KHOA CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG**

**🙞 🕮 🙜**



**LUẬN VĂN NGÀNH**

**NGÀNH KHOA HỌC MÁY TÍNH**

**Đề tài**

**HỆ THỐNG GỢI Ý**

**CÂU HỎI TƯƠNG TỰ**

**Giáo viên hướng dẫn: Sinh viên thực hiện:**

**TS.Lưu Tiến Đạo Nguyễn Tấn Pil**

**Mã số: B1812295**

**Khóa : 44**

Cần Thơ, 5/2022

**NHẬN XÉT CỦA GIẢNG VIÊN**

------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Cần Thơ, ngày tháng 12 năm 2022

(GVHD ký và ghi rõ họ tên)

**LỜI CẢM ƠN**

Để hoàn thành đề tài luận văn này, em xin được bày tỏ lòng biết ơn chân thành và sâu sắc đến Thầy Lưu Tiến Đạo – người đã trực tiếp tận tình hướng dẫn, giúp đỡ em trong suốt quá trình thực hiện luận văn, nhờ sự chỉ bảo và hướng dẫn quý giá đó mà bài luận văn này được hoàn thành một cách tốt nhất.

Em cũng xin gửi lời cảm ơn chân thành đến các Thầy Cô Giảng viên Đại học Cần Thơ, đặc biệt là các Thầy Cô ở Khoa CNTT & TT, những người đã truyền đạt những kiến thức quý báu trong suốt khoảng thời gian qua.

Em cũng xin chân thành cảm ơn bạn bè cùng với gia đình luôn luôn động viên, khích lệ và tạo điều kiện giúp đỡ trong suốt quá trình thực hiện để em có thể hoàn thành bài niên luận một cách tốt nhất.

Tuy thật sự quyết tâm và cố gắng trong suốt quá trình thực hiện đề tài, nhưng không thể tránh khỏi những sai sót. Em rất mong nhận được sự đóng góp ý kiến quý báu của quý Thầy Cô và các bạn để bài niên luận hoàn thiện hơn.

Cần Thơ, ngày tháng 12 năm 2022

Người viết

Nguyễn Tấn Pil

**MỤC LỤC**

[DANH MỤC HÌNH 3](#_Toc119275216)

[DANH MỤC BẢNG 4](#_Toc119275217)

[TÓM TẮT 5](#_Toc119275218)

[PHẦN GIỚI THIỆU 6](#_Toc119275219)

[1. Đặt vấn đề 6](#_Toc119275220)

[2. Lịch sử giải quyết vấn đề 7](#_Toc119275221)

[3. Mục tiêu đề tài 7](#_Toc119275222)

[4. Đối tượng và phạm vi nghiên cứu 8](#_Toc119275223)

[5. Phương pháp nghiên cứu 8](#_Toc119275224)

[6. Kết quả đạt được 8](#_Toc119275225)

[7. Bố cục luận văn 8](#_Toc119275226)

[PHẦN NỘI DUNG 9](#_Toc119275227)

[CHƯƠNG 1 - MÔ TẢ BÀI TOÁN 9](#_Toc119275228)

[1. Mô tả chi tiết bài toán 9](#_Toc119275229)

[2. Vấn đề và giải pháp liên quan đến bài toán 10](#_Toc119275230)

[2.1. Công cụ thiết kế website 10](#_Toc119275231)

[2.2. Hệ thống gợi ý 13](#_Toc119275232)

[CHƯƠNG 2 - THIẾT KẾ VÀ CÀI ĐẶT 29](#_Toc119275233)

[1. Thiết kế hệ thống 29](#_Toc119275234)

[1.1 Sơ đồ hệ thống 29](#_Toc119275235)

[1.2 Giới thiệu tập dữ liệu 32](#_Toc119275236)

[1.3 Cấu trúc của website 32](#_Toc119275237)

[2. Cài đặt giải thuật 32](#_Toc119275238)

[2.1 Thu thập dữ liệu 32](#_Toc119275239)

[2.2 Sơ lượt về dữ liệu 33](#_Toc119275240)

[2.3 Tiền xử lý dữ liệu 34](#_Toc119275241)

[2.4 Trích xuất các đặt trưng 34](#_Toc119275242)

[2.5. Xây dựng hệ thống gợi ý 34](#_Toc119275243)

[CHƯƠNG 3 – GIỚI THIỆU WEBSITE 36](#_Toc119275244)

[1. Giao diện chính hệ thống 36](#_Toc119275245)

[2. Đánh giá 36](#_Toc119275246)

[2.1 Given-N 36](#_Toc119275247)

[2.2 All-but-one 36](#_Toc119275248)

[PHẦN KẾT LUẬN 37](#_Toc119275249)

[1. Kết quả đạt được 37](#_Toc119275250)

[2. Hướng phát triển 37](#_Toc119275251)

[TÀI LIỆU THAM KHẢO 38](#_Toc119275252)

# DANH MỤC HÌNH

[Hình 1: Hôm nay ăn gì? 7](#_Toc103782722)

[Hình 2: Gợi ý từ website Amazon. 8](#_Toc103782723)

[Hình 3: Gợi ý từ Netflix 9](#_Toc103782724)

[Hình 4: Quy trình xây dụng hệ thống gợi ý 15](file:///C:\Users\DELL\OneDrive\Nien_Luan\Nguyen_Tan_Pil_Recipe_Recommendation.docx#_Toc103782725)

[Hình 5: Sơ đồ tổng quát hệ thống gợi ý 21](#_Toc103782726)

[Hình 6: Cấu trúc website 23](#_Toc103782727)

[Hình 7: Giải thuật đăng ký tài khoản 26](#_Toc103782728)

[Hình 8: Giải thuật đánh giá và trả lời đánh giá 27](#_Toc103782729)

[Hình 9: Giải thuật đưa ra gợi ý công thức nấu ăn 29](#_Toc103782730)

[Hình 10: Giải thuật cập nhật thông tin người dùng 30](#_Toc103782731)

[Hình 11: Dữ liệu công thức nấu ăn 31](#_Toc103782732)

[Hình 12: Dữ liệu người dùng 31](#_Toc103782733)

[Hình 13: Tạo ma trân quan hệ 32](#_Toc103782734)

[Hình 14: Tìm các công thức nấu ăn tương tự 32](#_Toc103782735)

[Hình 15: Lọc các công thức có độ tương tự cao nhất 32](#_Toc103782736)

[Hình 16: Giao diện trang chủ hệ thống 33](#_Toc103782737)

[Hình 17: Giao diện tất cả các công thức nấu ăn 33](#_Toc103782738)

[Hình 18: Giao diện các công thức nấu ăn phổ biến nhất 34](#_Toc103782739)

[Hình 19: Giao diện tìm kiếm công thức nấu ăn theo tên 34](#_Toc103782740)

[Hình 20: Giao diện đăng ký tài khoản 35](#_Toc103782741)

[Hình 21: Giao diện đăng nhập 35](#_Toc103782742)

[Hình 22: Giao diện thông tin người dùng 36](#_Toc103782743)

[Hình 23: Giao diện cập nhật thông tin người dùng 37](#_Toc103782744)

[Hình 24: Giao diện chi tiết công thức nấu ăn 38](#_Toc103782745)

[Hình 25: Giao diện đánh giá công thức nấu ăn 38](#_Toc103782746)

[Hình 26: Giao diện các công thức nấu ăn đã được lưu 39](#_Toc103782747)

[Hình 27: Giao diện tạo công thức nấu ăn 39](#_Toc103782748)

[Hình 28: Giao diện thay đổi theme 40](#_Toc103782749)

# DANH MỤC BẢNG

[Bảng 1: Ví dụ các thuộc tính của items 16](#_Toc104222538)

[Bảng 2: Profile người dùng 17](#_Toc104222539)

[Bảng 3: Các items của người dùng 17](#_Toc104222540)

[Bảng 4: Các items sau khi nhân với profile người dùng 18](#_Toc104222541)

[Bảng 5: Tổng điểm theo thành phần dinh dưỡng của người dùng 19](#_Toc104222542)

[Bảng 6: Scale các điểm thành phần về 1 19](#_Toc104222543)

[Bảng 7: Các items mới có thể gợi ý cho người dùng 19](#_Toc104222544)

[Bảng 8: Các items sau khi nhân với điểm thành phần dinh dưỡng người dùng 20](#_Toc104222545)

[Bảng 9: Điểm các items khi dự đoán 20](#_Toc104222546)

[Bảng 10: Cấu trúc dữ liệu công thức nấu ăn 21](#_Toc104222547)

[Bảng 11: Cơ sở dữ liệu công thức nấu ăn 23](#_Toc104222548)

[Bảng 12: Cơ sở dữ liệu người dùng 24](#_Toc104222549)

[Bảng 13: Cơ sở dữ liệu đánh giá 24](#_Toc104222550)

[Bảng 14: Cơ sở dữ liệu trả lời đánh giá 25](#_Toc104222551)

# TÓM TẮT

Ngày nay

# PHẦN GIỚI THIỆU

### 1. Đặt vấn đề

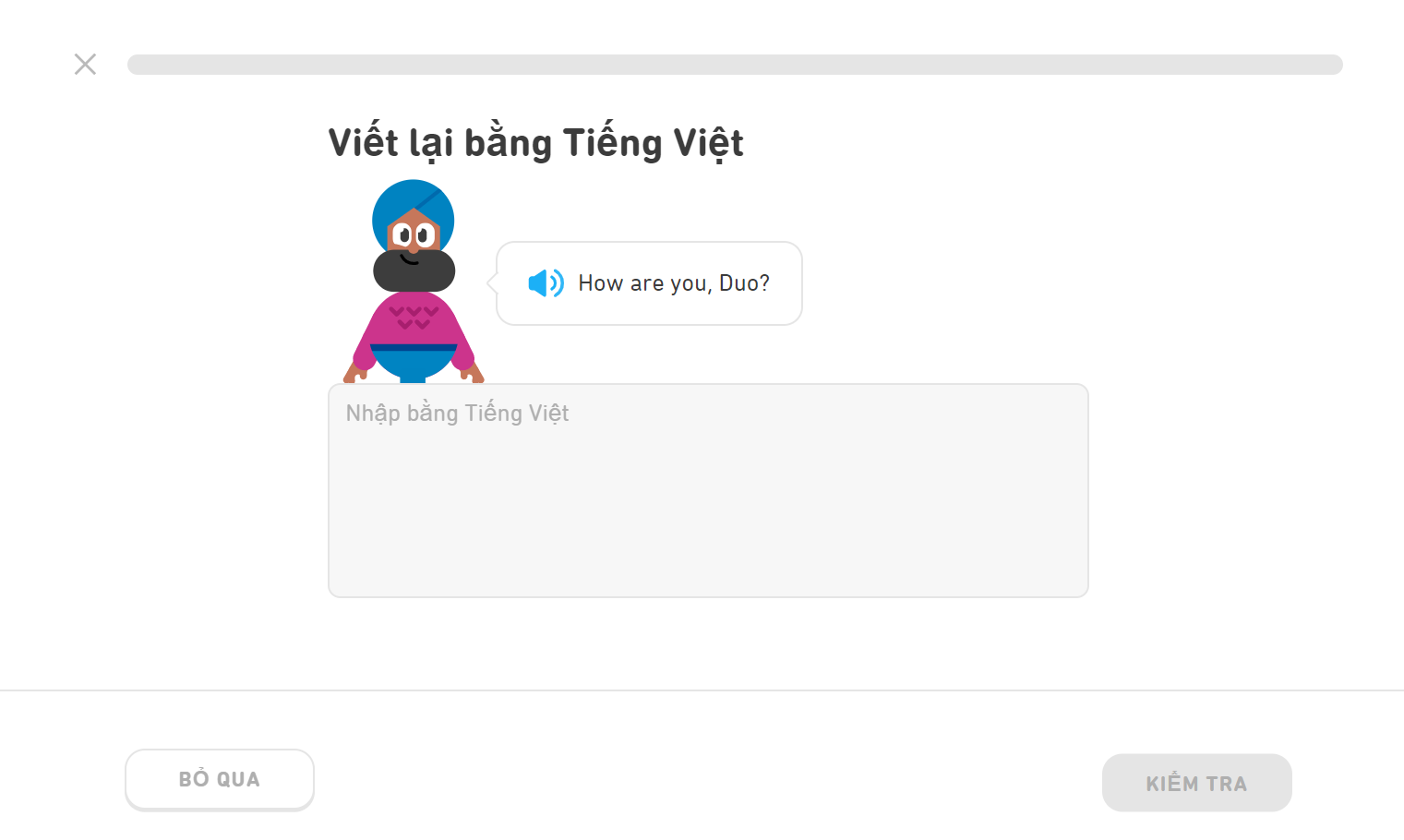
Ngày nay, với sự phát triển mạnh mẽ của Công Nghệ Thông Tin chúng ta có thể học tập, trao đổi và tìm kiếm thông tin một cách nhanh chóng thông qua Internet. Ngày càng có nhiều ứng dụng, website được xây dựng để hỗ trợ cho cuộc sống hằng ngày của chúng ta dễ dàng hơn, các ứng dụng, website có mặt hầu hết trên tất cả các lĩnh vực trong cuộc sống. Có thể kể đến như shoppe, tiki, lazada hỗ trợ việc mua sắm trở nên dễ dàng và thuận lợi hơn. Ngoài ra có website kahoot, quizizz hỗ trợ người dùng có thể tạo các câu hỏi trắc nghiệm và tạo ra các bài thì hoặc trò chơi đó vui một cách dễ dàng và nhanh chóng.

Các hệ thống câu hỏi trắc nghiệm trực tuyến, giảng dạy online đang được áp dụng rộng rãi và phổ biến hơn bao giờ hết. Các hệ thống trắc nghiệm online hỗ trợ rất tốt cho cả giáo viên và các bạn học sinh, sinh viên trong quá trình học tập và kiểm tra năng lực của học sinh, sinh viên, hệ thống trắc nghiệm giúp người ra đề tiết kiệm thời gian ra đề, cũng như thời gian trông coi thi, các câu hỏi thiết kế đa dạng và phong phú và các câu hỏi được hệ thống trộn ngẫu nhiên sẽ hạn chế được tình trạng gian lận trung lúc làm bài, nâng cao chất lượng của bài thi một cách đáng kể. Ngoài ra còn có nhiều ứng dụng, trang web sử dụng hình thức trắc nghiệm để giúp cho việc học tập một cách hiệu quả hơn, các hệ thống có tích hợp hệ thống gợi ý, giúp cho người học có thể học tốt hơn, ghi nhớ tốt hơn.

Nhờ có những hệ thống như trên giúp cho người học dễ dàng học tập và ôn luyện những kiến thức mà mình muốn tiếp thu. Hình thức trắc nghiệm giúp cho người học cảm thấy thích thú trong việc tiếp thu kiến thức, sau khi học những kiến thức bằng nhiều dòng văn bản, người học có thể sử dụng hệ thống trắc nghiệm để ôn luyện lại những kiến thức mà mình đã học, sau khi hoàn thành bài kiểm tra thì có thể biết được kết quả và những lỗi sai. Một số hệ thống còn giúp người học ghi nhớ những lỗi sai và chỉnh sủa giúp họ bằng cách đưa ra các câu hỏi tương tự để học có thể thực hiện lại và ghi nhớ kỹ hơn về câu hỏi đó.

### 2. Lịch sử giải quyết vấn đề

Hệ thống gợi ý đã và đang được sử dụng rộng rãi trong nhiều nền tảng khác nhau như **Duolingo** là trang web cũng như ứng dụng học Tiếng Anh phổ biến trên thế giới, hệ thống có tích hợp hệ thống gợi ý khi chúng ta trả lời sai, hệ thống sẽ đưa ra câu hỏi tương tự với câu hỏi đã trả lời sai.



Hình 2: Gợi ý từ website Duolingo.

### 3. Mục tiêu đề tài

Xây dựng một hệ thống gợi ý các câu hỏi tương tự khi người tham gia trả lời không chính xác và có thể ứng dụng vào các trang web có hệ thống câu hỏi trắc nghiệm.

Tạo ra được một website có giao diện bắt mắt, dễ sử dụng và ứng dụng được hệ thống gợi ý vào trong website.

Tìm hiểu được quy trình để phát triển một hệ thống, ứng dụng. Hiểu được các kiến thức về xây dụng một hệ thống gợi ý, đặc biệt củng cố và ứng dụng được phương pháp gợi ý dựa trên lọc theo nội dung.

Nguyên cứu cách thức hoạt động, mô hình, phương pháp và tầm quan trọng của hệ thống gợi ý.

### 4. Đối tượng và phạm vi nghiên cứu

Đối tượng nghiên cứu của đề tài là nghiên cứu về phương pháp gợi ý dựa trên nội dung bằng cách đưa các câu về các vector và sau đó tính độ tương tự của các câu với nhau. Sau khi có độ tương tự thì dựa vào ma trận xây dựng đưa ra danh sách các câu hỏi có độ tương tự cao nhất và thấp nhất.

Phạm vi nghiên cứu của đề tài là xử lý văn bản văn bản tiếng Việt, vector hóa văn bản và tính độ tương tự của các câu hỏi.

### 5. Phương pháp nghiên cứu

Trong đề tài này chúng ta tiếp cận bằng phương pháp gợi ý dựa trên nội dung, sử dụng các phương pháp vector hóa văn bản và các phương pháp tính độ tương tự của văn bản. Về phần thực hiện thì sử dụng ngôn ngữ lập trình Python, các thư viện tách từ, vector hóa văn bản và tính độ tương tự.

### 6. Kết quả đạt được

Xây dựng được trang web tạo và thực hiện kiểm tra các câu hỏi trắc nghiệm.

Xây dựng được mô hình và API gợi ý câu hỏi tương tự bằng phương pháp lọc theo nội dung. Học được cách phân công, lên kế hoạch và giải quyết vấn đề.

### 7. Bố cục luận văn

**Phần giới thiệu**

Giới thiệu tổng quát về đề tài.

**Phần nội dung**

**Chương 1** : Mô tả bài toán.

**Chương 2** : Thiết kế, cài đặt giải thuật, biễu diễn cơ sở dữ liệu, trình bày các bước xây dựng hệ thống bằng phương pháp lọc theo nội dung.

**Chương 3** : Kiểm thử hệ thống, đánh giá độ chính xác và demo trang web.

**Phần kết luận**

Trình bày kết quả đạt được và hướng phát triển hệ thống.

# PHẦN NỘI DUNG

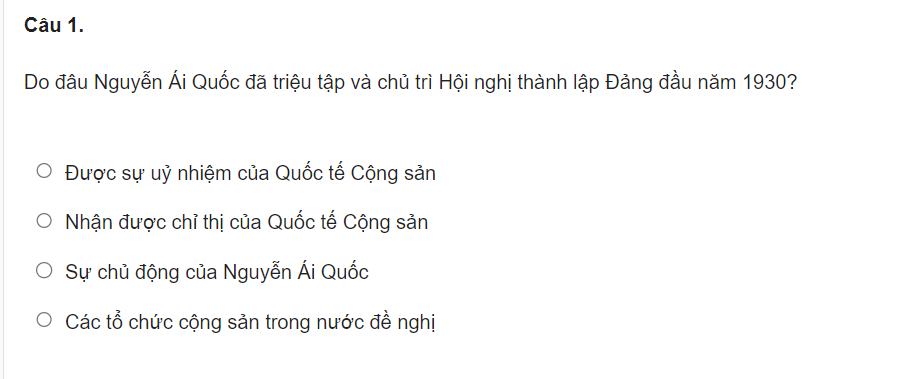
## CHƯƠNG 1 - MÔ TẢ BÀI TOÁN

### 1. Mô tả chi tiết bài toán

Ngày nay, việc ứng dụng Công Nghệ Thông Tin vào trong việc học tập và làm việc không còn xa lạ với chúng ta nữa, trên Internet có rất nhiều website ứng dụng các hệ thống thi và ôn tập bằng hình thức trắc nghiệm. Các hệ thống thi trắc nghiệm hỗ trợ cho người học có thể không cần phải đến tận nơi để làm bài thi hoặc bài kiểm tra mà có thể trực tiếp thực hiện bài thi tại bất cứ nơi đâu, chỉ cần có các thiết bị thông mình có thể kết nối Internet là được. Trong khoảng thời gian do dịch bệnh covid nhờ có các hệ thống giảng dạy online giúp kết nối giảng viên trong quá trình giảng dạy và thi cử trở nên dễ dạng hơn, cho thấy được hiệu quả của hệ thống giảng dạy online. Không chỉ trong thời gian dịch bệnh, các hệ thống thi và ôn luyện online vẫn là một lựa chọn hàng đầu trong các hình thức thi cử.

Để giúp có người dùng có thể đạt được hiệu quả ôn luyện cao nhất trong quá trình ôn tập thì webiste trắc nghiệm online cần có một hệ thống đưa ra các câu hỏi tương tự khi trả lời câu hỏi chưa chính xác. Đưa ra các câu hỏi tương tự với câu hỏi trước đó chưa chính xác giúp cho người dùng có thể thực hiện câu hỏi tương tự với câu hỏi đã trả lời chưa chính xác tăng khả năng ghi nhớ về câu hỏi đó nhiều hơn. Và khi người dùng trả lời đúng một câu thì hệ thống gợi ý sẽ đưa ra các câu hỏi có độ tương tự thấp nhất với câu hỏi trước đó đã trả lời đúng, để bài ôn tập của người dùng sẽ bao quát nhiều câu hỏi khác nhau trong bộ đề.

Hệ thống gợi ý câu hỏi tương tự được xây dựng được thực hiện bằng cách sử dụng các câu hỏi sau khi được tiền xử lý văn bản, sau khi tiền xử lý văn bản thì sử dụng các thư viện tách từ tiếng Viết để tách các từ trong câu. Câu hỏi sau khi được tách từ thì đưa về các vector số, vì máy tính không hiểu được ngôn ngữ dạng văn bản nên phải đưa các câu hỏi về dạng các dãy số thì máy tình mới có thể tính toán được. Sau khi có được các vector số thì sử dụng độ tương tự cosin để tính độ tương tự giữa các câu hỏi với nhau, sau đó chọn ra các câu hỏi có độ tương tự cao nhất và thấp nhất với từng câu hỏi và đưa ra gợi ý cho người dùng.



VnDoc là một trang web mạng xã hội học tập, chia sẻ tài liệu miễn phí, do có thể chia sẽ tài liệu, nên tài nguyên của website vô cùng đa dạng và phong phú. Trang web có thể giúp người dùng ôn tập kiến thức với nhiều chủ đề và nội dung khác nhau, có thể ôn tập để chuẩn bị cho các kỳ thi.

### 2. Vấn đề và giải pháp liên quan đến bài toán

Để xây dựng một hệ thống gợi ý thì có nhiều phương pháp như gợi ý dựa theo lọc cộng tác, gợi ý dựa trên lọc theo nội dung và kết hợp cả 2 phương pháp trên. Do dữ liệu có dạng văn bản nên trong đề tài này chúng tôi sẽ sử dụng phương pháp gợi ý dựa trên lọc theo nội dung.

Phương pháp lọc nội dung được triển khai rộng rãi nhất và thành công nhất trong thực tế. Phương pháp này phân tích và đánh giá dựa trên ngượi dùng hoặc mục dữ liệu sau đó tìm ra sự tương đồng thông qua các đánh giá và tạo ra các gợi ý cho người dùng nào đó. Trong đề tài chúng tôi dựa vào nội dung của các câu hỏi để phân tích và tính độ tương tự giữa các câu để đưa ra các gợi ý cho người dùng.

#### 2.1**. Công cụ thiết kế website**

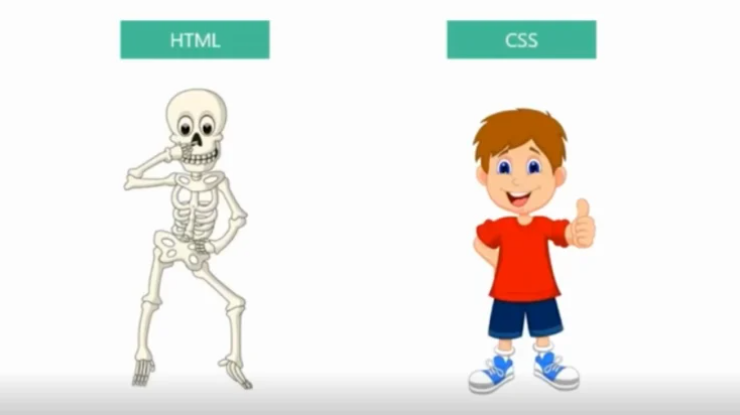
##### 2.1.1. HTML và CSS

**HTML (viết tắt của từ Hypertext Markup Language, hay là “Ngôn ngữ đánh dấu siêu văn bản”)** là một ngôn ngữ đánh dấu tiêu chuẩn cho các tài liệu được thiết kế hiển thị trên trình duyệt web. Nó có thể kết hợp với CSS và JavaScript, HTML là một trongnhững ngôn ngữ quan trọng trong lĩnh vực thiết kế website. HTML có thể nhận từ server hoặc từ local storage và sau đó thực hiện render vào trang web.

Phiên bản chính thức mới nhất của HTML là HTML 4.01 (1999). Sau đó, các nhà phát triển đã thay thế nó bằng XHTML. Hiện nay, phiên bản mới nhất của ngôn ngữ này là HTML5.

Khi làm việc với HTML, chúng ta sẽ sử dụng cấu trúc code đơn giản (tên thẻ và thuộc tính thể) để định nghĩa thẻ và hiển thỉ lên trình duyệt. Ví dụ, chúng ta có thể tạo một đoạn văn và định nghĩa thuộc tính class bằng cách đặt văn bản vào trong cặp tag mở và đóng văn bản <p class=”description”> và </p>.

**CSS** là ngôn ngữ tạo phong cách cho trang web – **Cascading Style Sheet** language. Nó dùng để tạo styles và định kiểu cho những thẻ được viết dưới dạng ngôn ngữ đánh dấu, như là HTML. Nó có thể định kiểu của nhiều trang web cùng lúc để tiết kiệm công sức và tăng khả năng tái sử dụng code cho người viết web. Nó phân biệt cách hiển thị của trang web với nội dung chính của trang bằng cách điều khiển bố cục, màu sắc, và font chữ.

****

Chúng ta có thể hiểu rằng HTM cung cấp cấu trúc cơ bản cho trang web và CSS cung cấp kiểu dáng, định dạng cho cấu trúc HTML đó. Như hình ở trên HTML xem như là bộ xương của cơ thể người và CSS giống như phần da và trang phục để làm cho bộ xương trở nên trông đẹp hơn lúc ban đầu.

##### 2.1.2. JavasSript

**JavaScript (gọi tắt là JS)** là ngôn ngữ lập trình kích bản phía client cũng như phía server (NodeJS), các mã lệnh được thực thi bởi trình duyệt của người dùng khi cho phép sử dụng JavaScript. Là một công nghệ lõi của **World Wide Web** cùng với HTML và CSS tạo nên một bộ 3 không thể thiếu của nhà phát triển website.

**JavaScript** sử dụng kết hợp với HTML và CSS tạo nên các tính năng tương tác, làm website sinh động hơn, tăng trải nghiệm người dùng. **Javascript** cho phép chúng kiểm soát các hành vi của website tốt hơn so với việc chỉ sử dụng HTML và CSS.

**JavaScript** là ngôn ngữ lập trình đơn giản, dễ học và tìm hiểu cũng như sử dụng nó. Cú pháp của ngôn ngữ có những phương thức đặt tên khá giống với tiếng Anh nên có thể dễ hiểu và sử dụng một cách nhanh chóng.

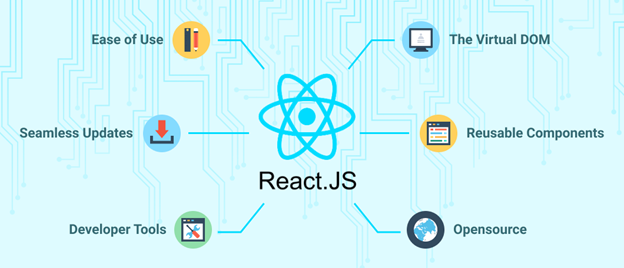


Sau khi có HTML và CSS thì tiếp theo chúng ta có Javascript giúp cho trang web có thể thực hiện tính năng có thể tương tác với người dùng làm cho wesite có tính tương tác và tăng trải nghiệm của người dùng một cách đáng kể. Giống như sau khi trở nên đẹp hơn với CSS giờ có thêm Javascript giúp người bên trên có thể thực hiện các chuyển động.

##### 2.1.3. ReactJS

**ReactJS** là một thư viện JavaScript xây dựng giao diện người dùng có tính hiệu quả và linh hoạt có thể tái sử sử dụng lại thông qua các thành phần gọi là components. ReactJS xây dụng các thành phần với tư tưởng phân chia, từ các thành phần lớn, phức tạp chuyển đổi thành các thành phần nhỏ hơn, dễ quản lý và tái sử dụng hơn. React được tạo ra bởi Jordan Walke, một kỹ sư phần mềm tại Facebook, phát triển và duy trì bởi Facebook và các ứng dụng nổi tiếng như WhatsApp & Instagram sử dụng công nghệ này.

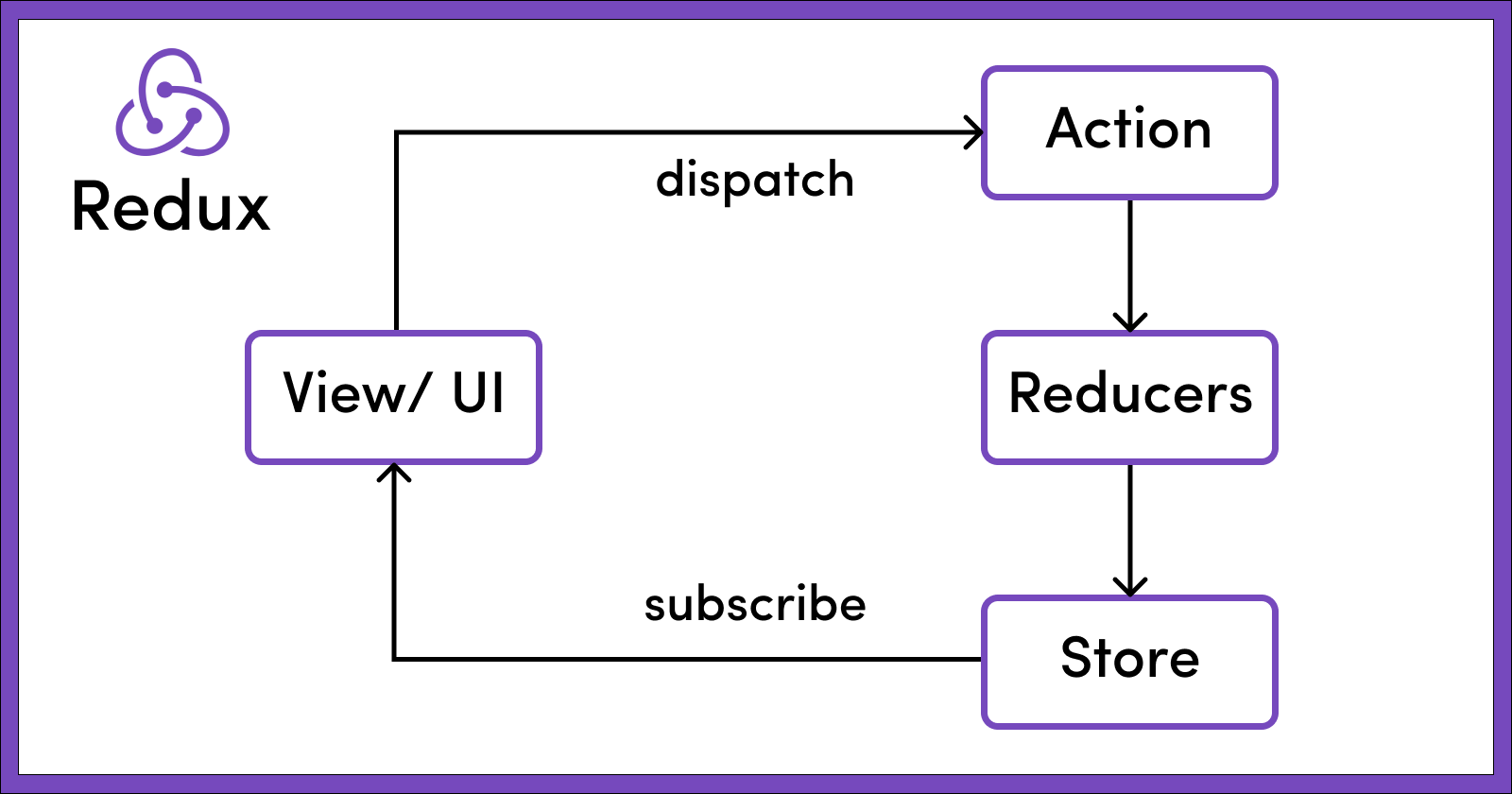
**ReactJS** được dùng để xây dựng các ứng dụng [Single Page Application] (SPA). Một trong những điểm hấp dẫn của ReactJS là nó không chỉ được xây dựng bên phía clients mà còn sử dụng được bên phía server[1].



ReactJs là một trong các frameworks javascript để xây dụng giao diện trang web dễ sử dụng, dễ học với cơ chế DOM ảo giúp tăng hiệu xuất của webiste. React giúp cho việc viết code nhanh chóng bằng các components sử dụng lại, có hệ sinh thái các các packages hỗ trợ và cộng đồng của framework này rất nhiều.

##### 2.1.4 Redux

Redux là một predictable state management tool cho các ứng dụng Javascript. Nó giúp chúng ta viết các ứng dụng hoạt động một cách nhất quán, chạy trong các môi trường khác nhau và dễ dàng kiểm tra, kiểm thử. Redux ra đời lấy cảm hứng từ tư tưởng của ngôn ngữ Elm và kiến trúc Flux của Facebook[20].



Redux store là một nơi để chứa các state của ứng dụng, khi các state của store có sự thay đổi thì UI cũng nhận biết và thay đổi theo. Khi ở phía UI muốn thay đổi, cập nhật một state nào đó trong store sẽ dispatch một action đi, action sẽ được xử lý ở các reducers và thực hiện việc cập nhật các state theo actions tương ứng.

#### 2.2. Hệ thống gợi ý

**Hệ thống gợi ý** (Recommendation system) là một lớp con của hệ thống lọc thông tin, tìm cách dự đoán “xếp hạng” hoặc “ưa thích” của một người dùng nào đó với một sản phẩm hoặc một đối tượng. Các hệ thống gợi ý thường được sử dụng trong nhiều lĩnh vực khác nhau, đặc biệt là trong các ứng dụng thương mại[4].

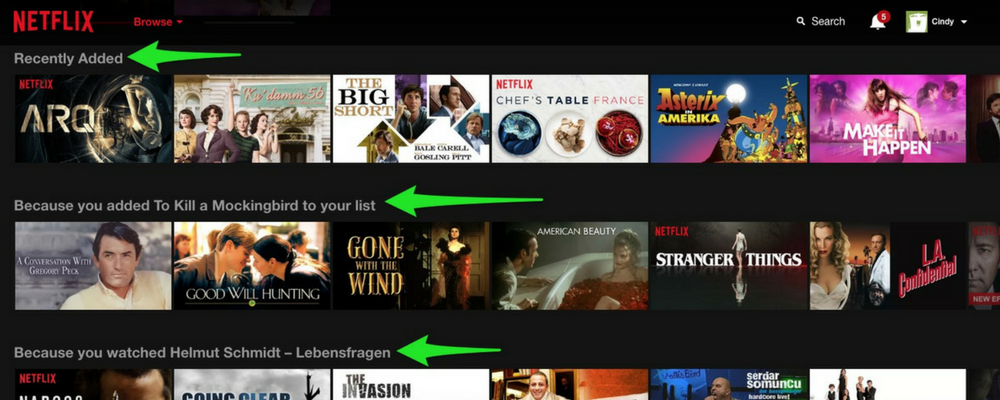
Các hệ thống gợi ý được sử dụng ở nhiều lĩnh vực khác nhau, các trường hợp phổ biến như gợi ý các video và bài hát, giới thiệu các sản phẩm cho người dùng mua hàng online trên các trang thương mại điện tử và các trang mạng xã hội. Sau khi người dùng mua một sản phẩm trên trang thương mại điện tử, hệ thống sẽ phân tích các sở thích, các sản phẩm tương tự với sản phẩm mà người dùng đã xem và đưa ra các sản phẩm mà người dùng có thể quan tâm.

Một số hệ thống gợi ý:

* Thương mại điện tử: Azazon, Alibaba, Tiki…
* Giải trí: Youtube, Netflix, Spotify…
* Mạng xã hội: Facebook, Zalo…
* Ứng dụng học tập: Duolingo,..

Một số minh chứng cụ thể cho thấy hệ thống gợi ý đóng một vai trò quan trọng trong cuộc sống hằng ngày[5]:

Hệ thống gợi ý đóng vai trò rất quan trọng trong các trang web nổi tiếng như Youtube, Amazon, Netflix… Bên cạnh đó, nhiều công ty truyền thông đang phát triển và triển khai các hệ thống gợi ý như một phần dịch vụ cung cấp cho các người dùng. Một trong số các hoạt động tiêu biểu đó là Netflix, nhà chung cấp dịch vụ xem các phim và chương trình trực tuyến theo hình thức đăng ký, đã trao giải thưởng 01 triệu đô la cho nhóm làm tăng độ chính xác lên 10.05%.



Trên đây là website Netflix, là một website cung cấp dịch vụ xem video trực tuyến, cho phép người dùng có thể xem các chương trình truyền hình và phim ảnh. Khi truy cập vào website, hệ thống gợi ý của Netflix sẽ hỗ trợ chúng ta tìm một chương trình hoặc một bộ phim tương tự để thưởng thức.

Hệ thống gợi ý của Netflix lấy ý kiến từ mỗi lần truy cập vào website của người dùng và liên tục thực hiện lại các thuật toán với các phản hồi để cải thiện độ chính xác của hệ thống gợi ý. Dữ liệu, thuật toán và hệ thống tính toán kết hợp với nhau để đưa ra các đề xuất mới phù hợp với người xem[19].

Có nhiều hội nghị, hội thảo chuyên về lĩnh vực hệ thống gợi ý, một trong số đó có thể để cập đến ACM Recommeder System(RecSyss), thành lập vào năm 2007 là sự kiện đáng chú ý hằng năm trong lĩnh vự nghiên cứu hệ thống gợi ý. Thêm đó các buổi họp dành riêng cho vấn đề hệ thống gợi ý được tổ chức trong các hội nghị cơ sở dữ liệu, hệ thống thông tin.

Hiện nay trong các trường đại học và sau đại học, các khóa học, học phần hệ thống gợi ý đang trở nên phổ biến, đặt biệt là trong chuyên ngành Khoa học máy tính. Số lượng tài liệu và cộng đồng về hệ thống gợi ý ngày càng tăng.

Các tạp chí khoa học tiến hành xuất bản các số đặt biệt đề cập về nghiên cứu và phát triển trong lính vực hệ thống gợi ý. Các bài báo tiêu biểu như AI Communication (2008), IEEE Intelligent System (2007), Interbational Jourrnal of Electronic Commerece (2006) và ACM Transactions of Inrfomation Systems (2004).

Trong đề tài luận văn này, chúng tôi xây dựng hệ thống gợi ý các câu hỏi tương tự trong quá trình ôn thi trắc nghiệm online bằng phương pháp gợi ý dựa trên nội dung để đưa ra gợi ý.

##### 2.2.1. Quy trình xây dựng hệ thống gợi ý

Để xây dựng một hệ thống gợi ý chúng ta cần phải thực hiện các bước như thu thập dữ liệu, chuẩn hóa dữ liệu, tính độ tương tự và đưa ra gợi ý, cuối cùng là đánh giá hệ thống.

Evaluate

Data

Standardized Data

Predict

Train Model

Hình 4: Quy trình xây dụng hệ thống gợi ý

Thu thập dữ liệu: Tùy vào bài toán mà chúng ta cần phải có chiến thuật thu thập dữ liệu. Đây là một bước có vai trò quan trọng, nếu thu thập thông tin không đầy đủ hoặc không chính xác, hệ thống sẽ đưa ra các gợi ý thiếu chính xác.

Chuẩn hóa dữ liệu: để có thể thực hiện quá trình tính độ tương tự cũng như dự đoán chúng ta phải chuẩn hóa dữ liệu về dạng vector số, máy tính không thể thực hiện tính toán trực tiếp thông qua các câu văn bản thuần túy mà phải là các con số.

Tính độ tương tự và dự đoán: dữ liệu trải qua quá trình tiền xử lý, tách từ và vector hóa văn bản, từ dữ liệu đã được xử lý chúng ta xây dựng ma trận độ tương tự của các câu hỏi. Cuối cùng lựa chọn ra top và bottom N item thỏa mãn điều kiện để gợi ý cho người dùng.

Đánh giá hệ thống: muốn biết được hệ thống có tốt hay không chúng ta cần phải thực hiện việc đánh giá hệ thống gợi ý. Có nhiều phương pháp để đánh giá nhưng chúng ta cần chọn ra phương pháp phù hợp với hệ thống hiện tại.

##### 2.2.2. Phương pháp xây dựng hệ thống gợi dựa trên nội dung

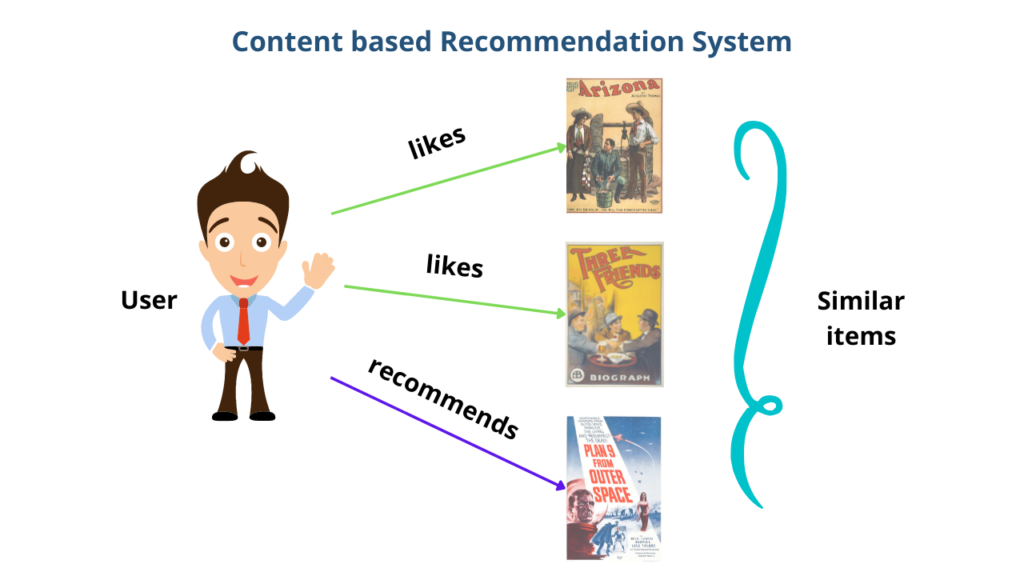
Lọc cộng tác (collaborative filtering) là phương pháp phân tích dữ liệu người dùng để tìm ra điểm tương tự giữa các người dùng có cùng sở thích, thói quen. Phương pháp này hoạt động bằng các xây dựng và phân tích dựa trên ma trận giữa người dùng(user) và mục dữ liệu(items)[7].

Hạn chế của phương pháp lọc cộng tác:

* Dữ liệu thưa.
* Sản phẩm mới.
* Gợi ý thiếu chính xác vì không quan tâm đến dặc điểm của sản phẩm mà người dùng đã chọn.
* Khả năng giải thích cho kết quả gợi ý kém.
* Thiếu minh bạch.

Hệ thống gợi ý dựa trên nội dung với ý tưởng là gợi ý các sản phẩm cho người dùng X tương tự với các **sản phẩm được đánh giá cao** cũng bởi những **người dùng X này**[7].

Tìm kiếm các mục dữ liệu tương tự với mục dữ liệu mà **người dùng** đã mua / chọn lựa / xem / đánh giá trước đó **dựa trên thuộc tính** của các mục dữ liệu (ví dụ như màu sắc, giá cả, mô tả mục dữ liệu..) hoặc dựa trên **hồ sơ người dùng**[7].



Ví dụ, người dùng sẽ thích quyển sách thứ nhất và quyển sách thứ 2, thông tin này được sử dụng để thực hiện việc tính toán và tìm ra quyển sách tương tự với hai quyển sách mà người dùng đã thích rồi đưa ra gợi ý cho người dùng là quyển sách thứ 3 ở trên.

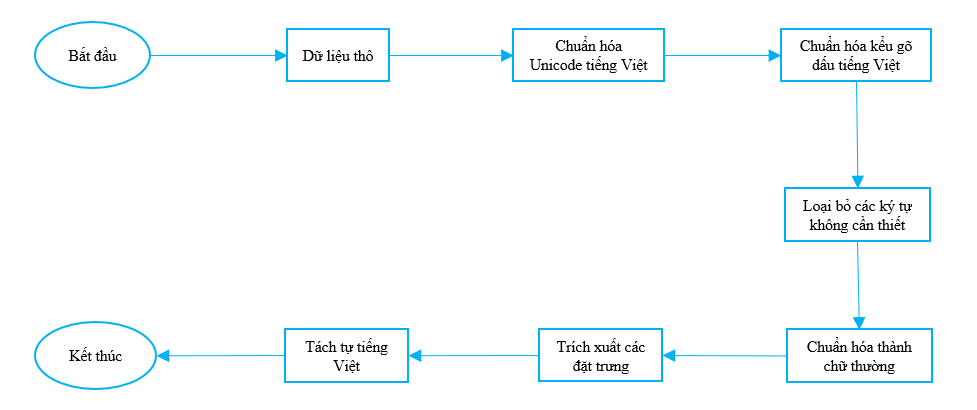
Dữ liệu xây dựng gợi ý dựa trên item description. Chia thành hai loại:

* Dữ liệu có cấu trúc:
  + Trang phục: loại vải, nhãn hiệu, giá cả…
  + Thức ăn: thể loại, thương hiệu, dinh dưỡng…
* Dữ liệu phi cấu trúc:
  + Nội dung bình luận.
  + Nội dung câu hỏi.

Trong đề tài này với dữ liệu là các câu hỏi thuộc kiểu dữ liệu phi cấu trúc chúng ta sẽ biểu diễn mỗi câu hỏi dưới dạng một vector, sau đó tính độ tương tự giữa các câu hỏi và đưa ra các gợi ý có độ tương tự cao cho người dùng.

##### 2.2.3 Tiền xử lý văn bản tiếng Việt

Dữ liệu sau khi được thu thập là dữ liệu thô chưa sử dụng để thực hiện việc tính toán hoặc khai phá. Vì thế cần thực hiện việc tiền xử lý dữ liệu thô, bước tiền xử lý dữ liệu này ảnh hưởng đến việc tính toán và đánh giá.



Dữ liệu thô được tiền xử lý thông qua các bước như chuẩn hóa Unicode tiếng Việt, chuẩn hóa kiểu gõ dấu tiếng Việt, loại bỏ các ký tự không cần thiết, chuẩn hóa thành chữ thường, trích xuất các đặc trung và cuối cùng là tách từ tiếng Việt. Trải qua các bước tiền xử lý thì dữ liệu có thể đưa vào để tính toán hoặc huấn luyện cũng như dự đoán.

a. Chuẩn hóa Unicode tiếng Việt

Unicode (hay gọi là mã thống nhất, mã đơn nhất) là bộ mã chuẩn quốc tế được thiết kế để dùng làm bộ mã duy nhất cho tất cả các ngôn ngữ khác nhau trên cả thế giới[9]. Hiện nay có 2 bảng Unicode để mã hóa chữ tiếng Việt là Unicode dựng sẵn và Unicode tổ hợp[8].

Với Unicode dựng sẵn: xem ký tự trong một từ như hợp chữ + dấu mũ + dấu thanh như một ký tự duy nhất.

Ví dụ: chữ “Việt” gồm 4 ký tự V, i, ệ, t.

Với Unicode tổ hợp: xem ký tự trong một từ như hợp chữ + dấu mũ như một ký tự duy nhất và dấu thanh được tính như một ký tự riêng.

Ví dụ: chữ “Việt” gồm 5 ký tự V, i, ê, t và dấu nặng.

Trong các câu hỏi có thể xuất hiện loại mã Unicode dựng sẵn hoặc mã Unicode tổ hợp nếu không được chuẩn hóa thì trong lúc tính toán hai từ nhìn hoàn toàn giống nhau nhưng thật ra máy tính sẽ hiểu 2 từ khác nhau.Vì lý do đó, chúng ta sẽ đưa tất cả về một kiểu mã hóa Unicode, trong đề tài chúng tôi sử dụng chuẩn Unicode phổ biến hơn là Unicode dựng sẵn.

b. Chuẩn hóa kiểu gõ dấu tiếng Việt

Hiện nay có hai quan điểm về cách đặt dấu thanh thường được gọi là “kiểu cũ” và “kiểu mới”. Trong đời sống, ví dụ như trong các bộ gõ tiếng Việt, hiện tại vẫn còn tồn tại hai cách đặt dấu thanh. Ví dụ từ “hóa” là cách đặt dấu thanh cũ và từ “hoá” là cách đặt dấu thanh mới[10].

Với các kiểu gõ khác nhau thì mắt thường chúng ta có thể thấy được sự khác biệt rõ ràng. Cũng giống như chuẩn hóa Unicode tiếng Việt chúng ta cần đưa các từ về kiểu gõ dấu cũ.

c. Loại bỏ các ký tự không cần thiết và chuẩn hóa thành chữ thường

Trong các câu hỏi trong bộ đề có thể xuất hiện các ký tự như “*!"#$%&'()\*+,-./:;<=>?@[\]^\_`{|}~*” các ký tự này không có tác dụng cho việc trong quá trình tính độ tương tự giữa các câu.

Việc đưa dữ liệu về chữ viết thường là một việc làm rất cần thiết. Chúng ta cần phải chuẩn hóa các từ trong câu thành dạng chữ thường, đưa về chung một dạng chữ thường giúp giảm số lượng đặt trưng vì máy tính phân biệt chữ hoa và chữ thường và tăng đọ chính xác hơn cho quá trình tính toán.

d. Loại bỏ các từ dừng

Từ dừng là các từ được lọc ra trước hoặc sau quá trình xử lý dữ liệu văn bản. Mặc dù những từ dừng được coi là các từ xuất hiện nhiều trong một ngôn ngữ, tuy nhiên chúng là các từ không có tác dụng cho việc phân loại văn bản.

Các từ dừng thường là các từ nối như và, của, là, có…, danh sách các từ dừng được xây dựng từ bộ dữ liệu văn bản rất lớn, trong đề tài luận văn này chúng tôi sử dụng danh sách từ dừng của tác giả Van-Duyet Le. Danh sách từ dùng được lưu trong tập tin txt, chúng ta chỉ việc lặp qua các từ trong một câu hỏi và loại bỏ các từ có trong danh sách từ dừng.

e. Trích xuất các đặt trưng với CountVectorizer

CountVectorizer là một công cụ được cung cấp bởi thư viện scikit-learning. Công cụ này được sử dụng để chuyển một câu thành một vector số dựa vào số lần xuất hiện của mỗi từ trong một câu[11].

CountVectorizer tạo ra một ma trận trong đó mỗi cột sẽ biễu diễn cho một từ trong toàn bộ dữ liệu. Mỗi dòng đại diện cho một câu trong toàn bộ dữ liệu, trong một dòng thì mỗi cột là số lần xuất hiện của từ đó trong câu, nếu các từ không xuất hiện trong câu thì mang giá trị là 0.

Ta có các câu như sau:

A: Hôm nay có bài tập về nhà.

B: Bài tập về nhà hôm nay rất khó.

C: Có nhiều bài tập về nhà trong hôm nay.

CounterVectorizer sẽ tạo ra một bộ từ vựng và một ma trận.

Vocabulary:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| index | 2 | 4 | 1 | 0 | 8 | 10 | 5 | 7 | 3 | 6 | 9 |
|  | hôm | nay | có | bài | tập | về | nhà | rất | khó | nhiều | trong |

Matrix:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| A | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| B | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 |
| C | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 |

f. Trích xuất các đặt trưng với TF-IDF (Term Frequency-Inverse Document Frequency)

TF-IDF là viết tắt của Thuật ngữ Tần suất Đảo ngược Tài liệu Tần suất của các từ trong văn bản. Là một kỹ thuật sử dụng trong việc khai phá dự liệu văn bản. Trọng số này được sử dụng để đánh giá tầm quan trọng của một từ trong văn bản. Giá trị cao thể hiện mức độ quan trọng cao và nó phụ thuộc vào số lần xuất hiện của từ đó trong văn bản, nhưng bù lại với tần số của từ đó trong tập dữ liệu[12].

TF-IDF(t, d, D) = TF(t, d) x IDF(t, D)

TF (tần suất xuất hiện của từ) là số lần xuất hiện của một từ trong trong một văn bản cho trước. Vì các văn bản có độ dài không giống nhau nên một từ có thể xuất hiện nhiều lần trong những văn bản có độ dài hơn và xuất hiện ít hơn trong những văn bản ngắn hơn. Do đó, tổng số lần xuất hiện của từ trong một văn bản được chia cho tổng số từ trong văn bản đó.

TF(t, d) = ( số lần xuất hiện từ t trong văn bản d) / (tổng số từ trong văn bản d).

IDF (nghịch đảo tần suất của văn bản) là phương pháp tính toán tầm quan trọng của một từ, nếu chỉ dựa vào tần suất xuất hiện của từ thì các từ có độ quan trọng như nhau. Trong đó, có một số từ thường xuyên xuất hiện trong các câu và có độ quan trọng không cao, nên phải giảm độ quan trọng của các từ đó.

IDF(t, D) = log(tổng số văn bản trong tập D / số văn bản có chứa từ t)

Ví dụ: Có 2 câu hỏi A, B, C trong mỗi câu có số lượng các từ “hôm”, “nay”, “có”, “bài”, “tập”, “về”, “nhà”, “rất”, “khó”, “nhiều”, “trong” được thống kê bên dưới:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | hôm | nay | có | bài | tập | về | nhà | rất | khó | nhiều | trong |
| A | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| B | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 |
| C | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 |

Tính giá trị TF: Lấy số lần xuất hiện của một từ trong câu hỏi chia cho tổng các từ trong câu hỏi đó.

TF của từ “hôm” cho câu hỏi A: tf(“hôm”, A) = 1 / 7 = 0.143

Tương tự sẽ tính cho các từ còn lại với các câu hỏi A, B, C ta có:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | hôm | nay | có | bài | tập | về | nhà | rất | khó | nhiều | trong |
| A | 0.143 | 0.143 | 0.143 | 0.143 | 0.143 | 0.143 | 0.143 | 0 | 0 | 0 | 0 |
| B | 0.125 | 0.125 | 0 | 0.125 | 0.125 | 0.125 | 0.125 | 0.125 | 0.125 | 0 | 0 |
| C | 0.111 | 0.111 | 0.111 | 0.111 | 0.111 | 0.111 | 0.111 | 0 | 0 | 0.111 | 0.111 |

Tính giá trị IDF: Lấy tổng số câu hỏi chia cho tổng số câu hỏi chứa từ nào đó.

IDF của từ “hôm” với 3 câu hỏi: IDF(“hôm”, D) = log(3 / 3) = 0

Tương tự tính cho các từ còn lại.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| hôm | nay | có | bài | tập | về | nhà | rất | khó | nhiều | trong |
| 0 | 0 | 0.176 | 0 | 0 | 0 | 0 | 0.477 | 0.477 | 0.477 | 0.477 |

Tính giá trị TF-IDF: Lấy giá trị TF của từng câu nhân với giá trị của các từ trong bảng IDF.

Trọng số của từ “hôm” trong câu hỏi A: TF-IDF(“hôm”, A, D) = 0.143 \* 0 = 0

Tương tự tính các từ còn lại trong các câu hỏi A, B, C ta có:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | hôm | nay | có | bài | tập | về | nhà | rất | khó | nhiều | trong |
| A | 0 | 0 | 0.025 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| B | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.059 | 0.059 | 0 | 0 |
| C | 0 | 0 | 0.019 | 0 | 0 | 0 | 0 | 0 | 0 | 0.052 | 0.052 |

g. Tách từ tiếng Việt

Tách từ là một quá trình xử lý nhằm mục đích xác định ranh giới của các từ trong văn bản, cũng có thể hiểu đơn giản rằng tách từ là quá trình xác định các từ đơn, từ ghép… trong câu. Đối với xử lý ngôn ngữ tự nhiên, có thể xác định cấu trúc ngữ pháp của câu, xác định từ loại của một từ trong câu, yêu cầu đặt ra là phải xác định được đâu là từ trong câu[13].

Hiện này có nhiều thư viện mã nguồn mở hỗ trợ việc tách từ tiếng Việt, trong đề tài này chúng tôi sử dụng thư viện Pyvi và VnCoreNLP.

Thư viện Pyvi (Python Vietnameese Toolkit) là một thư viện mã nguồn mở của tác giả Viet Trung Tran, Vu Vanh và Hieu Nguyen Van phát triển. Huấn luyện mô hình mã hóa trên tập dữ liệu lớn và đó chỉ số f1 là 0.985 với thuật toán trường ngẫu nhiên có điều kiện[14].

VnCoreNLP là một hệ thống cung cấp chú thích NLP nhanh và chính xác cho tiêng Việt, cung cấp các chú thích ngôn ngữ phong phú thông qua các thành phần NLP chính của phân đoạn từ, gắn thẻ POS, nhận dạng đối tượng có tên(NER) và phân tích cú pháp phụ thuộc. Là thư viện mã nguồn mở được phát triển bởi các tác giả Dat Quoc Nguyen, Thanh Vu[15].

Ví dụ ta có câu: “Hôm nay Nguyễn Tấn Pil có bài tập về nhà”.

Pyvi: “Hôm\_nay Nguyễn\_Tấn Pil có bài\_tập về nhà”

VnCoreNLP: “Hôm\_nay Nguyễn\_Tấn\_Pil có bài\_tập về nhà”

##### 2.2.4 Độ tương tự

Tương tự là một thuật ngữ bao trùm lớn bao hàm các yếu tố điểm số và thước đo để đánh giá sự khác biệt giữa các loại dữ liệu khác nhau. Trên thực tế, đọ tương tự đề cập đến nhiều thứ có thể bao gồm một hướng duy nhất trong đề tài này. Đối với đề tài này chúng ta sẽ tìm hiểu một kiểu tương tự liên quan tới phân tích văn bản. Độ tương tự này được đề cập trong đề tài này là lấy các đặc điểm dựa trên một tập hợp các tài liệu và đo mức độ tương tương đồng giữa các tài liệu trên khoảng cách của chúng với nhau trong không gian Descartes. Cụ thể, phương pháp này xác định sự khác biệt giữa các văn bản từ số lượng từ của chúng[16].

Ví dụ ta có:

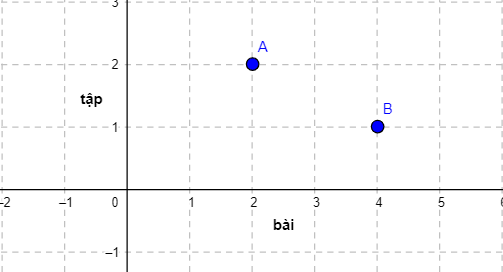
A: “Hôm nay có bài tập về nhà, thật nhiều bài tập.”

B: “Bài tập cô cho về nhà có 5 bài, 3 bài đầu tiên cơ bản và 2 bài còn lại là nâng cao.”

Trong ví dụng này là 2 câu trong tập dữ liệu, hai câu này có nhiều đặc điểm, thuộc tính của mẫu dữ liệu mà chúng ta có thể đo lường và biểu diễn bằng số. Chúng ta sẽ lấy từ “bài” và “tập” và đếm số lần xuất hiện của chúng trong mỗi câu.

|  |  |  |
| --- | --- | --- |
|  | bài | tập |
| A | 2 | 2 |
| B | 4 | 1 |

Có nhiều cách khác nhau để biểu diễn dữ liệu ở trên. Một trong những hệ tọa độ có tuổi đời và phổ biến nhất là hệ tọa độ Descartes, hệ tọa độ này cho phép chúng ta biểu diễn các đối tượng số dưới dạng tọa độ (điểm), thường là không gian hai chiều. Bảng dữ liệu ở trên có thể biểu diễn dưới dạng hệ tọa độ như sau:

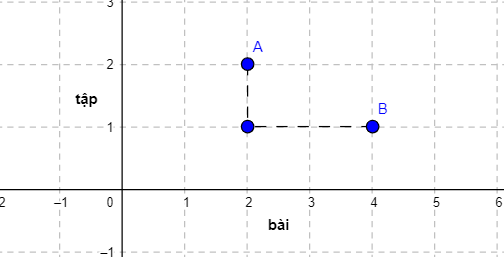


Trong biểu đồ này, A và B được biểu thị dưới dạng dữ liệu điểm phân bố dọc theo hai trục x (hoành) và y(tung). Trục x nằm ngang biểu thị các giá trị cho từ “bài” và trục y nằm dọc biểu thị có giá trị của từ “tập”. Cách biểu diễn này cho chúng ta hình dung được mối quan hệ của các điểm dữ liệu trong không gian 2 chiều dựa trên các đặc điểm của hai điểm này.

Để xác định hai điểm này có liên quan với nhau hay không, chúng ta có thể đặt ra vấn đề là hai điểm trên hệ tọa độ xa nhau hay gần nhau? Để trả lời cho câu hỏi này chúng ta sẽ tính khoảng cách giữa hai điểm này. Chúng ta có thể tính khoảng cách bằng cách vẽ một đường thẳng nối hai điểm trong hệ tọa độ và tính độ dài của nó.

a. Khoảng cách Manhattan

Khoảng cách Manhattan còn được gọi là khoảng cách L1 hay khoảng cách trong thành phố là khoảng cách giữa hai điểm trong không gian Euclid với hệ tọa độ Descartes. Đại lượng này được tính bằng tổng chiều dài của hình chiếu của đường thẳng nói hai điểm này trong hệ trục tọa độ Descartes.



Với hai điểm bất kỳ có giá trị (x1, y1) và (x2, y2), khoảng cách Mahattan được tính bằng biểu thức sau:

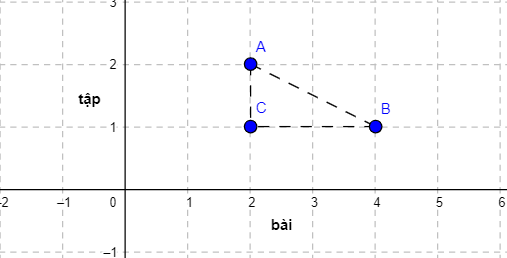
D = |x1 – x2| + |y1 – y2|

DA,B = |2 – 4| + |2 – 1| = 2 + 1 = 3

Vậy khoảng cách Manhattan của A và B trong hệ trục tọa độ Decartes là 3.

b. Khoảng cách Euclide

Khoảng cách Euclide là dạng thông dụng nhất của khoảng cách Minkowski, khi nói về khoảng cách thường đề cập tới khoảng cách Euclide. Khoảng cách Euclide áp dụng định lý Pythagoras để tính khoảng cách giữa 2 diểm như sau[16]:



Để đo khoảng cách của đường thẳng AB chúng ta có thể xem đường thằng AB là cạnh huyền của một tam giác vuông ACB hai cạnh còn lại là cạnh bình thường của tam giác.

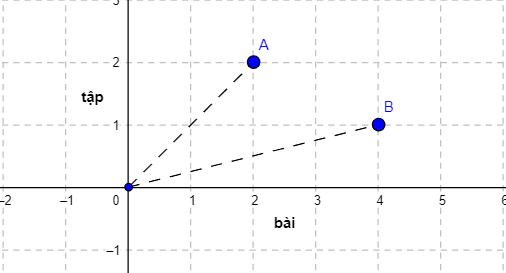
D = sqrt(AC2 + CB2) = sqrt((x2 – x1)2 + (y2 – y1)2)

DAB = sqrt((2 - 4)2 + (2 - 1)2) = sqrt(5) = 2.23.

Vậy khoảng cách Euclide của A và B trong hệ trục tọa độ Decartes là 2.23. Kết quả khoảng cách Euclide nhỏ hơn một ít so với khoảng cách Manhattan, mỗi độ đo khoảng cách cho ta biết mối quan hệ giữa hai điểm trên hệ tọa độ, nhưng mỗi độ đo cho chúng ta biết sự khác biệt về khoảng cách giữa có độ đo là khác nhau. Khoảng cách giữa hai điểm có thể khác nhau tùy thuộc vào cách chúng ta chọn độ đo khoảng cách ngay từ lúc bắt đầu tính toán.

c. Độ tương tự Cosine

Độ tương tự Cosine tính toán độ tương tự của hai vector, có nghĩa là tính độ lớn cosin góc của hai vector. Độ tương tự xác định hai vector có hướng gần như cùng một hướng hay không. Vì thế, nếu góc giữa hai vecotr là 0 thì độ tương tự cosine là cao nhất(1) ngược lại góc giữa hai vector càng lớn thì độ tương tự càng thấp[16].



Độ tương tự cosin rất khác với hai độ đo khoảng cách nói trên, phép đo này quan tâm đến hướng của hai điểm trong không gian hơn là khoảng cách thật sự của hai điểm với nhau.

Độ tương tự của hai điểm của hai điểm A, B trong hệ tọa độ là góc được tạo hai đường thằng OA và OB. Nếu hai điểm A, B cách nhau 90 độ, nghĩa là một trong hai nằm trên trục x(hoành) và y(tung) của biểu đồ càng xa thì càng tốt trong góc phần tư của biểu đồ này, thì độ tương tự cosin của A và B bằng không do cos(90) là 0. Nếu góc tạo bởi OA và OB là 0 độ thì có nghĩa là A và B nằm trên cùng một đường thẳng thì độ tương tự của A và B là 1 vì cos(0) là 1.

Cos(theta) = (x1x2 + y1y2) / sqrt(x12 + y12).sqrt(x22 + y22)

Cos(OA, OB) = (2 x 4 + 2 x 1) / sqrt(22 + 22) x sqrt(42 + 12)

= 10 / (sqrt(8) x sqrt(17)) = 0.85.

Ta tính ra được độ tương tự của A và B là 0.85 khá là cao, gần với giá trị 1, nhìn vào hệ tọa độ ta có thể thấy rằng góc được tạo thành giữa OA và OB khá là nhỏ. Vì A và B có hướng khá giống nhau và gần nhau nên độ tương tự cosin của chúng khá là cao. Nói một cách khác thì hai văn bản đã cho khá giống nhau.

Chúng ta có thể thấy rằng 0.85 là độ tương tự chứ không phải là khoảng cách, giá trị cao nhất là 1, nghĩa là hai điểm gần nhau nhất và ngược lại giá trị thấp nhất (xa nhất) là 0. Điều này khác với độ đo khoảng cách Euclide và Manhatan thì các giá trị nhỏ nhất nói lên rằng các điểm gần nhau nhất. Vì vấn đề đó, chúng ta có khoảng cách cosine được tính bằng cách lấy 1 trừ đi độ tương tự.

DA,B = 1 – 0.85 = 0.15

Khoảng cách cosin có giá trị thấp nhất là 0 và cao nhất là 1 dễ dàng thực hiện việc tính toán cũng như so sanh hơn độ đo Euclide và Manhattan.

d. Thước đo khoảng cách phù hợp

Như đã nói ở trên, các thước đo này hoàn toàn không giống nhau và chúng mang lại kết quả khác nhau. Tuy nhiên chúng điều là các thước đo khoảng cách, dựa vào các đo khoảng cách có thể ảnh hưởng đến lớn đến kết quả cuối cùng của chúng ta.

Trong trường hợp này chúng ta cần phải xác định là làm thế nào để đo mối quan hệ giữa hai điểm? Để trả lời cho câu hỏi này chúng ta cần dựa vào bản chất của dữ liệu mà chúng ta đang phân tích và tính toán.

##### 2.2.5. Phương pháp đánh giá

**K-fold**: là phương pháp đánh giá chia tập dữ liệu thành k phần (fold) bằng nhau, lặp lại k lần, mỗi lần sử dụng k-1 folds để học và 1 fold để kiểm tra và sau đó tính trung bình của k lần kiểm tra[7].

**Given-N**: Thường được sử dụng để đánh giá trong các lĩnh vực thương mại điện tử. Là một mở rộng của phương pháp k-fold thực hiện trên từng giao dịch thay vì toàn bộ giao dịch. Giao dịch phải có ít nhất **N + 1** dữ liệu thì mới có thể thực hiện phương pháp này. Chia danh sách sản phẩm trong items thành hai phần: một tập được gọi là <Given> và một tập là <Test>, so sánh các items gợi ý với những items mong muốn, độ chính xác của hệ thống sẽ tăng lên một đơn vị nếu các items gợi ý tồn tại trong các items mong muốn ngược lại thì không tăng[7].

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Items | Items mong muốn | Items gợi ý | Độ chính xác |
| 1 | A, B, C, D | A, C | B, C | +1 |
| 2 | B, D, E, F | D, B, F | A, G | 0 |
| 3 | A, C, D, G | D, G | B, G | +1 |

Với dòng dữ liệu 1 có danh sách các items là A, B, C và D, có các items mong muốn là A, C, sau khi tính toán ta thu được các items gợi ý là B và C. Do C tồn tại trong các items mong muốn nên đọ chính xác hệ thống gợi ý tăng thêm một. Với dòng dữ liệu thứ 2 độ chính xác không tăng lên do các items gợi ý A, G không tồn tại trong danh sách mong muốn thứ 2 là D, B, F.

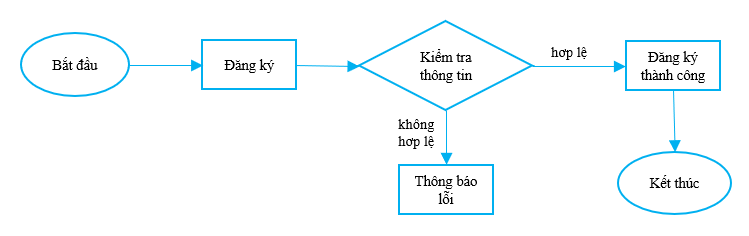
**All-but-one**: Là trường hợp đặc biệt của Given-N, phần <Given> là **N-1**, phần test luôn là 1. Số lượng các items phải ít nhất là một. Ưu điểm của phương pháp “All But One” là cho phép đánh giá các items có kích thước lớn hơn một sản phẩm.

## CHƯƠNG 2 - THIẾT KẾ VÀ CÀI ĐẶT

### 1. Thiết kế hệ thống

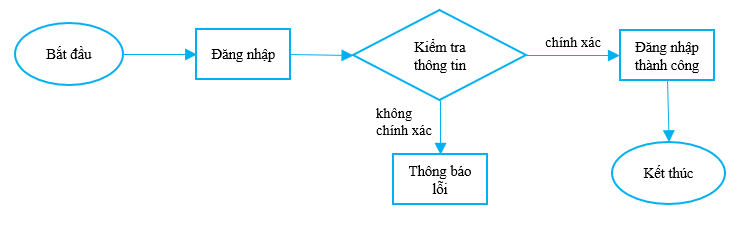
#### 1.1 Sơ đồ tổng quát

Để người dùng cho thể tham gia vào website thi trắc nghiệm online thì cần phải tạo một tài khoản, quy trình tạo một tài khoản của website:



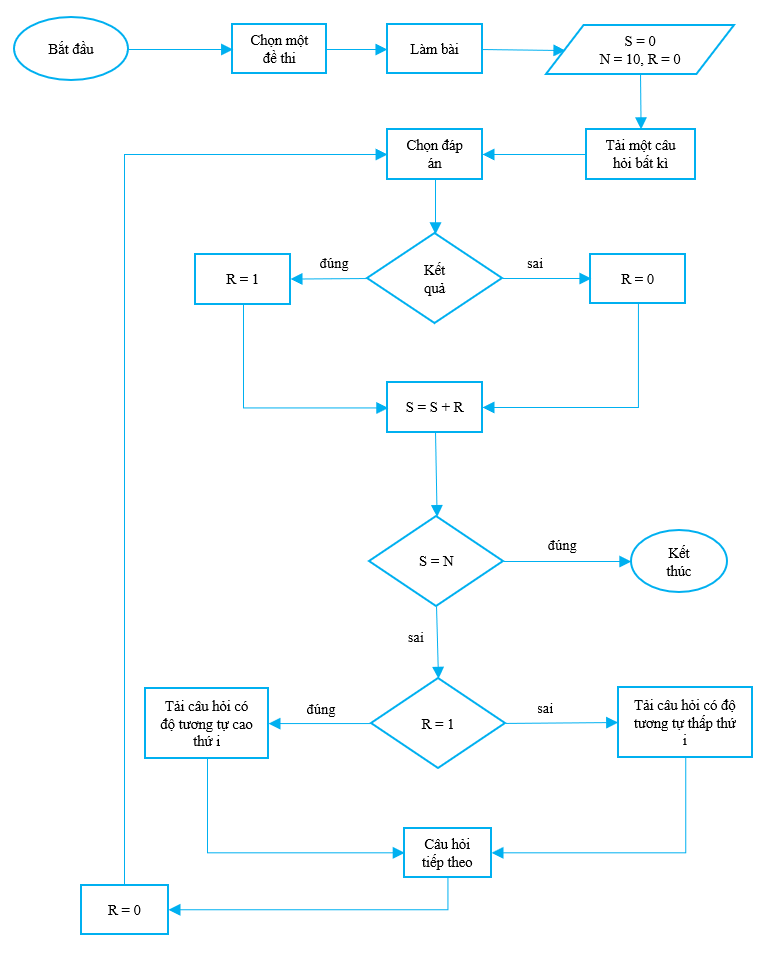
Sau khi người dùng cung cấp các thông tin cơ bản để đăng ký tài khoản, hệ thống sẽ kiểm tra thông tin, nếu tài khoản đã tồn tại hoặc thông tin không hợp lệ thì sẽ thông báo lỗi cho người dùng ngược lại thì đăng ký tài khoản cho người dùng.

Đối với người dùng đã có tài khoản thì sẽ thực hiện việc đăng nhập vào hệ thống, quy trình đăng nhập của website:



Khi người dùng cung cấp thông tin đăng nhập thì hệ thống sẽ kiểm tra thông tin tài khoản, nếu thông tin đăng nhập chính xác thì hoàn thành quá trình đăng nhập vào hệ thống.

Khi người dùng đã đăng nhập vào hệ thống thì có thể tham gia vào các đề thi, đề ôn đối với học viên, có thể tạo đề thi, tạo lớp học với người dùng với vai trò là giáo viên, giảng viên, người giảng dạy. Khi tham gia vào một đề thì, thì quy trình của hệ thống gợi ý:



Sau khi người dùng chọn một đề thi thì có thể bắt đầu làm bài, đối với lưu đồ bên trên chúng ta có:

S là score số điểm của người dùng, khởi tạo bằng 0.

N là số lượng câu hỏi, trong lưu đồ này lấy giá trị bằng 10.

R là số điểm của một câu hỏi, mang giá trị bằng 1 nếu người dùng trả lời đúng một câu hỏi hoặc bằng 0 nếu người dùng trả lời sai.

Khi bắt làm bài, trong tất cả câu hỏi ta chọn ngẫu nhiên một câu hỏi để người dùng thực hiện, người dùng sẽ chọn một trong các đáp án trong bốn đáp án của câu hỏi tương tự, nếu người dùng trả lời chính xác thì số điểm của câu hỏi đó là 1 ngược lại là 0. Số điểm của người dùng sẽ cộng với số điểm của câu hỏi đó, có thể là 1 hoặc 0 tùy vào kết quả của câu hỏi.

Sau khi tính số điểm của người dùng thì so sánh với số lượng câu hỏi cần phải trả lời đúng, nếu số điểm của người dùng bằng với số lượng câu hỏi cần trả lời đúng thì kết thúc bài làm, ngược lại thì dựa vào kết quả của câu hỏi đó, tải một câu hỏi có độ tương tự cao hay thấp thứ i.

|  |  |  |  |
| --- | --- | --- | --- |
| Id | Câu hỏi | Danh sách câu hỏi có độ tương tự cao | Danh sách câu hỏi có độ tương tự thấp |
| 1 | Câu hỏi thứ 1 | 2, 3, 6, 7, 9 | 4, 12, 21, 24, 54 |
| 2 | Câu hỏi thứ 2 | 1, 33, 44, 56, 72 | 11, 16, 20, 29, 99 |
| 3 | Câu hỏi thứ 3 | 1, 5, 10, 22, 68 | 13, 31, 42, 56, 79 |
| 4 | Câu hỏi thứ 4 | 38, 41, 52, 61, 80 | 15, 34, 48, 58, 67 |
| 5 | Câu hỏi thứ 5 | 3, 16, 36, 44, 55 | 23, 25, 69, 74, 81 |
| … | … | … | … |
| n | Câu hỏi thứ n | … | … |

Ví dụ trong lưu đồ số lượng câu hỏi mà người dùng cần phải trả lời đúng là 10 câu hỏi, tức là số điểm của người dùng phải là 10 thì mới hoàn thành quá trình làm bài. Lúc đầu giá trị của biến i(0, 1, 2, 3, 4) trong danh sách câu hỏi có độ tương tự cao và độ tương tự thấp là phần tử thứ nhất trong danh sách, danh sách có độ dài là 5 cho mỗi câu hỏi.

Đầu tiên, câu hỏi có id là 1 sẽ được tải lên, nếu người dùng chọn đáp án chính xác và số điểm nhỏ đạt được nhỏ hơn 10 thì tải câu hỏi tiếp theo. Khi người dùng trả lời chính xác thì sẽ lấy id của câu hỏi trong danh sách câu hỏi có độ tương tự cao, với lần thứ nhất thì lấy phần tử thứ 0 và tăng i lên một giá trị, nếu i là chỉ số của phần tử cuối cùng thì gán i lại giá trị 0 và cứ thế tiếp tục cho đến khi hoàn thành.

Với trường hợp người dùng trả lời không chính xác, giống như ở trên chúng ta sẽ lấy phần tử thứ 0 trong danh sách độ tương tự thấp với lần đầu tiên và tăng giá trị i thêm 1, nếu i là chỉ số của phần tử cuối cùng thì gán lại là 0. Thực hiện cho đến khi hoàn thành.

#### 1.2 Cấu trúc của website

.

### 2. Cài đặt giải thuật

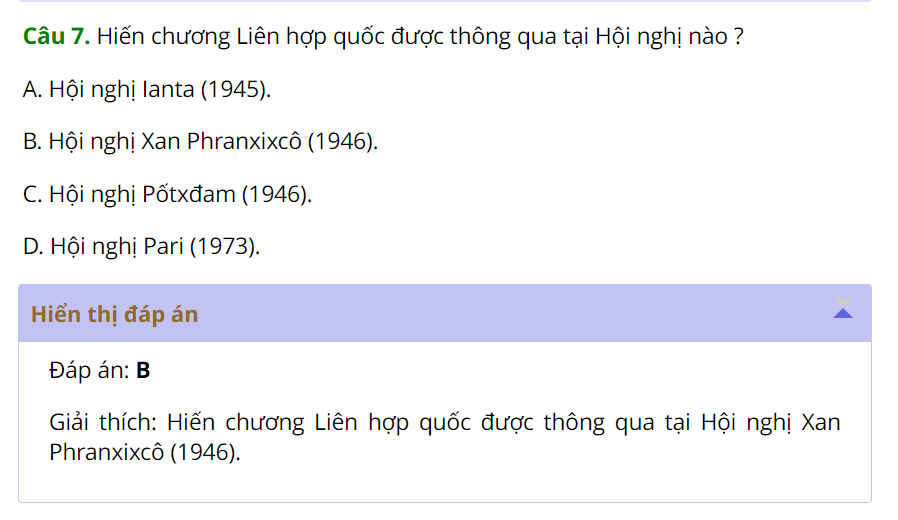
#### 2.1 Thu thập dữ liệu

Tập dữ liệu được sử dụng để thực hiện trong đề tài này là các câu hỏi trắc nghiệm lịch sử lớp 12 có đáp án tại <https://vietjack.com/bai-tap-trac-nghiem-lich-su-12/>.

Quá trình thu thập dữ liệu được thực hiện bằng phương pháp **Web crawling**, cho phép chúng ta tự động trích xuất các thông tin từ các trang web mà chúng ta cần và đưa về một định dạng mong muốn.

BeautifulSoup là một thư viện Python dùng nó để lấy dữ liệu từ các phần từ HTML và XML. Nó hoạt động cùa mới các parser (trình phân tích cú pháp) cung cấp cho chúng ta các cách để điều hướng, tìm kiếm và chỉnh sửa trong parse tree (cây phân tích được tạo từ parser).

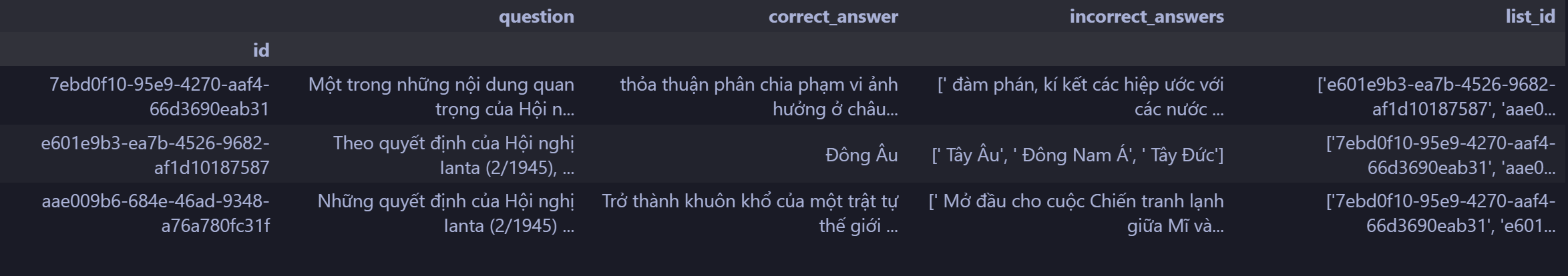
Sau khi cài đặt thư viện BeautifulSoup chúng ta tiến hành cào dữ liệu, dữ liệu là các câu hỏi, các câu trả lời và câu trả lời chính xác được thực hiện trên 52 bài ôn tập trắc nghiệm trên VietJack. Kết quả sau cuối cùng thu được hơn 1300 câu hỏi.



Trong mỗi câu hỏi sẽ có nội dung câu hỏi và các đáp án là A, B, C, D dưới mỗi câu hỏi sẽ có đáp án của câu hỏi đó. Trong quá trình cào dữ liệu, đầu tiên sẽ lấy nội dung câu hỏi, tiếp đến lấy tất cả các câu trả lời và lưu lại key cho mỗi đáp án, cuối cùng sẽ sử dụng đáp án chính xác chia danh sách thành hai phần là correct\_answer và incorrect\_answers.

#### 2.2 Sơ lượt về dữ liệu

Tập dữ liệu được thu thập từ website VietJack có 1396 mẫu tin, mỗi mẫu tin có các trường như question, level, correctAnswer, inCorrectAnswers và listIds. Dữ liệu sẽ được import vào cơ sở dữ liệu, dữ liệu sau khi import vào cơ sở dữ liệu thì sẽ có thêm trường id để xác định câu hỏi trong lúc đưa ra gợi ý.



Trong đó:

Id là định danh của một câu hỏi trong cơ sở dữ liệu.

Question là nội dung câu hỏi.

Correct\_answer là câu trả lời chính xác.

Incorrect\_answers là những câu trả lời không chính xác.

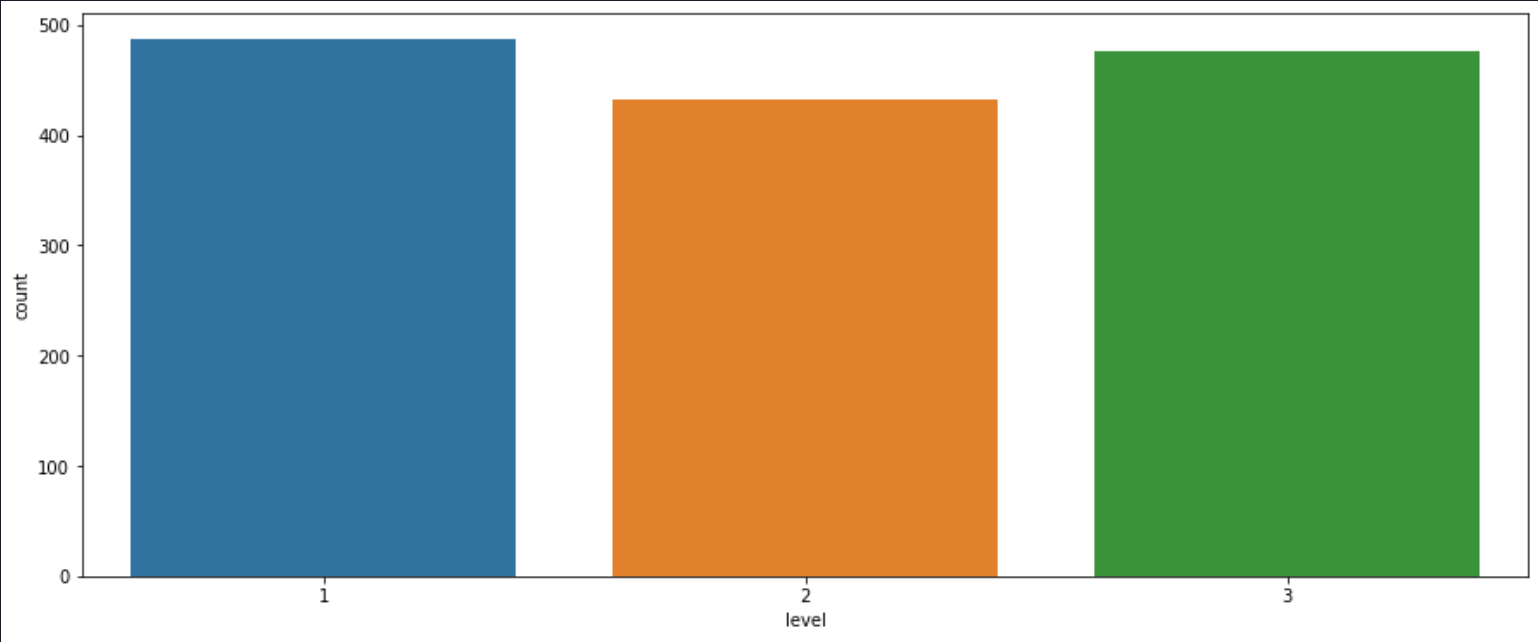
List\_ids là danh sách các id câu hỏi tương tự.

Trong 1396 mẫu tin có số lượng câu hỏi theo từng level như sau:

Easy level tương đương với số 1 trong tập dữ liệu.

Medium level tương đương với số 2 trong tập dữ liệu.

Hard level tương đương với số 3 trong tập dữ liệu.



Ta thấy số lượng của các câu hỏi theo từng level tương đối cân bằng.

#### 2.3 Tiền xử lý dữ liệu

Đối với dữ liệu mới thu thập là văn bản thô không thích hợp để thực hiện việc tính toán, dữ liệu phải đưa qua bước tiền xử lý. Quá trình tiền xử lý dữ liệu là một bước quan trọng trong việc khai phá dữ liệu văn bản, nó ảnh hưởng trực tiếp đến quá trình tính toán và đánh giá vì thế đây là bước thật sự cần thiết.

#### 2.4 Trích xuất các đặt trưng

Sử dụng CountVectorizer và TfidfVectorizer của thư viện sklearn để chuyển đổi các câu hỏi trong tập dữ liệu thành các vector số để thực hiện việc tính toán.

Ngoài ra còn sử dụng còn sử dụng tokenize bằng bộ Tokenizer của phoBert để trích xuất các đặt trưng. Chúng ta đưa các câu hỏi vào trong bộ Tokenizer của phoBert để để encode, sau khi encode nhận được 2 danh sách là các input\_ids và attention\_masks. Cuối cùng chúng ta đưa các input\_ids và attention\_masks vào trong model và lấy ra các output.

#### 2.5. Xây dựng hệ thống gợi ý

Dữ liệu sau khi được thu thập thì sẽ được thông qua các bước tiền xử lý văn bản cơ bản như: chuẩn hóa Unicode tiếng Việt, chuẩn hóa kiểu gõ dấu tiếng Việt, loại bỏ các ký tự không cần thiết, chuẩn hóa thành chữ thường. Tiếp theo là bước tách từ, chúng ta sử dụng tách từ với thư viện pyvi và VnTokenizer.

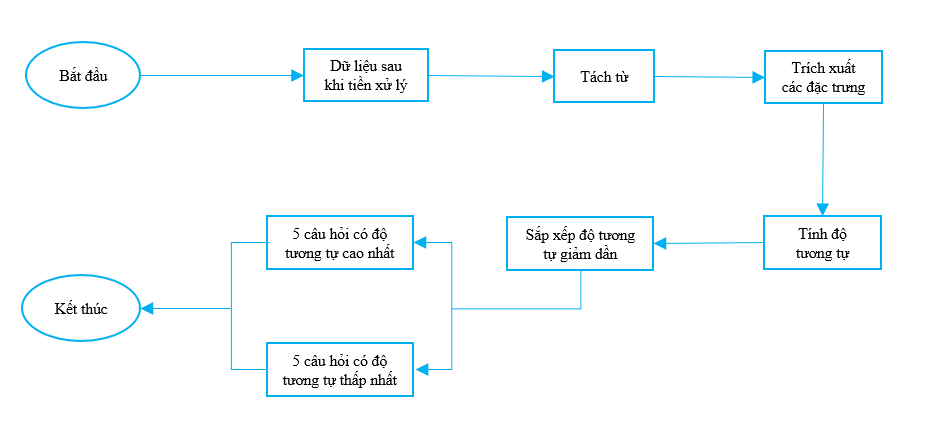
Ví dụ một câu trong bộ câu hỏi: “Nguyên nhân cơ bản nào làm cho chủ nghĩa xã hội ở Liên Xô và Đông Âu sụp đổ?”.

|  |  |  |
| --- | --- | --- |
|  | Tách từ với pyvi | Tách từ với VnTokenizer |
| 1 | nguyên\_nhân chủ\_nghĩa xã\_hội liên xô đông âu sụp\_đổ | nguyên\_nhân chủ\_nghĩa\_xã\_hội liên xô đông âu sụp\_đổ |

Chúng ta có thể thấy với 2 phương pháp tách từ thì đa số các cụm từ được tách giống nhau, chỉ có một chút khác nhau với từ “chủ nghĩa xã hội”.

Đến bước trích xuất các đặc trưng thì với đề tài này chúng ta sẽ sử dụng dữ liệu ở bước tiền xử lý cơ bản ở trên và trích xuất theo CountVector, Tfidf và tokenizer của phobert để thực hiện việc tính toán.

Trích xuất các đặc trưng với CountVector và Tfidf ta thu được một ma trận có 1396 dòng tương ứng với số lượng câu hỏi và 1372 cột tương ứng với số từ khóa đặc trưng. Còn với tokenizer của phobert ta thu được ma trận có số dòng là 1936 và số từ đặc trưng là 768 từ.



Sử dụng các độ đo khoảng cách để tính khoảng cách giữa các vector sau khi được trích xuất các đặc trưng, nêu khoảng cách giữa hai vector càng gần thì độ tương tự của hai vector đó cao và ngược lại khoảng cách càng xa thì độ tương tự càng thấp. Độ tương tự cao nhất là 1 và thấp nhất là 0.

Sau bước tính độ tương tự giũa các vector ta thu được một ma trận [1396 x 1396] chứa tất cả các vector câu hỏi và độ tương tự của chúng với các vector còn lại của toàn bộ vector.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1 | 0 | 0.64 | 0.93 | … | 0.87 |
| 2 | 0.64 | 0 | 0.98 | … | 0.99 |
| 3 | 0.93 | 0.98 | 0 | … | 0.89 |
| … | … | … | … | … | … |
| n | 0.87 | 0.99 | 0.89 | … | 0 |
|  | 1 | 2 | 3 | … | n |

Ví dụ chúng ta có n câu hỏi sau khi tính độ tương tự dựa vào độ đo cosine ta thu được ma trận như bên trên. Đối với độ đo khoảng cách thì giá trị càng nhỏ thì độ tương tự càng cao.

Đối với câu hỏi số 1, sau khi tính độ đo khoảng cách giữa câu 1 và n – 1 câu ta có được ma trận như trên. Để đưa ra gợi ý cho câu hỏi số 1 ta lấy các tất cả các giá trị của dòng thứ nhất đưa vào một mảng. Sau đó, chúng ta sắp xếp các giá trị trong mảng theo thứ tự từ nhỏ đến lớn. Chúng ta sẽ bỏ qua phần tử thứ 1, khi tính toán có tính luôn độ đo khoảng cách của chính câu số 1, vậy danh sách câu hỏi có độ tương tự cao nhất với câu 1 là phần tử thứ 2 đến phần tử thứ 6 và danh sách câu hỏi có độ tương tự thấp nhất với câu số 1 là 5 phần tử cuối cùng trong mảng độ đo khoảng cách sau khi được sắp xếp.

Ta sẽ thực hiện với n – 1 câu hỏi còn lại trong tập dữ liệu, trong quá trình tính toán chúng ta sẽ thực hiện với dữ liệu được vector hóa bằng CountVectorizer, TfidfVectorizer và Tokenizer của phobert thu được các cột:

recd\_phobert\_top: danh sách gợi ý có độ tương tự cao nhất của câu hỏi thứ i, vector hóa bằng Tokenzier của phobert.

recd\_phobert\_bottom: danh sách gợi ý có độ tương tự thấp nhất của câu hỏi thứ i, vector hóa bằng Tokenzier của phobert.

recd\_count\_top: danh sách gợi ý có độ tương tự cao nhất của câu hỏi thứ i, vector hóa bằng CountVectorizer.

recd\_count\_bottom: danh sách gợi ý có độ tương tự thấp nhất của câu hỏi thứ i, vector hóa bằng CountVectorizer.

recd\_tf\_top: danh sách gợi ý có độ tương tự cao nhất của câu hỏi thứ i, vector hóa bằng TfidfVectorizer.

recd\_tf\_bottom: danh sách gợi ý có độ tương tự thấp nhất của câu hỏi thứ i, vector hóa bằng TfidfVectorizer.

## CHƯƠNG 3 – GIỚI THIỆU WEBSITE

### 1. Giao diện chính hệ thống

### 2. Đánh giá

#### 2.1 Given-N

Tập dữ liệu có tổng là 1396 câu hỏi, trong đó có 102 câu hỏi có danh sách các câu hỏi tương tự dùng để đánh giá hệ thống gợi ý. Với mỗi câu hỏi có các trường dữ liệu sử dụng để đánh giá như sau:

id: định danh của câu hỏi.

list\_ids: danh sách ids của những câu hỏi tương tự và có 5 phần tử.

Để thực hiện việc đánh giá, mỗi danh sách ids của câu hỏi tương tự sẽ được chia thành 2 phần là phần <Given> và phần <Test>. Phần <Given> sẽ được sử dụng để sử dụng để tính toán và cho ra một danh sách gợi ý, sau đó gộp tất cả các phần tử được tính toán từ <Given> so sánh với các phần tử trong tập <Test>, nếu có một phần tử trong tập dự đoán tồn tại trong tập <Test> thì tăng độ chính xác hệ thống lên 1 đơn vị, ngược lại thì 0.

Việc đánh giá sẽ được thực hiện dựa trên dữ liệu được vector hóa theo nhiều cách khác nhau:

Dữ liệu tách từ bằng phương pháp pyvi và vector hóa bằng CountVectorizer.

Dữ liệu tách từ bằng phương pháp pyvi và vector hóa bằng TfidfVectorizer.

Dữ liệu tách từ bằng phương pháp Vncorenlp và vector hóa bằng Tokenizer của phobert.

Đối với luận văn này thì sử dụng phương pháp Given-1, Given-2 và Given-3 để đánh giá trên 100 câu hỏi. Đối với Given-1 thì lấy id của câu hỏi để tìm các câu hỏi có độ tương tự cao nhất rồi so sánh với danh sách ids tương tự từ trước. Với Given-2 thì lấy id của câu hỏi hiện tại và id của 1 phần tử trong danh sách tương tự để thực hiện việc tính toán, các phần tử còn lại thì dùng để so sánh như Given-1, áp dụng tương tự với Given-3.

Chia danh sách các công thức nấu ăn thành 2 phần: phần **given** và phần **test**. So sánh các gợi ý với dữ liệu trong tập test để đánh giá.

Trong đề tài này chúng tôi sử dụng **Given-1** và **Given-2** để đánh giá trên **30** người dùng. Sử dụng 1 công thưc (**Given-1**) và 2 công thức (**Given-2**) đầu tiên để dự đoán, nếu trong danh sách dự đoán có công thức nấu ăn của người dùng đã tạo thì là dự đoán đúng.

Với Given-1 ta có 6 dự đoán đúng: 6 / 30 = 0.2 = **20%**.

Với Given-2 ta có 7 dự đoán đúng: 7 / 30 = 0.23 = **23%**.

#### 2.2 All-but-one

Ưu điểm của phương pháp này là cho phép đánh giá các giỏ hàng có kích thước lớn hơn 1 sản phẩm.

Với phương pháp này ta đạt được 12 dự đoán đúng: 12 / 32 = 0.4 = **40%**.

# PHẦN KẾT LUẬN

### 1. Kết quả đạt được

### 2. Hướng phát triển

# TÀI LIỆU THAM KHẢO

[1] **Freetuts**, <https://freetuts.net/>

**[2] Viblo,** Giới thiệu về Hệ thống gợi ý – Recommendation Systems, <https://viblo.asia/p/gioi-thieu-ve-he-thong-goi-y-recommendation-systems-aWj53kEp56m>.

**[3] Lighthouselabs,** How Netflix Utilizes Data Science - Rachel Meltzer, [https://www.lighthouselabs.ca/en/blog/how-netflix-uses-data-to-optimize-their-product#:~:text=The%20Netflix%20Recommendation%20Engine,clusters%20based%20on%20user%20preferences](https://www.lighthouselabs.ca/en/blog/how-netflix-uses-data-to-optimize-their-product%23:~:text=The%20Netflix%20Recommendation%20Engine,clusters%20based%20on%20user%20preferences).

[4] **Wikipedia,** Hệ thống gợi ý, <https://en.wikipedia.org/wiki/Recommender_system>

[5] **Võ Thị Xuân Mai** – Luận văn tốt nghiệp Đại Học Cần Thơ.

[6] **Nguyễn Hùng Dũng, Nguyễn Thái Nghe,** Hệ thống gợi ý sản phẩm trong bán hàng trực tuyến sử dụng kỹ thuật lọc công tác.

[7] **Trần Nguyễn Minh Thư**, Hệ thống gợi ý dựa trên nội dung – slide.

[8] **Unicode, Dựng Sẵn hay Tổ Hợp, Vương Quang Khải**, <https://kilopad.com/May-tinh-Cong-nghe-c32/doc-sach-truc-tuyen-unicode-dung-san-hay-to-hop-b4903/chuong-1-ti1>

[9] **Wikipedia, Unicode**, <https://vi.wikipedia.org/wiki/Unicode>

[10] **Wikipedia, Quy tắc đặt dấu thanh trong chữ quốc ngữ**, <https://vi.wikipedia.org/wiki/Quy_t%E1%BA%AFc_%C4%91%E1%BA%B7t_d%E1%BA%A5u_thanh_trong_ch%E1%BB%AF_qu%E1%BB%91c_ng%E1%BB%AF>

[11] **Using CountVectorizer to Extracting Features from Text, GeeksforGeeks**, <https://www.geeksforgeeks.org/using-countvectorizer-to-extracting-features-from-text/>

[12] **TF-IDF là gì, Lập trình không khó**, <https://blog.luyencode.net/tf-idf-la-gi/>

[13] **Thuật toán tách từ**, <http://viet.jnlp.org/kien-thuc-co-ban-ve-xu-ly-ngon-ngu-tu-nhien/thuat-toan-tach-tu-tokenizer/thuat-toan-tach-tu>

[14] **Python Vietnamese Toolkit**, <https://github.com/trungtv/pyvi>

[15] **VnCoreNLP: A Vietnamese natural language processing toolkit**, <https://github.com/vncorenlp/VnCoreNLP>

[16] **Undertanding and Using Comom Simalarity Measures for Text Analysis**, <https://programminghistorian.org/en/lessons/common-similarity-measures>

[17] **Ultimate Guide To Text Similarity With Python – NewsCatCher**, <https://newscatcherapi.com/blog/ultimate-guide-to-text-similarity-with-python>

[18] **Wikipedia, Khoảng cách Manhattan**, <https://vi.wikipedia.org/wiki/Kho%E1%BA%A3ng_c%C3%A1ch_Manhattan>

[19] **How Netflix’s Recommendations System Works**, <https://help.netflix.com/en/node/100639#:~:text=We%20estimate%20the%20likelihood%20that,preferences%20on%20our%20service%2C%20and>

[20] **Redux là gì? Hiểu rõ cơ bản cách dùng Redux, Topdev**, <https://topdev.vn/blog/redux-la-gi/>