TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**KHÓA LUẬN TỐT NGHIỆP**

**XÂY DỰNG HỆ THỐNG HỎI ĐÁP**

**DỰA TRÊN CỘNG ĐỒNG**

*Người hướng dẫn*: **PGS.TS LÊ ANH CƯỜNG**

*Người thực hiện*: **NGÔ HÙNG PHÚC-51303129**

**NGUYỄN NHẬT NGUYÊN-51303352**

Lớp **: 13050302**

Khoá  **: 17**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2017**

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**KHÓA LUẬN TỐT NGHIỆP**

**XÂY DỰNG HỆ THỐNG HỎI ĐÁP**

**DỰA TRÊN CỘNG ĐỒNG**

*Người hướng dẫn*: **PGS.TS LÊ ANH CƯỜNG**

*Người thực hiện*: **NGÔ HÙNG PHÚC-51303129**

**NGUYỄN NHẬT NGUYÊN-51303352**

Lớp **: 13050302**

Khoá  **: 17**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2017**

LỜI CẢM ƠN

Lời đầu tiên,chúng em xin chân thành cảm ơn quý thầy cô trong khoa Công nghệ thông tin cũng như các quý thầy cô đang giảng dạy và công tác tại trường Đại học Tôn Đức Thắng đã truyền đạt những kiến thức quý báu cho chúng em trong những năm học vừa qua.

Đặc biệt, Chúng em xin gửi lời cảm ơn đến thầy Lê Anh Cường đã hỗ trợ chúng em trong suốt thời gian làm luận văn đã đưa ra những ý tưởng để cải tiến cho hệ thống của chúng em giúp chúng em có được một hệ thống hoàn chỉnh hơn, giúp đỡ và động viên em trong suốt thời gian thực hiện đề tài. Và để có được kết quả như ngày hôm nay, em rất biết ơn gia đình đã động viên, khích lệ, tạo mọi điều kiện thuận lợi nhất trong suốt quá trình học tập cũng như quá trình thực hiện đề tài tốt nghiệp này.

**CÔNG TRÌNH ĐƯỢC HOÀN THÀNH**

**TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

Tôi xin cam đoan đây là công trình nghiên cứu của riêng chúng tôi và được sự hướng dẫn khoa học của PGS.TS Lê Anh Cường. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong luận văn còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

**Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung luận văn của mình.** Trường đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

*TP. Hồ Chí Minh, ngày tháng năm*

*Tác giả*

*(ký tên và ghi rõ họ tên)*

*Ngô Hùng Phúc*

*Nguyễn Nhật Nguyên*

TÓM TẮT

Nhận thấy những khó khăn của các bạn sinh viên đặc biệt là các sinh viên mới vào trường khi có những thắc mắc trăn trở trong quá trình học tập sinh hoạt tại trường nhưng không biết trình bày những thắc mắc với ai.

Thậm chí nếu có thắc mắc và đặt câu hỏi thì đôi khi thông tin không chính xác làm các bạn có thể đánh mất quyền lợi của bản thân.

Do suy nghĩ đó nên chúng em đã đưa ra ý tưởng xây dựng một hệ thống hỏi đáp giành cho các bạn sinh viên sử dụng để đặt ra những câu hỏi và trò chuyện cùng nhau.

Hệ thống đã bước đầu đáp ứng được nhu cầu sử dụng của một diễn đàn cho phép hỏi đáp, tìm kiếm những câu hỏi chuẩn xác.

MỤC LỤC

[LỜI CẢM ƠN i](#_Toc476234882)

[TÓM TẮT iii](#_Toc476234883)

[MỤC LỤC 1](#_Toc476234884)

[DANH MỤC KÍ HIỆU VÀ CHỮ VIẾT TẮT 3](#_Toc476234885)

[DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ 4](#_Toc476234886)

[CHƯƠNG 1 – MỞ ĐẦU 5](#_Toc476234887)

[1.1 Giới thiệu hệ thống hỏi đáp tự động 5](#_Toc476234888)

[1.2 Mục đích đề tài 7](#_Toc476234891)

[1.3 Các nghiên cứu liên quan 7](#_Toc476234892)

[1.4 Đối tượng và phạm vi nghiên cứu 9](#_Toc476234893)

[CHƯƠNG 2 – XÂY DỰNG HỆ THỐNG 10](#_Toc476234894)

[2.2 Thiết kế hệ thống 11](#_Toc476234898)

[2.3. Thiết kế cơ sỡ dữ liệu 22](#_Toc476234913)

[2.3.1. Các bước xây dựng cơ sở dữ liệu. 22](#_Toc476234914)

[2.3.2. Xây dựng cơ sở dữ liệu 23](#_Toc476234915)

[2.4 Kiến trúc web 23](#_Toc476234917)

[2.4.1 Quản lý theo mô hình MVC 24](#_Toc476234918)

[2.4.2 Công cụ hiện thực 25](#_Toc476234919)

[2.5 Ứng dụng trên điện thoại 30](#_Toc476234925)

[2.5.1 Quản lý theo module 31](#_Toc476234926)

[2.5.2 Công cụ thực hiện 31](#_Toc476234927)

[2.6. Server 31](#_Toc476234928)

[CHƯƠNG 3 – XÂY DỰNG CƠ CHẾ TRẢ LỜI TỰ ĐỘNG 33](#_Toc476234931)

[3.1 Kiến trúc tổng quan 33](#_Toc476234932)

[3.2 Phương pháp: 35](#_Toc476234935)

[3.2.1 Lọc dữ liệu sử dụng Full Text Search 35](#_Toc476234936)

[3.2.2 Tách từ 36](#_Toc476234953)

[3.2.3 Phân loại câu hỏi sử dụng Maximum entropy classifier 36](#_Toc476234959)

[3.2.4 Đo độ tương tự giữa các câu hỏi. 38](#_Toc476234962)

[CHƯƠNG 4–NGHIÊN CỨU THỰC NGHIỆM 40](#_Toc476234963)

[4.1 Mục tiêu thử nghiệm 40](#_Toc476234964)

[4.2 Kế hoạch thử nghiệm Kế hoạch thử nghiệm sẽ bao gồm 4 phần: 40](#_Toc476234965)

[4.3 Đánh giá giải thuật: 41](#_Toc476234966)

[4.3.1 Tổng quan mô hình hiện thực giải thuật 41](#_Toc476234967)

[4.3.2 Mô hình 1-gram 41](#_Toc476234969)

[4.3.2 Mô hình 2-gram 42](#_Toc476234970)

[CHƯƠNG 5– TỔNG KẾT 45](#_Toc476234971)

DANH MỤC KÍ HIỆU VÀ CHỮ VIẾT TẮT

**CÁC KÝ HIỆU**

**CÁC CHỮ VIẾT TẮT**

* FTS: Full Text Search.
* MVC: Model View Controller.
* CSDL: Cơ sở dữ liệu.
* QA: Question Answering
* IR: Information Research

DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ

**DANH MỤC HÌNH:**

[Hình 2. 1: Kiến trúc tổng quan 10](#_Toc476233281)

[Hình 2. 2 Class Diagram hệ thống 11](#_Toc476233282)

[Hình 2. 3: Use case chưa đăng nhập 13](#_Toc476233283)

[Hình 2. 4: Use case đăng ký sau đó đăng nhập 14](#_Toc476233284)

[Hình 2. 5: Use case sau khi đăng nhập 15](#_Toc476233285)

[Hình 2. 6: Usecase sau khi đăng nhập và đặt câu hỏi 16](#_Toc476233286)

[Hình 2. 7: Usecase sau khi đăng nhập và trả lời câu hỏi 16](#_Toc476233287)

[Hình 2. 8: Sequence Diagram đăng nhập 17](#_Toc476233288)

[Hình 2. 9: Sequence Diagram đăng ký 18](#_Toc476233289)

[Hình 2. 10: Sequence Diagram đăng câu hỏi 19](#_Toc476233290)

[Hình 2. 11: Sequence Diagram trả lời câu hỏi 20](#_Toc476233291)

[Hình 2. 12: Sequence Diagram gửi tin nhắn 21](#_Toc476233292)

[Hình 2. 13: Mô hình thiết kế cơ sỡ dữ liệu 23](#_Toc476233293)

[Hình 2. 14: Mô Hình Hoạt Động Của Một Ứng Dụng Mean Js 30](#_Toc476233294)

[Hình 3. 1: Kiến trúc tổng quan cơ chế trả lời tự động 33](#_Toc476233161)

[Hình 4. 1: Tổng quan mô hình giải thuật 41](#_Toc476233166)

[Biểu đồ 2. 1: So sánh tốc độ insert dữ liệu giữa MongoDB và SQL Server 28](#_Toc476233295)

[Biểu đồ 2. 2: So sánh tốc độ truy vấn dữ liệu giữa MongoDB và SQL Server 29](#_Toc476233296)

CHƯƠNG 1 – MỞ ĐẦU

* 1. Giới thiệu hệ thống hỏi đáp tự động

Ngày nay hệ thống internet phát triển với một khối lượng dữ liệu khổng lồ dẫn đến việc tìm kiếm sẽ gặp khó khăn cùng với đó là sự nhiễu loạn thông tin cũng có thể dẫn tới việc nắm bắt sai thông tin.

Thông tin tìm kiếm đôi khỉ chỉ dừng ở mức tài liệu nghiên cứu, còn hệ thống hỏi đáp sẽ cho ta một câu trả lời ngắn gọn nhất có thể và đôi khi là 1 hướng giải quyết vấn đề từ những người đi trước.

Hiện nay có nhiều hệ thống hỏi đáp phục vụ nhu cầu của cộng đồng từ các mảng lập trình, sửa chữa máy tính, tư vấn học tập, sức khỏe và nhiều hơn nữa nhưng đa phần chỉ được phát triển bằng tiếng Anh.

Hệ thống hỏi đáp vẫn chưa thật sự phát triển mạnh ở nước ta nên nhận thấy việc xây dựng hệ thống hỏi đáp bằng tiếng việt rất có ý nghĩa và mang tính thực tế.

Hệ thống hỏi đáp là gì:

Hệ thống hỏi đáp là hệ thống cho phép người dùng đặt câu hỏi và nhận được câu trả lời từ cơ sở dữ liệu đã thu thập từ cộng đồng trước đó về những vấn đề mà họ đang gặp khó khăn và chưa tìm ra hướng giải quyết.

Hệ thống hỏi đáp tự động là hệ thống tự đưa ra câu trả lời cho người dùng đặt câu hỏi dựa trên cơ sở dữ liệu xây dựng sẵn từ trước.

Hệ thống hỏi đáp là nơi tập trung các kiến thức, các kinh nghiệm của những người đi trước chia sẽ lại, nó cũng là thế giới thông tin mở và là kho tàng kiến thức.

Hệ thống hỏi đáp được chia thành 2 loại chính:

* Hỏi đáp dựa trên cộng đồng (Community Question Answering System): Dữ liệu được thu thập chủ yếu từ những cơ sở tri thức trên internet để xây dựng nên hệ thống.
* Hỏi đáp dựa trên bộ sinh (Generative Question Answering System): Câu trả lời sẽ tự động được sinh ra đáp ứng nhu cầu hỏi đáp.Được xây dựng từ các tri thức có sẵn và sinh ra câu trả lời.

Hệ thống hỏi đáp trong để tài nghiên cứu này sẽ tập trung vào cộng đồng sinh viên trường đại học Tôn Đức Thắng.

Ý nghĩa hệ thống hỏi đáp và nhu cầu xã hội:

Hiện nay lượng người dùng internet rất là lớn và nhu cầu được giải đáp thắc mắc cũng tăng theo cùng với đó là các câu hỏi thì lập đi lập lại.Vấn đề đặt ra là phát triển 1 hệ thống trả lời tự động theo 2 hướng đưa ra cho người dùng trả lời hoặc từ hệ thống. Hệ thống hỏi đáp sẽ là nơi tập trung thông tin các vấn đề, thắc mắc, góp ý… Cung cấp một nguồn thông tin đáng tin cậy cho người dùng, tránh được những thông tin trái chiều không chính xác.

Nhu cầu được giải đáp thắc mắc trong quá trình học tập sinh hoạt luôn có ở bất cứ môi trường học tập nào vì vậy việc ứng dụng hệ thống hỏi đáp vào thực tế cho sinh viên sử dụng sẽ tạo tiền đề và sẽ là cảm hứng phát triển thêm nữa cho các hệ thống khác về sau.

Như đã giới thiệu do nhận thấy nhà trường đã có hệ thống hỏi đáp nhưng vẫn chưa có sự tương tác mạnh giữa sinh viên và nhà trường và số lượng sử dụng còn ít nên chúng em quyết định hiện thực hệ thống cho các bạn sinh viên sử dụng.

Hệ thống cho phép sinh viên đặt và trả lời các câu hỏi. Cùng với đó là khu vực trò chuyện cho các bạn sinh viên giúp các bạn tiết kiệm thời gian chờ đợi câu hỏi được trả lời.Hệ thống sẽ là một kênh thông tin hữu dụng cho các bạn sinh viên.

Hệ thống sẽ là một địa chỉ đáng tin cậy và tập trung thông tin cho toàn thể sinh viên trường Đại Học Tôn Đức Thắng sử dụng hệ thống để đặt câu hỏi và giải đáp thắc mắc cho nhau. Là cầu nối giữa nhà trường và sinh viên. Là nơi để sinh viên bày tỏ nguyện vọng của mình và cũng là nơi để nhà trường tiếp nhận ý kiến đóng góp của sinh viên để từng bước cải thiện quá trình đào tạo. Tránh làm mất quyền lợi của các bạn sinh viên khi không được giải đáp kịp thời.

1.2 Mục đích đề tài

Tạo ra diễn đàn kết nối giữa sinh viên và nhà trường, thông qua website và ứng dụng điện thoại trên nền tảng Android. Là nơi cung cấp lời, trao đổi, giải đáp các thắc mắc của các bạn sinh viên, hỗ trợ sinh viên tìm kiếm thông tin một cách nhanh chóng và chính xác nhất.

Nghiên cứu hệ thống Question Answering và cách hiện thực một hệ thống trả lời tự động áp dụng cho sinh viên trường Đại Học Tôn Đức Thắng nói riêng và có thể mở rộng cho toàn thể xã hội sử dụng.

1.3 Các nghiên cứu liên quan

Nghiên cứu được rút ra được từ bài báo Question Answering from Frequently Asked Question Files(*Robin D. Burke, Kristian J. Hammond, Vladimir Kulyukin,*

*Steven L. Lytinen, Noriko Tomuro, and Scott Schoenberg*)

Hệ thống tìm kiếm các câu hỏi thường được dựa trên giả định của bộ câu hỏi thường gặp được xây dựng theo chuẩn hỏi đáp(Kulyukin, Hammond, and Burke 1996). Tất cả thông tin cần để xác định sự liên quan của các cặp câu hỏi có thể tìm thấy trong bản thân của các cặp câu.Việc xác định sự liên quan giữ các cặp câu hỏi với nhau dựa trên những cơ sở tri thức sau: Tri thức rộng và nông của việc xử lý ngôn ngữ sẽ thích hợp cho việc so trùng câu hỏi.

Ví dụ khi hệ thống xử lý một câu hỏi người dùng nhập vào “Cách viết đơn xin nghỉ học?”.Bộ tìm kiếm câu hỏi của hệ thống sẽ tiến hành so sánh câu hỏi nhập vào với tập các câu hỏi trong hệ thống và trả về được xếp hạng dựa trên độ tương tử với câu hỏi của người dùng như:

* Em muốn xin nghỉ học 1 buổi, Vậy phải viết đơn như thế nào?
* Làm sao để xin phép nghỉ học tạm thời?

Khi bộ câu hỏi được chọn hệ thống sẽ tiến hành lặp qua cặp câu hỏi trong dữ liệu, so sánh câu hỏi trong bộ dữ liệu với câu hỏi của người dùng sau đó tính toán điểm tương tự.

Vậy hệ thống sẽ xử lý ra sao:

* Bước đầu tiên trong việc xử lý hệ thống sẽ thu hẹp phạm vi tìm kiếm dựa trên loại câu hỏi người dùng nhập vào.
* Tiếp theo từng cặp câu hỏi câu trả lời sẽ được so sánh với câu hỏi của người dùng
* Giai đoạn đầu của việc xử lý,hệ thống tìm kiếm sử dụng công nghệ rút trích thông tin tiêu chuẩn,miền chung của hệ thống SMART rút trích dữ liệu(Buckley 1985) để thực hiện bước khởi tạo trong việc thu hẹp lại thành một miền dữ liệu con từ tập câu hỏi trong hệ thống.SMART sẽ tiến hành lấy ra những từ biến thể trong câu hỏi của người dùng nhập.Sau đó sẽ tạo ra 1 vector từ câu hỏi truy vấn với vector câu hỏi tương tự trong hệ thống đã được đánh chỉ mục.
* Giai đoạn tiếp theo của việc xử lý sẽ là quá trình so trùng câu hỏi.Từng câu hỏi trong hệ thống sẽ được so sánh với câu hỏi của người dùng nhập vào và chấm điểm tương tự.Chúng ta sẽ sử dụng 3 cặp số liệu sau cho từng tập câu hỏi: điểm tương tự của vector (t),điểm tương tự về ngữ nghĩa(s) và độ phủ (c).Tổng quan độ trùng khớp m được tính như sau:

m =

Trong đó T,S,C là hằng số phụ thuộc được điều chỉnh dựa trên hệ thống của mỗi số liệu.

Điểm số thống kê tại từng cặp Question Answering(QA) theo tiêu chí tương tự như tài liệu so trùng của hệ thống SMART.Từng cặp QA được đại diện bởi 1 vector,các vector liên kết với giá trị trọng yếu trong từng cặp QA.Giá trị trọng yếu thường được biết đến bằng tên tdifdf(Salton và McGill 1983).Nếu n là tần số(số lần xuất hiện của điều khoản trong cặp QS), m là số cặp han QA xuất hiện trong tập dữ liệu,M là số cặp QA xuất hiện trong tập dữ liệu nên tfidf = n x log(M/n).Ý tưởng của việc so sánh này dùng để đánnh giá độ tương đối hiếm của điều khoản trong tài liệu và sử dụng như là yếu tố để tính toán tần suất của điều khoản trong tài liệu cụ thể.

Vector giới hạn số liệu cho phép hệ thống đánh giá độ tương tự của câu hỏi người dùng và cặp QA.Các biện pháp tfidf có một lịch sử khá dàitrong tìm kiếm thông tin và thường xem làm việc tốt nhất chỉ trên tài liệu tương đối dài vì chỉ tài liệu dài có đủ từ ngữ để thống kê so sánh được coi là có ý nghĩa.

Nghiên cứu được rút ra được từ bài báo An Analysis of the AskMSR Question-Answering System(Eric Brill, Susan Dumais and Michele Banko):

Tái định hình câu truy vấn:

Với một câu hỏi hệ thống tạo ra một các câu được viết lại từ câu hỏi gốc của người dùng.Ví dụ: “Kẹp bấm giấy được sáng chế vào lúc nào?” sẽ được viết lại thành “Kẹp giấy được sáng chế”. Sau đó chung ta sẽ sét qua tập các tài liệu trong tìm kiếm mô hình như vậy. Việc viết lại các chuỗi cũng làm giảm khả năng tìm được câu hỏi tương tự.

Khai thác N-gram:

Một khi tập các bộ câu hỏi tái định hình truy vấn được tạo ra,từng câu hỏi được định hình như là công cụ tìm kiếm và được gửi tới bộ công cụ tìm kiếm mà tại đó dữ liệu được tổng hợp và phân tích. Như vậy,điểm số cuối cùng cho một n-gram được dựa trên trọng lượng

kết hợp với các quy tắc viết lại sinh ra nó và số lượng bản tóm tắt duy nhất trong đó các câu hỏi xuất hiện.

Lọc N-Gram:

Tiếp theo,n-gram được lọc lại một cách phù hợp sao cho từng ứng viên khớp với kì vọng của loại trả lời.Hệ thống sẽ sử dụng bộ lọc theo các bước sau:

* Đầu tiên, truy vấn được phân tích và được gán thành một trong bảy loại câu hỏi, chẳng hạn như loại who-question, what-question, hoặc how-manyquestion
* Dựa vào loại truy vấn đã được gán, hệ thống sẽ xác định những tập bộ lọc nào sẽ áp dụng cho các của câu trả lời tiềm năng tìm thấy trong tập của n-gram.
* Các ứng viên n-grams được phân tích cho các tính năng liên quan đến các bộ lọc,sau đó sẽ được chấm điểm lại theo sự hiện diện của thông tin.

Độ bao phủ của N-Gram:

Cuối cùng áp dụng giải thuật bao phủ câu trả lời mà kết hợp cả việc gộp câu trả lời tương tự và tập hợp câu trả lời dài từ các đoạn trả lời chồng chéo nhau.Ví dụ như “A B C” và “B C D” sẽ được gộp lại thành “A B C D”. Giải thuật sẽ thực thi bài toán tham lam từ ứng viên có số điểm cao nhất đến các ứng viên con (lên đến một ngưỡng nhất định). Các ứng cử viên có số điểm cao hơn được thay thế bằng việc phủ n-gram,các ứng cử viên có điểm thấp hơn được loại bỏ. Thuật toán dừng lại chỉ khi không có n-gram có thể được tiếp tục bao phủ.

Nghiên cứu được rút ra từ bài báo Learning Surface Text Patterns

for a Question Answering System

Đa số các hệ thống mở về miền trả lời câu hỏi sử dụng tri thức bên ngoài và công cụ để làm xác định câu trả lời bao gồm những người gắn thẻ thực thể có tên, WordNet,

phân tích cú pháp,gán thẻ tay corpora, và danh sách bản thể học.Hệ thống dành chiến thắng là hệ thống sử dụng chỉ một nguồn: một danh sách khá rộng bề mặt các hình mẫu.

Ví dụ, đối với Ngày sinh (với những câu hỏi như "Khi nào X được sinh ra? "), câu trả lời điển hình là

"Mozart được sinh ra trong năm 1756."

"Gandhi (1869-1948) ..."

Những ví dụ này cho thấy cụm từ như

"<TÊN> sinh năm <Ngày Sinh>" và "<TÊN> <Ngày Sinh> “ khi được xây dựng như biểu thức chính quy có thể được dùng để xác định câu trả lời đúng.

Phương thức này sử dụng kỹ thuật học máy để xây dựng nên tập văn đươc gắn thể bắt đầu từ vài cặp QA mẫu.

Hệ thống sẽ giả định mỗi câu là một trình tự đơn giản của các từ và tìm kiếm

theo thứ tự từ lặp đi lặp lại làm bằng chứng cho những câu trả lời hữu ích.

1. Hướng đi của giải thuật:
2. Ví dụ các bước hiện thực giải thuật pattern-learning :

* Ta chọn “Morazrt sinh năm 1756” (“Mozart” sẽ đóng vai trò là giới hạn cho câu hỏi và “1756” sẽ là giới hạn cho câu trả lời)
* Gửi các câu hỏi và câu trả lời như các truy vấn vào công cụ tìm kiếm
* Lấy về khoảng 100 tài liệu được đưa ra từ hệ thống tìm kiếm
* Áp dụng ngắt câu cho tài liệu
* Giữ lại những câu chưa cả giới hạn của câu hỏi câu trả lời.
* Tokenize văn bản đầu vào và loại bỏ html và các thẻ liên quan khác, để cho phép các công cụ biểu thức chính quy đơn giản như egrep có thể sử dụng.
* Chuyển những câu lấy được qua hàm khởi tạo cây hậu tố cách này sẽ tìm ra những chuỗi con bao gồm độ dài của chúng. Ví dụ như câu “Nhà soạn nhạc vĩ đại Mozart(1756-171) đạt được danh tiếng khi tuổi đời còn trẻ”.Câu “Mozart(1756-1791) là thiên tài” và câu “Cả thế giới sẽ mang ơn nhà soạn nhạc vĩ đại Mozart(1756-1791)”. Chuỗi con trùng có độ dài lớn nhất trong 3 câu là “Mozart(1756-1791)” vậy cây hậu tố sẽ có điểm số là 3.
* Đưa từng câu trong cây hậu tố qua bộ lọc để giữ lại những câu nào bao gồm giới hạn câu hỏi và câu trả lời.
* Thay từ cho thẻ <Tên> và từ cho thẻ <Câu trả lời>

Và <Ngày Sinh> theo những bước trên sẽ cho ta kết quả sau:

1. Sinh vào <Câu trả lời>, <Tên>
2. <Tên> sinh vào <Câu trả lời>
3. <Tên> <Câu trả lời>
4. <Tên> <Câu trả lời>
5. …..
6. Giải thuật 2: Tính toán độ chính xác của từng hình mẫu :

* Truy vấn các công cụ tìm kiếm bằng cách chỉ sử dụng giới hạn câu hỏi (trong ví dụ này, chỉ một từ "Mozart").
* Tải về 1000 tài liệu web hàng đầu được cung cấp bởi các công cụ tìm kiếm.
* Như đề cập từ trước, phân đoạn các tài liệu vào từng loại câu hỏi.
* Giữ lại chỉ có những câu có chứa giới những hạn câu hỏi.
* Đối với mỗi mẫu thu được từ giải thuật, kiểm tra sự hiện diện của mỗi hình mẫu trong câu được lấy ra từ 2 trường hợp sau:

1. Sự hiện diện của hình mẫu với thẻ <Câu trả lời> trùng khớp với bất kỳ từ nào
2. Sự hiện diện của hình mẫu <Câu trả lời> trùng khớp với bất kỳ giới hạn trả lời nào

Trong ví dụ, với hình mẫu <Tên> được sinh vào <Câu trả lời> chúng ta sẽ kiểm sự hiện diện của các chuỗi sau trong câu trả lời:

1. Mozart được sinh vào <Bất kỳ>
2. Mozart được sinh vào 1756

Độ chính xác của từng hình mẫu được tính bằng công thức

P = Ca/ Co

Trong đó:

Ca = tổng số lương hình mẫu mà giới hạn câu trả lời đại diện

C0 = tổng số lượng hình mẫu xuất hiện mà giới hạn câu trả lời đại diện có thể thay thế bởi bất kỳ từ nào

* Chỉ giữ lại những hình mẫu nào trùng khớp đầy đủ với số lượng ví dụ
* Đối với từ <Sinh Nhật> chúng ta thu được kết quả như sau:

Hình mẫu tìm kiếm <Tên>( <Câu trả lời> - )

1. 0.85 < Tên > được sinh vào < Câu trả lời >,
2. 0.6 < Tên > được sinh vào < Câu trả lời >,
3. 0.59 < Tên > được sinh vào < Câu trả lời >,
4. 0.53 < Tên > < Câu trả lời >, sinh vào
5. 0.50 – < Tên > (< Câu trả lời >
6. 0.36 < Tên > (< Câu trả lời >-
7. Tìm câu trả lời:

* Xác định các loại câu hỏi của các câu hỏi mới.
* Giới hạn trong câu hỏi đã được xác định.
* Tạo một truy vấn từ hạn câu hỏi và thực hiện IR.
* Phân khúc các tài liệu thu được thành câu.
* Thay thế cụm từ câu hỏi trong mỗi câu bởi các câu hỏi có thẻ ("<TÊN>", trong trường hợp <Năm Sinh>).
* Sử dụng bảng mô hình phát triển cho rằng cho từng loại câu hỏi, tìm kiếm sự hiện diện của mỗi mẫu sau đó chọn từ phù hợp với từ khóa "<Câu trả lời>".
* Sắp xếp các câu trả lời bằng các hình mẫu với điểm chính xác. Loại ra những bản dữ liệu trùng lắp và trả về 5 câu trả lời có số điểm cao nhất.

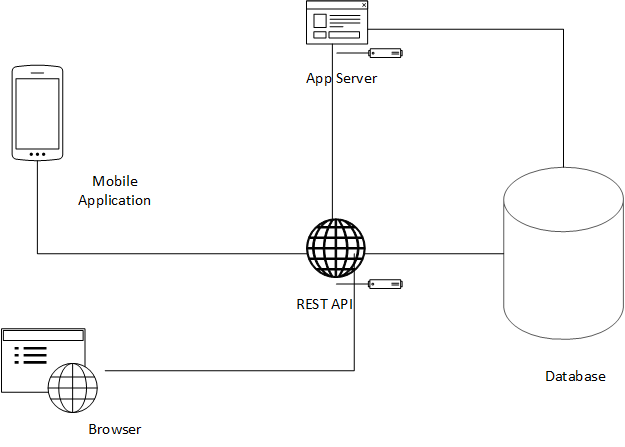
1.4 Đối tượng và phạm vi nghiên cứu

Đối tượng nghiên cứu là hệ thống hỏi đáp (Question Answering System) bao gồm:

* Nghiên cứu dữ liệu tương tác.
* Nghiên cứu xây dựng diễn đàn.
* Nghiên cứu xây dụng app android.
* Hệ thống tìm kiếm câu hỏi tương đương.
* Các hệ thống QA đã được xây dựng từ trước
* Các giải thuật đã được áp dụng cho các hệ thống khác

CHƯƠNG 2 – XÂY DỰNG HỆ THỐNG

2.1 Kiến trúc tổng quan hệ thống website dùng API và mobile



Hình 2. 1: Kiến trúc tổng quan

Hệ thống web sẽ là nơi cung cấp dịch vụ REST API cho mobile app.Ứng với từng loại yêu cầu của người dùng hệ thống mobile hoặc web sẽ gọi tới REST API và thực thi thao tác thêm xóa sửa cập nhật dữ liệu.

Trình duyệt web hoặc ứng dụng TDT Forum trên điện thoại sẽ gửi yêu cầu để lấy nội dung. REST API sẽ là nơi tiếp nhận xử lý các yêu cầu này. Tùy vào loại yêu cầu mà REST API sẽ xử lý khác nhau, nếu yêu cầu là lấy danh sách các câu hỏi, đăng ký, đăng nhập… thì REST API sẽ lấy dữ liệu trực tiếp từ Database và phản hồi về cho trình duyệt web hoặc ứng dụng điện thoại.

Ví dụ nếu yêu cầu của người dùng là đăng câu hỏi sau hoàn thành nhập dữ liệu yêu cầu cho câu hỏi hệ thống sẽ liên lạc với đường dẫn cho yêu cầu đăng câu hỏi.Sau khi liên lạc thành công hệ thống tiến hành thực thi logic trong phương thức đăng câu hỏi và thông báo kết quả cho người dùng.

2.2 Thiết kế hệ thống

Mô Hình Class Diagram



Hình 2. 2 Class Diagram hệ thống

Phân tích Class Diagram:

Khi người dùng đặt câu hỏi hệ thống sẽ tiến hành đánh giá, phân loại cho câu hỏi thuộc loại câu hỏi nào, chuyên mục gì.

Hệ thống sẽ dựa trên tiêu chí của hệ thống phân loại câu hỏi để quyết định đó là loại câu hỏi gì ? (What,where,when,which,how)

Khi đã phân loại sẽ gợi ý những câu hỏi tương tự vậy trong hệ thống để người dùng có thể tham khảo thêm.

Người dùng tìm kiếm câu hỏi trong hệ thống bằng việc nhập liệu trong khung tìm kiếm.Hệ thống sẽ tiến hành đánh giá dựa trên các giải thuật tính toán 1-gram 2-gram để đưa ra câu trả lời gần đúng nhất so với câu hỏi người dùng nhập vào.

Hệ thống sẽ tiến hành xếp hạng theo số điểm so sánh giữa câu hỏi người dùng nhập vào so với câu hỏi nguồn và đưa ra câu trả lời gần đúng nhất cho yêu cầu của người dùng.

Mô Hình Usecase

****

Hình 2.3: Use case chưa đăng nhập

Khi thành viên chưa đăng nhập sẽ bị giới hạn ở một số chức năng nhất định của diễn đàn:

* Xem chuyên mục câu hỏi:Các chuyên mục câu hỏi hiện có của diễn đàn.Chuyên mục giúp phân loại câu hỏi.
* Xem câu hỏi:Các câu hỏi được các thành viên gửi về diễn đàn
* Xem bài viết chưa trả lời:Các câu hỏi chưa được trả lời sẽ được hiển thị cho người dùng tham gia trả lời
* Xem câu hỏi được quan tâm nhiều nhất: Các câu hỏi có số lượt xem nhiều nhất sẽ được hiển thị và xếp hạng từ cao tới thấp.
* Tìm kiếm câu hỏi: Người dùng khi chưa đăng nhập sẽ có thể tìm kiếm các câu hỏi đã có trong hệ thống nếu có thì không cần phải đăng nhập và đặt câu hỏi.

****

Hình 2. 4: Use case đăng ký sau đó đăng nhập

Khi người dùng đăng ký tài khoản mới hệ thống sẽ yêu cầu nhập 3 thông tin chính sau

* Tên tài khoản
* Email sử dụng
* Mật khẩu tải khoản

Hệ thống sau khi kiểm tra dữ liệu hợp lệ hay không nếu dữ liệu hợp lệ sẽ tiến hành tạo tài khoản mới và đăng nhập cho người dùng bằng tài khoản mới tạo.



Hình 2. 5: Use case sau khi đăng nhập

Khi đăng nhập thành công người dùng sẽ có thể:

* Quản lý câu hỏi:Người dùng có thể xem lại các câu hỏi đã gửi lên cho hệ thống,có thể sửa hoặc xóa câu hỏi do mình đăng.
* Đặt câu hỏi: Khi người dùng có thắc mắc về một vấn đề nào đó có thể sử dụng hệ thống để đặt câu hỏi.Người dùng chỉ cần chọn chuyên đề của câu hỏi và nhập tiêu đề cũng như nội dung câu hỏi.
* Trả lời câu hỏi: Người dùng có thể tham gia trả lời các câu hỏi của các thành viên khác trong diễn đàn.
* Thích bài viết và không thích bài viết: Nếu cảm thấy câu trả lời hay hoặc không hay người dùng có thể bó phiếu tán thành hoặc phản đối cho câu trả lời đó để tăng độ tin cậy cho một câu trả lời.



Hình 2. 6: Usecase sau khi đăng nhập và đặt câu hỏi

Sau khi đăng nhập người dùng sẽ có quyền đặt câu hỏi.Việc đặt câu hỏi sẽ được tiến hành qua từng bước sau:

* Nhập tiêu đề câu hỏi:Tiêu đề câu hỏi sẽ ngắn gọn và nói lên vấn đề người dùng đang gặp phải.
* Chon chuyên mục câu hỏi: Việc chọn chuyên mục câu hỏi góp phần giúp phân loại bài viết tạo điều kiện câu hỏi nhận được trả lời nhanh hơn.
* Nhập nội dung câu hỏi: Nội dung câu hỏi sẽ cung cấp thêm thông tin cần thiết để các thành viên nắm bắt rõ vần đế của người đặt câu hỏi và đưa ra hướng giải quyết hợp lý hơn.

Sau khi hoàn thành xong người dùng tiến hành đăng câu hỏi và đợi câu trả lời



Hình 2. 7: Usecase sau khi đăng nhập và trả lời câu hỏi

Sau khi đăng nhập người dùng có thể chọn câu hỏi muốn trả lời.Câu trả lời càng chi tiết càng tốt.Việc trả lời câu hỏi chỉ yêu cầu người dùng nhập nội dung câu trả lời.

Mô Hình Sequence Diagram



Hình 2. 8: Sequence Diagram đăng nhập

Phân tích mô hình sequence đăng nhập:

Hệ thống sẽ hiển thị giao diện cho người dùng nhập thông tin đăng nhập gồm tài khoản và mật khẩu.Sau khi kiểm tra tính hợp lệ của dữ liệu hệ thống sẽ tiến hành truy vấn thông tin người dùng từ cơ sở dữ liệu.

Dữ liệu trả về sẽ có kết quả đăng nhập thành công hay không.Nếu đăng nhập không thành công hệ thống sẽ thống báo cho người dùng để tiến hành đăng nhập lại.



Hình 2. 9: Sequence Diagram đăng ký

Phân tích mô hình sequence đăng ký:

Khi người dùng đăng ký tài khoản mới hệ thống sẽ hiện thị ra nhưng thông tin bắt buộc gồm tài khoản,mật khẩu và email.

Người dùng tiến hành nhập thông tin sau đó hệ thống sẽ tiến hành kiểm tra tính hợp lệ của dữ liệu.

Nếu dữ liệu hợp lệ sẽ truy vấn dữ liệu và kiểm tra trùng thông tin đăng ký hay không.Nếu thông tin đăng ký trùng sẽ thông báo cho người dùng đăng kí lại đến khi thành công.



Hình 2. 10: Sequence Diagram đăng câu hỏi

Phân tích mô hình sequence đăng câu hỏi: Hệ thống sẽ hiện thị những trường dữ liệu cần thiết để người dùng đặt câu hỏi.Người dùng tiến hành nhập dữ liệu và hệ thống sẽ kiểm tra tính hợp lệ của dữ liệu.Khi dữ liệu hợp lệ sẽ gửi lên server và lưu câu hỏi vào hệ thống cơ sở dữ liệu và thông báo kết quả cho người dùng.



Hình 2. 11: Sequence Diagram trả lời câu hỏi

Phân tích mô hình sequence trả lời câu hỏi: Hệ thống sẽ hiện thị những trường dữ liệu cần thiết để người dùng đặt câu hỏi.Người dùng tiến hành nhập dữ liệu và hệ thống sẽ kiểm tra tính hợp lệ của dữ liệu.Khi dữ liệu hợp lệ sẽ gửi lên server và lưu câu hỏi vào hệ thống cơ sở dữ liệu và thông báo kết quả cho người dùng.



Hình 2. 12: Sequence Diagram gửi tin nhắn

Phân tích mô hình sequence gửi tin nhắn: Hệ thống sẽ hiện thị form cho người dùng nhập tin nhắn sau đó sẽ tiến hành kiểm tra dữ liệu.Khi dữ liệu hợp lệ sẽ gửi dữ liệu về server và tiến hành lưu tin nhắn vào cơ sở dữ liệu.

2.3. Thiết kế cơ sỡ dữ liệu

2.3.1. Các bước xây dựng cơ sở dữ liệu.

* Xác định vấn đề cần giải quyết: Tự đặt ra vấn đề hệ thống cần thiết kế như thế nào để lữu trữ.Cơ sở dữ liệu được sử dụng cho đối tượng sinh viên và các thông tin của sinh viên cùng với đó là các câu hỏi câu trả lời sinh viên
* Nghiên cứu hệ thống dữ liệu sẵn có: Các hệ thống hỏi đáp nổi tiếng như stackoverflow, askubuntu.
* Thiết kế cấu trúc dữ liệu: Xác định các đối tượng cần có cho cơ sỡ dữ liệu, các quan hệ giữa các đối tượng,các kiểu dữ liệu.
* Ràng buộc kiểu dữ liệu: Đảm bảo kiểu dữ liệu phù hợp với nhu cầu sử dụng tránh tạo kiểu dữ liệu sai hoặc nhiều hơn thực tế cần để tránh việc dư thừa dữ liệu.
* Tiến hành xây dựng cơ sỡ dữ liệu.

2.3.2. Xây dựng cơ sở dữ liệu



Hình 2. 13: Mô hình thiết kế cơ sỡ dữ liệu

Cơ sở dữ liệu sẽ gồm có 5 bảng chính:

* Đầu tiên là bảng Members, đây là nới lưu trữ thông tin của các thành viên trong diễn đàn, Id, Account, Password….Bảng này cũng sẽ chứa thông tin về tài khoản mạng xã hội trong trường hợp này là Facebook khi người dùng đăng nhập bằng hệ thống website bằng mạng xã hội.Một người dùng có thể có nhiều câu hỏi, câu trả lời, nhiều tin nhắn.
* Bảng Messages là nơi chứa các tin nhắn, trò chuyện của các thành viên.Lưu trư thông tin tin nhắn của như người gửi thời gian gửi.Một user có thể có nhiều tin nhắn
* Bảng Categories chứa thông tin các loại chủ đề trong diễn đàn ví dụ chủ đề nội quy, chủ đề học phí, quy chế đào tạo…Một chủ đề sẽ có nhiều bài viết về chủ đề đó.
* Bảng Questions chứa các câu hỏi do các thành viên đăng lên, Mỗi câu hỏi trong bảng sẽ thuộc một loại trong bảng Category.Mỗi câu hỏi sẽ có nhiều câu trả lời.
* Bảng Answers chưa trả lời cho từng câu hỏi cũng do các thành viên đăng lên.Một câu trả lời sẽ thuộc một câu hỏi.

2.4 Kiến trúc web

Ta cần một hệ thống cho phép người dùng đặt câu hỏi và trả lời, hệ thống cũng cho phép tìm kiếm câu hỏi, tìm kiếm những câu hỏi tương tự, trò chuyện để nhận được những phản hồi nhanh nhất có thể.

Phạm vi nghiên cứu của đề tài xoay quay 3 đối tượng chính là hệ thống web, ứng dụng di động và máy chủ (Server) để xử lý giải thuật và trao đổi thông tin với Web API

Vậy để hiện thực được một hệ thống như vậy chúng ta cần những gì:

2.4.1 Quản lý theo mô hình MVC

**Model**: có nhiệm vụ thao tác với cơ sở dữ liệu, nghĩa là nó sẽ chứa tất cả các hàm, các phương thức truy vấn trực tiếp với dữ liệu và controller sẽ thông qua các hàm, phương thức đó để lấy dữ liệu rồi gửi qua **View**

**View**: có nhiệm vụ tiếp nhận dữ liệu từ controller trả vể và hiển thị nội dung sang những gì mà người dùng có thể thấy được.

**Controller**: đóng vài trò trung gian giữa Model và View. Controller có nhiệm vụ diều hướng yêu cầu từ người dùng và trả dữ liệu từ tầng Model mà người dùng mong đợi.

Ý nghĩa của mô hình MVC:

* Dễ dàng quản lý mã nguồn,các tầng có nhiệm vụ riêng
* Dễ dàng nâng cấp mã nguồn
* Có thể dễ dàng tìm lỗi khi gặp sự cố

2.4.2 Công cụ hiện thực

Hệ thống web được xây dựng trên nền tảng công nghệ Javascript gồm 4 công nghệ Angular Js, Express , MongoDB, NodeJs.Nền tảng từ 4 công nghệ nêu trên được gọi tắt là MEAN Js

Angular Js

AngularJS là một thư viện Javascript dành cho giao diện người dùng, được phát triển bởi Google.

Ban đầu mục tiêu của Angular là để xây dựng các ứng dụng dựa trên tiêu chuẩn MVC (Model - View - Controller), sau đó Angular dần phát triển và tiến gần hơn về với MVVM và MVP. Sau đó Google đã định nghĩa nó lại là MVW (Model-View-Whatever) để ám chỉ Angular là một framework có tính chất làm mọi thứ mà ta cần.

Angular Js cho phép ta thao tác dữ liệu phía server trả về và hiển thị cho người dùng thấy ở phần giao diện.

Angular Js cho phép người dùng lập luận logic if, else, loop, hiển thị, ẩn… các thành phần của giao diện v.v….

Angular Js cũng cho phép chúng ta quản lý mã nguồn 1 cách linh hoạt bằng việc tổ chức mã nguồn theo tính năng và thành phần của hệ thống web.

Express JS

Express là một web application framework cho NodeJS, cung cấp các tính năng mạnh mẽ cho việc xây dựng một ứng dụng web.

ExpressJS là framework phổ biến và được sử dụng rộng rãi nhất của NodeJS. Ý tưởng đằng sau ExpressJS là đưa đến một framework nhẹ, dễ dàng tiếp cận để phát triển các ứng dụng web từ nhỏ đến lớn.

Express cũng có thể sử dụng để xây dựng một API mạnh mẽ và thân thiện với người dùng, vì nó cung cấp rất nhiều tiện ích HTTP và là một phương thức trung gian giúp cho các thành phần của hệ thống kết nối với nhau.

Express JS sẽ đóng vai trò trung gian cho hệ thống MEAN Js khi tiếp nhận dữ liệu phía người dùng gửi lên hoặc 1 yêu cầu do người dùng yêu cầu phía hệ thống xử lý.Từ đó Express sẽ điều hướng và quyết định nên làm gì tiếp theo cho yêu cầu đó.Express Js có thể chấp nhận hoặc từ chối yêu cầu từ phía người dùng nếu không đủ điều kiện thực hiện.

Node JS

NodeJS là một nền tảng server được xây dựng dựa trên Javascript Engine (V8 Engine)

NodeJs là một mã nguồn mở, đa nền tảng cho phát triển các ứng dụng phía Server và các ứng dụng liên quan đến mạng. Ứng dụng NodeJs được viết bằng Javascript và có thể chạy trong môi trường NodeJs trên hệ điều hành Window, Linux...

Node Js chạy phía server side cũng giống như các ngôn ngữ server side như .Net, PHP, Java cho phép xử lý logic, thao tác với cơ sở dữ liệu.

Node Js thực thi các tác vụ bất đồng bộ cho phép hệ thống chạy không theo thứ tự và có thể chạy nhiều tác vụ cùng 1 lúc giúp tiết kiệm thời gian đợi của hệ thống.

Dưới đây là các lĩnh vực mà Node.js được sử dụng như là một sự lựa chọn hoàn hảo:

* Các ứng dụng về I/O.
* Các ứng dựng về luồng dữ liệu.
* Các ứng dụng về dữ liệu hướng đến thời gian thực.
* Các ứng dụng dựa vào JSON APIs.
* Các ứng dụng Single Page Application.

MongoDB

MongoDb là cơ sở dữ liệu NoSQL được dùng cho việc lưu trữ dự liệu hệ thống.

NoSQL được phát triển trên Javascript Framework với kiểu dữ liệu là JSON và dạng dữ liệu theo kiểu key và value (1 đặc trưng về dữ liệu trong JSON). Với NoSQL bạn có thể mở rộng dữ liệu mà không lo tới những việc như tạo khóa ngoại, khóa chính, kiểm tra ràng buộc, .v.v…Vì NoSQL không hạn chế việc mở rộng dữ liệu nên tồn tại nhiều nhược điểm như: sự phục thuộc của từng bản ghi, tính nhất quán, toàn vẹn dữ liệu. NoSQL cho hiệu suất cao hơn khi giải quyết những bài toán lớn về hệ thống thông tin, phân tán hay lưu trữ dữ liệu.

Với các CSDL quan hệ, chi phí triển khai cũng như phát triển các ứng dụng sử dụng CSDL quan hệ cũng rất tốn kém và điều quan trọng được bộc lộ rõ khi truy vấn 1 lượng bản ghi lớn trong thời gian rất dài.

Tiếp đó những thiết bị cầm tay như smartphone thì không phù hợp để triển khai CSDL quan hệ vì dung lượng bộ nhớ thấp và khả năng xử lý kém hơn máy tính.



Biểu đồ 2. 1: So sánh tốc độ insert dữ liệu giữa MongoDB và SQL Server

(Nguồn: <https://quantrimang.com/so-sanh-hieu-suat-hoat-dong-cua-mongodb-va-sql-server-2008-84584>)

Ta có thể thấy việc MongoDB chèn dữ liệu nhanh hơn SQL Server tới hơn 100 lần



Biểu đồ 2. 2: So sánh tốc độ truy vấn dữ liệu giữa MongoDB và SQL Server

(Nguồn: <https://quantrimang.com/so-sanh-hieu-suat-hoat-dong-cua-mongodb-va-sql-server-2008-84584>)

Ta có thể thấy tốc độ truy vấn 50,000 dòng dữ liệ của MongoDB cao gần gấp 30 lần so với SQL Server

Mô Hình MEAN JS



Hình 2. 14: Mô Hình Hoạt Động Của Một Ứng Dụng Mean Js

(Nguồn google)

Hệ thống sẽ sử dụng 1 file server.js có tác dụng khởi động ứng dụng web

Từ file server.js hệ thống sẽ trả về giao diện cho người dùng.Từ phía người dùng ta cũng sẽ có mô hình MVC do Angular Js đảm nhận logic từ phía giao diện sau đó chuyển dữ liệu về lại cho phía server xử lý

Phía server cũng áp dụng mô hình MVC controller sẽ xử lý logic và model dùng để thao tác với đối tượng và cơ sở dữ liệu và sẽ trả dữ liệu về giao diện cho người dùng.

2.5 Ứng dụng trên điện thoại

Xây dựng trên nền tảng Android sử dụng ngôn ngữ lập trình Java đồng thời kết hợp với Web API của hệ thống để trao đổi dữ liệu.

2.5.1 Quản lý theo module

Hệ thống sẽ không tuân theo một mô hình nhất định mà được chia ra theo từng thành phần chức năng. Kiến trúc mô-đun (module) cho phép chia nhỏ bài toán (hay yêu cầu) của phần mềm thành các phần hầu như không trùng lắp. Các module sẽ cung cấp một giao diện (Interface) cho các thành phần khác trong hệ thống có thể gọi và sử dụng.

2.5.2 **Công cụ thực hiện**

Retrofit là một thư viên giúp hệ thống android thao tác trao đổi dữ liệu, thông tin với Web API. Thư viện cho phép người dùng thực thi các yêu cầu GET, POST, DELETE, PUT lên Web API.

Ngoài ra còn sử dụng các thư viện Design-support của google để thiết kế và hiện thực giao diện người dùng, thư viện facebook-sdk cho phép người dùng đăng nhập, đăng ký thông qua tài khoản Facebook.

2.6. Server

Server là nơi tiếp nhận những câu hỏi của người dùng từ Web API sau đó áp dụng, xử lý các giải thuật về phân loại câu hỏi, đo độ tương tự của câu hỏi đó với các câu hỏi trong cơ sở dữ liệu để tìm ra câu hỏi gần với câu người dùng tìm kiếm nhất và trả về lại cho Web API, để Web API xử lý và hiển thị lên cho người dùng cuối.

Công nghệ sử dụng:

Server được viết trên ngôn ngữ lập trình Java, sử dụng giao thức TCP trong lập trình mạng để trao đổi dữ liệu với Web API.

Thư viện sử dụng

vnTokenizer: thư viện cho phép chúng ta tách các từ trong trong văn bản Tiếng Việt thành các từ đơn hoặc từ ghép có nghĩa.

Stanford Classifier: là công cụ sử dụng trong học máy, hiện thực dựa trên phương pháp Maximum Entropy Classifier, ứng dụng trong việc phân loại các câu hỏi trong Tiếng việt thành các loại câu về “cái gì (what)”, “tại sao (why)”, “như thế nào (how)”…

MongoDB JDBC Driver: thư viện cho phép kết nối, lấy, truy vấn cơ sở dữ liệu MongoDB trên Java.

CHƯƠNG 3 – XÂY DỰNG CƠ CHẾ TRẢ LỜI TỰ ĐỘNG

3.1 Kiến trúc tổng quan



Hình 3. 1: Kiến trúc tổng quan cơ chế trả lời tự động

Các câu hỏi của người dùng khi đã được chuyển tới Web API sẽ tiếp tục được chuyển tới một máy chủ viết bằng ngôn ngữ Java để tiến hành xử lý, áp dụng giải thuật đo độ tương tự nhằm tìm ra các câu hỏi phù hợp với câu hỏi đó đồng thời trả kết quả về cho Web API để Web API xử lý cung cấp cho người dùng.

Câu hỏi nhận được sẽ đi qua 4 giai đoạn:

* Giai đoạn 1: Truy vấn các câu hỏi để lưu vào danh sách các câu hỏi tương tự sử dụng Full Text Search của MongoDB. Đây là bước đầu tiên để lọc dữ liệu, loại bỏ các câu không liên quan nhầm rút ngắn thời gian xử lý của giải thuật.
* Giai đoạn 2: Phân loại câu hỏi đã nhập vào. Đồng thời lọc lại danh sách câu hỏi tương tự đã có, chỉ lấy những câu cùng loại với câu do người dùng nhập vào.

Ví dụ: Học kỳ dự thính là gì?

→ Sau khi đi qua bộ phân loại, ta sẽ biết được câu hỏi này thuộc loại “what”.

* Giai đoạn 3: Tìm câu hỏi tương tự: Các câu hỏi từ người dùng nhập vào và các câu hỏi trong danh sách câu hỏi sẽ được chuyển thành chữ in thường và tách các từ có nghĩa thông qua công cụ VnTokenizer. Từng câu hỏi trong danh sách câu hỏi sẽ được so sánh với câu hỏi từ người dùng nhập vào thông qua giải thuật đo độ tương tự và được gán điểm số, điểm số càng cao thì càng chứng minh được câu này tương tự với câu người dùng nhập vào.
* Giai đoạn 4: In ra danh sách các câu hỏi tương tự. Ở đây chúng em sẽ lấy 3 câu hỏi được gán điểm số cao nhất và trả về cho hệ thống Web xử lý

3.2 Phương pháp:

3.2.1 Lọc dữ liệu sử dụng Full Text Search

Để tăng hiệu năng, thời gian truy xuất dữ liệu trong cơ sở dữ liệu, đồng thời loại bỏ những dữ liệu không liên quan đến câu hỏi từ người dùng nhập vào, chúng ta sẽ sử dụng một kỹ thuật gọi là Full Text Search (FTS)

FTS là một kỹ thuật tìm kiếm trên cơ sở dữ liệu dạng văn bản. Sở dĩ FTS có hiệu năng và tốc độ truy xuất cao hơn các kỹ thuật tìm kiếm thông thường là nhờ sử dụng phương pháp đánh chỉ mục (Indexing). Mỗi từ trong văn bản mới chèn vào sẽ được lưu vào một mảng gồm địa chỉ các văn bản chứa từ đó

Ví dụ:

D1= “Đây là văn bản thứ nhất”.

D2= “Đây là hai văn bản”.

D3=” Một hai”.

Việc tạo index như vậy sẽ giúp cho việc tìm kiếm nhanh hơn. Thay vì phải tìm kiếm từng văn bản 1, ta chỉ tìm kiếm dựa trên các phép kết.

Các câu hỏi sau khi tìm kiếm được sẽ là một danh sách mà mỗi câu hỏi trong danh sách này sẽ chứa ít nhất 1 từ trùng với các từ trong câu hỏi người dùng nhập vào.

3.2.2 Tách từ

Việc đầu tiên trong xử lý câu hỏi từ người dùng nhập vào đó là Tách từ. Tách từ là phương pháp xử lý xác định các từ có nghĩa trong một văn bản Tiếng Việt. Mục đích của việc tách từ nhầm lọc ra những từ có nghĩa trong Tiếng Việt để nâng cao khả năng tìm kiếm các câu hỏi tương tự.

Ví dụ:

Cho một câu: “Xã hội ngày càng phát triển”

Sau khi tiến hành xử lý tách từ, ta sẽ nhận được một câu: “Xã\_hội ngày càng phát\_triển”.

Trong luận văn này, chúng em sử dụng công cụ tách từ vnTokenizer của tác giả Lê Hồng Phương, cộng cụ này được sử dụng trong quá trình phân tích câu hỏi người dùng nhập vào và phân tích các câu hỏi trong cơ sở dữ liệu.

3.2.3 Phân loại câu hỏi sử dụng Maximum entropy classifier

Sau khi câu hỏi từ người dùng nhập vào được chuyển thành chữ in thường và đi qua giai đoạn tách từ, chúng ta sẽ phân loại, kiểm tra xem câu hỏi đó thuộc loại nào trong các loại “What”, “When”, “Where”,”Who”, ”How”, ”YesNo”. Để phân loại thanh các câu hỏi như vậy chúng ta sẽ sử dụng phương pháp tính Maximum entropy

Maximum entropy là gì?

Maximum entropy là một kỹ thuật học máy có mục địch. Nó cung cấp các ước tính ít sai lệch nhất có thể dựa vào những thông tin nhất định cho trước được gọi là tập dữ liệu mẫu.

Maximum entropy classifier

Maximum entropy classifier là một bộ phân loại thường được sử dụng trong xử lý ngôn ngữ tự nhiên. Nó cho phép xây dựng các đặc trưng từ tập dữ liệu huấn luyện. Tự động phân loại 1 câu theo 1 chủ đề dựa trên các đặc trưng đó.

-Để áp dụng Maximum entropy classifier chúng ta cần chọn ra một tập hợp các đặc trưng từ tập dữ liệu dữ liệu huẩn luyện để cài đặt các truy vấn. Cụ thể, chúng ta sử dụng số lượng từ xuất hiện như là các đặc trưng.

Với mỗi từ ta đưa ra được một đặc tính như sau:

Trong đó:

- là số lần từ *w* xuất hiện trong loại d.

- *N(d)* là số lượng từ trong loại d.

Như vậy nêu 1 từ w thường xuyên xuất hiện trong loại d nào đó, ta sẽ tính được trọng số của từ này. Nếu 1 từ có trọng số cao thì khả năng nó thuộc loại d cao.

Ví dụ: Trong lĩnh vực phân loại các loại câu hỏi trong bai toán này, chúng ta có 7 loại câu hỏi là What, Where, When, Which, How, Why, YesNo.

Các thống kê dữ liệu cho rằng 90% các loại câu hỏi When có chứa từ “khi nào”. Như vậy nếu D có chứa từ “khi nào” thì xác suất nó thuộc loại câu hỏi When là 90%, đây cũng gọi là 1 ràng buộc của mô hình và xác suất vào các loại còn lại là 2,5 %

Stanford classifier là một công cụ cho phép chúng ta chia dữ kiệu thành các chủ đề từ một tập các dữ liệu đào tạo cho trước, tạo ra một bộ phân loại dưa trên Maximum entropy classifier. Trong bài toán này, chúng ta sẽ phân loại các câu hỏi trong Tiếng Việt thành 6 chủ để: “What”, “Where”, “When”, “How”, “Why”, “YesNo”. Các bước thực hiện như sau :

- Trước tiên chúng ta cần tạo ra một tập dữ liệu đặc trưng, Mỗi chủ đề chúng ta sẽ lấy từ 25-50 câu. Tập dữ liệu đặc trưng là một tập tin có phẩn mở rộng là “.train”

- Tạo một tập tin có phần mở rộng là “.test” chứa nội dung câu hỏi và chủ đề chúng ta mong đợi sau khi chạy qua chương trình.

- Tiếp đó là tập tin có phần mở rộng là “.prop” trình bày các thuộc tính, các cấu trúc quy định trong tập tin .train mà chương trình sẽ học.

- Sau khi có cả 3 tập tin trên, chúng ta sẽ tiến hành cho chương trình tính toán.

3.2.4 Đo độ tương tự giữa các câu hỏi.

- Giải thuật dựa trên 3 yếu tố chính sau:

* Sự ánh xạ độ tương tự từ ngữ: 1 chuỗi ít khác biệt có thể xem là tương tự.
* Sự thay đổi trật tự từ: 2 chuỗi bao hàm từ giống nhau nhưng theo 1 trật tự khác có thể được xem là tương tự
* Không phụ thuộc vào ngôn ngữ : Giải thuật không phải chỉ hoạt động được với tiếng anh mà còn nhiều thứ tiếng khác
* Ví dụ ta có 2 câu hỏi như sau:

1. Bao nhiêu tiền một tín chỉ?
2. Một tín chỉ bao nhiêu tiền?

Là 2 câu có độ tường tự bằng nhau vì câu 1 và câu 2 chỉ khác nhau ở vị trí các từ.

* Sự giống nhau giữa hai câu s1 và s2 được tính bằng số lượng từ cùng xuất hiện ở trên 2 câu, nêu số lượng từ cùng xuất hiện càng nhiều chứng tỏ 2 câu đó có khả năng giống nhau càng cao, từ đó chúng ta có công thức:

Độ tương tự(s1,s2) =

* Đô tương tự đo được bằng số lượng từ (nếu sử dụng 1-gram) hoặc số lượng các cặp từ (nếu chúng ta sử dụng 2-gram)… chung ở cả hai chuỗi nhân 2 chia cho tổng số các cặp từ trong hai chuỗi. Giải thuật sẽ đo độ tương tự giữa 2 chuỗi bằng 0 nếu như 2 chuỗi đó khác nhau hoàn toàn, bằng 1 nếu như 2 chuỗi đó giống hệt nhau. Nêu kết quả cho ra càng gần bằng 1 thì khả năng 2 chuỗi đó tương tự nhau càng cao và ngược lại nếu kết quả càng gần với số 0 thì 2 chuỗi đó càng khác nhau.

Ví dụ : Ta có 1 bài toán sau:

- Đo độ tương tự giữa 2 chuỗi FRANCE và FRENCH.

- Chúng ta sẽ sử dụng mô hình 2-gram, tách từng chuỗi ra thành các cặp mỗi cặp 2 chữ cái

S1 = FRANCE: {FR, RA, AN, NC, CE}

S2 = FRENCH: {FR, RE, EN, NC, CH}

* Để so sánh từ FRANCE và FRENCH, các số liệu được tính như sau

= = 0.4

- Độ tương tự đo được luôn nằm trong khoảng từ 0 – 1.Ở giải thuật này ta tính được độ tương tự của 2 từ FRANCE và FRENCH là 40%.

CHƯƠNG 4–NGHIÊN CỨU THỰC NGHIỆM

4.1 Mục tiêu thử nghiệm

Để đo độ hiệu quả của hệ thống, kiểm trả độ chính xác của hệ thống khi hoạt động chúng em xây dựng bộ thử nghiệm.

Bộ thử nghiêm có 2 phần chính là:

* Phần cơ sở dữ liệu.
* Chương trình chạy để đánh giá.

4.2 Kế hoạch thử nghiệm Kế hoạch thử nghiệm sẽ bao gồm 4 phần:

- Lấy 1 câu hỏi tương tự với điểm số cao nhất sử dụng mô hình 1-gram.

- Lấy 3 câu hỏi tương tự với điểm số cao nhất với mô hình 1-gram.

- Lấy 1 câu hỏi tương tự sử dụng mô hình 2-gram.

- Lấy 3 câu hỏi tương tự với điểm số cao nhất sử dụng mô hình 2-gram.

4.3 Đánh giá giải thuật:

4.3.1 Tổng quan mô hình hiện thực giải thuật



Hình 4. 1: Tổng quan mô hình giải thuật

Từng câu hỏi thử nghiệm sẽ được con người gán từ 1-3 câu hỏi tương tự với nó. Sau đó từng câu hỏi thử nghiệm này sẽ được xử lý bởi hệ thống để tìm ra câu hỏi tương tự do hệ thống xử lý. Lấy câu hỏi tương tự do hệ thống xử lý đem so sánh với câu do người gán, nếu trùng nhau chứng tỏ giải thuật chạy đúng, ngược lại khác nhau thì giải thuật chạy sai. Lần lượt làm như vậy với 30 câu hỏi thử nghiệm để kiểm tra hệ thống.

4.3.2 Mô hình 1-gram

- Để đánh giá tính đúng đắn của giải thuật chúng ta sẽ chạy bộ đánh giá.Việc chạy bộ đánh giá được chia ra 2 trường hợp:

* Lấy 1 câu hỏi có số điểm cao nhất
* Lấy 3 câu hỏi có số điểm cao nhất
* Ví dụ đầu vào:

Câu hỏi nhập liệu: Làm gì khi kết quả điểm danh không chính xác?

Dữ liệu giải thuật truy vấn trong hệ thống:

* Em phải làm sao khi kết quả điểm danh TOEIC không chính xác? (0.7058823529411765)
* Em phải làm sao khi kết quả điểm danh Trung tâm TOEIC công bố không chính xác? (0.631578947368421)

|  |  |  |
| --- | --- | --- |
| **Tiêu chí đánh giá** | **Lấy 1 câu hỏi có số điểm cao nhất** | **Lấy 3 câu hỏi có số điểm cao nhất** |
| Số lượng câu đánh giá | 30 | 30 |
| Số lượng câu nguồn để kiểm tra | 44 | 44 |
| Số lượng câu đúng |  |  |
| Tỉ lệ đúng | 56% | 80% |

* Sau đây là bảng so sánh 2 trường hợp so sánh:
* Vậy ta có thể thấy rõ việc lấy 1 câu hỏi có số điểm cao nhất cho kết quả đúng thấp hơn .
* Việc lấy ra 3 câu hỏi tương tự được khuyến khích hơn do tỉ lệ xuất hiện câu hỏi tương tự trong cở sở dữ liệu cao hơn.
* Tạo điều kiện cho người sử dụng dễ dàng tìm được câu trả lời cho thắc mắc của mình hơn.

4.3.2 Mô hình 2-gram

- Để đánh giá tính đúng đắn của giải thuật chúng ta sẽ chạy bộ đánh giá.Việc chạy bộ đánh giá được chia ra 2 trường hợp:

* Lấy 1 câu hỏi có số điểm cao nhất
* Lấy 3 câu hỏi có số điểm cao nhất
* Ví dụ đầu vào:

Câu hỏi nhập liệu: Cách làm đơn rút môn học như thế nào ?

Dữ liệu giải thuật truy vấn trong hệ thống :

* Muốn rút môn học phải làm đơn như thế nào? (0.46153846153846156)
* Em muốn hỏi là bây giờ em muốn rút môn học vì lý do thời gian học không phù hợp với em. Vậy ở thời điểm này em có được làm đơn xin rút môn học không? Và nếu được thì cách làm đơn như thế nào ạ? Em mong văn phòng hỗ trợ em. Em cám ơn (0.18181818181818182)

|  |  |  |
| --- | --- | --- |
| Tiêu chí đánh giá | Lấy 1 câu hỏi có số điểm cao nhất | Lấy 3 câu hỏi có số điểm cao nhất |
| Số lượng câu đánh giá | 30 | 30 |
| Số lượng câu nguồn để kiểm tra | 44 | 44 |
| Số lượng câu đúng |  |  |
| Tỉ lệ đúng | 60% | 63% |

* Vậy ta có thể thấy rõ việc lấy 1 câu hỏi có số điểm cao nhất cho kết quả đúng thấp hơn
* Việc lấy ra 3 câu hỏi tương tự được khuyến khích hơn do tỉ lệ xuất hiện câu hỏi tương tự trong cở sở dữ liệu cao hơn.
* Tỉ lệ đúng không chênh lệch nhau nhiều.

CHƯƠNG 5– TỔNG KẾT

Các vấn đề đã làm được:

* Xây dựng hệ thống hỏi đáp bằng web và mobile app.
* Xây dựng được hệ thống API cho cả web và mobile cùng sử dụng.
* Hiện thực được các chức năng cần có của một hệ thống hỏi đáp.
* Hiện thực được giải thuật tìm kiếm cho phép hệ thống trả lời tự động người dùng.
* Phòng chat cho phép người dùng nhận được câu trả lời nhanh hơn không phải chờ đợi.
* Tối ưu tốc độ web và truy vấn cơ sở dữ liệu.

Các vấn đề chưa làm được:

* Hệ thống chưa kiếm các câu hỏi tương tự đúng 100%.
* Không có quản lý ảnh đại diện cho người dùng.
* Chưa có thông báo cho người dùng câu hỏi đã có người trả lời hay chưa.
* Xử lý bài viết không hợp lệ với tiêu chí của diễn đàn.
* Bảo mật cho hệ thống.

Hướng phát triển cho tương lai:

Áp dụng thêm công nghệ thời gian thực thông báo cho người dùng khi có bài viết mới. Phát triển robot trả lời tự động cho người dùng những câu hỏi đơn giản.

**TÀI LIỆU THAM KHẢO**

**Tiếng Việt**

1. Blog kỹ thuật máy tính, Full Text Search, *Từ khái niệm đến thực tiễn (Phần 1)*, <http://ktmt.github.io/blog/2013/10/27/full-text-search-engine/>
2. Lưu Tuấn Anh, *Thuật toán Entroy cực đại*, <http://viet.jnlp.org/kien-thuc-co-ban-ve-xu-ly-ngon-ngu-tu-nhien/machine-learning-trong-nlp/thuat-toan-entropy-cuc-dai>

**Tiếng Anh**

1. Kamal Nigam, John Lafferty, Andrew McCallum, Using Maximum Entropy for Text Classification

<http://www.kamalnigam.com/papers/maxent-ijcaiws99.pdf>

1. [Le-Hong, P.](http://mim.hus.vnu.edu.vn/phuonglh/publications?f%5bauthor%5d=5), [T M H. Nguyen](http://mim.hus.vnu.edu.vn/phuonglh/publications?f%5bauthor%5d=2), [A. Roussanaly](http://mim.hus.vnu.edu.vn/phuonglh/publications?f%5bauthor%5d=10), and [T V. Ho](http://mim.hus.vnu.edu.vn/phuonglh/publications?f%5bauthor%5d=11), A hybrid approach to word segmentation of Vietnamese texts (2008)
2. Simon White, How to Strike a Match http://www.catalysoft.com/articles/StrikeAMatch.html
3. Robin D. Burke, Kristian J. Hammond, Vladimir Kulyukin,Steven L. Lytinen, Noriko Tomuro, and Scott Schoenberg in Question Answering from Frequently Asked Question Files
4. An Analysis of the AskMSR Question-Answering System(Eric Brill, Susan Dumais and Michele Banko):
5. Deepak Ravichandran and Eduard HovyLearning Surface Text Patterns for a Question Answering System