**TRƯỜNG ĐẠI HỌC HÀNG HẢI VIỆT NAM  
KHO****A CÔNG NGHỆ THÔNG TIN**

**-----\*\*\*-----**

****

**BÁO CÁO BÀI TẬP LỚN**

**HỌC PHẦN “TRÍ TUỆ NHÂN TẠO”**

***Đề tài:***

***“XÂY DỰNG CHƯƠNG TRÌNH CHUẨN ĐOÁN***

***BỆNH THƯỜNG GẶP”***

***GVHD: TS. Nguyễn Duy Trường Giang***

***Sinh viên thực hiện: Vũ Thị Mùi - MSV: 93917***

***Nguyễn Thị Ngọc Lan- MSV: 91154***

***Đinh Thị Thanh Thư - MSV: 92805***

***Hải Phòng, tháng 12 năm 2024***

**LỜI MỞ ĐẦU**

Trong suốt quá trình nghiên cứu và thực hiện đề tài bài tập lớn, mặc dù đối mặt với không ít khó khăn và thử thách trong quá trình nghiên cứu, nhưng nhờ sự hướng dẫn tận tình và chỉ bảo chu đáo của thầy, chúng em đã có thể hoàn thành được nhiệm vụ này đúng tiến độ được giao. Đây là một trải nghiệm quý báu, giúp em tích lũy nhiều kiến thức và kỹ năng quan trọng trong lĩnh vực Trí tuệ nhân tạo.

Trước tiên, cho phép em được gửi lời cảm ơn chân thành và sâu sắc nhất tới Thầy TS. Nguyễn Duy Trường Giang – người đã trực tiếp hướng dẫn và truyền đạt kiến thức cùng những phương pháp nghiên cứu hiệu quả. Những buổi học trên lớp, cùng với sự hỗ trợ không ngừng của thầy, đã giúp em vượt qua mọi trở ngại và hoàn thành xuất sắc đề tài này.

Quá trình thực hiện đề tài không chỉ giúp nhóm chúng em hiểu sâu hơn về các khái niệm và kỹ thuật trong Trí tuệ nhân tạo mà còn rèn luyện cho chúng em các kỹ năng quan trọng như tư duy phản biện, khả năng giải quyết vấn đề và kỹ năng làm việc nhóm. Những kiến thức này không chỉ là hành trang cho em trong lĩnh vực học tập mà còn là nền tảng vững chắc cho sự phát triển nghề nghiệp sau này.

Mặc dù đã nỗ lực hết mình, nhưng chắc chắn vẫn còn những thiếu sót và hạn chế trong quá trình thực hiện. Chúng em rất mong nhận được những ý kiến đóng góp quý báu từ của các thầy– người có chuyên môn và kiến thức sâu rộng để em có thể rút kinh nghiệm hoàn thành tốt hơn các đề tài sau này.

Một lần nữa, nhóm chúng em xin chân thành cảm ơn Thầy TS. Nguyễn Duy Trường Giang đã tạo điều kiện thuận lợi nhất để chúng em hoàn thành đề tài này.

**TRƯỜNG ĐẠI HỌC HÀNG HẢI**

**KHOA CÔNG NGHỆ THÔNG TIN**

**BỘ MÔN KHOA HỌC MÁY TÍNH**

**-----\*\*\*-----**

**BÀI TẬP LỚN**

**HỌC PHẦN: TRÍ TUỆ NHÂN TẠO**

1. **Tên đề tài**

Đề tài: **“*Xây dựng chương trình chuẩn đoán bệnh thường gặp”***

1. **Mục đích**

Xây dựng chương trình tự đông chuẩn đoán bệnh thường gặp nhằm ứng dụng những kiến thức đã học của học phần.

1. **Công việc cần thực hiện**

* Xây dựng chương trình chuẩn đoán bệnh thường gặp
* Làm báo cáo bài tập lớn của học phần
* Bảo vệ bài tập lớn

1. **Yêu cầu**

* Kết quả làm bài tập lớn: một chương trình chuẩn đoán bệnh thường gặp thành công, báo cáo bài tập lớn và slide thuyết trình
* Hạn nộp báo cáo bài tập lớn: …/12/2024.

*Hải Phòng, tháng 12 năm 2024*

**NGƯỜI HƯỚNG DẪN**

TS. Nguyễn Duy Trường Giang

**MỤC LỤC**

[CHƯƠNG 1. CƠ SỞ LÝ THUYẾT 1](#_Toc184723259)

[1.1. Giới thiệu 1](#_Toc184723260)

[1.2. Công trình liên quan 1](#_Toc184723261)

[1.3. Mô tả hệ thống đề xuất 2](#_Toc184723262)

[1.4. Phương pháp nghiên cứu 3](#_Toc184723263)

[CHƯƠNG 2: PHÂN TÍCH BÀI TOÁN 5](#_Toc184723264)

[2.1. Mô tả bài toán 5](#_Toc184723265)

[*2.1.1. Tổng quan* 5](#_Toc184723266)

[*2.1.1. Giải quyết bài toán* 5](#_Toc184723267)

[2.2. Mô hình và thuật toán sử dụng 6](#_Toc184723268)

[*2.2.1 Machine learning* 6](#_Toc184723269)

[*2.2.2 SVM (*Support Vector Machine) 9](#_Toc184723270)

[CHƯƠNG 3: XÂY DỰNG CHƯƠNG TRÌNH 10](#_Toc184723271)

[3.1. Môi trường phát triển 10](#_Toc184723272)

[*3.1.1. Ngôn ngữ lập trình* 10](#_Toc184723273)

[*3.1.2. Thư viện sử dụng và cài đặt* 10](#_Toc184723274)

[3.2. Quá trình thực hiện 12](#_Toc184723275)

[*3.2.1 Dữ liệu Dataset* 12](#_Toc184723276)

[*2.3.2 Xây dựng chương trình dự đoán bệnh thường gặp ở người* 17](#_Toc184723277)

[3.3. Kết quả đạt được 20](#_Toc184723278)

[KẾT LUẬN 25](#_Toc184723279)

[4.1. Kết luận 25](#_Toc184723280)

[*4.1.1 Ưu điểm* 25](#_Toc184723281)

[*4.1.2 Nhược điểm* 25](#_Toc184723282)

[TÀI LIỆU THAM KHẢO 27](#_Toc184723283)

**DANH MỤC HÌNH**

Hình 2.1. Quy trình vận hành của Machine Learning 7

Hình 2.2. Supervised Machine Learning 8

Hình 2.3. Unsupervised Machine Learning 8

Hình 2.4.Semi-supervised Learning 8

Hình 2.5 Minh họa thuật toán SVM 9

Hình 3.6: Tên các bệnh có trong dữ liệu 13

Hình 3.7: Gán nhãn các triệu chứng bệnh 14

Hình 3.8 Dữ liệu file symtoms\_df.csv 15

Hình 3.9 Dữ liệu file precautions\_df.csv 15

Hình 3.10 Dữ liệu file diets.csv 16

Hình 3.11 Dữ liệu file workout.csv 16

Hình 3.12 File dữ liệu description.csv 17

Hình 3.13. Mô hình chương trình 18

Hình 3.14: Chuẩn bị dữ liệu 19

Hình 3.15 Huấn luyện mô hình SVM với bệnh và triệu chứng 19

Hình 3.16 Giao diện chính của chương trình 20

Hình 3.17: Giao diện phần nhập triệu chứng và nút dự đoán 21

Hình 3.18 Kết quả hiển thị khi ấn nút mô tả 22

Hình 3.19 Kết quả hiển thị khi ấn nút biện pháp 22

Hình 3.20 Kết quả hiển thị khi ấn nút thuốc chữa 23

Hình 3.21 Kết quả hiển thị khi ấn nút chế độ ăn 23

Hình 3.22 Kết quả hiển thị khi ấn nút hoạt động 24

Hình 3.23 Cảnh báo khi chưa nhập triệu chứng đã ấn dự đoán 24

**DANH MỤC CÁC TỪ VIẾT TẮT**

|  |  |
| --- | --- |
| **[Từ](#_Toc163552185)** | **[Ý nghĩa](#_Toc163552185)** |
| [MIML](#_Toc163552185) | [Multi-Instance Multi-Label](#_Toc163552185) |
| [SVM](#_Toc163552185) | [Support Vector Machine](#_Toc163552185) |
| [AI](#_Toc163552185) | [Artificial Intelligence](#_Toc163552185) |
| [ML](#_Toc163552185) | [Machine Learning](#_Toc163552185) |

**CHƯƠNG 1. CƠ SỞ LÝ THUYẾT**

**1.1. Giới thiệu**

Trong vài năm qua, các hệ thống chuẩn đoán bệnh đã được phát triển để hỗ trợ bác sĩ với nhiều nghiên cứu [1, 2] dựa trên quy tắc từ các hướng dẫn lâm sàng và kinh nghiệm thực tế, nhằm phát triển các hệ thống chuẩn đoán. Tuy nhiên, cách này rất tốn kém thời gian,công sức, và không đủ linh hoạt cho những trường hợp bệnh phức tạp hoặc hiếm gặp.

Tuy nhiên, các hệ thống này gặp phải vấn đề khi phải đối phó với các bệnh lý chưa được bao gồm trong bộ dữ liệu huấn luyện, dẫn đến việc thiếu khả năng chuẩn đoán chính xác cho những bệnh nhân có triệu chứng khác biệt hoặc hiếm gặp. Ngoài ra, các phương pháp học máy có giám sát không thể phát hiện ra những thay đổi nhanh chóng trong thực hành y học lâm sàng, nơi mà các bệnh lý mới và các biến thể của bệnh có thể xuất hiện.

Việc phân biệt triệu chứng chồng lấn và xử lý các yếu tố nguy cơ bất thường vẫn là thách thức lớn, làm giảm độ chính xác của hệ thống trong các tình trạng bệnh lý phức tạp, đòi hỏi tiếp tục cải tiến để nâng cao hiệu quả và độ tin cậy. Các nghiên cứu [3, 4] đã chỉ ra rằng việc xác định đúng bệnh lý ở những bệnh nhân này rất quan trọng, vì nếu không chính xác, bệnh nhân có thể phải chịu những nguy cơ sức khỏe nghiêm trọng.

**1.2. Công trình liên quan**

Việc phát triển và sử dụng các hệ thống khuyến nghị y tế và thuốc cá nhân hóa đã trở nên ngày càng phổ biến, nhằm cải thiện kết quả sức khỏe cho bệnh nhân và hỗ trợ bác sĩ trong việc đưa ra các phương pháp điều trị phù hợp. Các thuật toán trí tuệ nhân tạo (AI) và máy học (ML) được ứng dụng để tìm kiếm và đề xuất các phương pháp điều trị cho các bệnh lý. Tuy nhiên, việc áp dụng các thuật toán này gặp phải không ít thách thức.

Các nghiên cứu trước đây đã chỉ ra một số thành công trong việc sử dụng AI và ML để xây dựng các hệ thống khuyến nghị y tế. Cụ thể, Gupta và cộng sự đã thử nghiệm ba thuật toán khai thác dữ liệu [5], bao gồm Decision Tree Classifier, Random Forest Classifier, và Naive Bayes Classifier. Trong đó, Naive Bayes đạt độ chính xác cao nhất với khoảng 98%, vượt trội hơn so với hai thuật toán còn lại. Bên cạnh đó, Bao et al. đã phát triển một hệ thống khuyến nghị thuốc bằng cách sử dụng SVM, cây quyết định ID3 và mạng nơ-ron BP, và mô hình SVM đã được chọn nhờ độ chính xác cao (95%).

Hơn nữa, Zhang et al. kết hợp ANN và CBR (Lý luận Dựa trên Ca bệnh) để giúp bác sĩ đa khoa trong việc kê đơn lâm sàng [6], loại bỏ được các thách thức trong việc phân tích mối liên hệ giữa thuốc và triệu chứng. Cuối cùng, Bhimavarapu et al. [7] đề xuất một hệ thống khuyến nghị thuốc dựa trên mạng nơ-ron nhân tạo xếp chồng, sử dụng tiền sử sức khỏe và lối sống của bệnh nhân, giúp giảm thiểu tác dụng phụ và đạt độ chính xác 97,5%. Mặc dù các hệ thống này đã đạt được những kết quả đáng khích lệ, nhưng vẫn còn những thách thức trong việc cải thiện độ chính xác và tin cậy của các dự đoán, cũng như đảm bảo tính đầy đủ và chính xác của dữ liệu, ảnh hưởng trực tiếp đến hiệu quả thực tế.

**1.3. Mô tả hệ thống đề xuất**

Trong đề tài này, hệ thống chúng em sử dụng mô hình máy học SVM (Support Vector Machine) để dự đoán bệnh lý từ các triệu chứng mà người dùng cung cấp, với dữ liệu bệnh lý bao gồm triệu chứng, biện pháp phòng ngừa, thuốc chữa trị, chế độ ăn, và các bài tập thể dục tương ứng với từng bệnh lý.

Mô hình được huấn luyện trên dữ liệu liên kết giữa bệnh lý và triệu chứng, cho phép phân tích và đưa ra dự đoán dựa trên danh sách triệu chứng mà người dùng nhập vào. Khi người dùng nhập các triệu chứng, hệ thống sẽ phân tích và dự đoán bệnh lý có khả năng xảy ra, sau đó cung cấp thông tin chi tiết và các biện pháp phòng ngừa, thuốc chữa trị, chế độ ăn, và các bài tập thể dục.

Hệ thống này áp dụng giao diện người dùng đồ họa (GUI) qua thư viện Tkinter, cho phép người dùng dễ dàng nhập triệu chứng và nhận kết quả chuẩn đoán bệnh. Hệ thống có khả năng mở rộng và cải thiện bằng cách cập nhật dữ liệu và cải tiến thuật toán học máy để đáp ứng tốt hơn nhu cầu của các bệnh nhân có tình trạng bệnh lý phức tạp.

**1.4. Phương pháp nghiên cứu**

Để phát triển hệ thống chuẩn đoán bệnh thường gặp dựa trên các triệu chứng, nghiên cứu này áp dụng phương pháp tiếp cận kết hợp giữa lý thuyết học máy, xử lý dữ liệu y tế và phát triển phần mềm. Các phương pháp nghiên cứu chính bao gồm:

1. *Thu thập và xử lý dữ liệu y tế:*

Dữ liệu triệu chứng và bệnh lý: Thu thập từ nguồn y tế uy tín bao gồm danh sách các bệnh lý phổ biến và các triệu chứng lâm sàng tương ứng, lưu dưới dạng bảng, liên kết bệnh lý với triệu chứng.

Xử lý dữ liệu: Dữ liệu được làm sạch, chuẩn hóa, phân loại và gán nhãn để huấn luyện mô hình học máy.

1. *Ứng dụng thuật toán học máy (SVM - Support Vector Machine):*

Mô hình SVM: SVM được sử dụng để phân loại bệnh lý, giúp xác định biên phân cách tối ưu giữa các lớp dựa trên triệu chứng. Mô hình SVM được huấn luyện với dữ liệu triệu chứng và bệnh lý đã được chuẩn bị.

Chuẩn hóa và tối ưu hóa mô hình: Dữ liệu được chuẩn hóa (ví dụ: chuyển đặc trưng triệu chứng thành giá trị nhị phân), tối ưu mô hình bằng cách tinh chỉnh các tham số như hàm kernel và độ phức tạp.

1. *Xử lý giao diện người dùng (GUI) với Tkinter:*

Thiết kế giao diện người dùng: Giao diện người dùng của hệ thống được xây dựng bằng thư viện Tkinter trong Python. Hệ thống yêu cầu người dùng nhập các triệu chứng bệnh lý, sau đó hệ thống sẽ dự đoán bệnh dựa trên các triệu chứng này.

Chức năng giao diện: Giao diện cho phép người dùng nhập triệu chứng, nhận dự đoán bệnh, xem thông tin chi tiết về bệnh lý (bao gồm mô tả, biện pháp phòng ngừa, thuốc chữa trị, chế độ ăn uống và các bài tập thể dục) và có thể dễ dàng tương tác với hệ thống để nhận được các gợi ý chi tiết hơn.

1. *Đánh giá mô hình:*

Đánh giá độ chính xác: Hiệu quả của SVM được đo lường qua các chỉ số như độ chính xác (accuracy), độ nhạy (recall) và độ đặc hiệu (precision), bằng cách so sánh kết quả dự đoán với nhãn bệnh lý thực tế trên dữ liệu kiểm thử.

Kiểm thử và tinh chỉnh: Mô hình được thử nghiệm trên nhiều bộ dữ liệu và điều chỉnh tham số để tối ưu hiệu suất và đảm bảo độ ổn định.

1. *Phát triển tính năng lưu trữ và tự động hóa:*

Lưu trữ kết quả: Hệ thống được phát triển để tự động lưu lại các kết quả chẩn đoán, từ đó giúp người dùng theo dõi tình trạng sức khỏe của mình hoặc lưu trữ kết quả để tham khảo sau.

Tự động hóa quá trình nhận diện: Các ảnh hoặc video có thể được tự động lưu khi hệ thống nhận diện các triệu chứng hoặc nụ cười từ webcam, giúp người dùng theo dõi và ghi lại các kết quả theo thời gian.

1. *Phát triển và triển khai hệ thống:*

Triển khai ứng dụng: Hệ thống được triển khai dưới dạng một ứng dụng phần mềm cài đặt trên máy tính, cho phép người dùng dễ dàng sử dụng mà không cần phải có kiến thức chuyên sâu về lập trình hay y học.

Hỗ trợ đa nền tảng: Đảm bảo rằng hệ thống có thể chạy trên các hệ điều hành phổ biến (Windows, macOS) và hỗ trợ đa dạng các thiết bị đầu vào như webcam, video và các hệ thống giám sát y tế.

**CHƯƠNG 2: PHÂN TÍCH BÀI TOÁN**

**2.1. Mô tả bài toán**

***2.1.1. Tổng quan***

Ngày nay, mối quan tâm đến sức khỏe và chẩn đoán y tế ngày càng tăng, nhưng vì nhiều lí do mà những sai lầm trong việc kê đơn thuốc do bác sĩ vẫn gây ra nhiều hậu quả nghiêm trọng. Để giải quyết vấn đề này, nghiên cứu phát triển hệ thống khuyến nghị thuốc ứng dụng khai thác dữ liệu, gồm các mô-đun: cơ sở dữ liệu, mô hình khuyến nghị, đánh giá và trực quan hóa. Các thuật toán như SVM, mạng nơ-ron LBP và cây quyết định ID3 được thử nghiệm, với SVM được chọn nhờ cân bằng tốt giữa độ chính xác và hiệu quả, đảm bảo khuyến nghị

Thông tin liên quan đến sức khỏe là một trong những chủ đề được quan tâm rộng rãi nhất trên Web. Một cuộc khảo sát vào năm 2013 của Pew Internet và American Life Project cho thấy 59% người trưởng thành đã tìm kiếm trực tuyến các chủ đề sức khỏe và với 35%tập trung vào chuẩn đoán y tế. Tuy nhiên, sai sót thuốc gây hàng trăm nghìn ca tử vong mỗi năm tại Mỹ và Trung Quốc, chủ yếu do bác sĩ kê đơn dựa trên kinh nghiệm hạn chế. Những thách thức như thiếu chuyên gia và dữ liệu y tế đòi hỏi giải pháp công nghệ để khắc phục những hạn chế đó một cách nhanh chóng hiệu quả.

Vì thế, chúng em giới thiệu đề xuất một chương trình dự đoán bệnh và khuyến nghị thuốc cũng như chế độ ăn và các hoạt động thể chất để nâng cao sức khỏe,… hiệu quả và chính xác để dự đoán thích hợp cho bệnh nhân. Hệ thống tích hợp khai thác cơ sở dữ liệu, phân tích thông tin và hỗ trợ dự đoán chính xác, hữu ích cho bác sĩ và người dùng.

***2.1.1. Giải quyết bài toán***

Hệ thống xây dựng cơ sở dữ liệu phong phú và cập nhật, thu thập thông tin từ nhiều nguồn y tế đáng tin cậy bao gồm triệu chứng bệnh, lịch sử điều trị, phản ứng thuốc, và các yếu tố nguy cơ. Đồng thời, cơ sở dữ liệu hiện đại được sử dụng để quản lý và lưu trữ thông tin, đảm bảo khả năng truy xuất nhanh chóng và chính xác khi cần thiết.

Áp dụng các thuật toán học máy tiên tiến như SVM (Support Vector Machine) để xây dựng mô hình khuyến nghị, đảm bảo độ chính xác cao trong chẩn đoán và kê đơn thuốc.

Hiệu quả của mô hình được đánh giá thông qua dữ liệu thực tế, giúp tối ưu hóa các thuật toán và cải thiện hệ thống. Đồng thời, phản hồi từ bác sĩ và bệnh nhân được thu thập, phân tích thường xuyên nhằm điều chỉnh và hoàn thiện hệ thống.

Hệ thống cung cấp giao diện trực quan và dễ sử dụng, hỗ trợ bác sĩ và bệnh nhân dễ dàng tra cứu thông tin và nhận khuyến nghị chính xác. Đồng thời, hệ thống được thiết kế để tích hợp và tương thích với các hệ thống quản lý y tế hiện hành, góp phần tối ưu hóa quy trình chăm sóc sức khỏe.

**2.2. Mô hình và thuật toán sử dụng**

***2.2.1 Machine learning***

Machine Learning hay còn gọi được là học máy, một nhánh của trí tuệ nhân tạo, sử dụng thuật toán huấn luyện trên bộ dữ liệu để tạo mô hình giúp máy thực hiện các nhiệm vụ như phân loại hình ảnh, phân tích dữ liệu và dự đoán giá.

Đối với mọi bài toán, Machine Learning Workflow giúp bạn hiểu rõ về quy trình làm việc đồng nhất của Machine Learning.

*Bước 1: Data Collection hay Gathering Data*

Thu thập dữ liệu được xem là bước quan trọng nhất trong cả quy trình. Thông thường, bước này sẽ chiếm khoảng 70% đến 80% thời gian cho toàn bộ quá trình vận hành. Vì tệp dữ liệu này sẽ ảnh hưởng rất lớn đến mô hình máy học của doanh nghiệp.

*Bước 2: Data Processing*

Data processing hay còn được gọi là xử lý dữ liệu sơ khởi. Công việc này tương đương với bước thu thập dữ liệu. Trong quá trình vận hành, bước xử lý dữ liệu sẽ thực hiện các công việc loại bỏ thuộc tính dư thừa, mã hóa và thu gọn dữ liệu,…

*Bước 3: Training model*

Training model được hiểu là huấn luyện mô hình. Ở giai đoạn này, bạn sẽ để cho mô hình học trên các tệp dữ liệu mà bạn đã thu thập và chắt lọc được.

*Bước 4: Evaluating model*

Evaluating model có nghĩa là đánh giá mô hình. Công việc này sẽ được thực hiện sau khi mô hình đã học xong tệp dữ liệu và bắt đầu cho kết quả

*Bước 5: Improve*

Cái thiện mô hình sẽ là giai đoạn cuối cùng cần thực hiện. Vì sau khi chọn được các mô hình phù hợp, số còn lại sẽ thuộc nhóm chưa phù hợp và cần được huấn luyện lại sao cho đạt được kỳ vọng mà doanh nghiệp đã đề ra.

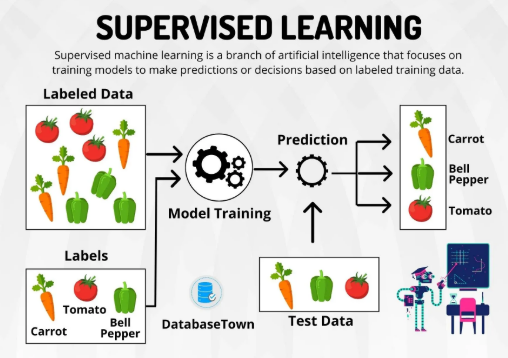


Hình 2.1 .Quy trình vận hành của Machine Learning

Các loại học máy khác nhau hỗ trợ nhiều loại sản phẩm và dịch vụ kỹ thuật số và với mục tiêu chung là tạo ra hệ thống hoạt động tự động. Tuy nhiên, phương pháp sử dụng của từng loại có sự khác biệt:

1. *Học máy có giám sát (Supervised machine learning):*

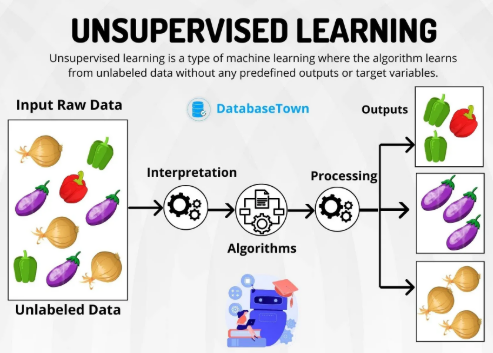
Học máy có giám sát huấn luyện thuật toán trên dữ liệu có nhãn, trong đó mỗi phần dữ liệu đều có gắn thẻ mô tả.



Hình 2.2. Supervised Machine Learning

1. *Học máy không giám sát (Unsupervised machine learning)*

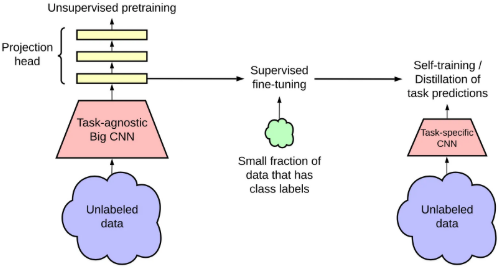
Học máy không giám sát huấn luyện thuật toán trên dữ liệu không có nhãn, giúp thuật toán tự phát hiện mẫu dữ liệu.



Hình 2.3. Unsupervised Machine Learning

1. *Học máy bán giám sát (Semi-supervised machine learning)*

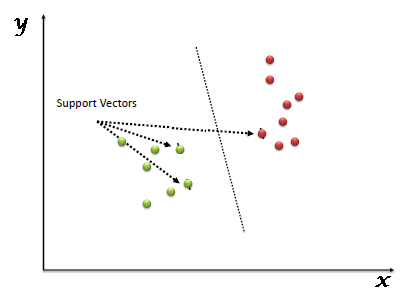
Học máy bán giám sát kết hợp dữ liệu có nhãn và không có nhãn để huấn luyện thuật toán. Thuật toán đầu tiên huấn luyện trên dữ liệu có nhãn, sau đó tiếp tục huấn luyện với dữ liệu không có nhãn.



Hình 2.4. Semi-supervised Learning

***2.2.2 SVM (*Support Vector Machine)**

Thuật toán SVM là một phương pháp thuật toán phân loại, trong đó bạn vẽ biểu đồ dữ liệu thô dưới dạng các điểm trong không gian N chiều (với n là số đối tượng bạn có). Sau đó, giá trị của mỗi đối tượng địa lý được gắn với một tọa độ cụ thể, giúp dễ dàng phân loại dữ liệu.



Hình 2.5. Minh họa thuật toán SVM

Hiệu quả trong phân loại: SVM đặc biệt mạnh trong việc xử lý các bài toán phân loại, ngay cả khi dữ liệu không tuyến tính. Với việc sử dụng kernel trick, SVM có thể ánh xạ dữ liệu lên không gian nhiều chiều hơn, giúp tìm được siêu phẳng phân cách tốt nhất.

Khả năng tổng quát hóa tốt: So với các thuật toán khác, SVM thường đạt độ chính xác cao trên tập dữ liệu kiểm tra nhờ khả năng tránh hiện tượng overfitting, đặc biệt khi làm việc với tập dữ liệu nhỏ hoặc có nhiều đặc trưng.

Xử lý dữ liệu phức tạp: SVM hoạt động hiệu quả trên dữ liệu có nhiều chiều và các loại biến phức tạp. Điều này rất phù hợp với các bài toán y tế, nơi dữ liệu có thể bao gồm nhiều yếu tố từ lịch sử bệnh, triệu chứng đến phản ứng thuốc.

Hiệu suất vượt trội trong thử nghiệm: Qua quá trình thử nghiệm, em đã so sánh SVM với các thuật toán khác như mạng nơ-ron và cây quyết định. SVM cho thấy sự cân bằng tốt hơn giữa độ chính xác và thời gian xử lý, đặc biệt là trong bối cảnh cần đưa ra các khuyến nghị nhanh và chính xác.

**CHƯƠNG 3: XÂY DỰNG CHƯƠNG TRÌNH**

**3.1. Môi trường phát triển**

***3.1.1. Ngôn ngữ lập trình***

Python là ngôn ngữ lập trình cao cấp, ra đời năm 1991 bởi Guido van Rossum, nổi bật với cú pháp đơn giản, dễ đọc và tính linh hoạt cao. Python được ứng dụng rộng rãi trong nhiều lĩnh vực:

* Phát triển web: Hỗ trợ mạnh mẽ với các framework như Django và Flask, giúp xây dựng ứng dụng web nhanh chóng, hiệu quả.
* Khoa học dữ liệu: Là ngôn ngữ phổ biến nhất trong phân tích dữ liệu, máy học và trí tuệ nhân tạo.
* Đồ họa và trò chơi: Sử dụng các thư viện như Pygame, PyOpenGL để phát triển trò chơi và ứng dụng đồ họa.
* Ứng dụng hệ thống: Cung cấp công cụ mạnh mẽ cho quản lý cơ sở dữ liệu và viết ứng dụng hệ thống.

Đặc biệt, Python 3.10 cải thiện hiệu năng nhờ tối ưu hóa cấu trúc dữ liệu Dictionary, giảm thời gian thao tác và truy cập. Vì vậy, sử dụng Python 3.10 có thể tăng đáng kể hiệu suất chương trình.



*Hình 3.1. Hình ảnh minh họa ngôn ngữ Python*

***3.1.2. Thư viện sử dụng và cài đặt***

1. *Pandas*

Pandas là một thư viện mạnh mẽ và phổ biến trong Python giúp xử lý và phân tích dữ liệu một cách dễ dàng và hiệu quả. Pandas cung cấp các cấu trúc dữ liệu chính như **Series** (một mảng một chiều) và **DataFrame** (một bảng dữ liệu hai chiều), giúp tổ chức và thao tác với dữ liệu dưới dạng hàng và cột tương tự như bảng trong Excel hoặc cơ sở dữ liệu.

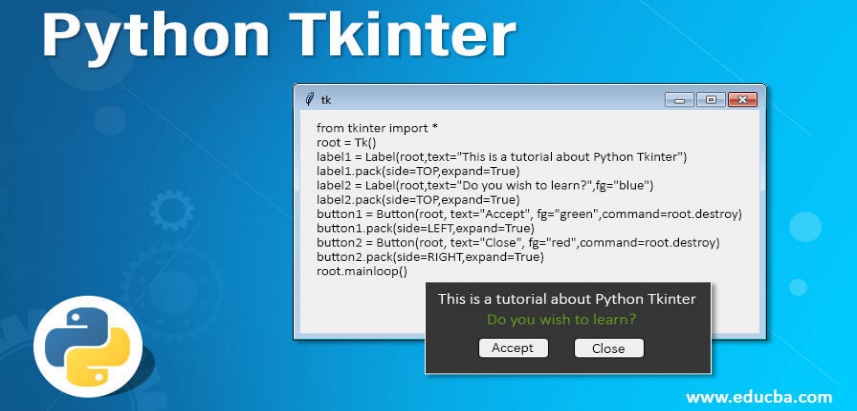
Thư viện này hỗ trợ đọc, ghi dữ liệu từ nhiều định dạng khác nhau (CSV, Excel, SQL, JSON), lọc, nhóm, và trực quan hóa dữ liệu.



*Hình 3.2. Hình ảnh minh họa thư viện Pandas*

1. *Python Tkinter*

Tkinter là một thư viện đồ họa tích hợp sẵn trong Python, được sử dụng để xây dựng các ứng dụng GUI trên nền tảng desktop từ đơn giản đến phức tạpĐể cài đặt Tkinter và các thư viện liên quan, có thể sử dụng lệnh: pip install opencv-python pillow tk.



*Hình 3.3. Hình ảnh minh họa thư viện Tkinter trong Python*

1. *Các thư viện khác*

**NumPy** là một thư viện cung cấp các hỗ trợ cho các mảng và ma trận lớn đa chiều, cùng với một tập hợp lớn các hàm toán học để thao tác với các mảng này. Nó thường được sử dụng trong các phép tính khoa học và kỹ thuật.

**Scikit-learn** là thư viện chuyên về học máy, xây dựng trên NumPy, SciPy và matplotlib. Nó cung cấp các công cụ hiệu quả cho phân loại, hồi quy, phân cụm, và giảm kích thước dữ liệu.

**3.2. Quá trình thực hiện**

Bộ dữ liệu liên quan đến bệnh án và thông tin chăm sóc sức khỏe là những tài liệu nhạy cảm, thuộc phạm vi bảo vệ của các quy định về quyền riêng tư. Việc tiết lộ thông tin y tế của bệnh nhân mà không có sự đồng ý rõ ràng là hành vi vi phạm pháp luật.

Trong nghiên cứu của chúng em, để đảm bảo tuân thủ các quy định về bảo mật và quyền riêng tư, chúng em đã sử dụng một bộ dữ liệu triệu chứng và bệnh bằng tiếng Anh, sau đó tiến hành dịch thuật sang tiếng Việt.

***3.2.1 Dữ liệu Dataset***

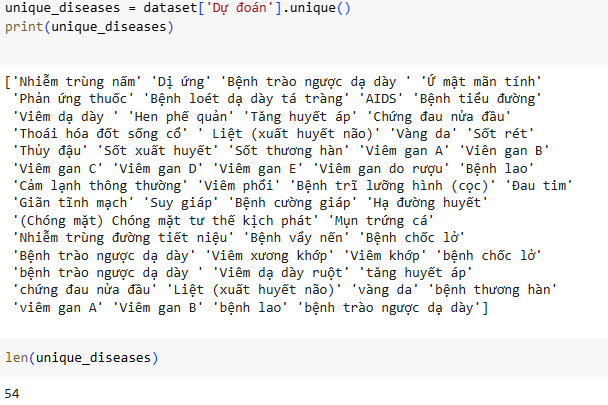
1. *Thu thập dữ liệu*

Tập dữ liệu mà nhóm em sử dụng bao gồm các tệp CSV chứa thông tin phong phú và đa dạng, phục vụ cho mục đích nghiên cứu và phân tích. Cụ thể:

* **symtoms\_df\_trans.csv**: Tệp chứa danh sách các triệu chứng đã được chuẩn hóa và dịch sang tiếng Việt.
* **precautions\_df.csv**: Tệp lưu trữ các biện pháp phòng ngừa đối với từng loại bệnh cụ thể.
* **description.csv**: Cung cấp mô tả chi tiết về các bệnh, giúp hiểu rõ đặc điểm và biểu hiện của từng loại bệnh.
* **medications.csv**: Tệp chứa thông tin về các loại thuốc, liều lượng và hướng dẫn sử dụng phù hợp cho từng triệu chứng hoặc bệnh.
* **diets.csv**: Danh sách các chế độ ăn uống được khuyến nghị để hỗ trợ quá trình điều trị và phục hồi.
* **workout\_df.csv**: Tệp bao gồm các bài tập và hoạt động thể chất phù hợp, góp phần cải thiện sức khỏe tổng thể.

Những tệp dữ liệu này không chỉ giúp nhóm em xây dựng mô hình chính xác và đa chiều mà còn đảm bảo tính hệ thống và khả năng ứng dụng thực tế trong nghiên cứu.

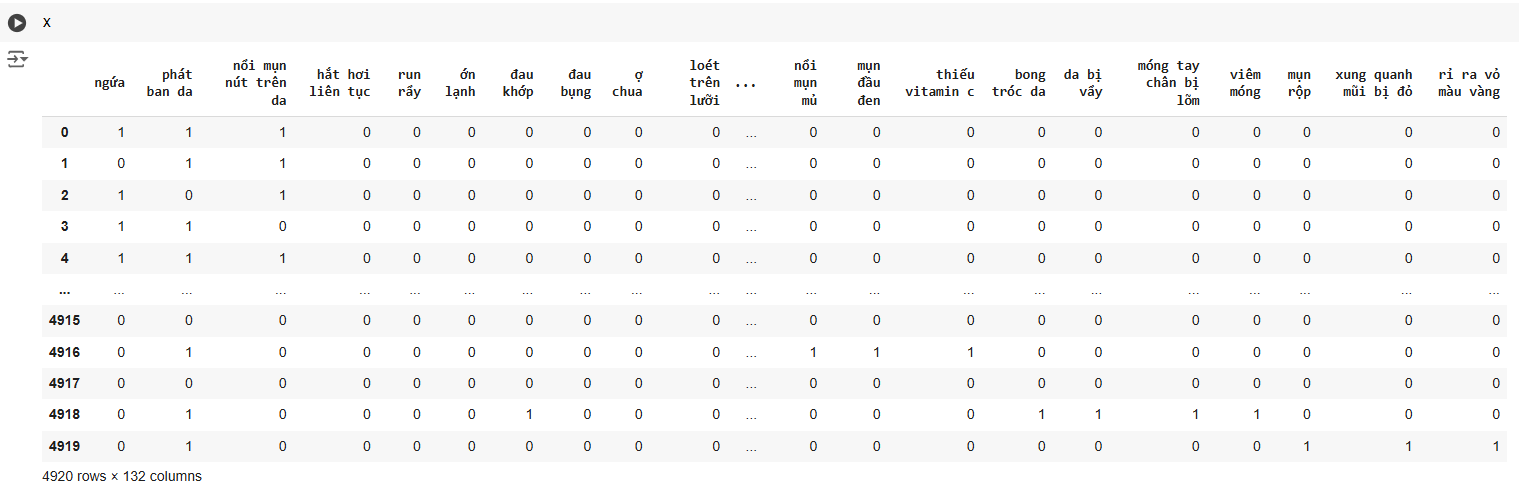
Bộ dữ liệu các triệu chứng biểu hiện gồm các bệnh cơ bản hay thường gặp cụ thể trong dữ liệu gồm 54 bệnh khác nhau đã được chúng em thu thập:



Hình 3.4. Tên các bệnh có trong dữ liệu

Và các triệu chứng, biểu hiện trong file triệu chứng gồm 4920 hàng các triệu chứng tương ứng với 54 bệnh 1 bệnh có thể có 1 hoặc nhiều triệu chứng cũng như 1 triệu chứng có thể là một trong các dự đoán của các bệnh

Sau đó thực hiện gán nhãn các triệu chứng biểu hiện của các bệnh với 4920 hàng và 132 cột triệu chứng:



Hình 3.5. Gán nhãn các triệu chứng bệnh

Chúng em đã xử lý trước và đã hợp nhất tập dữ liệu, các triệu chứng, biểu hiện để đáp ứng nhu cầu của chúng em nhằm có được thông tin sâu sắc và xác định xem tập dữ liệu có chứa dữ liệu cần thiết để giải quyết vấn đề.

1. *Xử lý dữ liệu*

Chúng em đã xử lý từng bảng dữ liệu riêng biệt và kết hợp chúng để tạo thành một tập dữ liệu hợp nhất, bao gồm các cột cần thiết sau khi thu thập dữ liệu thô và xác định cấu trúc của cả bốn tệp CSV. Để xử lý dữ liệu, chúng em đã áp dụng các hướng dẫn cụ thể, chẳng hạn như đếm số lượng giá trị duy nhất, số lượng mục trong mỗi hàng, cũng như loại bỏ hoặc kết hợp các cột không cần thiết hoặc không chứa dữ liệu hữu ích.

Sau khi hoàn thành các bước xử lý ban đầu, chúng em nhận thấy bốn bảng dữ liệu được tổ chức theo mô hình lược đồ, với bảng chính là ***symtoms\_df.csv*** và các bảng bao gồm ***precautions\_df.csv, description.csv, medications.csv, diets.csv, workout\_df.csv***  Mối quan hệ giữa các bảng là mối quan hệ một-nhiều, theo mô hình lược đồ hình sao. Cuối cùng, chúng em đã kết hợp bốn bảng này thành một bảng duy nhất sau khi xử lý các hàng và cột một cách hợp lý để đảm bảo tính chính xác và tính liên kết của dữ liệu.

Dữ liệu ***symtoms\_df.csv*** *bao gồm 4920 hàng cùng 5 cột là cột tên bệnh cùng 4 biểu hiện triệu chứng.*



Hình 3.6. Dữ liệu file symtoms\_df.csv

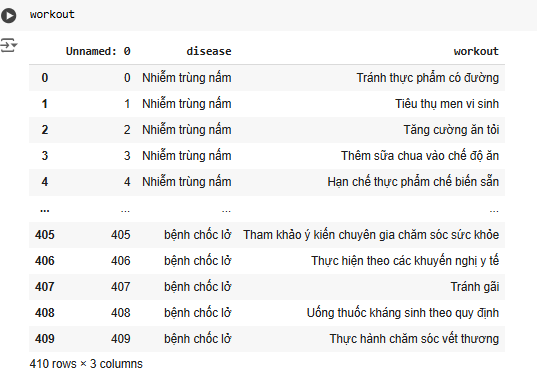
Dữ liệu ***precautions.csv*** *có* mỗi cột chứa một biện pháp phòng ngừa hoặc khuyến nghị để quản lý bệnh tương ứng. Các biện pháp bao gồm hành động cụ thể như: ngừng kích ứng, tư vấn bệnh viện gần nhất, bôi calamine, tránh đồ ăn cay...



Hình 3.7. Dữ liệu file precautions\_df.csv



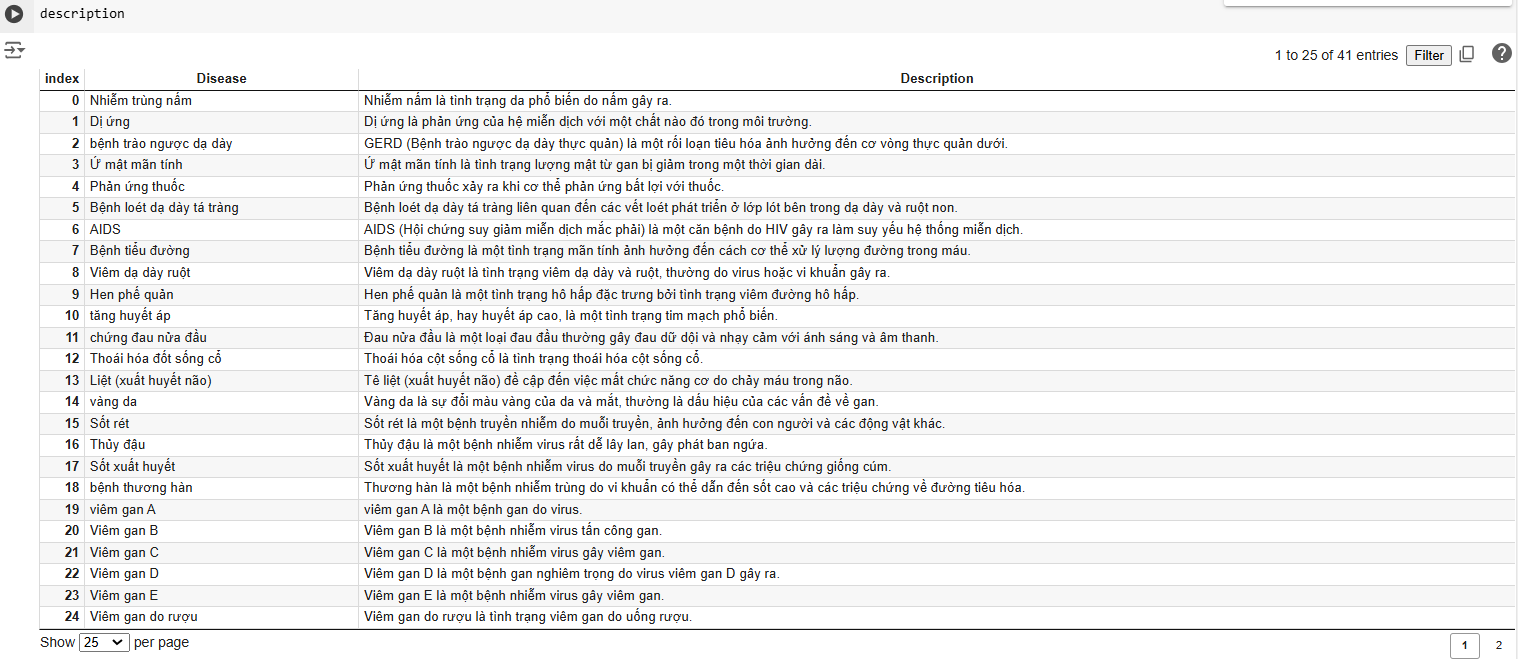
Hình 3.8. Dữ liệu file diets.csv



Hình 3.9. Dữ liệu file workout.csv

File workout.csv có tổng cộng có 410 dòng dữ liệu. Cột "disease" là danh sách các loại bệnh hoặc tình trạng y tế, ví dụ: *Nhiễm trùng nấm, Bệnh chốc lở...*Cột "workout" chứa các gợi ý hoặc biện pháp liên quan đến thói quen, chế độ sinh hoạt, hoặc lời khuyên chăm sóc sức khỏe phù hợp với từng bệnh.

Các bệnh được sắp xếp theo thứ tự chỉ số từ 0 đến 409, bao gồm cả bệnh phổ biến như *Nhiễm trùng nấm* và bệnh nghiêm trọng hơn như *Bệnh chốc lở* và sau đó đưa ra gợi ý các biện pháp chăm sóc sức khỏe hoặc chế độ sinh hoạt cho từng bệnh lý. Và cuối cùng là file description.csv mô tả giải thích các bệnh:



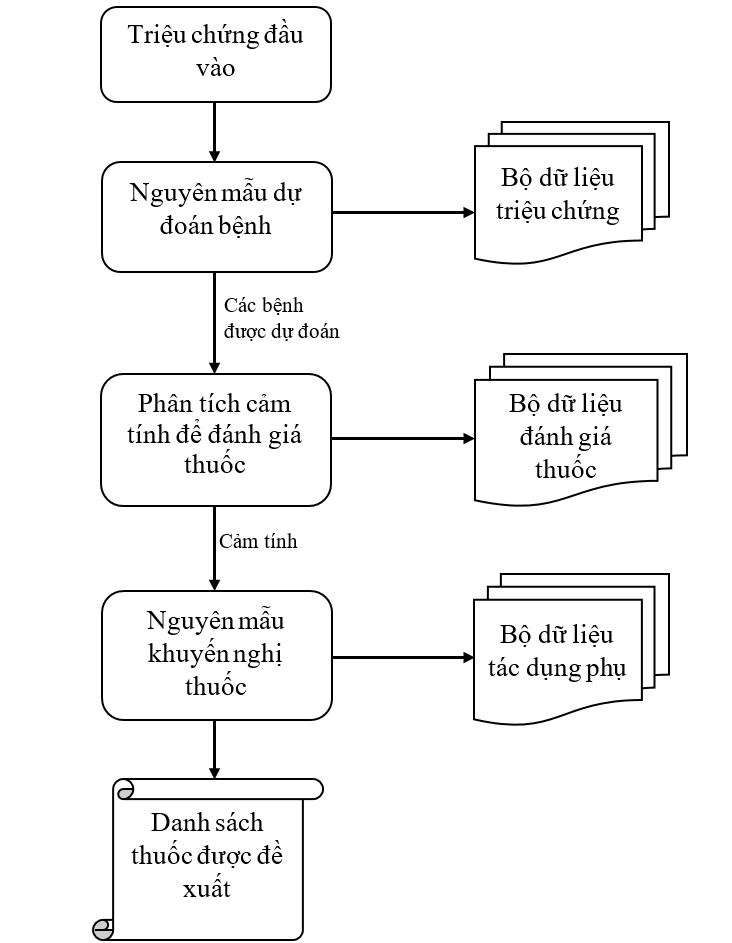
Hình 3.10. File dữ liệu description.csv

***3.3.2 Xây dựng chương trình dự đoán bệnh thường gặp ở người***

Một mô hình xác suất dựa trên các dự đoán về triệu chứng được sử dụng như một nguyên mẫu để dự đoán bệnh tật. Chúng em đã xây dựng một cơ sở dữ liệu kiến thức về các triệu chứng và bệnh lý, bao gồm 132 triệu chứng liên quan đến hơn 40 bệnh khác nhau với mục đích ứng dụng trong dự đoán.

Trong nghiên cứu, chúng em đã thử nghiệm nhiều phương pháp dự báo bệnh lý, do dữ liệu chỉ chứa một tính năng duy nhất là các triệu chứng, điều này làm giảm độ chính xác của các mô hình phân loại.Để cải thiện, chúng em đã cố gắng tạo thêm nhiều điểm dữ liệu cho cùng một bệnh lý bằng cách xem xét toàn bộ các triệu chứng, tầm quan trọng và sự xuất hiện của chúng.

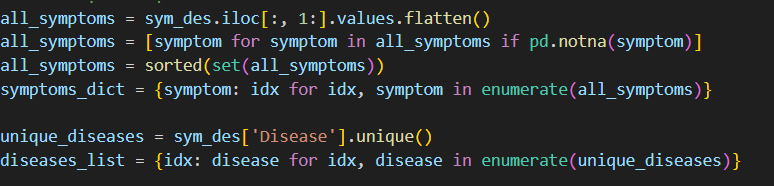
Dữ liệu được huấn luyện với mô hình phân loại máy vectơ hỗ trợ (SVM), với các triệu chứng đóng vai trò là tính năng đào tạo và bệnh lý là nhãn. Các mô hình phân loại sau đó được sử dụng để dự báo bệnh tật dựa trên một, hai hoặc ba đặc điểm thậm chí bốn.



Hình 3.11. Mô hình chương trình

Đây là mô hình học máy tuyến tính mạnh mẽ để phân loại, được sử dụng để tìm ra ranh giới tốt nhất giữa các lớp trong không gian nhiều chiều. Trong đoạn mã, hàm svc.fit(X, y) huấn luyện mô hình này dựa trên ma trận triệu chứng (X) và nhãn bệnh (y).

Đầu tiên sẽ lấy tất cả các triệu chứng từ cột thứ hai trở đi trong bảng dữ liệu sym\_des (ngoại trừ cột đầu tiên là tên bệnh), sau đó "làm phẳng" các giá trị này thành một mảng một chiều. Sau đó loại bỏ các giá trị bị thiếu (NA/null) sắp xếp theo bàng chũ cái và đánh số thứ tự mỗi triệu chứngđể đảm bảo chỉ giữ lại các triệu chứng hợp lệ.

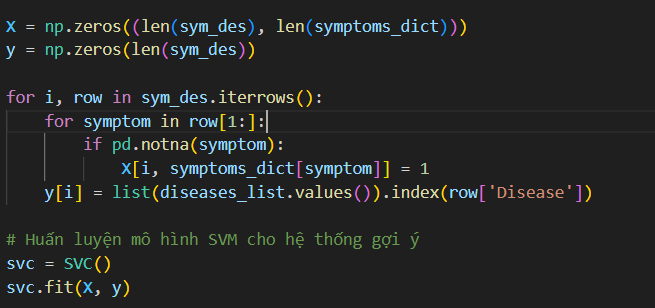


Hình 3.12. Chuẩn bị dữ liệu

Trước khi huấn luyện mô hình, cần xây dựng các ma trận đặc trưng biến đổi dữ liệu triệu chứng và bệnh thành ma trận đầu vào (X) và vector mục tiêu (y) cho mô hình học máy:

X = np.zeros((len(sym\_des), len(symptoms\_dict))): Tạo ma trận đầu vào X với kích thước số dòng của sym\_des x số triệu chứng. Tất cả các giá trị ban đầu được khởi tạo bằng 0.

y = np.zeros(len(sym\_des)): Tạo vector mục tiêu y để lưu nhãn bệnh của mỗi dòng.



Hình 3.13. Huấn luyện mô hình SVM với bệnh và triệu chứng

Dữ liệu triệu chứng được gán giá trị 1 nếu liên quan đến bệnh cụ thể và 0 nếu không. Một khung dữ liệu mới được xây dựng, trong đó số lượng hàng cho mỗi bệnh được nhân lên để đảm bảo cân bằng giữa các bệnh. Phương pháp xác suất được áp dụng để ánh xạ các giá trị, ưu tiên triệu chứng quan trọng nhất với giá trị 1. Triệu chứng thứ hai được ánh xạ tới 95% tổng số hàng, trong khi khoảng 40% hàng được ánh xạ với triệu chứng ít quan trọng nhất.

Cuối cùng, các tập dữ liệu nhỏ với các giá trị triệu chứng khác nhau được hợp nhất lại tạo thành một tập dữ liệu lớn, chứa nhiều hàng và điểm dữ liệu đa dạng cho từng bệnh. Chiến lược này đã giúp nhóm chúng em cải thiện đáng kể hiệu quả và độ chính xác của các mô hình phân loại.

**3.3. Kết quả đạt được**

Giao diện của chương trình dự đoán bệnh được thiết kế đơn giản, dễ sử dụng và trực quan. Các thành phần trong giao diện được bố trí hợp lý và có chức năng rõ ràng, giúp người dùng dễ dàng nhập triệu chứng, nhận kết quả dự đoán và tìm hiểu thông tin chi tiết về bệnh. Chương trình không chỉ cung cấp thông tin về bệnh mà còn giúp người dùng có những hướng dẫn chi tiết về cách phòng ngừa, điều trị và chăm sóc sức khỏe.

Các thành phần chính trong giao diện bao gồm:



Hình 3.14. Giao diện chính của chương trình

Ở đầu giao diện, chương trình hiển thị một tiêu đề lớn với tên gọi "Chuẩn đoán bệnh thường gặp". Đây là phần giới thiệu chương trình và giúp người dùng biết mục đích sử dụng của ứng dụng. Tiêu đề sử dụng phông chữ **Arial**, cỡ chữ **22**, và được in đậm để tạo sự chú ý. Màu nền của tiêu đề là màu trắng, trong khi màu chữ là màu đen, tạo nên sự rõ ràng và dễ đọc.

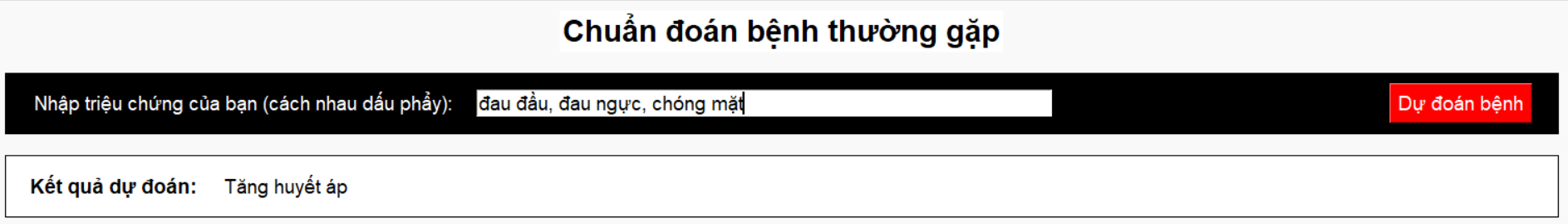
Khung nhập liệu triệu chứng là phần quan trọng nhất trong giao diện, nơi người dùng có thể nhập các triệu chứng của mình để hệ thống thực hiện dự đoán. Phần này có một **Entry** để nhập liệu và một **Button** để thực hiện việc dự đoán.

Label: Hiển thị thông báo yêu cầu người dùng nhập triệu chứng, với nội dung "Nhập triệu chứng của bạn (cách nhau dấu phẩy)".

Entry: Ô nhập liệu dài có thể chứa đến 50 ký tự, giúp người dùng dễ dàng nhập các triệu chứng của mình.

Button: Nút "Dự đoán bệnh" khi được nhấn sẽ thực hiện việc phân tích và đưa ra kết quả bệnh lý tương ứng với các triệu chứng đã nhập.

Sau khi người dùng nhập triệu chứng và nhấn nút dự đoán, chương trình sẽ hiển thị kết quả dự đoán bệnh trong một khung riêng biệt.



Hình 3.15. Giao diện phần nhập triệu chứng và nút dự đoán

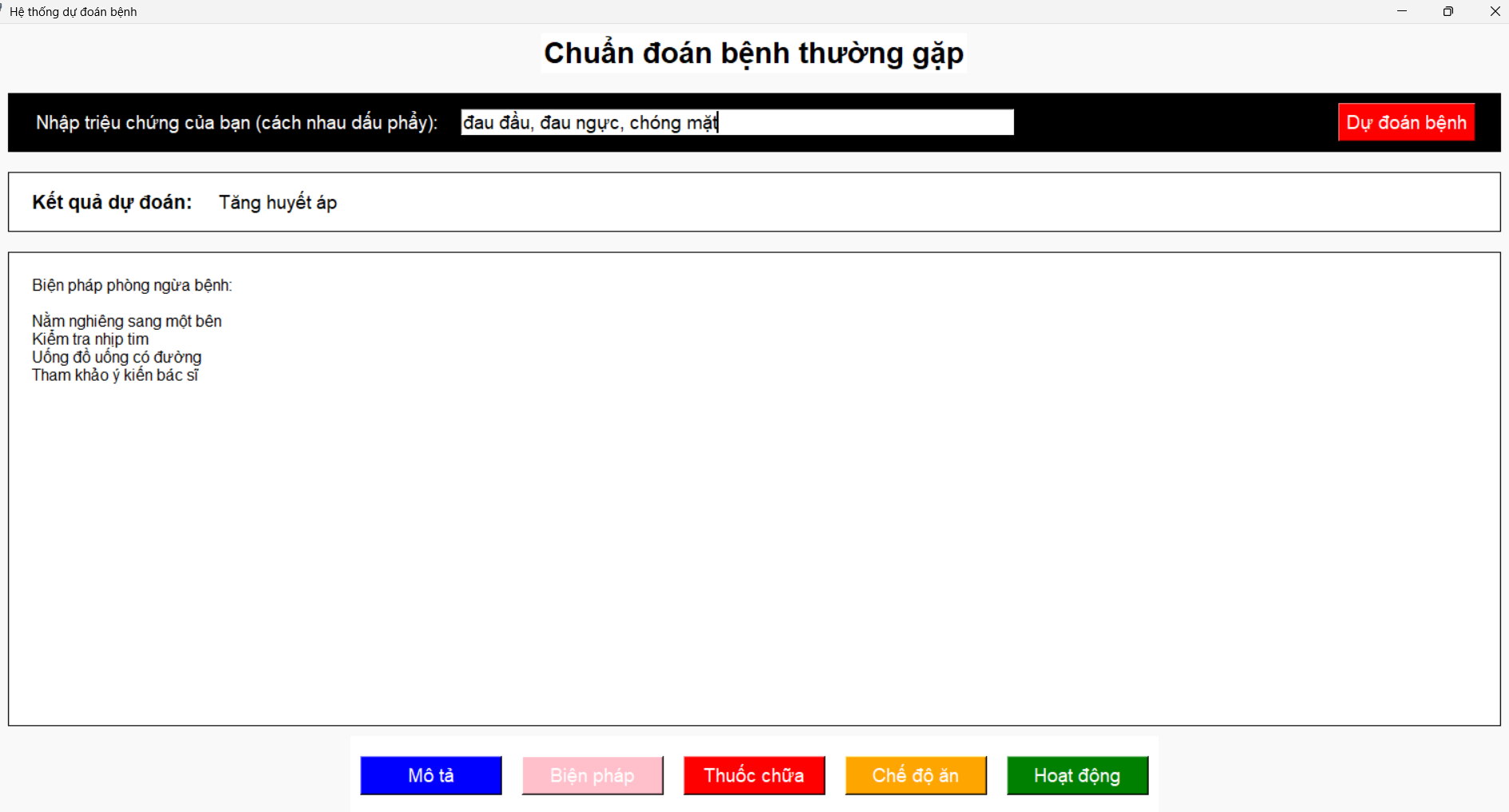
Đây là phần quan trọng để cung cấp thông tin chi tiết về bệnh lý đã được dự đoán. Sau khi dự đoán bệnh, người dùng có thể nhấn vào các nút chức năng để tìm hiểu thêm về bệnh.

* **Mô tả bệnh**: Người dùng có thể xem mô tả chi tiết về bệnh lý, bao gồm các triệu chứng, nguyên nhân và các yếu tố liên quan.



Hình 3.16. Kết quả hiển thị khi ấn nút mô tả

* **Biện pháp phòng ngừa**: Cung cấp các biện pháp phòng ngừa để tránh mắc bệnh hoặc để giảm nhẹ các triệu chứng.



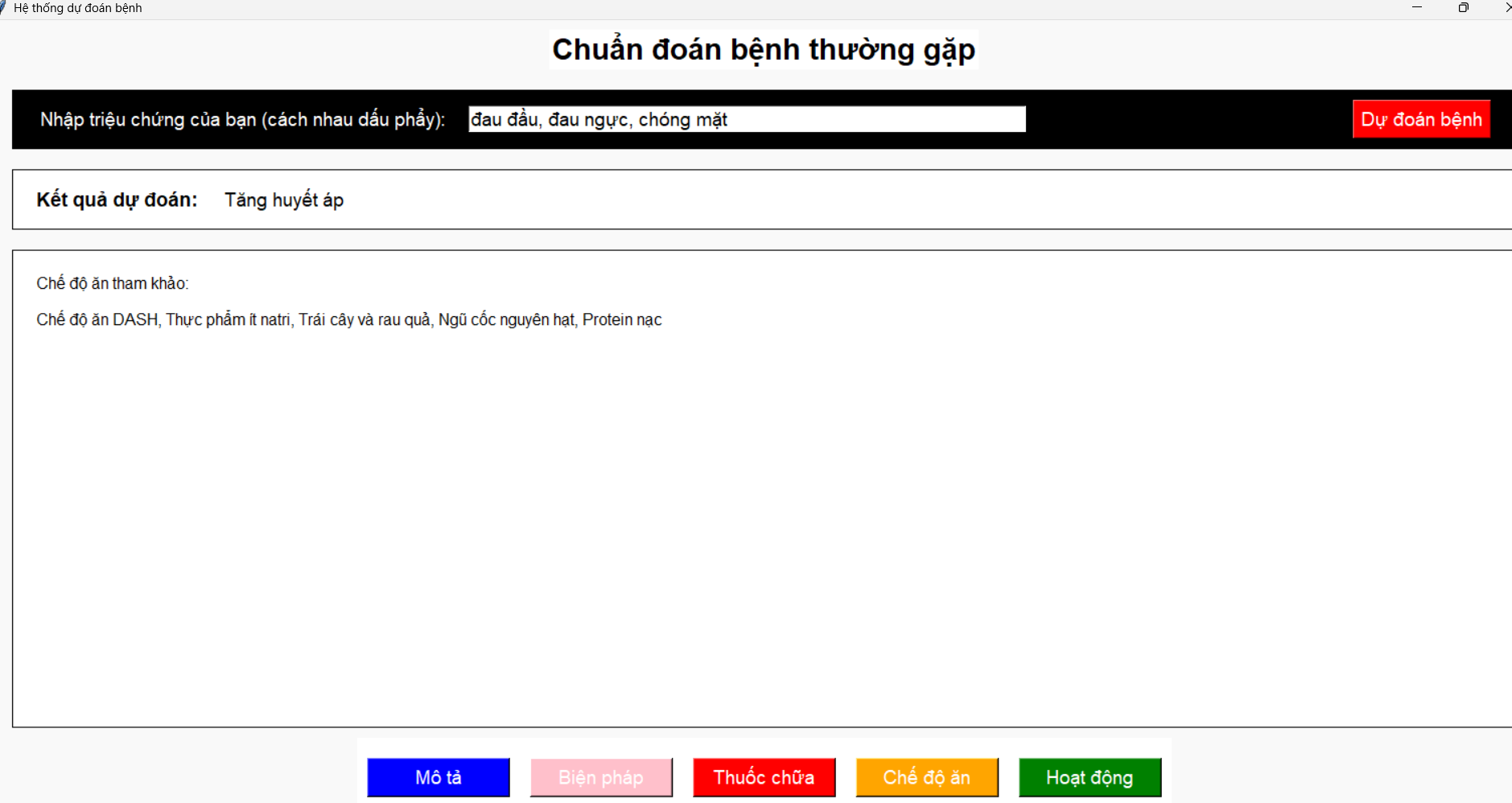
Hình 3.17. Kết quả hiển thị khi ấn nút biện pháp

* **Thuốc chữa**: Hiển thị danh sách các loại thuốc có thể điều trị cho bệnh lý đã dự đoán.



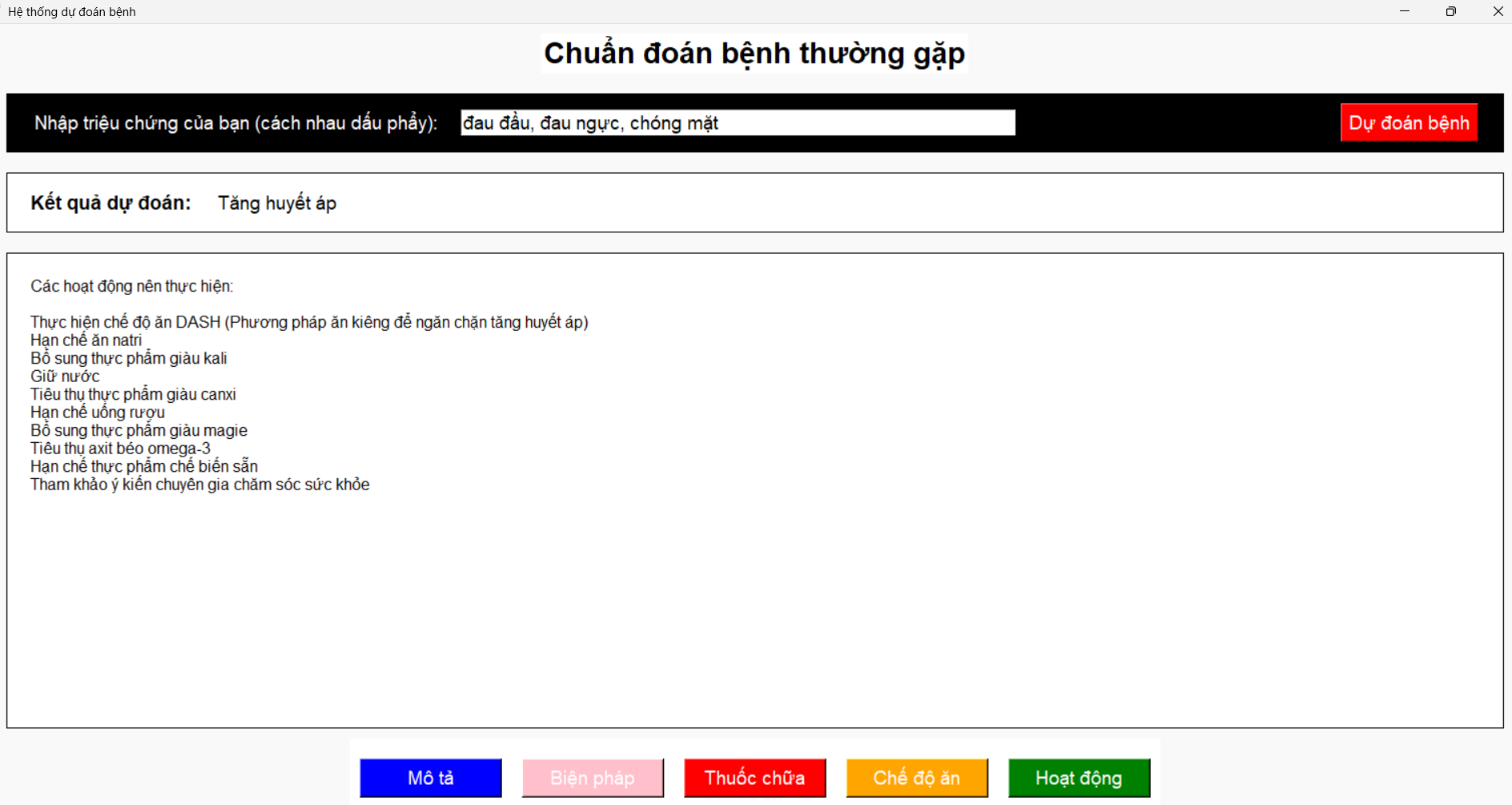
Hình 3.18. Kết quả hiển thị khi ấn nút thuốc chữa

* **Chế độ ăn tham khảo**: Đưa ra các gợi ý về chế độ ăn phù hợp cho bệnh nhân với bệnh lý cụ thể.



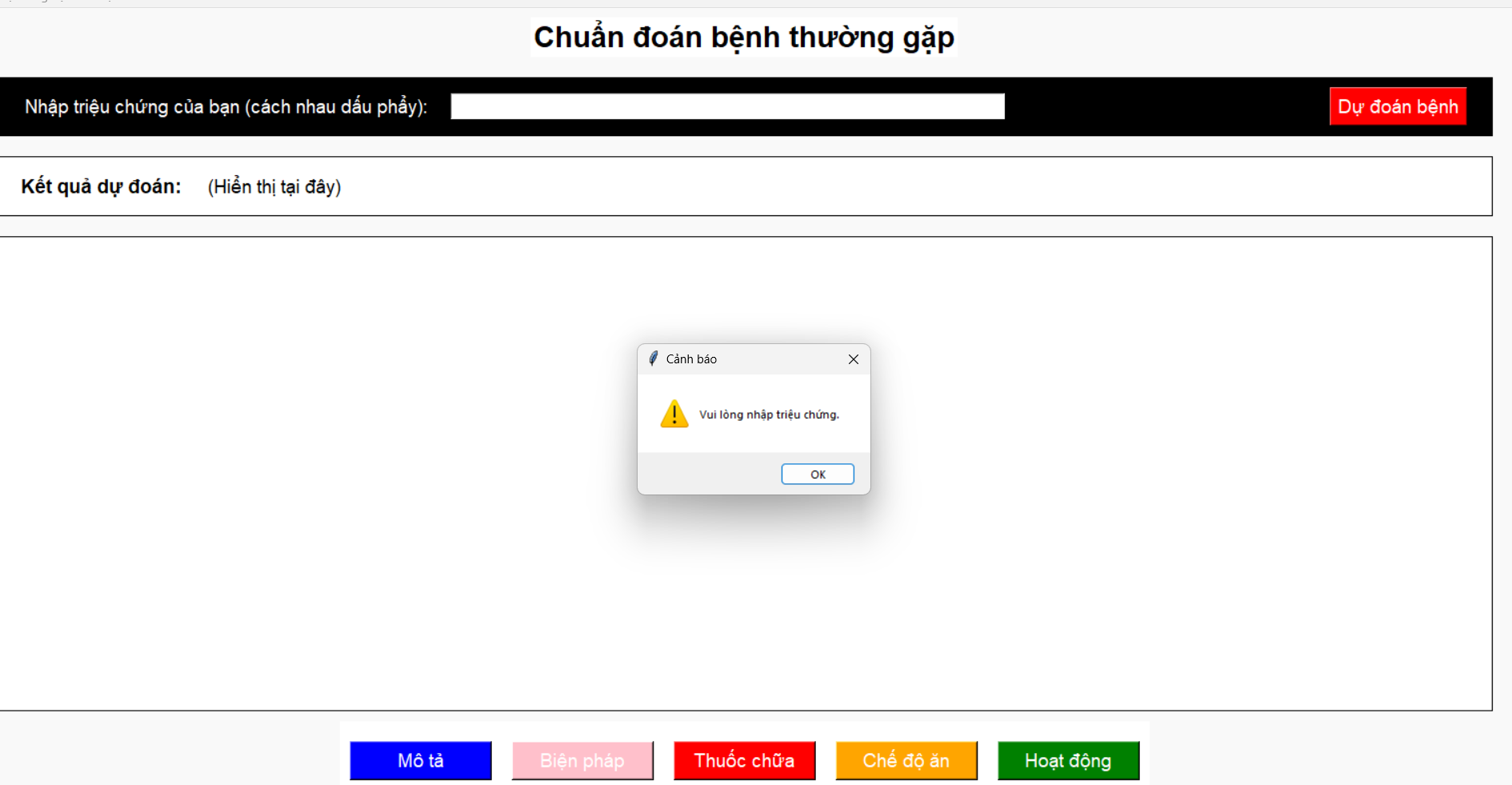
Hình 3.19. Kết quả hiển thị khi ấn nút chế độ ăn

* **Các hoạt động thể chất**: Giới thiệu các hoạt động thể chất hay lời khuyên có thể giúp bệnh nhân phục hồi sức khỏe hoặc giảm nhẹ triệu chứng của bệnh.



Hình 3.20. Kết quả hiển thị khi ấn nút hoạt động

Chương trình cũng có các cơ chế thông báo lỗi để đảm bảo rằng người dùng có thể nhận biết được những vấn đề xảy ra trong quá trình sử dụng: Nếu người dùng không nhập triệu chứng và nhấn nút "Dự đoán bệnh", hệ thống sẽ hiển thị thông báo yêu cầu nhập triệu chứng. Nếu triệu chứng nhập vào không hợp lệ hoặc không có trong cơ sở dữ liệu, hệ thống sẽ hiển thị một thông báo lỗi. Cảnh báo khi chưa dự đoán bệnh: Nếu người dùng chưa thực hiện việc dự đoán bệnh mà nhấn các nút chức năng chi tiết (Mô tả, Biện pháp, Thuốc chữa, Chế độ ăn, Hoạt động), hệ thống sẽ thông báo yêu cầu dự đoán bệnh trước.



Hình 3.21. Cảnh báo khi chưa nhập triệu chứng đã ấn dự đoán

# **KẾT LUẬN**

**4.1. Tổng kết**

Trong công việc này, chúng em đã xây dựng thành công một chương trình chuẩn đoán bệnh dựa trên phương pháp học máy **SVM (Support Vector Machine)**. Đây là một hệ thống có khả năng phân tích dữ liệu, học hỏi từ các đặc trưng đầu vào, và đưa ra kết luận về tình trạng sức khỏe của bệnh nhân một cách chính xác. Phương pháp SVM được lựa chọn do khả năng phân tách tối ưu và hiệu quả trong việc giải quyết các bài toán phân loại. Thông qua quá trình thực hiện, hệ thống đã được kiểm chứng trên dữ liệu thực tế, đem lại kết quả đáng khích lệ, mở ra nhiều tiềm năng ứng dụng trong lĩnh vực chăm sóc sức khỏe thông minh. Nhìn chung, chương trình chuẩn đoán bệnh sử dụng SVM đã đạt được những kết quả tích cực và khẳng định tiềm năng phát triển lâu dài. Dù còn một số hạn chế cần khắc phục, hệ thống hứa hẹn sẽ là một công cụ hữu ích trong việc hỗ trợ y tế, góp phần nâng cao chất lượng dịch vụ chăm sóc sức khỏe và cải thiện đời sống con người. Với sự nỗ lực không ngừng, em tin rằng trong tương lai, chương trình sẽ được hoàn thiện và có thể trở thành một giải pháp công nghệ thực sự hiệu quả và tiên tiến trong lĩnh vực y học.

***4.1.1 Ưu điểm***

Một trong những ưu điểm nổi bật của hệ thống là khả năng hoạt động với độ chính xác cao. Nhờ sử dụng phương pháp SVM, mô hình có khả năng tìm ra ranh giới phân loại tối ưu giữa các lớp dữ liệu, giúp tăng độ tin cậy trong chẩn đoán. Đồng thời, hệ thống cũng thể hiện sự linh hoạt khi hoạt động tốt với dữ liệu không đồng nhất hoặc có sự nhiễu nhẹ, đảm bảo hiệu quả tổng quát trong các tình huống thực tế. Quá trình xử lý và dự đoán được thực hiện nhanh chóng, phù hợp với yêu cầu thời gian trong các ứng dụng thực tiễn, đặc biệt là trong môi trường y tế, nơi mỗi giây phút đều quý giá.

***4.1.2 Nhược điểm***

Tuy nhiên, bên cạnh những ưu điểm, hệ thống vẫn tồn tại một số hạn chế. Đầu tiên, chất lượng đầu ra phụ thuộc rất lớn vào tập dữ liệu huấn luyện. Nếu dữ liệu không đủ đa dạng hoặc bị thiếu các đặc trưng quan trọng, độ chính xác của mô hình có thể bị ảnh hưởng. Ngoài ra, với các tập dữ liệu lớn hoặc có nhiều chiều, quá trình huấn luyện SVM có thể yêu cầu nhiều tài nguyên tính toán, gây ra thách thức khi triển khai trên các hệ thống có giới hạn về phần cứng. Cuối cùng, việc áp dụng SVM cho các bài toán phức tạp với nhiều lớp phân loại cũng gặp khó khăn, đòi hỏi sự tùy chỉnh mô hình để đạt được hiệu quả tối ưu.

**4.2. Định hướng tương lai**

Dựa trên những kết quả và bài học thu được, trong tương lai, em định hướng sẽ cải thiện và phát triển hệ thống theo nhiều hướng khác nhau. Một trong những mục tiêu quan trọng là tăng cường chất lượng và đa dạng hóa tập dữ liệu bằng cách thu thập thêm các mẫu từ nhiều nguồn khác nhau, đảm bảo rằng hệ thống có thể xử lý được các trường hợp đa dạng hơn. Bên cạnh đó, việc tối ưu hóa mô hình cũng là một ưu tiên hàng đầu. Các phương pháp hiện đại như **Grid Search** hay **Bayesian Optimization** sẽ được áp dụng để tinh chỉnh các tham số của SVM, nâng cao hiệu suất của hệ thống. Em cũng sẽ nghiên cứu tích hợp SVM với các phương pháp học sâu (Deep Learning) để tận dụng lợi thế của cả hai công nghệ, mở rộng khả năng ứng dụng của hệ thống trong những bài toán phức tạp hơn.

Không chỉ dừng lại ở việc cải tiến mô hình, em còn hướng tới việc đưa hệ thống vào ứng dụng thực tế. Một số lĩnh vực có tiềm năng ứng dụng lớn bao gồm hỗ trợ bác sĩ trong việc ra quyết định chẩn đoán, xây dựng các công cụ theo dõi sức khỏe cá nhân, và phát triển các phần mềm chăm sóc sức khỏe thông minh. Hệ thống cũng sẽ được thử nghiệm trên nhiều loại bệnh khác nhau để kiểm chứng tính hiệu quả và khả năng mở rộng, từ đó tối ưu hóa cho từng trường hợp cụ thể.

# **TÀI LIỆU THAM KHẢO**

[1] Z. Chen, K. Marple, E. Salazar, G. Gupta, L. Tamil, A physician advisory system for chronic heart failure management based on knowledge patterns, Theory and Practice of Logic Programming 16 (5-6) (2016) 604–618.

[2] D. Almirall, S. N. Compton, M. Gunlicks-Stoessel, N. Duan, S. A. Murphy, De signing a pilot sequential multiple assignment randomized trial for developing an adaptive treatment strategy, Statistics in Medicine 31 (17) (2012) 1887–1902.

[3] I. R. Edwards, J. K. Aronson, Adverse drug reactions: definitions, diagnosis, and management, The Lancet 356 (9237) (2000) 1255–1259.

[4] L. L. Leape, D. W. Bates, D. J. Cullen, J. Cooper, H. J. Demonaco, T. Gallivan, R. Hallisey, J. Ives, N. Laird, G. Laffel, et al., Systems analysis of adverse drug events, Journal of The American Medical Association 274 (1) (1995) 35–43.

[5] J. P. Gupta, A. Singh, and R. K. Kumar, ‘‘A computer-based disease prediction and medicine recommendation system using machine learning approach,’’ Int. J. Adv. Res. Eng. Technol. (IJARET), vol. 12, no. 3, pp. 673–683, 2021.

[6] Y. Bao and X. Jiang, ‘‘An intelligent medicine recommender system framework,’’ in Proc. IEEE 11th Conf. Ind. Electron. Appl. (ICIEA), Jun. 2016,

pp. 1383–1388.

[7] U. Bhimavarapu, N. Chintalapudi, and G. Battineni, ‘‘A fair and safe usage drug recommendation system in medical emergencies by a stacked ANN,’’ Algorithms, vol. 15, no. 6, p. 186, May 2022.