BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC CẦN THƠ**

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

**LUẬN VĂN TỐT NGHIỆP ĐẠI HỌC**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

****

ĐỀ TÀI

**XÂY DỰNG HỆ THỐNG**

**QUẢN LÝ SÂN CẦU LÔNG**

**VÀ BÁN SẢN PHẨM ONLINE**

**KẾT HỢP CHAT BOT**

**SINH VIÊN THỰC HIỆN:**

**Lê Minh Chiến**

**MSSV: B1606935**

**Khóa: 42**

**Cần Thơ, Tháng 01/2021**

BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC CẦN THƠ**

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

**BỘ MÔN TIN HỌC ỨNG DỤNG**

**LUẬN VĂN TỐT NGHIỆP ĐẠI HỌC**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

****

ĐỀ TÀI

**XÂY DỰNG HỆ THỐNG**

**QUẢN LÝ SÂN CẦU LÔNG**

**VÀ BÁN SẢN PHẨM ONLINE**

**KẾT HỢP CHAT BOT**

**GIÁO VIÊN HƯỚNG DẪN: SINH VIÊN THỰC HIỆN:**

**Th.S Nguyễn Cao Hồng Ngọc Lê Minh Chiến**

**MSSV: B1706563**

**Khóa: 43**

**Cần Thơ, Tháng 01/2021**

LỜI CẢM ƠN

Đầu tiên, em xin gửi lời cảm ơn chân thành đến gia đình và bạn bè đã động viên em trong suốt quá trình học tập.

Em xin cảm ơn Bộ môn Công nghệ thông tin, khoa Công nghệ thông tin và Truyền thông, Trường Đại học Cần Thơ đã tạo điều kiện tốt nhất để em thực hiện đề tài này.

Dưới sự hướng dẫn của ThS. Nguyễn Cao Hồng Ngọc, em đã hoàn thành đề tài “Xây dựng web quản lý sân cầu lông kết hợp chatbot”. Em xin bày tỏ lời cảm ơn sâu sắc đến cô vì đã tận tình giúp đỡ, định hướng cho em trong suốt thời gian qua.

Cảm ơn thầy dạy cầu lông của em – thầy Diệp Xuân Tài đã đưa ra đề nghị cũng như ý tưởng để em có ý tưởng thực hiện đề tài luận văn này.

Trong quá trình thực hiện, em đã tích lũy được thêm nhiều kiến thức, giúp bản thân em dần hoàn thiện hơn. Đồng thời, kết hợp với kiến thức nền tảng đã có trong thời gian học tập những năm qua, em đã hoàn thành đề tài này.

Em rất mong nhận được sự góp ý, nhận xét từ quý thầy để đề tài được hoàn thiện nhất.

Cuối lời, em xin chúc quý thầy dồi dào sức khỏe, hạnh phúc và đạt được nhiều thành công trong công việc và cuộc sống.

Trân trọng!

Cần Thơ, ngày 19 tháng 12 năm 2020

Sinh viên thực hiện

Lê Minh Chiến

**NHẬN XÉT CỦA CÁN BỘ HƯỚNG DẪN**

Cần Thơ, ngày … tháng … năm ….  
Cán bộ hướng dẫn

Mục lục

[Mục lục i](#_Toc62855773)

[Danh mục hình ảnh ii](#_Toc62855774)

[Danh mục bảng iii](#_Toc62855775)

[Danh mục các ký hiệu, chữ viết tắt iv](#_Toc62855776)

[Tóm tắt v](#_Toc62855777)

[Abstract vi](#_Toc62855778)

[PHẦN GIỚI THIỆU 1](#_Toc62855779)

[1. Đặt vấn đề 1](#_Toc62855780)

[2. Các nghiên cứu liên quan 2](#_Toc62855781)

[3. Mục tiêu đề tài 3](#_Toc62855782)

[4. Đối tượng và phạm vi nghiên cứu 3](#_Toc62855783)

[5. Phương pháp nghiên cứu 3](#_Toc62855784)

[5.1. Về lý thuyết 3](#_Toc62855785)

[5.2. Về kỹ thuật 4](#_Toc62855786)

[6. Bố cục luận văn 4](#_Toc62855787)

[PHẦN NỘI DUNG 5](#_Toc62855788)

[CHƯƠNG 1: CƠ SỞ LÝ THUYẾT 5](#_Toc62855789)

[1.1. Trí tuệ nhân tạo 5](#_Toc62855790)

[1.2. Chatbot 6](#_Toc62855791)

[1.2.1. Giới thiệu chatbot 6](#_Toc62855792)

[1.2.2. Các khái niệm trong chatbot 7](#_Toc62855793)

[1.2.3. Các phương pháp xây dựng chatbot 7](#_Toc62855794)

[1.2.4. Xử lý ngôn ngữ tự nhiên (NLP) 9](#_Toc62855795)

Danh mục hình ảnh

[Hình 1.1 Trợ lý ảo Siri tích hợp trên Iphone 2](#_Toc62855765)

[Hình 1.2 Chatbot FPT.AI trên nền tảng Facebook 2](#_Toc62855766)

[Hình 1.1 Mối liên hệ giữa AI, Machine learning và Deep learning 5](#_Toc62855767)

[Hình 1.2 Cơ chế hoạt động chung của một chatbot 6](#_Toc62855768)

[Hình 1.3 Minh họa một Artificial Neural Network 8](#_Toc62855769)

[Hình 1.4 Các ứng dụng của NLP 10](#_Toc62855770)

Danh mục bảng

**No table of contents entries found.**

Danh mục các ký hiệu, chữ viết tắt

|  |  |
| --- | --- |
| **Ký hiệu, chữ viết tắt** | **Diễn giải** |
| AI | Artificial Intelligence |
| API | Application Program Interface |
| CSDL | Cơ sở dữ liệu |
| DL | Deep Learning |
| HTTP | HTTP HyperText Transfer Protocol |
| JSON | JavaScript Object Notation |
| ML | Machine Learning |
| MVC | Model - View - Controller |
| NLP | Natural Language Processing |
| REST | REpresentational State Transfer |
| SVM | Support Vector Machine |
| URL | Uniform Resource Locator |
| XML | eXtensible Markup Language |

Tóm tắt

Nhận thấy nhu cầu của chủ sân cầu lông là cần một ứng dụng để giúp họ quản lý dữ liệu sân tốt hơn khi phải cạnh tranh với nhau trong thời buổi công nghệ. Việc xây dựng một hệ thống website cho phép họ có thể dễ dàng quản lý dữ liệu kinh doanh của mình cũng như việc kết hợp bán sản phẩm online trên hệ thống là lý do giúp em thực hiện đề tài “Xây dựng hệ thống quản lý sân cầu lông và buôn bán sản phẩm kết hợp với chatbot”.

Giao diện website được xây dựng bằng thư viện ReactJS được phát triển và cung cấp bởi Facebook dựa trên nền tảng nodeJS, một thư viện rất nổi tiếng với việc sử dụng component, render dữ liệu dựa vào state và tạo ứng dụng dưới dạng single page. Back-end của hệ thống được xây dựng bằng NodeJS kết nối đến cơ sử dữ liệu NoSQL MongoDB.

Về phần AI (Artificial intelligence) – trí tuệ nhân tạo đã được áp dụng vào nhiều lĩnh vực trong cuộc sống và nâng cao chất lượng con người. Trong hệ thống này em sẽ áp dụng công nghệ này để tạo chat bot giúp trả lời câu hỏi tự động liên quan đến hệ thống.

Kết quả đề tài luậ văn này sẽ là hệ thống website dành cho hai nhóm người dùng chính, chủ sân cầu lông và người dùng mua các sản phẩm trên hệ thống. Các chủ sân có thể dùng các chức năng chính của hệ thống như quản lý sân, quản lý hoá đơn, xem doanh thu, đăng bán sản phẩm trên hệ thống,… Đối với người dùng thông thường họ có thể dùng chatbot để giúp họ giải đáp thắc mắc, mua sắm trên website và thanh toán online. Về phần quản trị viên có thể huấn luyện mô hình chatbot nhằm mở rộng kiến thức của chatbot.

Abstract

PHẦN GIỚI THIỆU

* 1. Đặt vấn đề

Ngày nay, việc quản lý dữ liệu trong kinh doanh một cách số hoá không chỉ được các chủ doanh nghiệp lớn quan tâm mà còn có những người kinh doanh nhỏ lẻ rất quan tâm và trong đó có các chủ sân cầu lông. Họ cần một ứng dụng giúp mình quản lý các dữ liệu trong lúc kinh doanh như doanh thu, hoá đơn, hàng hoá,… Và bên cạnh đó họ cũng bán các sản phẩm liên quan tới cầu lông như quần, áo, giày, vợt,… để tăng doanh thu. Những đối tượng này quan tâm việc khách hàng sử dụng dịch vụ thuê sân hơn và họ chỉ xem việc kinh doanh các sản phẩm còn lại như một cách tăng doanh thu nên họ rất ít đầu tư tiền bạc để nhờ các công ty phát triển cho mình một phần mềm quản lý hay một trang web để buôn bán các sản phẩm mà thường đăng các sản phẩm của mình trên facebook. Vì vậy một hệ thống giúp họ có thể vừa quản lý dữ liệu của sân cầu lông của mình và cho họ một route riêng trên hệ thống để đăng tải các sản phẩm tiện cho việc kinh doanh là một nhu cầu thiết yếu.

Khái niệm chatbot được nhắc đến đầu tiên vào năm 1950, bởi Alan Turing với ý tưởng “xây dựng một chương trình máy tính giả dạng con người để thực hiện cuộc trò chuyện giữa chương trình và con người thật”. Và những năm gần đây, độ tiện lợi của chatbot ngày càng cao khi nó đã cho thấy được tiện ích của mình và xuất hiện ngày càng nhiều hơn trên các mạng xã hội, website, cho đến các thiết bị di động. Chatot có thể giúp cho con người thực hiện một hành động hoặc cung cấp tri thức cho người dùng. Chatbot có thể giap tiếp nhận mệnh lệnh từ người dùng thông qua tin nhắn hoặc giọng nói. Để xác định ý định (intent) của người dùng, chatbot sẽ dùng hệ thống xử lý ngôn ngữ tự nhiên (NLP) để phân tích và dùng các thuật toán máy học (ML) để phân loại câu (text classification) và đưa ra hành động hoặc câu trả lời phù hợp nhất.

Việc kết hợp một chatbot giúp trả lời khách hàng hiểu rõ hơn về hệ thống sẽ giúp cho họ tiếp cận hệ thống một cách thuận tiện hơn và giúp cho chủ sân đỡ phải trả lời những câu hỏi thường gặp của khách hàng nếu chúng lặp đi lặp lại. Việc lựa chọn website để xây dựng hệ thống cùng nhầm giúp cho khách hàng dễ tiếp cận hơn với hệ thống thay vì phải cài đặt các phầm mềm trên máy tính của họ cũng giúp cho họ dễ tiếp cận với hệ thống hơn. Đây cũng chính là lý do đề tài “Xây dựng hệ thống quản lý sân cầu lông và buôn bán sản phẩm kết hợp chatbot” được thực hiện.

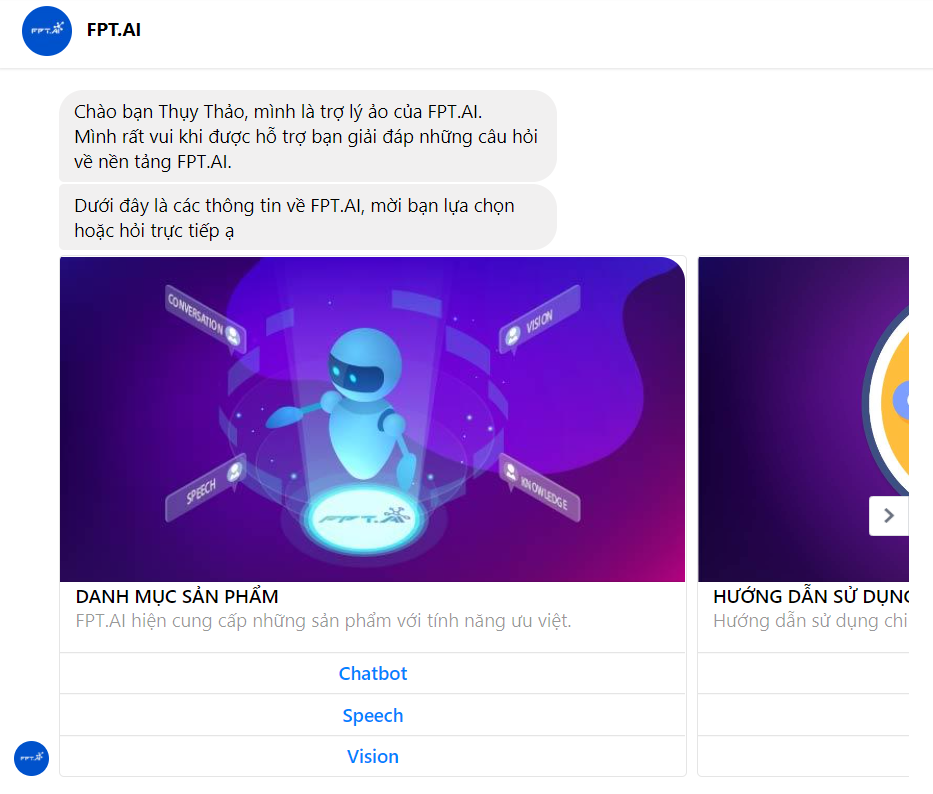
* 1. Các nghiên cứu liên quan

Trợ lý ảo là một dạng chatbot. Siri của Apple, Cortana của Windows hay Google Assistant của Google là những trợ lý ảo rất nổi tiếng được tích hợp vào thiết bị di động hay máy tính để hỗ trợ người dùng. Các trợ lý ảo có thể nhận lệnh từ giọng nói và thay người dùng thao tác như đọc thông báo, đặt lịch hẹn, báo thức, thực hiện cuộc gọi, gửi tin nhắn, …



* + - * 1. Trợ lý ảo Siri tích hợp trên Iphone

Hiện nay, trên nền tảng mạng xã hội Facebook cho phép xây dựng các chatbot. Chatbot dạng này thường đưa ra các lựa chọn cho người dùng chọn để thao tác.



* + - * 1. Chatbot FPT.AI trên nền tảng Facebook
  1. Mục tiêu đề tài

Mục tiêu của đề tài là xây dựng được một website hệ thống quản lý sân cầu lông gồm các năng: quản lý sân, quản lý sản phẩm, quản lý hoá đơn, xem các báo cáo doanh thu, mua hàng và thanh toán online cho khách hàng,.. kết hợp với một chatbot có thể trả lời các câu hỏi liên quan tới hệ thống giúp cho khách hàng tiện sử dụng trong quá trình sử dụng hệ thống. Và trang quản lý cho các chủ sân giúp họ dễ dàng quản lý dữ liệu của bản thân. Bên cạnh đó là một trang admin quản lý toàn bộ cơ sở dữ liệu của hệ thống tiện cho việc quản lý dữ liệu của người dùng cũng như huấn luyện cho chatbot.

* 1. Đối tượng và phạm vi nghiên cứu

Đề tài tập trung nghiên cứu các vấn đề:

* Tìm hiểu các chatbot, hệ thống gợi ý hiện có
* Tìm hiểu xử lý ngôn ngữ tự nhiên, thuật toán để huấn luyện mô hình, phân loại ý định người dùng trong chatbot và thiết kế quy trình đưa ra hành động thích hợp.
* Xây dựng thuật toán cho hệ thống gợi ý.
* Xây dựng website dành cho người dùng.
* Xây dựng website dành cho người quản trị.
* Tích hợp chatbot, hệ thống gợi ý vào website.
  1. Phương pháp nghiên cứu
     1. Về lý thuyết
* Tìm hiểu ngôn ngữ Python, các package, thư viện cần thiết cho phần core AI.
* Tìm hiểu các thuật toán sử dụng trong chatbot và hệ thống gợi ý.
* Tìm hiểu framework Flask cho việc xây dựng server cho service AI.
* Tìm hiểu thư viện React và các thư viện hỗ trợ khác cho việc xây dựng frond end của web.
* Tìm hiểu NodeJS và các thư viện hỗ trợ khác xây dựng server backend cho web.
* Tìm hiểu MongoDB và Mongoose cho việc lưu trữ và truy vấn dữ liệu.
* Tìm hiểu kỹ thuật lập trình một website và tích hợp với core AI.
  + 1. Về kỹ thuật
* Sử dụng ngôn ngữ Python cho việc lập trình phần core AI.
* Sử dụng MongoDB và Mongoose để quản lý cơ sở dữ liệu.
* Sử dụng React để xây dựng giao diện và NodeJS để xây dựng server.
  1. Bố cục luận văn

Bố cục quyển báo cáo luận văn gồm 3 phần, mỗi phần có những nội dung như sau:

* Phần giới thiệu: Gồm các nội dung như sau:

+ Đặt vấn đề

+ Các nghiên cứu liên quan

+ Mục tiêu đề tài

+ Đối tượng và phạm vi nghiên cứu

+ Phương pháp nghiên cứu

* Phần nội dung: Gồm các chương 1, 2, 3 với các nội dung như sau:

+ Chương 1: Cơ sở lý thuyết

+ Chương 2: Nội dung, phương pháp nghiên cứu

+ Chương 3: Đánh giá kiểm thử

* Phần kết luận: Gồm các nội dung như sau:

+ Kết luận

+ Hướng phát triển.

* 1. Kế hoạch làm việc

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Tuần 1-4 | Tuần 5-8 | Tuần 9-10 | Tuần 11-15 | Tuần 16-17 | Tuần 18-20 |
| Thiết kế các chức năng, cơ sở dữ liệu của hệ thống và viết back-end. | Viết các chúc năng của hệ thống ở phía front end. | Tìm hiểu cơ sở lý thuyết về chatbot | Viết chatbot và tích hợp với hệ thống | Kiểm thử, sửa chửa lỗi, hoàn thiện chức năng | Hoàn thiện báo cáo |

PHẦN Nội DUNG

Chương 1: Cơ sở Lý Thuyết

* 1. Phân tích hệ thống thông tin
     1. Web service

Web Service đã mở ra một hướng mới cho việc phát triển các ứng dụng Internet. Web Service có thể tạm hiểu là dịch vụ web. Web Service kết hơp sử dụng nhiều công nghệ khác nhau cho phép các ứng dụng khác nhau được viết bằng nhiều ngôn ngữ khác nhau, vận hành trên các nền tảng khác nhau những vẫn có thể trao đổi thông tin qua lại thông qua môi trường Internet. Web Service là một khái niệm rộng hơn so với khái niệm web thông thường, nó cung cấp thông tin thô, và khó hiểu với đa số người dùng, chính vì vậy nó được sử dụng bởi các ứng dụng. Các dữ liệu này thường được trả về dưới dạng XML hoặc phổ biến hơn là JSON.

* + 1. RESTful API

RESTfull API là một tiêu chuẩn dùng trong việc thiết kế API cho các ứng dụng web (thiết kế Web Service) để tiện cho việc quản lý resource. Nó chú trọng vào tài nguyên hệ thống (tệp văn bản, âm thanh, video, hoặc dữ liệu động, …), bao gồm các trạng thái tài nguyên được định dạng và được truyền tải qua HTTP.

Diễn giải các thành phần:

* API (Application Programming Interface) là một tập hợp các quy tắc và cơ chế mà theo đó, một ứng dụng hay một thành phần sẽ tương tác với một ứng dụng hay một thành phần khác. API có thể trả về dữ liệu mà bạn cần cho ứng dụng của mình ở những kiểu dữ liệu phổ biến như JSON hay XML.
* REST (Representational State Transfer) là một dạng chuyển đổi cấu trúc dữ liệu, một kiến trúc để viết API. Nó sử dụng phương thức HTTP đơn giản để tạo giao tiếp giữa các máy. Vì vậy, thay vì sử dụng một URL cho việc xử lý một số thông tin người dùng, REST gửi một yêu cầu HTTP như GET, POST, DELETE,…vv đến một URL để xử lý dữ liệu.
* RESTful API là một tiêu chuẩn dùng trong việc thiết kế các API cho các ứng dụng web để quản lý resource. RESTful API ngày nay được sử dụng phổ biến cho các ứng dụng (web, mobile,…) khác giao tiếp với nhau.
* Chức năng quan trọng nhất của REST là quy định các sử dụng HTTP method (như GET, POST, PUT, DELETE,..) và cách định dạng các URL cho ứng dụng web để quản lý resource. RESTful không quy định logic code ứng dụng và không giới hạn bởi ngôn ngữ lập trình ứng dụng, bất kỳ ngon ngữ hoặc framework nào cũng có thể sử dụng để thiết RESTfull API.

Các thức hoạt động



* + - * 1. Cách thức hoạt động của REST

REST hoạt động chủ yếu dựa vào giao thức HTTP. Các hoạt động cơ bản nêu trên sẽ sử dụng những phương thức HTTP riêng.

* GET (SELECT): Trả về một tài nguyên hoặc danh sách tài nguyên
* POST (CREATE): Tạo mới một tài nguyên.
* PUT (UPDATE): Cập nhật thông tin cho tài nguyên.
* DELETE (DELETE): Xoá một tài nguyên

Xác thực dữ liệu trả về:

RESTfull API không sử dụng session và cookie, nó sử dụng một access\_token với mỗi request. Dữ liệu trả về thường có cấu trúc sau:

{

‘id’: ‘B1706563,

‘name’: ‘Lê Minh Chiến’,

‘class’: ‘DI17V7A2’

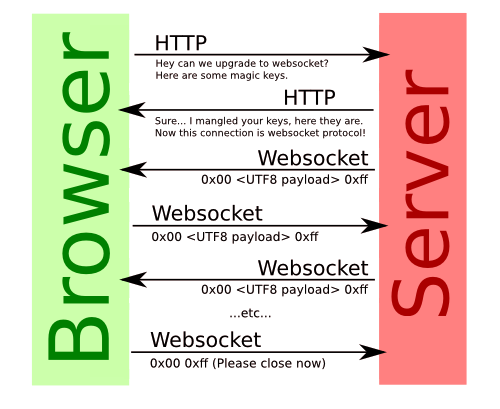
}

* + 1. Socket và Websocket

**Socket** là điểm cuối trong liên kết truyền thông tin hai chiều (two-way-communication) biểu diễn kết nối giữa Client-Server. Các lớp Socket được rang buộc với một cổng port (thể hiện là một con số cụ thể) để các tầng TCP (TCP Layer) có thể định danh ứng dụng mà dữ liệu sẽ được gửi tới.

Chức năng của socket là kết nối giữa client và server thông qua TCP/IP và UDP để truyền và nhận dữ liệu qua Internet. Giao diện lập trình ứng dụng mạng này chỉ có thể hoạt động khi đã có thông tin về thông số IP và số liệu cổng của 2 ứng dụng cần trao đổi dữ liệu cho nhau.

**Websocket** là giao thức hỗ trợ giao tiếp hai chiều giữa client và server dể tạo một kết nối trao đổi dữ liệu. Giao thức này không sử dụng HTTP mà thực hiện nó qua TCP. Mặc dù thiết kế để chuyên sử dụng cho các ứng dụng web, lập trình viên vẫn có thể đưa chúng vào bất kỳ loại ứng dụng nào.



* + - * 1. Mô tả cách thức hoạt động của Websocket
    1. Các nền tảng phía Client
       1. ReactJS

ReactJS là một thư viện của JavaScript, dùng để xây dựng giao diện cho người dùng được phát hành bởi facebook. React được phát triển để giúp lập trình viên tạo ra một single-page application một cách dễ dàng. React chia các thành phần của một trang web thành các component nhỏ dễ viết hơn nhờ việc sử dụng JSX, mở rộng cú pháp tuỳ chọn của JavaScript cho phép người dùng kết hợp HTML và JavaScript, người dùng chỉ đơn giản xử lý logic với JavaScript trong các component thay vì phải quan tâm đến toàn bộ ứng dụng. Việc chia ứng dụng web thành nhiều component cũng giúp người dùng có thể tái sử dụng một các dễ dàng nếu phát triển các chức năng tương tự. Hơn thế, ReactJS mang lại hiệu suất tốt hơn với Virtual DOM (Virtual Document Object Model – Mô hình đối tượng tài liệu), cho phép người dùng xây dựng các virtual DOMs và host chúng lại trong bộ nhớ. Mỗi khi component cần render lại virtual DOM sẽ được so sánh với DOM thật để tìm ra sự khác biệt từ đó thay thế chúng trong DOM thật.

* + - 1. Boostrap

Boostrap là một framework automatic CSS, cung cấp cho người dùng một loạt các class CSS được viết sẵn một cách có quy tắc, người dùng sẽ chỉ đơn giản quan tâm đến việc sử dụng chúng như thế nào và tuỳ chỉnh để phù hợp với trang web của mình thay vì phải viết lại từng style cho trang web. Một giải pháp tiện lợi giúp cho người dùng đỡ mất thời gian hơn trong việc thiết kế giao diện website.

* + 1. Các nền tảng phía Server
       1. MongoDB

**Định nghĩa:**

MongoDB là một cơ sở dữ liệu mã nguồn mở và là cơ sử dữ liệu NoSQL hàng đầu, được hàng triệu người sử dụng được viết bởi ngôn ngữ C++.

NoSQL là 1 dạng CSDL mã nguồn mở không sử dụng Transact-SQL để truy vấn thông tin. NoSQL được viết tắt bởi: Non-Relationship SQL, hay có nơi thường gọi là Not-Only SQL. CSDL này được phát triển trên JavaScript Framework với kiểu dữ liệu JSON. (Cú pháp của JSON là “key: value”) NoSQL ra đời như một mảnh vá cho những khiếm khuyết và thiết xoát cũng như hạn chế của mô hình dữ liệu quan hệ. RDBMS về tốc độ, tính năng, khả năng mở rộng, memory cache,…

**Các thuật ngữ trong MongoDB:**

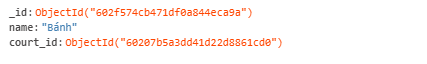
**\_id –** Là một trường bắt buộc được tự động sinh đại diện cho giá trị duy nhất trong MongoDB. Có thể được dùng để làm khoá chính cho document nhờ vào tính duy nhất của nó. Khi một document mới được thêm vào collection MongoDB sẽ tự động cấp cho nó một \_id để định danh document đó.

**Collection –** Là nhóm nhiều document trong MongoDB. Collection có thể được hiể là một bảng tương ứng trong cơ sở dữ liệu RDMS (Relationship Database Management System). Collection nằm trong một cơ sở dữ liệu duy nhất. Các collection không phải định nghĩa các cột, các hàng hay kiểu dữ liệu trước.

**Database –** Nơi chức các Collection, giống với cơ sở dữ liệu quan hệ, chúng chức các bảng. Mỗi Database có một tập tin riêng lưu trữ trên bộ nhớ vật lý. Một máy chủ MongoDB có thể chứa nhiều Database.

**Document –** Một bản ghi thuộc một Collection thì được gọi là một Document. Các Document lần lượt bao gồm các trường tên và giá trị.

**Field –** Là một cặp năm – value trong một document. Một document có thể có không hoặc nhiều trường. Các trường giống các cột ở cơ sở dữ liệu quan hệ.



* + - * 1. Một document trong MongoDB
      1. Express

Express là một framework nhỏ nhưng linh hoạt, được xây dựng trên nền tảng NodeJS. Nó cung cấp nhiều tính năng mạnh mẽ để phát triển web và ứng dụng di động. Express có nhều package hỗ trợ lập trình. Một chức năng của express chính là thiết lập các lớp trung gian để trả về các request, đjnh nghĩa router cho phép sử dụng với các hành động khác nhau dựa trên phương thức HTTP và URL, cho phép trả về các trang HTML dựa vào tham số.

* + - 1. NodeJS

NodeJS là một mã nguồn mở được xây dựng trên các nền tảng JavaScript V8 Engine. NodeJS sử dụng rộng bởi hàng ngàn lập trình viên trên toàn thế giới. Nó có thể chạy trên nhiều nền tảng điều hành như Windows, Linux, MacOS. NodeJS cung cấp các thư viện phong phú ở dạng JavaScript Module khác nhau giúp đơn giản hoá việc lập trình và giảm thời gian ở mức thấp nhất. Đặc tính nổi trội của NodeJS là tính bất đồng bộ. Điều này giúp các request riêng lẽ được xử lý đồng thời và gần như là ngay lập tức.

* + 1. Các nền tảng phía Service AI
       1. Ngôn ngữ Python

Python là một ngôn ngữ lập trình bậc cao cho các mục đích lập trình đa năng do Guido van Rossum tạo ra và lần đầu ra mắt năm 1991. Python được thiết kế với ưu điểm mạnh là dễ độc, dễ học và dễ nhớ. Python là ngôn ngữ có hình thức, cấu trúc rõ rang, thuận tiện cho người mới học lập trình. Python còn được tích hợp các thư viện và công cụ mạnh mẽ trong việc xây dựng các service AI, nơi thiên về toán học và giải thuật hơn. Nhờ hai điều này mà mà Python được ưa chuộng khi phát triển các service AI.

* + - 1. Framework Flask

Flask là một web frameworks thuộc loại micro-framework, được xây dựng bằng ngôn ngữ lập trình Python Flask cho phép xây dựng các ứng dụng web từ đơn giản tới phức tạp. Flask có thể xây dựng các api nhỏ, ứng dụng web như các trang web, blog, trang wiki hoặc một website dựa theo thời gian hay thậm chí là một trang web thương mại điện tử. Flask là một micro-framework. Điều này có nghĩa Flask là một môi trường độc lập, ít sử dụng các thư viện bên ngoài. Do vậy, flask có ưu điểm là nhẹ, có rất ít lỗi cũng như dễ dàng phát hiện và xử lý các lỗi bảo mật.

* + - 1. Thư viện Scikit-learn

Như đã được đề cập, Python được tích hợp các thư viện mạnh để phát triển service về AI và một trong số chúng là Scikit-learn. Thư viện cung cấp một tập các công cụ xử lý các bài toán về machine-learning và statistical modeling gồm: classification, regression, clustering,… Thư viện được cấp phép bản quyền chuẩn FreeBSD và chạu được trên nhiều nền tảng. Scikit-learn được sử dụng như một tài liệu để học tập.

* + - 1. Pymongo

Để kết nối đến CSDL trong python, cần một driver để có thể truy cập vào CSDL. Pymongo là một driver cho phép kết nối đến MongoDB.

* 1. Thương mại điện tử

Thương mại điện tử được hiểu là việc thực hiện hoạt động kinh doanh qua các phương tiện điện tử. Nó dựa trên việc xử lý và truyền dữ liệu điện tử dưới dạng văn bản (text), âm thanh và hình ảnh. Thương mại điện tử gồm nhiều hành vi trong đó hoạt động mà bán hàng hoá và dịch vụ qua phương tiện điện tử, giao nhận các nội dung kỹ thuật số trên mạng chuyển tiền điện tử. Thương mại điện tử được thực hiện với các thương mại hàng hoá và thương mại dịch vụ, các hoạt động truyền thống và các hoạt động mới (ví dụ như siêu thị ảo).

* 1. Chatbot
     1. Các khái niệm về chatbot

Các khái niệm chính được sử dụng trong chatbot gồm:

* Intent: Là ý định của người dùng trong câu nói.

Ví dụ: “Tôi muốn tra cứu sản phẩm Áo cầu lông Yonex” từ người dùng, thì ý định của họ ở đây là mong muốn biết các thông tin về sản phẩm Áo cầu lông Yonex như tên đầy đủ, mô tả sản phẩm, giá, …

* Entity: Là các thực thể bổ nghĩa cho một ý định

Ví dụ: “Tôi muốn tra cứu sản phẩm Áo cầu lông Yonex” từ người dùng, thực thể ở đây là “Áo cầu lông Yonex”, là tên của một sản phẩm.

* Stop words: Là những từ xuất hiện nhiều trong văn bản nhưng không có ý nghĩa quan trọng trong việc phân lớp.
* Negative words: Là các từ mang sắc thái tiêu cực
* Positive words: Là các từ mang sắc thái tích cực
  + 1. Các phương pháp xấy dựng chatbot

Chatbot được xây dựng trên các phương pháp sau:

* **Pattern matchers:** Chaatbot xây dựng dựa trên phương pháp pattern matchers sẽ tìm kiếm câu hỏi của người dùng có khớp với các câu hỏi đã được huấn luyện hay không. Phương pháp này dựa vào cấu trúc tiêu chuẩn của ngôn ngữ lập trình “Artificial Intelligence Markup Language” (AIML).

<category>

<pattern>Chatbot là gì </pattern>

<template>

Chatbot là một chương trình máy tính được tạo ra để thực hiện cuộc trò chuyện một cách tự động với người dùng

</template>

</category>

<category>

<pattern>Thế nào là chatbot </pattern>

<template>

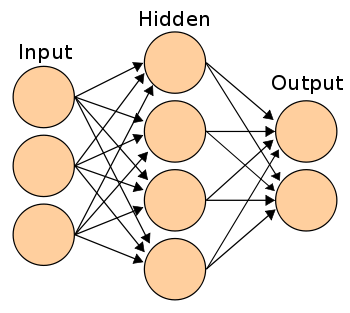
<srai>Chatbot là gì</srai>

</template>

</category>

Nếu người dùng hỏi “Chatbot là gì” hoặc “Thế nào là chatbot” thì sẽ nhận được câu trả lời “Chatbot là một chương trình máy tính được tạo ra dể thực hiện cuộc trò chuyện một cách tự động với người dùng”.

* **Sử dụng thuật toán:** Trong trường hợp có rất nhiều câu hỏi riêng biệt vơi ý định khác nhau đến từ khách hàng, mà chatbot chỉ có thể phản hồi khi tất cả câu hỏi này đã được thiết lập, huấn luyện cho chatbot chỉ có thể phản hồi khi tất cả câu hỏi này đã được thiết lập, huấn luyện cho chatbot với các kịch bản đối thoại đều được lưu trữ trong cơ sở dữ liệu. Do đó khi người dùng hỏi những câu hỏi phức tạp hơn thì cần một phương pháo khác để nhận dạng câu. Vì vậy các nhà phát triển đã đưa các thuật toán vào chatbot để phân loại câu, từ đó tăng độ chính xác cho các phản hồi đưa đến cho người dùng. Các thuật toán có thể sử dụng như SVM, KNN, Navie Bayes,…
* **Artificial Neural Network (ANN – mạng nơ ron nhân tạo):** Phương pháp này là tính toán tỉ lệ chính xác dữ liệu đầu ra từ dữ liệu đầu vào bằng cách tính toán các trọng số cho mỗi kết nối (connection) từ các lần lặp lại trong khi huấn luyện dữ liệu cho chatbot. Mỗi bước huấn luyện đầu ra được xuất ra với độ chính xác cao. Mỗi câu sẽ được phân thành từng từ và mỗi từ chính là dữ liệu đầu vào input đưa vào ANN. Các từ sẽ được kết nối với nhau khi huấn luyện dữ liệu Chatbot, các kết nối này sẽ được đo bằng trọng số cho hàng ngàn lần lặp lại ở các thời điểm khác nhau. Mỗi lần lặp lại trọng số có thể tăng và độ chính xác sẽ cao.



* + - * 1. Minh hoạ một mạng nơ ron nhân tạo
    1. Xử lý ngôn ngữ tự nhiên (NLP)

Xử lý ngôn ngữ tự nhiên (NL) là một nhánh của trí tuệ nhân tạo tập trung vào các ứng dụng trên ngôn ngữ con người. Trong trí tuệ nhân tjao thì xử lý ngôn ngữ tự nhiên là một trong những phần khó nhất vì nó liên quan đến việ phải hiểu ý nghĩa ngôn ngữ - công cụ hoàn hảo nhất của tư duy và giao tiếp.

* + - 1. Các bước xử lý trong NLP
* **Phân tích hình thái câu:** Trong bước này từng từ sẽ được phân tích và ký tự không phải chữ (như các dấu câu) sẽ được tách ra khỏi các từ. Trong tiếng Anh và nhiều ngôn ngữ khác, các từ được phân tách dùng để bằng dấu cách. Tuy nhiên trong tiếng Việt, dấu cách lại được dùng để phân tách tiếng (âm tiết) chứ không phải từ. Cùng với các ngôn ngữ như tiếng Trung, tiếng Hàn, tiếng Nhật, phân tacsh từ trong tiếng Việt là một công việc không hề đơn giản.
* **Phân tíc cú pháp:** Dãy các từ sẽ được biến đổi thành các cấu trúc thể hiện sự liên kết giữa các từ này. Sẽ có những dãy từ bị loại bỏ do vi phạm các luật văn phạm.
* **Phân tích ngữ nghĩa:** Thêm ngữ nghĩa vào các cấu trúc có thể phụ thuộc vào những câu đứng trước, đồng thời nó cũng có thể ảnh hưởng đến câu phía sau.
* **Tích hợp văn bản:** Ngữ nghĩa của một câu riêng biệt có thể phụ thuộc vào bộ những câu đứng trước, đồng thời nó cũng có thể ảnh hưởng đến các câu phía sau.
* **Phân tích thực nghĩa:** Cấu trúc thể hiện điều được phát ngôn sẽ được thông dịch lại để xác định nó thật sự có nghĩa là gì.

Tuy nhiên, ranh giới giữa 5 bước xử lý này cũng rất mỏng manh. Chúng có thể được tiến hành cùng một bước, hoặc tiến hành cùng lúc – tuỳ thuộc vào giải thuật và ngữ cảnh cụ thể.

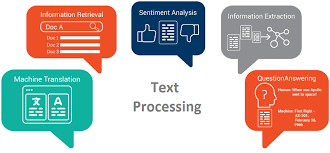
* + - 1. Các ứng dụng cơ bản của NLP và các bài toán trong xử lý tiếng Việt

NLP được ứng dụng vào thực tế như sau:

* Nhận dạng chữ viết
* Nhận dạng tiếng nói
* Tổng hợp tiếng nói
* Dịch tự động
* Tìm kiếm thông tin
* Tóm tắt văn bản
* Khai phá dữ liệu

Trong tiếng việt, NLP được ứng dụng vào các bài toán như:

* Phân tích câu
* Phân tách từ
* Tự động thêm dấu



* + - * 1. Các ứng dụng của NLP
      1. Phương pháp Bag-of-word

**Bag of Words** (BoW – mô hinh túi từ) là một thuật toán hỗ trợ xử lý ngôn ngữ tự nhiên và mục đích của BoW là phân loại câu hay văn bản. Ý tưởng của BoW là phân tích và phân nhóm dựa theo “Bag of Words” (corpus).

Xét hai ví dụ sau:

1. Tôi thích cầu lông. Em tôi cũng vậy.
2. Bạn tôi thì thích bóng đá.

Dưaj vào hai văn bản trên, xây dựng được danh sách như sau:

1. “tôi”, “thích”, “cầu”, “lông”, “em”, “tôi”, “cũng”, “vậy”
2. “bạn”, “tôi”, “thì”, “thích”, “bóng”, “đá”

Dựa vào hai danh sách trên, ta xây dựng được một từ điểm gồm 10 từ như sau:

1. [“tôi”, “thích”, “cầu”, “lông”, “em”, “cũng”, “vậy”, “bạn”, “thì”, “bóng”, “đá”]

Mỗi văn bản sẽ tạo ra một vector đặc trưng có số chiều bằng 10, mỗi phần tử đại diện cho số từ tương ứng xuất hiện trong văn bản đó. Hai văn bản trên sẽ có hai vector đặc trưng là:

1. [2,1,1,1,1,1,1,0,0,0]
2. [1,1,1,0,0,0,0,1,1,1]

Tuy nhiên, vector đại diện này không bao gồm thứ tự qua các từ trong câu gốc.

* + - 1. Phương pháp TF-IDF

**TF-IDF** (Temp Frenquecy – Inverse Document Frenquency) là một kỹ thuật sử dụng trong khai phá dữ liệu văn bản. Trọng số này được sử dụng để đánh giá tầm quan trọng của một từ trong một văn bản. Giá trị cao thể hiện độ quan trọng cao và nó phụ thuộc vào só lần xuất hiện trong văn bản nhưng bù lại bởi tần suất của từ đó trong tập dữ liệu. Một vài biến thể thường được sử dụng trong các hệ thống tìm kiếm như một công cụ để đánh giá và sắp xếp văn bản dựa vào truy vấn của người dùng. Tf-idf cũng được sử dụng để lọc những từ stopwords trong các bài toán như tóm tắt văn bản và phân loại văn bản.

**TF:** Term Frequency (Tần suất xuất hiện của từ) là số lần xuất hiện trong văn bản. Vì các văn bản có thể có độ dài ngắn khác nhau nên một số từ có thể xuất hiện nhiều lần trong một văn bản dài hơn là một văn bản ngắn, Như vậy, term frequency thường được chia cho độ dài văn bản (tổng số từ trong một văn bản).

Trong đó:

* tf (t,d): tần suất xuất hiện của từ t trong văm bản
* f(t, d): số lần xuất hiện của từ t trong văn bản d
* max(f{w, d) : w }): số lần xuất hiện của từ có số lần xuất hiện nhiều nhất trong văn bản d

**IDF:** Inverse Document Frequency (Nghịch đảo tần suất của văn bản), giúp đánh giá tầm quan trọng của một từ. Khi tính toán TF, tất cả các từ được coi như có độ quan trọng bằng nhau. Nhưng một số từ “là”, “hoặc”, “và” thường xuất hiện rất nhiều nhưng độ quan trọng lại không cao. Như vậy cần giảm độ quan trọng của những từ này xuống.

Trong đó:

* idf (t, D): giá trị idf của từ trong tập văn bản
* |D| : tổng số văn bản trong tập D
* |{d D : t d}| : thể hiện số văn bản trong tập D có chứa từ t

Cơ số logarit trong côcng thức này không thay đổi giá trị idf của từ mà chỉ thu hẹp khoảng giá trị của từ đó. Vì thay đổi cơ số sẽ dấn đến việc trị của các từ thay đổi bởi một số nhất định và tỷ lệ giữa chúng thì không thay đổi. Việc sử dụng logarit nhằm giúp giá trị tf-idf của một từ nhỏ hơn, do công thức tính của một từ trong 1 văm bản là tích của tf và idf của từ đó.

Xét với ví dụ trên, sau khi thực hiện vector hoá bằng BoW, ta có kết quả TF-IDF như sau:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Tôi | Thích | Cầu | Lông | Em | Cũng | Vậy | Bạn | Thì | Đá | Bóng |
| 0.00 | 0.00 | 0.00 | 0.09 | 0.09 | 0.09 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.12 | 0.12 | 0.12 | 0.12 |

1. Nội dung, Phương pháp Nghiên cứ
   1. Hệ thống quản lý sân cầu lông và bán hàng cá nhân
      1. Chức năng
         1. Website dành cho chủ sân cầu lông

* **Đăng ký:** Người dùng truy cập vào trang đăng nhập và chọn mục đăng ký, sau đó điền đủ thông tin yêu cầu. Dữ liệu này sẽ gửi về phía server và kiểm tra trong CSDL, nếu tên tài khoản chưa có, server sẽ trả về thông báo tạo tài khoản thành công. Ngược lại sẽ thông báo tài khoản đã tồn tại và yêu cầu nhập lại.
* **Đăng nhập:** Người dùng truy cập vào trang đăng nhập, nhập tên đăng nhập và mật khẩu. Dữ liệu này sẽ gửi về phía server kiểm tra trong CSDL, nếu tài khoản tồn tại (tên đăng nhập và mật khẩu), server sẽ trả về đăng nhập thành công và chuyển hướng người dùng đến trang dành cho chủ sân. Ngược lại sẽ thông báo tài khoản và mật khẩu sai, yêu cầu nhập lại hoặc đăng ký mới.
* **Trang thông tin về hệ thống:** Đăng các thông tin về hệ thống, ưu đã khi sử dụng dịch vụ từ hệ thống cho chủ sân.
* **Trang quản lý sân cầu lông:** Chủ sân có thể mở sân khi có khách yêu cầu để tạo mới hoá đơn, trong quá trình sử dụng nếu có yêu cầu về dịch vụ tại sân chủ sân có thể thêm các dịch vụ đó vào hoá đơn. Khi thanh toán sẽ lưu hoá đơn vào hệ thống và đưa sân về trạng thái đóng.
* **Trang Admin riêng cho chủ sân:** Chủ sân có thể quản lý các dữ liệu của mình tiện cho việc phục vụ cho sân cầu lông như (giá giờ thuê sân, các loại sản phẩm, …). Ngoài ra còn cung cấp thông tin về các hoá đơn sử dụng dịch vụ tại sân và hoá đơn mua hàng online của khách. Chủ sân có thể thêm sửa xoá dữ liệu riêng biệt của mình trên hệ thống khi các dịch vụ tại sân có sự thay đổi tiện cho việc quản lý.
* **Trang đăng bán sản phẩm:** Tại trang này, chủ sân có thể đưa các sản phẩm dịch vụ tại sân cầu lông của mình từ trạng thái offline sang online để bán chúng trên trên bán hàng cá nhân của hệ thống, cũng như có thể thêm mới các sản phẩm đó trực tiếp trên trang này. Ngoài ra hệ thống cũng cho chủ sân tuỳ chỉnh logo và banner của trang bán hàng cá nhân để tiện cho việc trang trí.
* **Trang thống kê:** Thống kê doanh thu từ việc kinh doanh dịch vụ tại sân và doanh thu từ việc bán hàng trên cửa hàng cá nhân. Chủ sân có thể chọn các mốc ngày để lọc doanh thu.
  + - 1. Trang cho khách hàng
* **Trang chủ:** Liệt kê một số sản phẩm đang có trên hệ thống, sản phẩm đang giảm giá tiện cho khách hàng lựa chọn khi mới truy cập vào hệ thống
* **Trang chi tiết sản phẩm:** Liệt kê chi tiết sản phẩm (tên, loại, giá, chủ shop,..), có đường dẫn để truy cập vào cửa hàng cá nhân của sân để xem các sản phẩm sân cung cấp.
* **Trang cửa hàng cá nhân của chủ sân:** Khách hàng có thể vào trang cửa hàng cá nhân để xem các sản phẩm mà shop đăng bán.
* **Giỏ hàng:** Khi đã chọn xong các sản phẩm mình muốn mua khách có thể vào xem lại giỏ hàng và tiến hành thanh toán qua paypal.
* **Trang thông tin cá nhân của khách hàng:** Khách hàng có thể đổi mật khẩu tài khoản, địa chỉ giao hàng, xem lại các hoá đơn mà mình đã thanh toán.
  + - 1. Trang quản lý dữ liệu của toàn hệ thống cho người quản trị
* **Thêm, sửa, xoá, tìm kiếm:** Người quản trị có thể thao tác thêm, cập nhật, xoá dữ liệu, nếu chưa có sẽ thêm vào cơ sở dữ liệu, ngược lại nếu có sẽ thông báo thêm thất bại. Khi xoá một dữ liệu, nếu dữ liệu con thì sẽ thông báo cho chủ sân biết trước việc dữ liệu con cũng sẽ bị mất để chủ sân biết trước.
* **Huấn luyện mô hình:** Người quản trị chuẩn bị dữ liệu huấn luyện theo định dạng JSON có sẵn, sau đó gửi yêu cầu cho server. Server sẽ gửi yêu cầu cho service AI. Sau khi huấn luyện thành công, service AI trả về dữ liệu cho server web rồi gửi thông báo cho người dùng.
* **Cập nhật danh sách stop words, negative words, positive words, entity:** Người quản trị có thể chuẩn bị dữ liệu cập nhật theo format JSON có sẵn, sau đó sẽ gửi yêu cầu cập nhật dữ liệu mới đến server. Server sau đó sẽ gửi yêu cầu tiếp qua service AI. Sau khi cập nhật thành công, việc Service AI trả về dữ liệu cho web rồi gửi thông báo cho người dùng.
  + 1. Thiết kế mô hình
       1. Mô hình dữ liệu mức quan niệm CDM
       2. Mô hình dữ liệu mức vật lý PDM

Bảng thực thể TÀI KHOẢN CHỦ SÂN

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Mô tả |
| 1 | ID\_TK | ObjectID | Mã tài khoản |
| 2 | TEN\_TK | String | Tài khoản đăng nhập |
| 3 | HO\_TEN | String | Họ tên người dùng |
| 4 | DIA\_CHI | String | Địa chỉ người dùng |
| 5 | SO\_DT | Number | Số điện thoại |
| 6 | GIOI\_TINH | String | Giới tính |
| 7 | VAI\_TRO | String | Vai trò |

Bảng thựcc thể SÂN

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | ID\_SAN | String | Mã sân |
| 2 | TEN\_SAN | String | Tên sân cầu lông |
| 3 | DIA\_CHI\_SAN | String | Địa chỉ sân |
| 4 | SO\_DT\_SAN | Number | Số điện thoại sân |
| 5 | MO\_TA | String | Mô tả sân |
| 6 | LOGO | Object | Logo của shop |
| 7 | BANNER | Object | Banner của shop |
| 8 | SO\_LUONG\_KV | Number | Số lượng khu vực sân |
| 9 | EMAIL\_CARD\_ID | String | Email tài khoản nhận tiền bán hàng từ shop cá nhân |
| 10 | ID\_TK | ObjectID | Mã tài khoản (khoá ngoại) |
| 11 | LINK\_SHOP | String | Đường dẫn tới shop cá nhân |

Bảng thực thể SÂN\_KHU VỰC

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | ID\_KHU\_VUC | ObjectID | Mã khu vực trong sân |
| 2 | TRANG\_THAI | Boolean | Trạng thái sân (đang mở hoặc đang tắt) |
| 3 | MO\_TA | Mô tả | Mô tả về khu vực |
| 4 | ID\_SAN | ObjectID | Mã sân (khoá ngoại liên kết với bảng thực thể SÂN) |

Bảng thực thể HOÁ ĐƠN SÂN

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | ID\_HOA\_DON | ObhectID | Mã hoá đơn |
| 2 | THOI\_GIAN\_VAO | Date | Thời gian vào thuê sân |
| 3 | THOI\_GIAN\_RA | Date | Thời gian kết thúc thuê sân |
| 4 | ID\_KHU\_VUC | ObjectID | Mã khu vực của sân (khoá ngoại liên kết với bảng thực thể SÂN\_KHU VỰC) |
| 5 | TONG\_TIEN | Number | Tổng tiền thuê sân và sử dụng các dịch vụ khác |
| 6 | TRANG\_THAI | Boolean | Trang thái hoá đơn thuê sân (đã thanh toán, chưa thanh toán) |

Bảng thực thể CHI TIẾT HOÁ ĐƠN SÂN

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | ID\_CT\_HOA\_DON | ObjectID | Mã chi tiết hoá đơn thuê sân |
| 2 | ID\_SAN\_PHAM | ObjectID | Mã sản phẩm sử dụng khi thuê sân (khoá ngoại liên kết với bảng thực thể sản phẩm) |
| 3 | ID\_HOA\_DON | ObjectID | Mã hoá đơn thuê sân (khoá ngoại liên kết với bảng thực thể hoá đơn) |
| 4 | SO\_LUONG | Number | Số lượng sản phẩm của chi tiết hoá đơn thuê sân |

Bảng thực thể LOẠI SẢN PHẨM

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | ID\_LOAI | ObjectID | Mã loại sản phẩm |
| 2 | TEN\_LOAI | String | Tên loại sản phẩm |
| 3 | ID\_SAN | ObjectID | Mã sân (khoá ngoại liên kết với bảng thực thể SÂN) |

Bảng thực thể SẢN PHẨM

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | ID\_SAN\_PHAM | ObjectID | Mã sản phẩm |
| 2 | TEN\_SAN\_PHAM | String | Tên sản phẩm |
| 3 | GIA\_SAN\_PHAM | Number | Giá sản phẩm |
| 4 | MO\_TA | String | Mô tả về sản phẩm |
| 5 | SO\_LUONG | Number | Sô lượng sản phẩm hiện có trong kho |
| 6 | TRANG\_THAI\_BAN | Boolean | Trạng thái của sản phẩm (đang bán online hoặc chỉ bán trong lúc khách thuê sân) |
| 7 | ID\_LOAI | ObjectID | Mã loại sản phẩm (khoá ngoại liên kết với bảng thực thể LOẠI\_SẢN\_PHẨM) |
| 8 | ID\_KHUYEN\_MAI | ObjectID | Mã khuyến mãi (khoá ngoại liên kết với bảng thực thể KHYẾN MÃI) |

Bảng thực thể KHUYẾN MÃI

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | ID\_KHUYEN\_MAI | ObjectID | Mã khuyến mãi |
| 2 | TEN | String | Tên chương trình khuyến mãi |
| 3 | NGAY\_BAT\_DAU | Date | Ngày bắt đầu khuyến mãi |
| 4 | NGAY\_KET\_THUC | Date | Ngày kết thúc khuyến mãi |
| 5 | TI\_LE\_KM | Number | Giá trị khuyến mãi |
| 6 | MO\_TA | String | Mô tả về khuyến mãi |
| 7 | MA\_SAN | ObjectID | Mã sân (khoá ngoại liên kết với bảng thực thể sân) |

Bảng thực thể TÀI KHOẢN KHÁCH

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | ID\_TK\_KHACH | ObjectID | Mã tài khoản khách hàng |
| 2 | MAT\_KHAU | String | Mật khẩu tài khoản |
| 3 | HO\_TEN | String | Tên đầy đủ của khách hàng |
| 4 | DIA\_CHI | String | Địa chỉ nhận hàng thường trưc của khách |
| 5 | SO\_DIEN\_THOAI | Number | Số điện thoại của sân |
| 6 | GIOI TINH | Boolean | Giới tính của khách hàng |
| 7 | NGAY\_SINH | Date | Ngày tháng năm sinh của khách hàng |

Bảng thực thể ĐƠN HÀNG

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | ID\_DON\_HANG | ObjectID | ID đơn hàng |
| 2 | TG\_THANH\_TOAN | Date | Ngày thanh toán đơn hàng |
| 3 | MO\_TA | String | Mô tả về đơn hàng |
| 4 | TONG TIEN | Number | Tổng tiền của hoá đơn |
| 5 | ID\_TK\_KHACH | ObjectID | ID khách (khoá ngoại liên kết bảng thực thể TK\_KHÁCH |
| 6 | ID\_SAN | ObjectID | Mã sân (khoá ngoại liên kết bảng thực thể SÂM) |

Bảng thực thể CHI TIẾT ĐƠN HÀNG

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | ID\_CHI\_TIET\_DH | ObjectID |  |
| 2 | ID\_SAN\_PHAM | ObjectID | ID sản phẩm (khoá ngoại liên kết với bảng thực thể SẢN PHẨM) |
| 3 | SO\_LUONG | Nubmer | Số lượng sản phẩm đã mua |
| 4 | ID\_TK\_KHACH | ObjectID | Mã tải khoản khách hàng (khoá ngoại liên kết tới bảng thực thể KHÁCH\_HÀNG |

* + 1. Cài đặt website
       1. Thiết kế API server cơ sở dữ liệu theo chuẩn RESTful

**Server cơ sở dữ liệu trong cổng 4000**

Bảng danh sách API

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Địa chỉ** | **Subpath** | **Phương thức** | **Chức năng** |
| 1 | /court | Không | GET | Lấy danh sách tất cả các sân |
|  | POST | Thêm sân mới vào cơ sở dữ liệu |
| DELETE | Xoá danh sách dữ liệu đã chọn |
| /{id} | PUT | Sửa nội dung của dữ liệu dựa vào id tương ứng |
|  |  | /{court\_id} | GET | Lấy dữ liệu thông tin của sân dựa vào id sân |
| 2 | /court\_area | Không | GET | Lấy danh sách tất cả các khu vực |
| POST | Thêm khu vực mới vào cơ sử dữ liệu |
| DELETE | Xoá danh sách khu vực đã được chọn |
| /{id} | PUT | Sửa nội dung khu vực dựa vào id tương ứng |
| /{court\_id} | GET | Lấy danh sách các khu vực thuộc một sân dựa vào id của sân |
| 3 | /court\_bill | Không | GET | Lấy danh sách tất cả các hoá đơn thuê sân |
| POST | Thêm hoá đơn thuê sân mới vào cơ sở dữ liệu |
| DELETE | Xoá danh sách hoá đơn thuê sân đã được chọn |
| /{id} | PUT | Sửa nội dung hoá đơn đã chọn dựa trên id tương ứng |
| /{court\_id} | GET | Lấy hết hoá đơn thuê sân của một sân dựa trên id sân |
| 4 | /court\_bill\_detail | Không | GET | Lấy danh sách tất cả chi tiết hoá đơn thuê sân |
|  | POST | Thêm một chi tiết hoá đơn thuê sân mới vào cơ sở dữ liệu |
|  | DELETE | Xoá danh sách chi tiết hoá đơn thuê sân đã được chọn |
| /{id} | PUT | Chỉnh sửa nội dung chi tiết hoá đơn thuê sân dựa trên id đã chọn |
|  |  | /{court\_id} | GET | Lấy hết tất cả chi tiết hoá đơn thuê sân của một sân dựa vào id sân |
| 5 | /product\_category | Không | GET | Lấy danh sách tất cả loại sản phẩm |
| POST | Thêm một loại sản phẩm mới vào cơ sở dữ liệu |
| DELETE | Xoá danh sách loại sản phẩm đã chọn |
| /{id} | PUT | Chỉnh sửa thông tin loại sản phẩm dựa vào id tương ứng |
| /{court\_id} | GET | Lấy danh sách tất cả loại sản phẩm của một sân dựa vào id sân tương ứng |
| 6 | /product | Không | GET | Lấy danh sách tất cả sản phẩm |
| POST | Thêm sản phẩm mới vào cơ sở dữ liệu |
| DELETE | Xoá danh sách sản phẩm đã chọn |
| /{id} | PUT | Chỉnh sửa nội dung sản phẩm dựa trên id tương ứng |
| /{court\_id} | GET | Lấy danh sách sản phẩm của một sân dựa trên id sân tương ứng |
| 7 | /user\_bill | Không | GET | Lấy danh sách tất cả các đơn hàng |
| POST | Thêm một đơn hàng mới vào cơ sử dữ liệu |
| DELETE | Xoá danh sách đơn hàng đã được chọn |
| /{id} | PUT | Sửa nội dung đơn hàng dựa trên id tương ứng |
| /{court\_id} | GET | Lấy danh sách đơn hàng của sân dựa trên id sân tương ứng |
|  |  | /{user\_id} | GET | Lấy danh sách đơn hàng của khách hàng dựa trên id khách hàng tương ứng |
| 8 | /user\_bill\_detail | Không | GET | Lấy tất cả chi tiết đơn hàng |
| POST | Thêm chi tiết đơn hàng vào cơ sở dữ liệu |
| DELETE | Xoá danh sách chi tiết đơn hàng đã chọn |
| /{id} | PUT | Chỉnh sửa nội dung chi tiết đơn hàng dựa trên id tượng ứng |
| /{court\_id} | GET | Lấy hết chi tiết đơn hàng của sân dựa trên id của sân tương ứng |
| 9 | /promotion | Không | GET | Lấy hết tất cả khuyến mãi trong cơ sở dữ liệu |
|  |  |  | POST | Thêm một khuyến mãi vào cơ sở dữ liệu |
|  |  |  | DELETE | Xoá danh sách khuyến mãi đã chọn |
|  |  | /{id} | PUT | Chỉnh sửa nội dung khuyến mãi dựa trên id tương ứng |
|  |  | /{court\_id} | GET | Lấy hết khuyến mãi của một sân |
| 10 | /user | Không | GET | Lấy hết danh sách tài khoản khách hàng |
| POST | Thêm một tài khoản khách hàng vào cơ sở dữ liệu |
| DELETE | Xoá danh sách tài khoản khách hàng đã chọn |
| /{id} | PUT | Chỉnh sửa thông tin tài khoảng khách hàng dựa trên id tương ứng |
| 11 | /owner | Không | GET | Lấy tất cả tài khoản chủ sân |
| POST | Thêm một tài khoản chủ sân vào cơ sở dữ liệu |
| DELETE | Xoá danh sách tài khoản chủ sân đã chọn |
| /{id} | PUT | Chỉnh sửa tài khoản chủ sân dựa trên id tương ứng |
| /check\_login | GET | Kiểm tra tài khoản và mật khẩu của tài khoản để trả về thông tin và vai trò của tài khoản |
| 12 | /ai-service | /{function} | PUT | Cập nhật danh sách stopwords, entities, negative words, positive words |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

* + - 1. Cài đặt phía Server

Khởi tạo thư mục server:

Cài đặt các package: npm install express body-parser cors mongoose –save

Cấu trúc thư mục:

* Models: chứa các file js tạo model kết nối đến cơ sở dữ liệu MongoDB
* Controllers: chứa các file js xử lý yêu cầu người dùng
* Routes: chứa các file js khai báo các route của API
* File package.json: chứa thông tin cấu hình npm
* File package-lock.json: chứa thông tin các lần thay đổi của node modules
* Thư mục node\_modules: chứa các module cho chương trình
* File server.js: chứa mã lệnh khởi tạo server
* File db.js chứa mã lệnh kết nối đến cơ sở dữ liệu
  + - 1. Cài đặt phía Client

Khởi tạo thư mục React:

Cài đặt môi trường để chạy React: npm install -g create-react-app

Khởi tạo thư mục: npx-react-app lvtn

Cài đặt module react-router-dom để thêm thực viện xử lý chuyển trang, đảm bảo tính chất single page của ReactJS: npm install react-router-dom –save

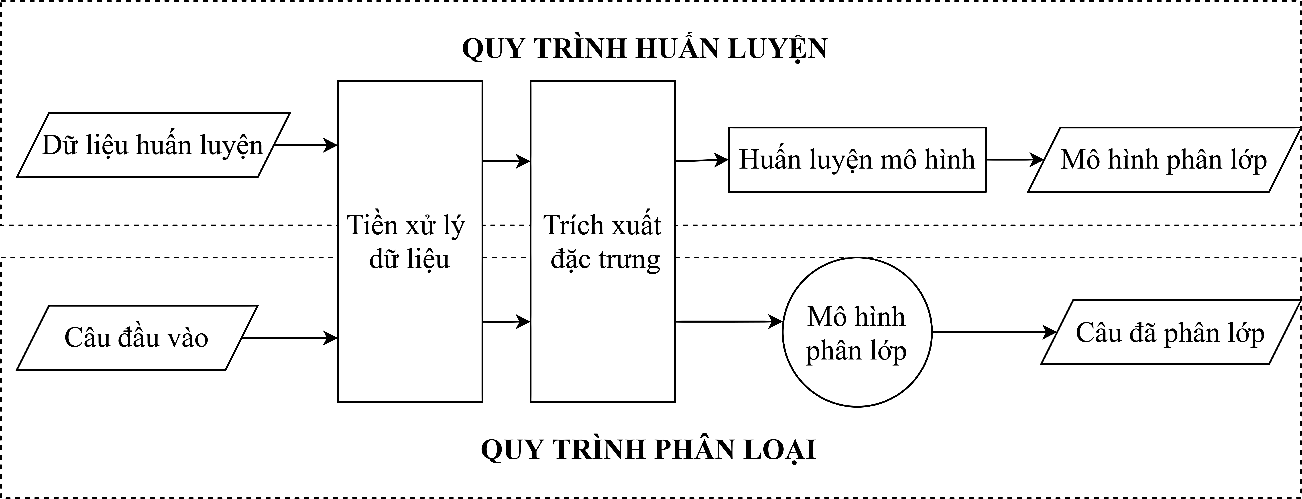
Cài đặt thư viện Axios để gửi request đến API: npm install axios –save

**Cấu trúc thư mục:**

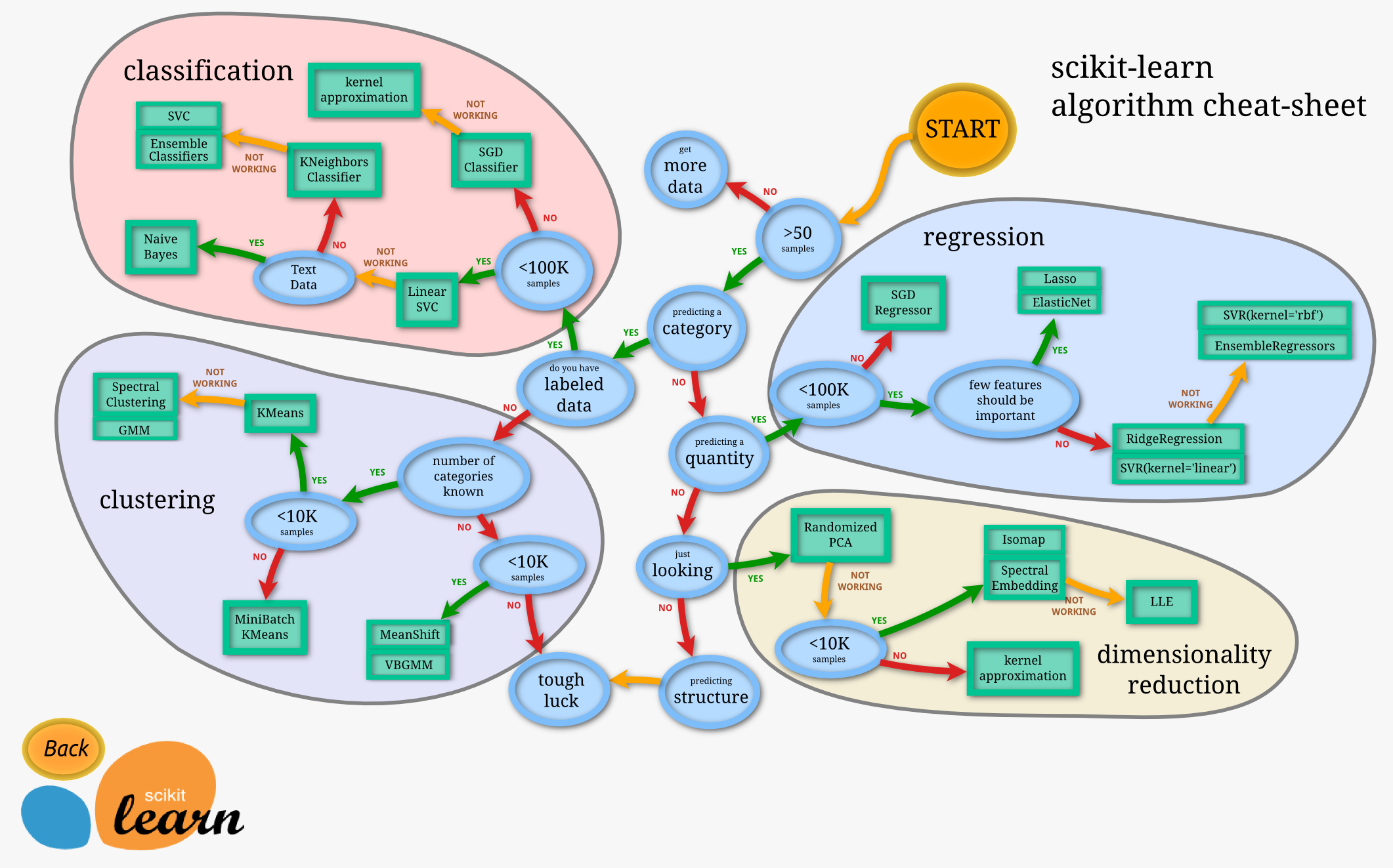
* Thư mục public: chứa các file hình ảnh, css, ..
* Thư mục src: chứa các file khởi tạo component và các file xử lý phía client
* File package.json: chứa thông tin cấu hình npm
* File package-lock.json: chứa thông tin các lần thay đổi của node modules
* Thư mục node\_modules: chứa các module cho chương trình
  1. Bài toán phân lớp ý định người dùng và huấn luyện chat bot

Để xây dựng mô hình phân lớp ý định người dùng, cần thực hiện các bước sau:

* Xây dựng tập dữ liệu huấn luyện
* Tiền xử lý văn bản
* Trích xuất đặc trưng và vector hoá văn bản
* Chọn thuật toán và huấn luyện mô hình máy học



* + - * 1. Quy trình huấn luyện và phân loại



* + - * 1. Cách lựa chọn thuật toán ML phù hợp với dữ liệu
    1. Xây dựng dữ liệu huấn luyện

Để huấn luyện cho chatbot, dữ liệu huấn luyện gồm danh sách các câu tự xây dựng theo format json. Một câu có các trường sau:

- intent\_name: tên intent

- training\_phrase: tập câu trả lời cho câu hỏi đầu vào tương ứng

- query: có truy vấn CSDL hay không

- description: mô tả intent

{

“intent\_name: “chao hoi”

“training\_phrases: [“xin chào”, “hi”, “hello”, “alo”]

“response: [

“Xin chào bạn!”,

“Xin chào, rất vui được gặp bạn!

]

“query”: “”,

“description”: “casual”

}

* + 1. Tiền xử lý văn bản

Tiền xử lý văn bản cho một câu đầu vào gồm các bước sau:

* Xử lý chữ viết thường: biến đổi toàn bộ văn bản về dạng chữ thường
* Loại bỏ dấu câu: loại bỏ các dấu câu, ký tự đặc biệt, …
* Chuẩn hoá câu: đồng nhất các từ viết tắt, teencode, … về một từ chuẩn hoá.
* Loại bỏ stop words: Loại bỏ các từ dừng trong văn bản.
* Tách từ: xác định các từ đơn, từ ghép, …
* Xét ví dụ sau:

Câu đầu vào: “Xin chào bạn, tôi muốn đky thành viên?”

Câu đã xử lý: “xin\_chào\_đăng\_ký\_thành\_viên

Sau khi tiền xử lý, câu chỉ còn lại những từ quan trọng. Đây có thể xem là những đặc trưng của văn bản và quá trình phân loại sẽ dựa trên các đặc trưng này.

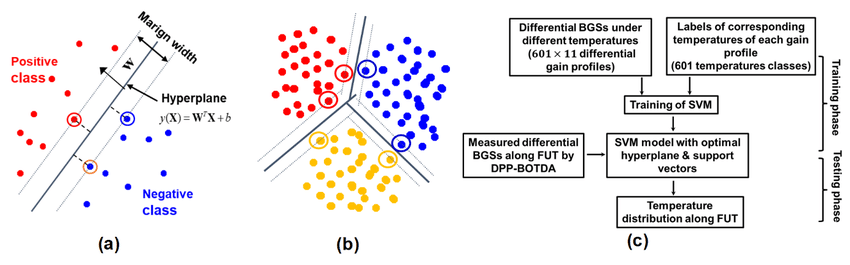
* + 1. Trích xấut đặc trưng và vector hoá văn bản

Để trích xuất đặc trưng và vector hoá cho văn bản, mô hình Bag-of-Word đã được sử dụng để xây dựng vector đặc trưng. Sau đó, thuật toán TF-IDF được sử dụng để xác định giá trị từng phần của phần tử trong vector.

* + 1. Thuật toán Multi-Class SVM cho huấn luyện mô hinh

Máy học vector hỗ trợ (Support vector machine – SVM) được Vapnik nghiên cứu những năm 1965, mãi đến những năm 1990 thì giải thuật này mới chính thức phát triển mạnh, trở thành công cụ đắc lực cho nhiều ứng dụng nhận dạng mặt người, phân loại văn bản, phân loại bệnh ung thư, … Bằng việc kết hợp với phương pháp hàm nhân, SVM cung cấp các mô hình hiệu quả chính xác cho các vấn đề phân lớp và hồi quy tuyến tính và phi tuyến tính trong thực tế.

Giải thuật SVM là một giải thuật tìm một siêu phẩm (hyperplane) tối ưu cho phép chia các điểm (phần tử) dữ liệu thành hai phần sao cho các điểm cùng một lớp nằm về một phía của siêu phẳng này (bài toán hai lớp – Binary Class). Để phân lớp nhiều hơn hai lớp (Multi Class), kỹ thuật SVM nguyên thuỷ sẽ chi không gian dữ liệu này thành 2 phần và quá trình này lặp lại nhiều lần. Giả sử bài toán cần phân loại có k lớp (k>2), chiến lược “một-dối-một” sẽ tiến hành lần phân lớp nhị phân, mỗi phân lớp sẽ tiến hành phân tách với k-1 lớp còn lại để xác định k-1 hàm phân tách dựa vào bài toán phân hai lớp bằng SVM.



* + - * 1. Phân loại 2 lớp (a) và phân loại nhiều lớp (b) bằng SVM

Dữ liệu huấn luyện gồm nhiều lớp intent, tương ứng với nhiều lớp cần huấn luyện. Trong đó, mỗi câu training là một điểm, SVM giúp tìm ra các siêu phẳng để có thể phân các điểm này thành các lớp phân biệt.

* 1. Xây dựng Chatbot
     1. Chức năng
* Trả lời các câu hỏi về giao tiếp thông thường (chào hỏi, hỏi thăm sức khoẻ, hỏi tên) và các câu hỏi tương tự sau khi được huấn luyện.
* Tra cứu thông tin về sản phẩm tại cửa hàng cá nhân bằng cách truy vấn dữ liệu và đưa ra thông tin sản phẩm theo yêu cầu người dùng.
* Khi người dùng yêu cầu
  + 1. Thiết kết và cài đặt quy trình
       1. Quy trình xử lý của chatbot

Câu đầu vào từ người dùng sẽ được xử lý biến đổi thành chữ viết thường, loại bỏ dấu câu, loại bỏ khoảng trắng dư thừa và chuẩn hoá. Sau đó câu được xử lý sẽ qua bước nhận dạng entity và tiếp tục được xử lý loại bỏ stop words và tiến hành tách từ. Câu sẽ qua bước phân loại câu, kết quả trả ra cả bước này là tên intent và độ tin cậy:

* Nếu độ tin cậy > giá trị ngưỡng: Kiểm tra intent có phải là “Yêu cầu xem thông tin của sản phẩm”:

+ Nếu đúng: tiến hành thực hiện bước truy vấn CSDL và trả ra thông tin cho người dùng.

+ Nếu không: tiến hành thực hiện bước lấy câu trả lời.

* Nếu độ tin cậy ≤ giá trị ngưỡng: tiến hành thực hiện nhận dạng sắc thái câu.

+ Nếu câu có sắc thái: trả về câu trả lời có sắc thái tương ứng

+ Nếu câu không có sắc thái: trả về câu trả lời không biết.

* + - 1. Quy trình nhận dạng entity

Quy trình nhận dạng entity được mô tả như sau:

Câu đầu vào của người dùng sau khi xử lý viết thường, loại bỏ dấu câu, loại bỏ khoảng trắng dư thừa, chuẩn hoá câu sẽ được thay thế các giá trị entity trong câu bằng entity tổng quát, đồng thời giá trị entity gốc sẽ được lưu lại. Thuật toán sử dụng cho việc thay thế này là thuật toán tìm kiếm tương đối, nhằm xử lý một số trường hợp người dùng gõ sai một vài ký tưj. Sau đó bước nhận dạng “và”, “hoặc” được thực hiện. Nếu giữa các entity key có các từ “và’, ‘hoặc” thì kết quả đó sẽ được lưu lại. Kết quả trả ra gồm danh sách các entities có trong câu (entities), kết quả “và”,, “hoặc” (sign) và các câu đầu vào đã được thay thế bằng entity tổng quản (sen\_result).

* + - 1. Quy trình nhận dạng sắc thái câu

Quy trình nhận dạng sắc thái câu:

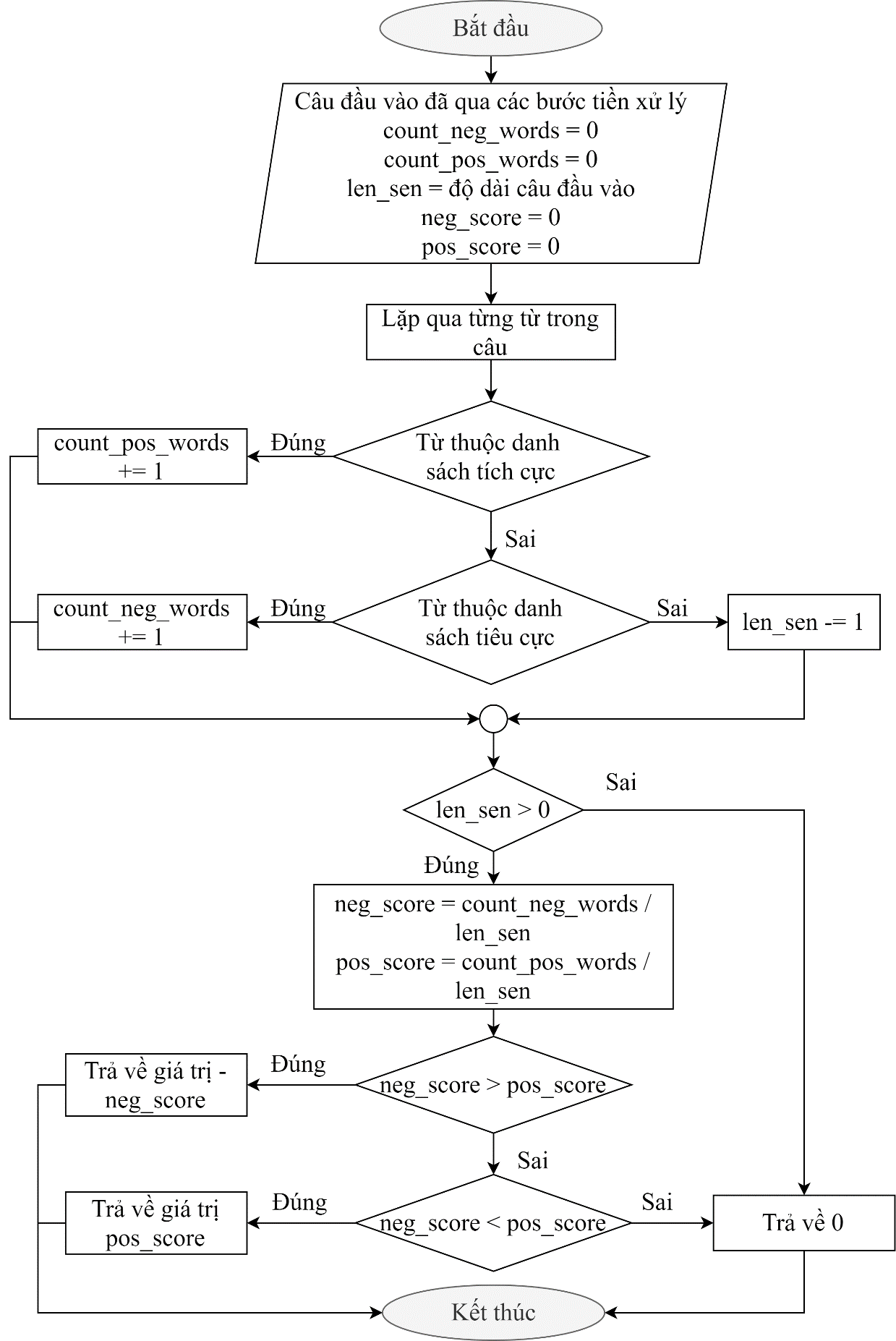
Câu đầu vào đã qua các bước tiền xử lý sẽ được chia thành các từ, ngăn cách bởi khoảng trắng. Sau đó xét từng từ, nếu từ nằm trong danh sách từ tiêu cực thì tổng từ tiêu cực (count\_neg\_words) tăng lên, còn nếu từ nằm trong danh sách từ tích cực thì tổng số từ tích cực (count\_pos\_words) tăng lên. Còn nếu từ không nằm trong hai danh sách trên thì biến độ dài câu (len\_sen) sẽ trừ đi 1. Sau khi lặp lại qua hết các từ, nếu giá trị len\_sen lớn hơn 0 thì sẽ tính điểm, ngược lại thì trả về 0 và kết thúc:

Neg\_score = count\_neg\_words / len\_sen

Pos\_score = count\_pos\_words / len\_sen

Tiếp theo sẽ xét neg\_score và pos\_score:

* Nếu neg\_score > pos\_score, trà về neg\_score
* Nếu pos\_score > pos\_score, trả về pos\_score
* Nếu neg\_score = pos\_score, trả về 0



* + - * 1. Quy trình nhận dạng sắc thái câu

1. Đánh giá Kiểm thử
   1. Môi trường cài đặt

Tất cả thực nghiệm được thực hiện trên môi trường laptop ASUS TUF A15 F506ii (AMD Ryzen 5 4600H, Base clock 3.0Ghz – Max boost clock 4.0Ghz, 16GB RAM), hệ điều hành Windows 10

* 1. Kiểm thử hệ thống sân cầu lông và đăng bán sản phẩm
     1. Phần quản lý cho chủ sân

Bảng kiểm thử chức năng quản lý của chủ sân

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Chức năng** | **Mô tả dữ liệu kiểm thử** | **Kết quả mong đợi** | **Thành công/ Thất bại** |
| 1 | Đăng nhập | Đăng nhập đúng tên đăng nhập và mật khẩu tài khoản chủ sân | Hiển thị kết quả “Đăng nhập thành công” và chuyển hướng đến trang quản lý của tài khoản | Thành công |
| 2 | Đăng nhập sai tên đăng nhập hoặc mật khẩu | Hiển thị “Tài khoản hoặc mật khẩu sai, vui long nhập lại” | Thành công |
| 3 | Quản ly sân | Bật sân khi có khách ra vào | Hoá đơn mới được tạo ra với giờ check in là giờ mở sân | Thành công |
| 4 | Thêm sản phẩm hoặc dịch vụ tại sân vào khu vực đã mở | Dịch vụ được thêm vào hoá đơn | Thành công |
| 5 | Thanh toán khi khách hàng ngừng thuê sân | Hoá đơn chuyển trạng thái thành “Đã thanh toán” và trở về trạng thái rỗng, sân trở về trạng thái đang tắt | Thành công |
| 6 | Admin | Thêm dữ liệu mới vào các bảng của chủ sân | Hiển thị kết quả “Đã thêm thành công” | Thành công |
| 7 | Sửa dữ liệu các bảng của chủ sân | Hiển thị kết quả “Đã sửa thành công” | Thành công |
| 8 | Xoá dữ liệu không có ràng buộc với các bảng khác | Hiển thị kết quả “Đã xoá thành công” | Thành công |
| 9 | Xoá dữ liệu có ràng buộc với các bảng khác | Hiển thị kết quả “Đã xoá thành công” nếu chủ sân xác nhận xoá, không thao tác gì nếu chủ sân xác nhận không | Thành công |
| 10 | Thống kê | Thống kê | Hiển thị kết quả thống kê của hoá đơn thuê sân và hoá đơn bán hàng theo thời gian được chọn | Thành công |
| 11 | Đăng bán sản phẩm | Thêm dữ liệu sản phẩm mới | Hiển thị kết quả “Đã thêm thành công” và trạng thái sản phẩm vừa thêm vào là “Đang đăng bán” | Thành công |
| 12 | Sửa thông tin sản phẩm | Hiển thị kết quả “Đã sửa thành công” | Thành công |
| 13 | Chỉnh sửa logo, banner | Hiển thị kết quả “Đã sửa thành công” | Thành công |

Bảng kiểm thử chức năng của khách hàng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Chức năng** | **Mô tả dữ liệu kiểm thử** | **Kết quả mong đợi** | **Thành công/ Thất bại** |
| 1 | Đăng nhập | Đăng nhập đúng tên đăng nhập và mật khẩu tài khoản chủ sân | Hiển thị kết quả “Đăng nhập thành công” và chuyển hướng đến trang quản lý của tài khoản | Thành công |
| Đăng nhập sai tên đăng nhập hoặc mật khẩu | Hiển thị “Tài khoản hoặc mật khẩu sai, vui long nhập lại” | Thành công |
| 2 | Đăng ký | Đăng ký với đầy đủ thông tin và tài khoản chưa tồn tại | Hiển thị “Đăng ký thành công” và chuyển hướng đến trang chủ mua hàng | Thành công |
| Đăng ký với đầy đủ thông tin và tài khoản tồn tại | Thông báo “Tài khoản đã tồn tại” | Thành công |
| 3 | Thêm sản phẩm vào giỏ hàng | Thêm sản phẩm với số lượng trong giới hạn hàng trong kho của sản phẩm | Giỏ hàng được thêm sản phẩm mới vào | Thành công |
| Thêm sản phẩm vượt quá số lượng giới hạn của kho | Thông báo “Sản phẩm hiện tại không đủ số lượng” | Thành công |
| 4 | Thanh toán | Thanh toán khi chưa đăng nhập | Thông báo “Vui lòng đăng nhập trước khi thanh toán” | Thành công |
|  |  | Thanh toán khi đã đăng nhập và đăng nhập thành công tài khoản paypal | Tiền được chuyển vào tài khoản của chủ sân và hiển thị kết quả “Thanh toán thành công”, giỏ hàng trở về trạng thái rỗng, đơn hàng mới được thêm vào. | Thành công |

* 1. Đánh giá kiểm thử huấn luyện mô hình phân lớp ý định người dùng

Tập dữ liệu gồm 10 lớp (intent) và 116 câu gồm câu mẫu tự tạo có đầy đủ dấu tiếng Việt và những câu mẫu tự tạo tự động nhân đôi và loại bỏ dấu, nhằm tăng cường thêm dữ liệu cho việc huấn luyện.

Phương pháp phân lớp sử dụng giải thuật Multi-Class SVM trên python có sử dụng thư viện scikit-learn. Thực nghiệm cũng sẽ so sánh giải thuật Multi-Class với các giải thuật khác nhau như: Navie Bayes, k-Nearest Neighbors, Decsion Trê.

Để đánh giá mô hình phân lớp, các chỉ số Accuracy, Macro-average Precision, Macro-average Recall và Macro-average F1-Score được sử dụng. Trong đó:

* **Accuracy:** được tính bằng tỉ lệ giữa số điểm dự đoán đúng và tổng số điểm trong tập dữ liệu kiểm thử
* **Macro-average Precision:** tính bằng trung bình cộng các precision theo mỗi lớp.
* **Macro**-**average Recall:** tính bằng trung bình cộng các recall theo mỗi lớp
* **Macro**-**average F1-Score:** tính bằng trung bình cộng các F1-Score mỗi lớp.

Kết quả được trình bày trong bảng sau:

Bảng so sánh kết quả phân loại giữa các giải thuật

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Chỉ số | SVM | Navie Bayes | Decision Tree | kNN |
| Accuracy | 0.97 | 0.9 | 0.9 | 0.8 |
| Macro-average Precision | 0.97 | 0.93 | 0.94 | 0.83 |
| Macro-average Recall | 0.98 | 0.94 | 0.91 | 0.81 |
| Macro-average F1-Score | 0.97 | 0.91 | 0.9 | 0.77 |

Dựa vào kết quả trên, có thể thấy giải thuật phân lớp SVM cho kết quả cao hơn các giải thuật còn lại.

* + 1. Đánh giá kiểm thử phân lớp ý định người dùng

Bảng kiểm thử phân loại lớp “Tra cứu thông tin product”

|  |  |  |
| --- | --- | --- |
| STT | Câu đầu vào | Thành công/ Thất bại |
| 1 | Muốn tra cứu sản phẩm Quần Yonex | Thành công |
| 2 | Tôi muốn tra cứu thông tin sản phẩm Quần Yonex | Thành công |
| 3 | Muốn coi sản phẩm Quần Yonex | Thành công |
| 4 | Muốn coi sản phẩm | Thành công |
| 5 | Muốn xem sản phẩm | Thành công |

Bảng kiểm thử phân lớp “Tra cứu loại sản phẩm categories”

|  |  |  |
| --- | --- | --- |
| STT | Câu đầu vào | Thành công/ Thất bại |
| 1 | Muốn tra cứu loại sản phẩm quần áo | Thành công |
| 2 | Muốn xem loại sản phẩm quần áo | Thành công |
| 3 | Thông tin loại sản phẩm quần áo | Thành công |
| 4 | Muốn coi loại sản phẩm | Thành công |
| 5 | Muốn đề xuất sản phẩm | Thành công |

Bảng kiẻm thử phần lớp “Chào hỏi”

|  |  |  |
| --- | --- | --- |
| STT | Câu đầu vào | Thành công/ Thất bại |
| 1 | Xin chào bạn | Thành công |
| 2 | Hi | Thành công |
| 3 | Hello | Thành công |
| 4 | Chào bạn nhe | Thành công |
| 5 | Alo bạn ơi | Thành công |

Bảng kiểm thử phân lớp “Tạm biệt”

|  |  |  |
| --- | --- | --- |
| STT | Câu đầu vào | Thành công/ Thất bại |
| 1 | Bye nhé | Thành công |
| 2 | Tạm biệt bạn nhen | Thành công |
| 3 | Bye bye bạn | Thành công |
| 4 | Gặp bạn sau nhé | Thành công |
| 5 | Hẹn khi khác gặp nhé | Thành công |

Bảng kiểm thử phân lớp “Hỏi thăm sức khoẻ”

|  |  |  |
| --- | --- | --- |
| STT | Câu đầu vào | Thành công/ Thất bại |
| 1 | Bạn khoẻ không? | Thành công |
| 2 | Sức khoẻ như thế nào? | Thành công |
| 3 | Bạn thế nào rồi? | Thành công |
| 4 | Dạo này bạn thế nào? | Thành công |
| 5 | Bạn có khoẻ không vậy? | Thành công |

Bảng kiểm thử phân lớp “Chatbot tên gì?”

|  |  |  |
| --- | --- | --- |
| STT | Câu đầu vào | Thành công/ Thất bại |
| 1 | Bạn tên gì vậy? | Thành công |
| 2 | Tên bạn là gì? | Thành công |
| 3 | Tôi có thể gọi bạn là gì? | Thành công |
| 4 | Họ tên bạn là gì dạ? | Thành công |
| 5 | Bạn là ai | Thành công |

Bảng kiểm thử phân lớp “Upin là gì?”

|  |  |  |
| --- | --- | --- |
| STT | Câu đầu vào | Thành công/ Thất bại |
| 1 | Upin nghĩa là gì vậy? | Thành công |
| 2 | Upin là gì? | Thành công |
| 3 | Mình không biết Upin là gì? | Thành công |
| 4 | Upin có nghĩa ra sao? | Thành công |
| 5 | Upin là gì hả? | Thành công |

Bảng kiểm thử phân lớp “Nhiệm vụ của chatbot”

|  |  |  |
| --- | --- | --- |
| STT | Câu đầu vào | Thành công/ Thất bại |
| 1 | Bạn giúp gì được? | Thành công |
| 2 | Nhiệm vụ của bạn gồm những gì? | Thành công |
| 3 | Nhiệm vụ bạn là gì? | Thành công |
| 4 | Bạn giúp gì cho tôi? | Thành công |
| 5 | Bạn có thể làm được những việc nào? | Thành công |

Bảng kiểm thử nhận dạng sắc thái câu

|  |  |  |
| --- | --- | --- |
| STT | Câu đầu vào | Thành công/ Thất bại |
| 1 | Bạn giỏi quá | Thành công |
| 2 | Thông minh quá vậy | Thành công |
| 3 | Bạn dở quá | Thành công |
| 4 | Bạn ngốc ghê | Thành công |
| 5 | Sao bạn giỏi quá chừng dạ | Thành công |

Bảng kiểm thử dữ liệu chưa được huấn luyện

|  |  |  |
| --- | --- | --- |
| STT | Câu đầu vào | Thành công/ Thất bại |
| 1 | Bạn đến từ đâu? | Thành công |
| 2 | Bạn sinh ra ở đâu? | Thành công |
| 3 | Bạn có thích nghe nhạc không? | Thành công |
| 4 | Bạn có thích đi du lịch không? | Thành công |
| 5 | Gia đình bạn có mấy thành viên? | Thành công |

* 1. Đánh giá kiểm thử Chatbot khi tích hợp với hệ thống

Tập dữ liệu kiểm thử là các sản phẩm được tự thêm vào thông qua tài khoản chủ sân thử nghiệm trên hệ thống quản lý sân cầu lông.

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Mô tả dữ liệu kiểm thử | Kết quả mong đợi | Thành công/ Thất bại |
| 1 | Khi kết nối tới service thất bại | Trả về thông báo “Kết nối tới chatbot có vấn đề” | Thành công |
| 2 | Yêu cầu cung cấp thông tin về sản phẩm có trong CSDL | Trả về thông tin sản phẩm cần tra cứu | Thành công |
| 3 | Yêu cầu cung cấp thông tin sản phẩm không có trong CSDL | Trả về kết quả không biết | Thành công |
| 4 | Yêu cầu gợi ý sản phẩm đối với người dùng | Trả về danh sách sản phẩm ngẫu nhiên trong shop | Thành công |
| 5 | Yêu cầu cung cấp sản phẩm theo loại | Trả về danh sách sản phẩm theo loại sản phẩm người dùng cung cấp | Thành công |

1. Kết quả Thực hiện