TRƯỜNG ĐẠI HỌC LAO ĐỘNG XÃ HỘI
KHOA CÔNG NGHỆ THÔNG TIN

BÁO CÁO HỌC PHẦN NHÓM 3
TRÍ TUỆ NHÂN TẠO

Đề Tài 8: Ứng Dụng Kỹ Thuật Học Máy Trong Bài Toán Dự Đoán Ung Thư Phổi

Sinh viên thực hiện: Nguyễn Văn Nam
Đinh Quang Nam
Nguyễn Đức Mạnh

Lớp tín chỉ : D18CN01

Giảng viên hướng dẫn: Phạm Đức Trọng

TRƯỜNG ĐẠI HỌC LAO ĐỘNG XÃ HỘI
KHOA CÔNG NGHỆ THÔNG TIN

BÁO CÁO HỌC PHẦN NHÓM 3
TRÍ TUỆ NHÂN TẠO

Đề Tài 8: Ứng Dụng Kỹ Thuật Học Máy Trong Bài Toán Dự Đoán Ung Thư Phổi

Sinh viên thực hiện: Nguyễn Văn Nam
Đinh Quang Nam
Nguyễn Đức Mạnh

Lớp tín chỉ : D18CN01

Giảng viên hướng dẫn: Phạm Đức Trọng

LỜI CẢM ƠN

Lời đầu tiên, nhóm 8 chúng em xin gửi lời cảm ơn chân thành đến thầy Phạm Đức Trọng. Thầy đã tận tâm giảng dạy trên lớp, cung cấp những thông tin hữu ích và giải đáp những thắc mắc của nhóm trong suốt quá trình thực hiện đề tài. Qua các bài giảng về môn học, nhóm em đã học hỏi được nhiều kiến thức quý báu và áp dụng vào thực tiễn. Nhờ đó, nhóm có thể hiểu rõ hơn về quy trình phát triển phần mềm và ứng dụng các kỹ thuật hiện đại vào việc xây dựng hệ thống hoàn chỉnh.

Đề tài "Ứng Dụng Kỹ Thuật Học Máy Trong Bài Toán Dự Đoán Ung Thư Phổi" là kết quả của quá trình nghiên cứu nghiêm túc của nhóm, dưới sự hướng dẫn và chia sẻ tài liệu của thầy trong suốt thời gian giảng dạy. Những kiến thức này đã giúp nhóm hoàn thành bài tiểu luận một cách tốt nhất.

Cuối cùng, nhóm rất mong nhận được những ý kiến đóng góp quý báu từ thầy để bài tiểu luận này có thể hoàn thiện hơn nữa. Những góp ý của thầy, cô sẽ là kinh nghiệm quý giá để nhóm có thể áp dụng trong các dự án tiếp theo, đặc biệt là trong đồ án tốt nghiệp và những công việc sau này.

Chúng em xin chân thành cảm ơn!

MỤC LỤC

DANH MỤC VIẾT TẮT

Các Từ Viết Tắt	Tiếng Anh	Tiếng Việt
HTML	HTML	Ngôn ngữ đồ họa HTML
Bot	BOT	Trình duyệt ảo
User	User	Người dùng
JS	Java cpirit	Ngôn ngữ lập trình Java-Cpirit
ANN	Artificial Neural Network	Mạng nơ-ron nhân tạo
AI	Artificial Intelligence	Trí tuệ nhân tạo

LỜI MỞ ĐẦU

Lời đầu tiên, nhóm 8 chúng em xin gửi lời cảm ơn chân thành đến thầy Phạm Đức Trọng. Thầy đã tận tâm giảng dạy trên lớp, cung cấp những thông tin hữu ích và giải đáp những thắc mắc của nhóm trong suốt quá trình thực hiện đề tài. Qua các bài giảng về môn học, nhóm em đã học hỏi được nhiều kiến thức quý báu và áp dụng vào thực tiễn. Nhờ đó, nhóm có thể hiểu rõ hơn về quy trình phát triển phần mềm và ứng dụng các kỹ thuật hiện đại vào việc xây dựng hệ thống hoàn chỉnh.

Đề tài "Ứng Dụng Kỹ Thuật Học Máy Trong Bài Toán Dự Đoán Ung Thư Phổi" là kết quả của quá trình nghiên cứu nghiêm túc của nhóm, dưới sự hướng dẫn và chia sẻ tài liệu của thầy trong suốt thời gian giảng dạy. Những kiến thức này đã giúp nhóm hoàn thành bài tiểu luận một cách tốt nhất.

Cuối cùng, nhóm rất mong nhận được những ý kiến đóng góp quý báu từ thầy để bài tiểu luận này có thể hoàn thiện hơn nữa. Những góp ý của thầy, cô sẽ là kinh nghiệm quý giá để nhóm có thể áp dụng trong các dự án tiếp theo, đặc biệt là trong đồ án tốt nghiệp và những công việc sau này.

Chúng em xin chân thành cảm ơn!

1. Lý do chọn đề tài

Ngày nay, với sự phát triển không ngừng của công nghệ, việc giao tiếp giữa con người và máy tính ngày càng trở nên dễ dàng và hiệu quả hơn. Nhờ vào các kiến trúc học máy tiên tiến như mạng neural nhân tạo (ANN), mạng neural học sâu (DNN), các phương pháp này đã được ứng dụng rộng rãi trong nhiều lĩnh vực, từ thị giác máy tính, nhận dạng giọng nói, xử lý ngôn ngữ tự nhiên đến y khoa. Những ứng dụng này đã mang lại kết quả vượt trội so với các phương pháp truyền thống, giúp cải thiện đáng kể hiệu quả trong nhiều ngành nghề.

Đặc biệt, với sự phát triển mạnh mẽ của phần cứng, các máy tính ngày nay có khả năng thực hiện hàng tỷ phép toán trong một giây, tạo ra một nền tảng vững chắc cho các mạng học sâu như mạng tích chập (CNN) trở nên phổ biến hơn bao giờ hết. Mạng CNN không chỉ giúp máy tính có khả năng nhận dạng hình ảnh của các đối tượng mà còn có thể phân tích, xử lý thông tin hình ảnh một cách chính xác qua các lớp với bộ lọc tích chập.

Trong y khoa, đặc biệt là trong việc phát hiện ung thư, mạng CNN đã chứng minh được sự hiệu quả vượt trội. Hàng năm, ung thư vẫn là nguyên nhân gây tử vong hàng đầu trên thế giới, với hàng triệu người mắc và hàng triệu ca tử vong. Thời gian sống sót của các bệnh nhân ung thư giai đoạn muộn thường rất ngắn, chỉ khoảng 6 tháng kể từ khi phát hiện bệnh. Vì vậy, việc phát triển các phương pháp chẩn đoán nhanh chóng và chính xác là cực kỳ quan trọng.

Lịch sử chẩn đoán ung thư đã trải qua nhiều giai đoạn phát triển, từ các phương pháp thủ công như sờ, nghe đến các kỹ thuật hình ảnh hiện đại như siêu âm, X-quang, CT, MRI và các phương pháp chẩn đoán y học hạt nhân như SPECT, PET. Những công nghệ này đã giúp nâng cao độ chính xác trong việc phát hiện và điều trị ung thư.

Vì lẽ đó, việc áp dụng các kỹ thuật học sâu, đặc biệt là mạng CNN, trong nhận dạng và chẩn đoán ung thư phổi là rất cần thiết và cấp bách. Các hệ thống thông minh này sẽ hỗ trợ các bác sĩ trong việc phát hiện sớm ung thư, từ đó nâng cao cơ hội điều trị và cải thiện chất lượng cuộc sống cho bệnh nhân.

2. Mục tiêu đề tài

- Trình bày và giải thích được thuật toán CNN
- Sử dụng được công cụ Google Colab
- Ứng dụng ngôn ngữ python để giải quyết bài toán ***"Chẩn đoán ung thư phổi"***

3. Phương pháp nghiên cứu

- Xây dựng dữ liệu để huấn luyện mô hình cho phương pháp và đánh giá phương pháp.
- Tiến hành thực nghiệm và đánh giá kết quả dựa trên mô hình và dữ liệu.

4. Đối tượng và phạm vi nghiên cứu

- Các triệu chứng của bệnh ung thư phổi

5. Tổng quát

Tài liệu được chia làm 4 chương:

Chương 1 – TỔNG QUAN VỀ TRÍ TUỆ NHÂN TẠO: Chương này tập trung vào việc nghiên cứu và phân tích thị trường cũng như nhu cầu của người dùng đối với "Ứng Dụng Kỹ Thuật Học Máy Trong Bài Toán Dự Đoán Ung Thư Phổi". Chức năng chính trong chương này bao gồm nghiên cứu thị trường, thu thập yêu cầu.

Chương 2 – TỔNG QUAN VỀ Scikit-learn: Chương này sẽ tập trung vào việc thiết kế kiến trúc tổng thể của hệ thống, từ cơ sở dữ liệu đến giao diện người dùng. Một số chức năng chính gồm phân tích biểu đồ hệ thống, thiết kế cơ sở dữ liệu.

Chương 3 – DỰ ĐOÁN UNG THU PHỔI: Chương này sẽ hướng tới việc phát triển và thử nghiệm. Một số chức năng chính bao gồm Phát triển giao diện người dùng,...

Chương 4 – TỔNG KẾT: Chương cuối cùng này sẽ đánh giá chất lượng và hiệu suất của phần mềm như báo cáo kết quả đánh giá và định hướng trong tương lai.

CHƯƠNG I. TỔNG QUAN VỀ TRÍ TUỆ NHÂN TẠO

1.1 Khái niệm về trí tuệ nhân tạo

Trong lĩnh vực Công nghệ thông tin, Trí tuệ nhân tạo (TTNT) cũng có thể hiểu là "thông minh nhân tạo", tức là sự thông minh của máy móc do con người tạo ra, đặc biêt tạo ra cho máy tính, robot, hay các máy móc có các thành phần tính toán điện tử. TTNT là một ngành mới, nhưng phát triển rất mạnh mẽ và đem lại nhiều kết quả to lớn. Mùa hè 1956, tại hội thảo ở Darmouth John McCarthy đã đưa ra thuật ngữ trí tuệ nhân tạo (Artificial Intelligence - AI). Mốc thời gian này được xem là thời điểm ra đời thực sự của lĩnh vực nghiên cứu TTNT.

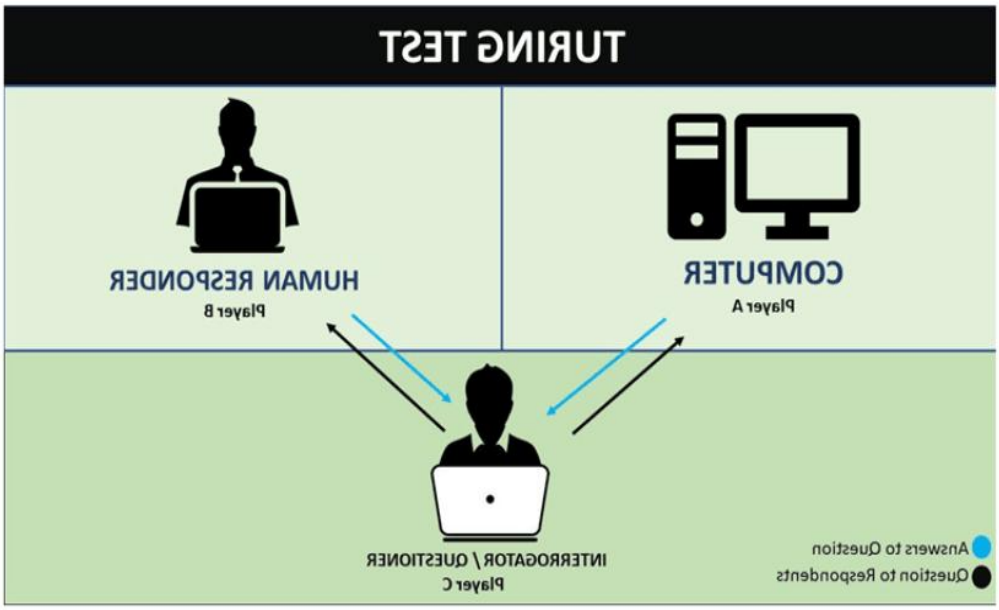
TTNT là một lĩnh vực nghiên cứu của khoa học máy tính và khoa học tính toán nói chung. Có nhiều quan điểm khác nhau về TTNT. Do đó có nhiều định nghĩa khác nhau về lĩnh vực này. Sau đây là một số định nghĩa [3]:

- "Sự nghiên cứu các năng lực trí tuệ thông qua viêc sử dụng các mô hình tính toán" (Charniak và McDormott, 1985).
- "Nghệ thuật tạo ra các máy thực hiên các chức năng đòi hỏi sự thông minh khi được thực hiên bởi con người" (Kurzweil, 1990).
- "Lĩnh vực nghiên cứu tìm cách giải thích và mô phỏng các hành vi thông minh trong thuật ngữ các quá trình tính toán" (Schalkoff, 1990).
- "Sự nghiên cứu các tính toán để có thể nhận thức, lập luận và hành động" (Winston, 1992).
- "Một nhánh của khoa học máy tính liên quan đến sự tự động hóa các hành vi thông minh" (Luger and Stubblefield, 1993).
- "TTNT là sự nghiên cứu thiết kế các tác nhân thông minh" (Poole, Mackworth and Goebel, 1998).

Trí tuệ nhân tạo là một nhánh của khoa học và công nghệ liên quan đến việc làm cho máy tính có những năng lực của trí tuệ con người, tiêu biểu như các khả năng biết suy nghĩ và lập luận để giải quyết vấn đề, biết giao tiếp do hiểu ngôn ngữ và tiếng nói, biết học và tự thích nghi,…[2].

Mong muốn làm cho máy có những khả năng của trí thông minh con người đã có từ nhiều thế kỷ trước, tuy nhiên TTNT chỉ xuất hiện khi con người sáng tạo ra máy tính điện tử. Alan Turing – nhà toán học lỗi lạc người Anh, người được xem là cha đẻ của Tin học do đưa ra cách hình thức hóa các khái niệm thuật toán và tính toán trên máy Turing – một mô hình máy trừu tượng mô tả bản chất viêc xử lý các ký hiêu hình thức -

có đóng góp quan trọng và thú vị cho TTNT vào năm 1950, gọi là phép thử Turing. Theo Turing: "Trí tuệ là những gì có thể đánh giá được thông qua các trắc nghiêm thông minh".

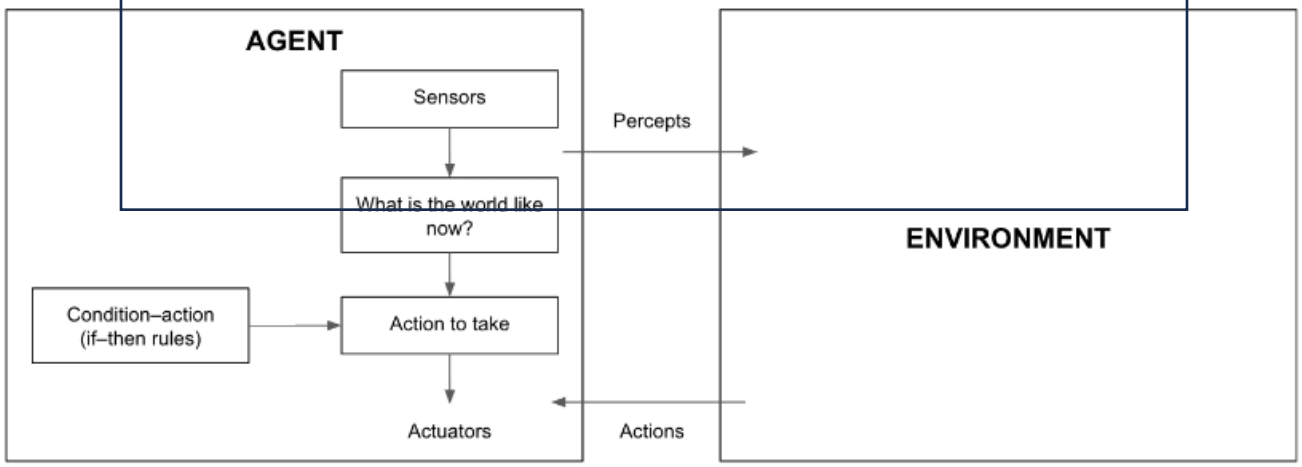


Phép thử Turing là một cách để trả lời câu hỏi "máy tính có biết nghĩ không?". Alan Turing đề xuất bộ kiểm thử (Turing test): Trong trắc nghiệm này, một máy tính và một người tham gia trắc nghiệm được đặt vào trong các căn phòng cách biệt với một người thứ hai (người thẩm vấn). Người thẩm vấn không biết được chính xác đối tượng nào là người hay máy tính, và cũng chỉ có thể giao tiếp với hai đối tượng đó thông qua các phương tiện kỹ thuật như một thiết bị soạn thảo văn bản, hay thiết bị đầu cuối. Người thẩm vấn có nhiệm vụ phân biệt người với máy tính bằng cách chỉ dựa trên những câu trả lời của họ đối với những câu hỏi được truyền qua thiết bị liên lạc này. Trong trường hợp nếu người thẩm vấn không thể phân biệt được máy tính với người thì khi đó theo Turing máy tính này có thể được xem là thông minh.

Khái niệm trí tuệ đưa ra trong từ điển bách khoa toàn thư:

Trí tuệ là khả năng: Phản ứng một cách thích hợp những tình huống mới thông qua hiệu chỉnh hành vi một cách thích đáng. Hiểu rõ những mối liên hệ qua lại của các sự kiện của thế giới bên ngoài nhằm đưa ra những hành động phù hợp đạt tới một mục đích nào đó.

Hiện nay nhiều nhà nghiên cứu quan niệm rằng, TTNT là lĩnh vực nghiên cứu sự thiết kế các tác nhân thông minh (intelligent agent). Tác nhân thông minh là bất cứ cái gì tồn tại trong môi trường và hành động một cách thông minh.



Theo M.Minskey: “Trí tuệ nhân tạo mô phỏng bằng máy tính để thí nghiêm một mô hình nào đó “.

TTNT là một ngành của khoa học máy tính - nghiên cứu xử lý thông tin bằng máy tính, do đó TTNT đặt ra mục tiêu nghiên cứu: làm thế nào thể hiện được các hành vi thông minh bằng thuật toán, rồi nghiên cứu các phương pháp cài đặt các chương trình có thể thực hiện được các hành vi thông minh bằng thuật toán, tiếp theo chúng ta cần chỉ ra tính hiệu quả, tính khả thi của thuật toán thực hiện một nhiệm vụ, và đưa ra các phương pháp cài đặt.

Mục tiêu của ngành TTNT: Nhằm tạo ra các máy tính có khả năng nhận thức, suy luận và phản ứng. Xây dựng TTNT là tìm cách biểu diễn tri thức và phát hiện tri thức từ các thông tin có sẵn để đưa vào trong máy tính. Để máy tính có các khái niêm nhận thức, suy luận, phản ứng thì ta cần phải cung cấp tri thức cho nó.

1.2. Vai trò của trí tuệ nhân tạo

Trí tuệ nhân tạo (AI) đóng vai trò ngày càng quan trọng trong nhiều lĩnh vực và ngành nghề, từ y tế, tài chính, giáo dục đến sản xuất và dịch vụ khách hàng. Dưới đây là một số vai trò chủ yếu của trí tuệ nhân tạo:

1. Tự động hóa các tác vụ lặp đi lặp lại

- AI giúp tự động hóa những công việc tốn thời gian và lặp đi lặp lại mà trước đây yêu cầu sự tham gia của con người. Điều này giúp giảm bớt khối lượng công việc cho con người, tăng hiệu suất và tiết kiệm thời gian. Ví dụ: trong các nhà máy sản xuất, các hệ thống AI có thể tự động kiểm tra chất lượng sản phẩm, lắp ráp các bộ phận, hoặc quản lý kho hàng.

2. Phân tích và xử lý dữ liệu lớn

- AI có khả năng xử lý và phân tích khối lượng dữ liệu khổng lồ một cách nhanh chóng và chính xác. Trong các lĩnh vực như tài chính, marketing, và y tế, AI có thể tìm ra các mẫu dữ liệu quan trọng và đưa ra các quyết định dựa trên những phân tích này. Điều này giúp các công ty đưa ra các quyết định chiến lược, tối ưu hóa chiến dịch marketing hoặc phát hiện bệnh sớm từ dữ liệu y tế.

3. Cải thiện khả năng ra quyết định

- AI hỗ trợ trong việc phân tích dữ liệu, mô phỏng các tình huống và đưa ra các quyết định dựa trên các mô hình học máy. Ví dụ trong y tế, AI có thể hỗ trợ bác sĩ trong việc chẩn đoán bệnh, trong tài chính có thể giúp các nhà đầu tư dự đoán xu hướng thị trường. Việc sử dụng AI giúp tăng độ chính xác và nhanh chóng của các quyết định, đồng thời giảm thiểu sự sai sót của con người.

4. Hỗ trợ trong các lĩnh vực y tế

- AI đã và đang giúp cải thiện chất lượng chẩn đoán và điều trị bệnh tật. Các hệ thống AI có thể phân tích hình ảnh y tế (như X-quang, MRI, CT) để phát hiện các dấu hiệu bệnh sớm, từ đó giúp bác sĩ đưa ra chẩn đoán chính xác hơn. AI cũng có thể hỗ trợ trong việc phát triển các phương pháp điều trị mới, tối ưu hóa quá trình chăm sóc bệnh nhân và quản lý bệnh viện.

5. Cải thiện trải nghiệm người dùng

- Trong các dịch vụ khách hàng, AI như chatbot và trợ lý ảo có thể trả lời nhanh chóng các câu hỏi của khách hàng, giải quyết vấn đề của họ ngay lập tức mà không cần sự can thiệp của con người. Điều này không chỉ giúp giảm tải cho các nhân viên mà còn nâng cao trải nghiệm của người dùng bằng cách cung cấp hỗ trợ 24/7.

6. Tối ưu hóa quy trình sản xuất và chuỗi cung ứng

- AI có thể giúp các công ty tối ưu hóa quy trình sản xuất và chuỗi cung ứng bằng cách dự đoán nhu cầu thị trường, quản lý kho, và phân phối sản phẩm một cách hiệu quả. Ví dụ, các hệ thống AI có thể dự báo khi nào một sản phẩm sẽ hết hàng và tự động lên kế hoạch cho việc sản xuất và phân phối.

7. Phát triển xe tự lái

- AI là công nghệ chủ đạo đằng sau sự phát triển của các xe tự lái. Các hệ thống AI có thể phân tích hình ảnh từ cảm biến và camera, nhận diện các vật thể xung quanh (xe, người đi bộ, biển báo giao thông) và đưa ra các quyết định lái xe an toàn. Điều này giúp giảm thiểu tai nạn giao thông và cải thiện tính hiệu quả của giao thông.

8. Giảm thiểu rủi ro và tăng cường bảo mật

- AI có thể được sử dụng để phát hiện các hành vi gian lận trong các giao dịch tài chính, cải thiện an ninh mạng và bảo vệ hệ thống khỏi các mối đe dọa. Các thuật toán học máy có thể phân tích các mô hình giao dịch để phát hiện các bất thường, giúp các tổ chức ngăn ngừa hành vi gian lận.

9. Học và cải tiến liên tục

- AI có khả năng học hỏi và cải thiện theo thời gian thông qua các thuật toán học máy. Điều này có nghĩa là hệ thống AI có thể trở nên thông minh hơn và hiệu quả hơn khi làm việc với nhiều dữ liệu hơn, từ đó tối ưu hóa quá trình và kết quả công việc.

10. Tạo ra các ứng dụng sáng tạo

- AI không chỉ giúp con người giải quyết các vấn đề thực tế mà còn có thể được sử dụng để tạo ra các tác phẩm sáng tạo. Ví dụ, AI có thể sáng tác nhạc, tạo ra nghệ thuật, viết văn bản, thậm chí tạo ra các bộ phim hoạt hình. Điều này mở ra một tiềm năng vô cùng lớn trong lĩnh vực nghệ thuật và sáng tạo.

So sánh kỹ thuật lập trình truyền thống và kỹ thuật xử lý tri thức trong TTNT:

Chương Trình Truyền Thống	Kỹ Thuật TTNT
Xử lý dữ liệu	Xử lý tri thức
Bản chất chương trình là tính toán, xử lý theo các thuật toán	Bản chất chương trình là lập luận, xử lý theo các thuật giải heuristics
Xử lý tuần tự theo lô	Xử lý theo chế độ tương tác
Xử lý thông tin chính xác đầy đủ	Xử lý được các thông thin không chắc chắn, không chính xác
Chương trình = Cấu trúc dữ liệu + Giải thuật	AI = Tri thức + Suy diễn
Không giải thích trong quá trình thực hiện	Có thể giải thích hành vi hệ thống trong quá trình thực hiện

1.3. Các kỹ thuật cơ bản

Có nhiều kỹ thuật nghiên cứu, phát triển ngành khoa học TTNT. Tuy vậy, các kỹ thuật TTNT thường khá phức tạp khi cài đăt cụ thể, lý do là các kỹ thuật này thiên về xử lý các ký hiệu tượng trưng và đòi hỏi phải sử dụng những tri thức chuyên môn thuộc nhiều lĩnh vực khác nhau. Do vậy, các kỹ thuật TTNT hướng tới khai thác những tri thức về lĩnh vực đang quan tâm được mã hoá trong máy sao cho đạt được mức độ tổng quát, dễ hiểu, dễ diễn đạt thông qua ngôn ngữ chuyên môn gần gũi với ngôn ngữ tự nhiên, dễ khai thác nhằm thu hẹp các khả năng cần xét để đi tới lời giải cuối cùng.

Các kỹ thuật Trí tuê nhân tạo cơ bản bao gồm:

- **Lý thuyết giải bài toán và suy diễn thông minh:** Lý thuyết giải bài toán cho phép viết các chương trình giải câu đố, các trò chơi thông qua các suy luận mang tính người.
- **Lý thuyết tìm kiếm may rủi:** Lý thuyết này bao gồm các phương pháp và kỹ thuật tìm kiếm với sự hỗ trợ của thông tin phụ để giải bài toán một cách có hiệu quả.
- **Các ngôn ngữ về TTNT:** Để xử lý các tri thức người ta không chỉ sử dụng các ngôn ngữ lập trình dùng cho các xử lý dữ liệu số, mà cần có ngôn ngữ khác. Các ngôn ngữ chuyên dụng này cho phép lưu trữ và xử lý thông tin ký hiệu. Một số ngôn ngữ được nhiều người biết đến là LISP, PROLOG,...
- **Lý thuyết thể hiện tri thức và hệ chuyên gia:** Trí tuệ nhân tạo là khoa học về thể hiện và sử dụng tri thức. Mạng ngữ nghĩa, logic vị từ, Frame,… là các phương pháp biểu diễn tri thức thông dụng. Việc gắn liền cách thể hiện và sử dụng tri thức là cơ sở hình thành hệ chuyên gia.
- **Lý thuyết nhân dạng và xử lý tiếng nói:** Giai đoạn phát triển đầu của TTNT gắn với lý thuyết nhận dạng. Ứng dụng của phương pháp này trong việc nhận dạng chữ viết, âm thanh,…
- **Người máy:** Cuối những năm 70, người máy trong công nghiệp đã đạt được nhiều tiến bộ. Người máy có bộ phận cảm nhận và các cơ chế hoạt động được nối ghép theo sự điều khiển thông minh. Khoa học về cơ học và TTNT được tích hợp trong khoa học người máy.
- **Tâm lý học xử lý thông tin :** Các kết quả nghiên cứu của tâm lý học giúp Trí tuệ nhân tạo xây dựng các cơ chế trả lời theo hành vi, có ý thức; nó giúp cho viêc thực hiện các suy diễn mang tính người.
- **Ngoài ra, xử lý danh sách, kỹ thuât đệ quy, kỹ thuật quay lui và xử lý cú pháp**

hình thức là những kỹ thuật cơ bản của tin học truyền thống có liên quan trực tiếp đến TTNT.

1.4. Lịch sử phát triển

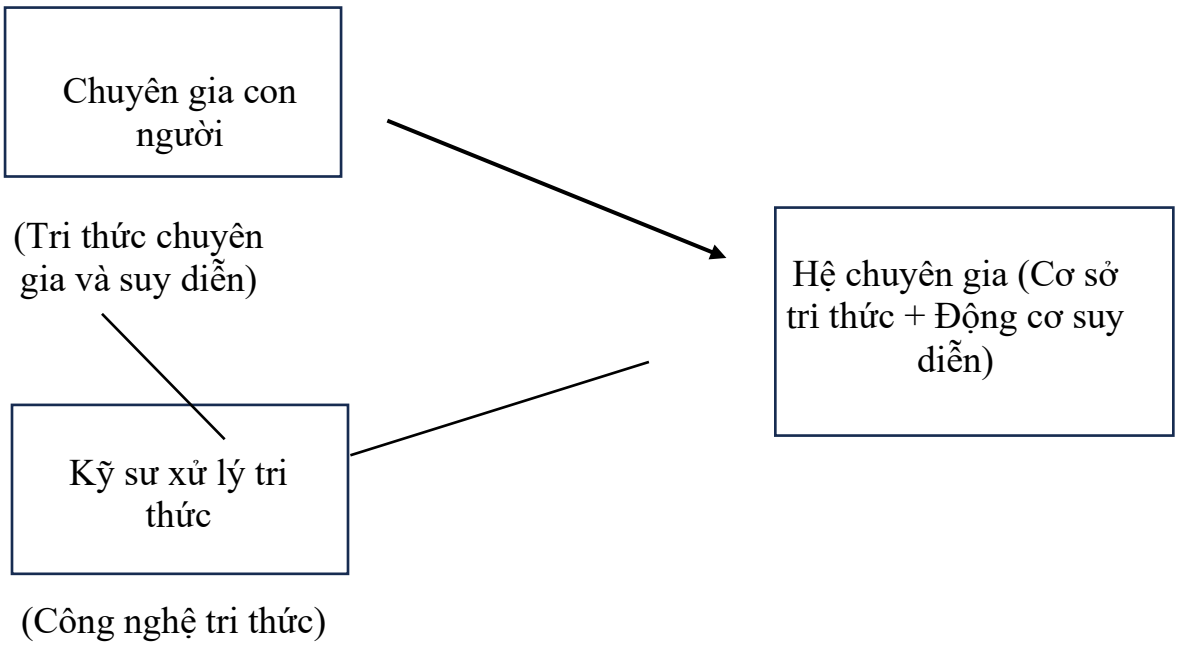
Lịch sử của TTNT cho thấy ngành khoa học này có nhiều kết quả đáng ghi nhận. Theo các mốc phát triển, người ta thấy TTNT được sinh ra từ những năm 50 với các sự kiện sau:

- Turing được coi là người khai sinh ngành TTNT bởi phát hiện của ông về máy tính có thể lưu trữ chương trình và dữ liệu.
- Tháng 8/1956 J.Mc Carthy, M. Minsky, A. Newell, Shannon. Simon,... đưa ra khái niệm "trí tuệ nhân tạo".
- Vào khoảng năm 1960 tại Đại học MIT (Massachusetts Institute of Technology) ngôn ngữ LISP ra đời, phù hợp với các nhu cầu xử lý đặc trưng của trí tuệ nhân tạo - đó là ngôn ngữ lập trình đầu tiên dùng cho trí tuệ nhân tạo.
- Thuật ngữ TTNT được dùng đầu tiên vào năm 1961 cũng tại MIT.
- Những năm 60 là giai đoạn lạc quan cao độ về khả năng làm cho máy tính biết suy nghĩ. Trong giai đoạn này người ta đã được chứng kiến máy chơi cờ đầu tiên và các chương trình chứng minh định lý tự động. Cụ thể:

    - 1961: Chương trình tính tích phân bất định
    - 1963: Các chương trình heuristics: Chương trình chứng minh các định lý hình học không gian có tên là "tương tự", chương trình chơi cờ của Samuel.
    - 1964: Chương trình giải phương trình đại số sơ cấp, chương trình trợ giúp ELIZA (có khả năng làm việc giống như một chuyên gia phân tích tâm lý).
    - 1966: Chương trình phân tích và tổng hợp tiếng nói
    - 1968: Chương trình điều khiển người máy (Robot) theo đồ án "Mắt – tay", chương trình học nói.

- Vào những năm 60, do giới hạn khả năng của các thiết bị, bộ nhớ và đặc biệt là yếu tố thời gian thực hiện nên có sự khó khăn trong việc tổng quát hoá các kết quả cụ thể vào trong một chương trình mềm dẻo thông minh.
- Vào những năm 70, máy tính với bộ nhớ lớn và tốc độ tính toán nhanh nhưng các phương pháp tiếp cận TTNT cũ vẫn thất bại do sự bùng nổ tổ hợp trong quá trình tìm kiếm lời giải các bài toán đặt ra.
- Vào cuối những năm 70 một vài kết quả như xử lý ngôn ngữ tự nhiên, biểu diễn tri thức và giải quyết vấn đề. Những kết quả đó đã tạo điều kiện cho sản phẩm thương mại đầu tiên của TTNT ra đời đó là Hệ chuyên gia, được đem áp dụng trong các lĩnh vực khác nhau (Hệ chuyên gia là một phần mềm máy tính chứa các

thông tin và tri thức về một lĩnh vực cụ thể nào đó, có khả năng giải quyết những yêu cầu của người sử dụng trong một mức độ nào đó, ở một trình độ như một chuyên gia con người có kinh nghiệm khá lâu năm). Hệ chuyên gia thay thế con người / trợ giúp con người ra quyết định.



Hình 1.4. Mô hình hê chuyên gia

- Một sự kiện quan trọng vào những năm 70 là sự ra đời ngôn ngữ Prolog, tương tự LISP nhưng nó có cơ sở dữ liệu đi kèm.
- Vào những năm 80 chứng kiến sự hồi sinh, bùng nổ và thi đua quốc tế trong ngành TTNT. Ý tưởng cơ bản để phát triển TTNT khi này là sự thông minh của máy tính không thể chỉ dựa trên việc suy diễn logic mà phải dựa cả vào (Công nghệ tri thức) (Tri thức chuyên gia và suy diễn) Chuyên gia con người Kỹ sư xử lý tri thức Hệ chuyên gia (Cơ sở tri thức + Động cơ suy diễn) tri thức của con người, và dùng khả năng suy diễn của máy để khai thác tri thức này. Cốt lõi của TTNT có thể diễn giải bởi công thức
- **TTNT = Tri thức + Suy diễn**
- Thành quả và nỗ lực tiêu biểu trong giai đoạn này là sự phát triển của các hệ chuyên gia. Mỗi hệ chuyên gia về cơ bản gồm hai thành phần: Cơ sở tri thức chứa các tri thức chuyên gia trong một lĩnh vực và một cơ chế suy diễn nhằm vận dụng các hiểu biết này để giải quyết các vấn đề cụ thể với hiệu quả như chính chuyên gia giải quyết. Hai hệ chuyên gia tiêu biểu là DENDRAL và MYCIN. Hệ DENDRAL giúp các nhà nghiên cứu hóa học hữu cơ xác định các phần tử hữu cơ chưa biết dựa trên phân tích phổ của chúng và các tri thức hóa học. MYCIN là hệ

chuyên gia y học có cơ sở tri thức khoảng 600 luật.

- Đề án máy tính thế hệ thứ 5 FGCS (Fifth Generation Computer Systems) của Nhật Bản do Bộ Ngoại thương và Công nghiệp phát động. FGCS kéo dài trong 10 năm (1982-1992). Đề án FGCS nhằm làm ra các hệ máy tính có khả năng suy diễn và giao tiếp bằng ngôn ngữ tự nhiên trên nền tính toán song song. Mặc dù cuối cùng được đánh giá là thất bại do không đạt được mục tiêu, nhưng đề án FGCS đã kích thích một cuộc thi đua quốc tế trong giai đoạn hồi sinh của TTNT. Đề án này cũng đặt ra và thách thức nhiều vấn đề cho giới nghiên cứu trên toàn thế giới.
- Những năm 90 cho đến nay, các nghiên cứu nhằm vào cài đặt thành phần thông minh trong các hệ thống thông tin, gọi chung là cài đặt TTNT, làm rõ hơn các ngành của khoa học TTNT và tiến hành các nghiên cứu mới, đặc biệt là nghiên cứu về cơ chế suy lý, về các mô hình tương tác. Các nghiên cứu về AI phân tán, mạng nơron nhân tạo, logic mờ, thuật giải di truyền, khai phá dữ liệu, web ngữ nghĩa, tin sinh học, mạng xã hội,...

1.5. Các thành phần trong hệ thống của TTNT

Hệ thống trí tuệ nhân tạo bao gồm hai thành phần cơ bản đó là biểu diễn tri thức và tìm kiếm tri thức trong miền biểu diễn:

TTNT = Tri thức + Suy diễn

Tri thức của bài toán có thể được phân ra làm ba loại cơ bản đó là tri thức mô tả, tri thức thủ tục và tri thức điều khiển.

Để biểu diễn tri thức người ta sử dụng các phương pháp sau đây:

- Phương pháp biểu diễn nhờ luật
- Phương pháp biểu diễn nhờ mạng ngữ nghĩa
- Phương pháp biểu diễn nhờ bộ ba liên hợp OAV
- Phương pháp biểu diễn nhờ Frame
- Phương pháp biểu diễn nhờ logic vị tư

1.6. Phân loại công nghệ TTNT

Loại 1: Công nghệ AI phản ứng (Reactive Machine)

Công nghệ AI phản ứng (Reactive Machine) dùng để phân tích những động thái khả thi – của chính nó và đối thủ và chọn hành động chiến lược nhất.

Ví dụ: Chương trình tự động chơi cờ vua của IBM đã đánh bại kì thủ thế giới Garry Kasparov vào những năm 1990. Công nghệ AI của Deep Blue có thể xác định các nước cờ và dự đoán những bước đi tiếp theo. Nhưng nó không có ký ức và không thể sử dụng những kinh nghiệm trong quá khứ để tiếp tục huấn luyện trong tương lai.

Deep Blue và AlphaGO (chơi cờ vây) của Google được thiết kế cho các mục đích hẹp và không thể dễ dàng áp dụng cho tình huống khác.

Loại 2: Công nghệ AI với bộ nhớ hạn chế

Các ứng dụng AI này được sử dụng những kinh nghiệm trong quá khứ để đưa ra các quyết định trong tương lai.

Một số chức năng ra quyết định này có mặt trong các loại thiết bị không người lái như xe, máy bay drone hoặc tàu ngầm. Kết hợp các cảm biến môi trường xung quanh công nghệ AI này có thể dự đoán được tình huống và đưa ra những bước hành động tối ưu cho thiết bị. Sau đó chúng sẽ được sử dụng để đưa ra hành động trong bước tiếp theo.

Loại 3: Lý thuyết về trí tuệ nhân tạo

Đây là công nghệ AI có thể tự mình suy nghĩ và học hỏi các thứ xung quanh để áp dụng cho chính bản thân nó cho một việc cụ thể. Tuy nhiên công nghệ AI này chưa khả thi trong hiện tại.

Loại 4: Tự nhận thức

Hệ thống AI tự nhận thức về bản thân, có ý thức và hành xử như con người. Các ứng dụng AI này còn có cảm xúc và hiểu được cảm xúc của những người khác. Tuy nhiên, loại công nghệ AI này chưa khả thi.

1.7. Các lĩnh vực nghiên cứu

Có nhiều nội dung nghiên cứu và phát triển của TTNT, từ cách để máy có thể suy diễn logic và nhận thức, cách ra quyết định và giải quyết vấn đề, cách biểu diễn tri con người trong máy, cách lập kế hoạch hành động, hay biết cách tự học để tạo ra tri thức mới, đến dịch tự động các ngôn ngữ, tìm kiếm thông tin trên Internet, robot thông minh. Ta nói về một vài lĩnh vực của TTNT trong những năm qua [2].

- Xử lý ngôn ngữ tự nhiên (natural language processing)

Liệu máy có thể nói được như người?

Đây là bài toán tổng hợp tiếng nói, tức việc làm cho máy biết đọc các văn bản thành tiếng người. Có thể hình dung nếu ta đưa cho máy các luật phát âm tiết, bài toán này sẽ là việc áp dụng các luật này vào các âm tiết trong một từ để tạo ra cách đọc từ này. Đã có nhiều hệ thống tạo ra được giọng đọc tự nhiên của con người hoặc đọc giống giọng một người nào đấy, nhất là cho các ngôn ngữ được nghiên cứu nhiều như tiếng Anh.

Liệu máy có thể nhận biết được tiếng người nói?

Đây là bài toán nhận dạng tiếng nói, tức việc làm cho máy biết chuyển tiếng nói của người từ microphone thành dãy các từ. Đây là bài toán rất khó, vì âm thanh người nói là liên tục và các âm quyện nối vào nhau, vì mỗi người mỗi giọng,... Với tiếng nói chuẩn, các hệ hiện đại cung mới nhận dạng đúng được khoảng 60%-70%.

Liệu máy có hiểu được tiếng nói và văn bản của con người?

Hiểu ngôn ngữ là một đặc trưng tiêu biểu của trí tuệ và việc làm cho máy hiểu được ngôn ngữ là một trong những vấn đề khó nhất của TTNT nói riêng và của CNTT nói chung. Để hiểu nghĩa một câu, máy không chỉ cần biết nghĩa của từng từ, mà trước hết phải biết phân tích được câu này về mặt ngữ pháp. Để làm việc này, máy phải tách câu thành các từ đong lẻ hay cụm từ, nhận viết chúng là các loại từ gì rồi xác định cấu trúc của câu, đoán nghĩa của từng từ và giải nghĩa của câu. Ngôn ngữ thường này trở nên vô cùng khó đối với máy.

- Dịch tự động: Liên quan đến hiểu ngôn ngữ là dịch tự động từ tiếng này sang tiếng khác. Việc dịch này đòi hỏi máy không chỉ phải hiểu đúng nghĩ a câu tiếng Việt mà còn phải tạo ra được câu tiếng Anh tương ứng.

- Tìm kiếm thông tin trên mạng: Đây là lĩnh vực có sự chia sẻ nhiều nhất giữa TTNT và Internet, và ngày càng trở nên hết sức quan trọng. Sẽ sớm đến một ngày, mọi sách bào của con người được số hóa và để lên mạng hay các thư viện số cực lớn. Chẳng hạn để tìm các tài liệu có liên quan đến "trí tuệ nhân tạo và ứng dụng trong khoa học". Với bài toán này có ít nhất hai cách để TTNT đóng góp vào. Một là hệ tìm kiếm các văn bản trong thư viện theo nghĩa này. Hai là hệ tìm kiếm sẽ mô hình các từ "trí tuệ nhân tạo", "khoa học", mỗi mô hình là tập hợp các từ khác kèm theo phân bố xác suất của chúng theo quy luật thông kê. Thay vì tìm kiếm trên mạng hay trong thư viện với hai tập hợp từ khóa, hệ sẽ tìm kiếm với ba tập hợp từ.

- Thị giác máy (computer vision): nghiên cứu về việc thu nhận, xử lý, nhận

dạng thông tin hình ảnh thành biểu biễn mức cao hơn như các đối tượng xung quanh để máy tính có thể hiểu được.

- Lý thuyết tìm kiếm heuristics: bao gồm các phương pháp và các kỹ thuật tìm kiếm, sử dụng các tri thức đặc biệt nảy sinh từ bản thân lĩnh vực của bài toán cần giải để từ đó nhanh chóng đưa ra kết quả mong muốn. Kỹ thuật cơ bản dựa trên các tri thức Heuristics hay được sử dụng trong thực tiễn là tạo các hàm đánh giá.

- Lý thuyết biểu diễn tri thức và kỹ nghệ xử lý tri thức: Logic mệnh đề, logic vị từ, các hệ sản xuất, biểu diễn bằng Frame, mạng ngữ nghĩa.

- Kỹ thuật suy diễn (inference): Quá trình sinh ra kết luận hoặc sự kiện mới từ những sự kiện và thông tin đã có.

- Học máy (machine learning): Làm tăng hiệu quả giải quyết vấn đề trên dữ liệu và kinh nghiệm đã có.

1.8. Ứng dụng

Ứng dụng AI trong ngành vận tải:

- Trí tuệ nhân tạo Al được ứng dụng trên những phương tiện vận tải tự lái, điển hình như là ô tô. Ứng dụng này góp phần mang lại lợi ích kinh tế cao hơn nhiều nhờ khả năng cắt giảm chi phí đặc biệt hạn chế những tai nạn nguy hiểm đến tính mạng con người.
- Vào năm 2016, phát triển xe tự lái thuộc hãng Uber đã vận chuyển thành công 50.000 lon bia Budweisers bằng chiếc xe tự lái trên quãng đường dài 193 km. Theo dự đoán công ty tư vấn công nghệ thông tin Gartner, trong tương lai, những chiếc xe đó có thể kết nối với nhau thông qua Wifi để đưa ra những lộ trình vận tải tốt nhất.

Ứng dụng trong sản xuất

- Trí tuệ nhân tạo còn được ứng dụng để xây dựng những quy trình sản xuất tối ưu hơn. Công nghệ AI còn có khả năng phân tích cao, làm cơ sở định hướng cho khả năng ra quyết định trong sản xuất.

Ứng dụng trong y tế

- Ứng dụng tiêu biểu của trí tuệ nhân tạo trong lĩnh vực y tế đó chính là máy bay thiết bị bay không cần người lái được sử dụng trong những trường hợp cứu hộ khẩn cấp. Thiết bị bay không người lái có thể đạt được tốc độ nhanh hơn xe chuyên dụng đến 40% và cực kì thích hợp để sửdụng Ở những nơi có đia hình hiểm trở.

Ứng dụng trong giáo dục

- Sự ra đời của trí tuệ nhân tạo đã tạo ra những thay đổi lớn trong lĩnh vực giáo dục. Các hoạt động giáo dục như là chấm điểm hay dạy kèm cho học sinh có thể được tự động hóa hoàn toàn nhờ công nghệ AI. Nhiều trò chơi, phần mềm giáo dục được ra đời đáp ứng nhu cầu cụ thể của từng học sinh, giúp cho học sinh cải thiện rất nhiều về tình hình học tập theo tốc độ riêng của mình.
- Trí tuệ nhân tạo còn có thể chỉ ra được những vấn đề mà các khóa học cần phải cải thiện. Chẳng hạn khi nhiều học sinh được phát hiện việc gửi đáp án sai cho bài tập, hệ thống sẽ thực hiện thông báo cho giáo viên đồng thời gửi thông điệp đến cho học sinh để chỉnh sửa đáp án phù hợp. Công nghệ AI còn có khả năng giúp theo dõi sự tiến bộ của các học sinh và thông báo đến giáo viên khi phát hiện vấn đề đối với kết quả học tập của học sinh.
- Hơn thế nữa, sinh viên có thể học hỏi được bất cứ nơi nào trên thế giới thông qua việc sử dụng phần mềm có hỗ trợ AI. Công nghệ AI còn cung cấp dữ liệu nhằm giúp sinh viên lựa chọn được những khóa học tốt nhất, phù hợp nhất cho mình.

Ứng dụng trong truyền thông

- Đối với lĩnh vực truyền thông thì sự phát triển của trí tuệ nhân tạo đã góp phần làm thay đổi cách thức tiếp cận với khách hàng mục tiêu. Nhờ ưu điểm của công nghệ AI, các công ty đã có thể cung cấp quảng cáo vào đúng thời điểm, cho đúng khách hàng tiềm năng, dựa trên việc phân tích những đặc điểm về nhân khẩu học, thói quen hoạt động trực tuyến những nội dung khách hàng thường xem trên quảng cáo.

Ứng dụng trong ngành dịch vụ

- Công nghệ AI còn giúp ngành dịch vụ hoạt động một cách tối ưu hơn từ đó góp phần mang đến những trải nghiệm mới mẻ hơn, tốt hơn cho khách hàng. Thông qua thu thập và phân tích dữ liệu, công nghệ AI còn nắm bắt thông tin về hành vi sử dụng những dịch vụ của khách hàng, từ đó đã mang lại những giải pháp phù hợp nhất với nhu cầu của từng khách hàng.

❖ TÓM TẮT CHƯƠNG 1

- Như vậy TTNT là là một lĩnh vực của khoa học và công nghệ nhằm làm cho máy có những khả năng của trí tuệ con người, tiêu biểu như biết suy nghĩ và lập luận để giải quyết vấn đề, biết giao tiếp do hiểu ngôn ngữ tự nhiên và tiếng nói, biết học và tự thích nghi,...
- Sự phát triển của TTNT đã tạo ra một bước nhảy vọt về chất trong kỹ thuật và kỹ nghệ xử lý thông tin. Trí tuệ nhân tạo chính là cơ sở của công nghệ xử lý thông tin mới.
- TTNT có vai trò rất quan trọng trong việc đưa ra lời giải cho các bài toán có không gian tìm kiếm lớn.
- TTNT gồm hai thành phần cơ bản: tri thức và suy diễn AI = Tri thức + Suy diễn
- TTNT được ứng dụng trong hầu hết các lĩnh vực: Kinh tế, địa chất, y học, hóa học,...

CHƯƠNG 2: TỔNG QUAN VỀ SCIKIT_LEARN

2.1. Giới thiệu về SCIKIT_LEARN

- Scikit-learn (Sklearn) là thư viện mạnh mẽ nhất dành cho các thuật toán học máy được viết trên ngôn ngữ Python. Thư viện cung cấp một tập các công cụ xử lý các bài toán machine learning và statistical modeling gồm: classification, regression, clustering, và dimensionality reduction.
- Thư viện được cấp phép bản quyền chuẩn FreeBSD và chạy được trên nhiều nền tảng Linux. Scikit-learn được sử dụng như một tài liệu để học tập.
- Để cài đặt scikit-learn trước tiên phải cài thư viện SciPy (Scientific Python). Những thành phần gồm:
  - Numpy: Gói thư viện xử lý dãy số và ma trận nhiều chiều
  - SciPy: Gói các hàm tính toán logic khoa học
  - Matplotlib: Biểu diễn dữ liệu dưới dạng đồ thị 2 chiều, 3 chiều
  - IPython: Notebook dùng để tương tác trực quan với Python
  - SymPy: Gói thư viện các kí tự toán học
  - Pandas: Xử lý, phân tích dữ liệu dưới dạng bảng
- Những thư viện mở rộng của SciPy thường được đặt tên dạng SciKits. Như thư viện này là gói các lớp, hàm sử dụng trong thuật toán học máy thì được đặt tên là scikit-learn.
- Scikit-learn hỗ trợ mạnh mẽ trong việc xây dựng các sản phẩm. Nghĩa là thư viện này tập trung sâu trong việc xây dựng các yếu tố: dễ sử dụng, dễ code, dễ tham khảo, dễ làm việc, hiệu quả cao.
- Mặc dù được viết cho Python nhưng thực ra các thư viện nền tảng của scikit-learn lại được viết dưới các thư viện của C để tăng hiệu suất làm việc. Ví dụ như: Numpy(Tính toán ma trận), LAPACK, LibSVM và Cython.
- Scikit-learn được biết đến với tính dễ phát triển tương đối nhờ các API được thiết kế nhất quán và hiệu quả, tài liệu mở rộng cho hầu hết các thuật toán và nhiều hướng dẫn online.

Hình 2.1: scikit learn

2.2. Lịch sử hình thành và phát triển

- Dự án Scikit-learn bắt đầu vào năm 2007 như một phần của chương trình Google Summer of Code, do nhà khoa học dữ liệu người Pháp David Cournapeau khởi xướng.
- Tên gọi "Scikit" xuất phát từ "SciPy Toolkit", ám chỉ đây là một phần mở rộng của SciPy.
- Năm 2010, các nhà phát triển từ Viện Nghiên cứu Khoa học Máy tính và Tự động hóa Pháp (INRIA) đã tiếp quản và phát hành phiên bản công khai đầu tiên vào ngày 1 tháng 2 năm 2010.
- Kể từ đó, Scikit-learn đã trở thành một trong những thư viện học máy phổ biến nhất trong cộng đồng Python.

David Cournapeau

2.3. Cách hoạt động của scikit learn

- Scikit-learn được viết chủ yếu bằng Python và sử dụng NumPy cho đại số tuyến tính hiệu suất cao cũng như cho các phép tính array. Một số thuật toán Scikit-learn cốt lõi được viết bằng Cython để tăng hiệu suất tổng thể.
- Là một thư viện cấp cao bao gồm một số triển khai các thuật toán học máy khác nhau, Scikit-learn cho phép người dùng xây dựng, đào tạo và đánh giá mô hình bằng một vài dòng mã.
- Scikit-learn cung cấp một bộ API cấp cao thống nhất để xây dựng quy trình hoặc quy trình ML.

- Người dùng sử dụng ML Scikit-learn Pipeline để truyền dữ liệu qua các transformers nhằm trích xuất các đặc điểm và công cụ ước tính để tạo ra mô hình, sau đó đánh giá các dự đoán để đo lường độ chính xác của mô hình.

- 

Trợ lí ảo

- Transformer: Đây là thuật toán biến đổi hoặc nhập vào dữ liệu để xử lý.
- Estimator: Đây là thuật toán học máy đào tạo hoặc điều chỉnh dữ liệu để xây dựng mô hình, mô hình này có thể được sử dụng để dự đoán.
- Pipeline: Một đường dẫn kết nối các Transformer và Estimator lại với nhau để chỉ định quy trình làm việc ML.

2.4. Các thuật toán chính của scikit learn

Sau đâu là một số nhóm thuật toán được xây dựng bởi thư viện scikit-learn:

❖ Clustering

- Phân cụm là kỹ thuật học máy không giám sát, trong đó các mẫu dữ liệu được nhóm lại thành các cụm (clusters) sao cho các mẫu trong cùng một cụm có tính chất tương tự nhau. Một trong những thuật toán phổ biến cho phân cụm là **KMeans**.
- Thuật toán này phân chia dữ liệu thành k nhóm sao cho các điểm trong cùng một nhóm có độ tương đồng cao nhất.
- Ví dụ, trong một bài toán phân cụm khách hàng, chúng ta có thể sử dụng KMeans để nhóm khách hàng thành các cụm dựa trên các đặc điểm như độ tuổi, thu nhập, thói quen tiêu dùng, v.v. Việc phân cụm khách hàng có thể giúp các doanh nghiệp tối ưu hóa chiến lược tiếp thị và chăm sóc khách hàng.

❖ Cross Validation:

- Kiểm thử chéo (cross-validation) là một kỹ thuật để đánh giá hiệu suất của mô

hình học máy. Phương pháp này chia bộ dữ liệu thành nhiều phần (folds), sau đó huấn luyện mô hình trên một phần của dữ liệu và kiểm tra trên phần còn lại.
- Kỹ thuật này giúp giảm thiểu rủi ro của việc mô hình học trên một tập dữ liệu không đại diện.
- **KFold** và **StratifiedKFold** là các thuật toán phổ biến trong scikit-learn để thực hiện kiểm thử chéo.

❖ Datasets:
- Scikit-learn cung cấp các bộ dữ liệu tích hợp sẵn, cho phép người dùng dễ dàng tải và sử dụng để huấn luyện mô hình. Những bộ dữ liệu này đã được chuẩn hóa và thường được sử dụng trong các bài học và thử nghiệm. Ví dụ, bộ dữ liệu Iris là một bộ dữ liệu nổi tiếng trong học máy với 150 mẫu dữ liệu về các loài hoa Iris và ba nhãn khác nhau.

❖ Dimensionality Reduction:
- Giảm chiều dữ liệu là một kỹ thuật giúp giảm số lượng biến trong dữ liệu mà không làm mất thông tin quan trọng. Một trong những thuật toán phổ biến là PCA (Principal Component Analysis), giúp giảm số chiều của dữ liệu bằng cách giữ lại các thành phần chính. PCA có thể giúp đơn giản hóa mô hình, giảm thời gian huấn luyện và tăng cường hiệu suất.

❖ Ensemble methods:
- Phương pháp tập hợp là kỹ thuật kết hợp nhiều mô hình để cải thiện hiệu suất dự đoán. Một ví dụ điển hình là **Random Forest**, kết hợp nhiều cây quyết định (decision trees) để tạo ra một mô hình tổng hợp có độ chính xác cao hơn. Phương pháp này đặc biệt hữu ích khi các mô hình đơn lẻ dễ bị overfitting (quá khớp).

❖ Feature extraction:
- Trích xuất đặc trưng là quá trình chuyển đổi dữ liệu thô thành các thuộc tính quan trọng giúp mô hình học máy học hiệu quả hơn. Đối với dữ liệu hình ảnh, trích xuất đặc trưng có thể bao gồm các thuộc tính như màu sắc, hình dạng hoặc kết cấu. Đối với dữ liệu ngôn ngữ, trích xuất đặc trưng có thể bao gồm các từ khóa hoặc các đặc trưng ngữ nghĩa.

❖ Feature selection:
- Trích chọn đặc trưng là quá trình lựa chọn các đặc trưng quan trọng nhất để huấn luyện mô hình. Điều này giúp giảm bớt nhiễu và làm cho mô hình trở nên đơn giản và hiệu quả hơn. Scikit-learn cung cấp các công cụ như **SelectKBest** để chọn ra các đặc trưng có ảnh hưởng lớn nhất đến dự đoán của mô hình.

❖ Parameter Tuning:
- Việc tinh chỉnh tham số (hyperparameter tuning) là bước quan trọng để tối ưu hóa hiệu suất của mô hình. Các phương pháp như **GridSearchCV** và **RandomizedSearchCV** giúp tìm kiếm và tối ưu hóa các tham số của mô hình, từ đó cải thiện kết quả.

❖ Manifold Learning:

- Manifold learning là một nhóm các kỹ thuật học máy dùng để phân tích và giảm chiều dữ liệu đa chiều phức tạp. Các thuật toán như **t-SNE (t-Distributed Stochastic Neighbor Embedding)** và **Isomap** là những ví dụ điển hình. Các thuật toán này giúp khám phá các mối quan hệ giữa các điểm dữ liệu trong không gian chiều cao mà không làm mất đi cấu trúc của dữ liệu.

❖ Supervised Models:

- Các mô hình học giám sát như **Linear Regression**, **Logistic Regression**, **Decision Trees**, **Support Vector Machines (SVM)**, và **Naive Bayes** là các thuật toán phổ biến để phân loại hoặc dự đoán giá trị liên tục. Những mô hình này học từ dữ liệu có nhãn để đưa ra dự đoán cho dữ liệu chưa thấy.

2.5. Ứng dụng của Scikit-learn

Scikit-learn là một thư viện mã nguồn mở mạnh mẽ trong Python, hỗ trợ nhiều thuật toán học máy cho các nhiệm vụ như phân loại, hồi quy, phân cụm và giảm chiều dữ liệu. Dưới đây là một số ứng dụng phổ biến của Scikit-learn:

2.5.1. Phân loại (Classification):

Phân loại là quá trình xác định nhãn cho các mẫu dữ liệu dựa trên các đặc trưng của chúng. Scikit-learn cung cấp nhiều thuật toán phân loại như:

- **Hồi quy logistic (Logistic Regression):** Dự đoán xác suất của các lớp nhãn.
- **Máy vector hỗ trợ (Support Vector Machines - SVM):** Tìm kiếm siêu phẳng phân chia tối ưu giữa các lớp.
- **Cây quyết định (Decision Trees):** Xây dựng mô hình phân loại dựa trên các quy tắc phân chia dữ liệu.
- **Rừng ngẫu nhiên (Random Forests):** Kết hợp nhiều cây quyết định để cải thiện độ chính xác.
- **K láng giềng gần nhất (K-Nearest Neighbors - KNN):** Phân loại dựa trên các láng giềng gần nhất trong không gian đặc trưng.

2.5.2. Hồi quy (Regression):

Hồi quy được sử dụng để dự đoán giá trị liên tục dựa trên các đặc trưng đầu vào. Scikit-learn hỗ trợ các thuật toán như:

- **Hồi quy tuyến tính (Linear Regression):** Mô hình hóa mối quan hệ tuyến tính giữa biến độc lập và phụ thuộc.
- **Hồi quy Ridge và Lasso:** Thêm điều kiện ràng buộc để giảm thiểu hiện tượng quá khớp (overfitting).

- **Hồi quy cây quyết định (Decision Tree Regression):** Sử dụng cây quyết định để dự đoán giá trị liên tục.

2.5.3. Phân cụm (Clustering):

Phân cụm là quá trình nhóm các mẫu dữ liệu có đặc trưng tương tự nhau. Scikit-learn cung cấp các thuật toán như:

- **K-means:** Phân chia dữ liệu thành K cụm dựa trên khoảng cách Euclidean.
- **DBSCAN:** Phân cụm dựa trên mật độ, có khả năng phát hiện các cụm có hình dạng bất kỳ.
- **Phân cụm phân cấp (Hierarchical Clustering):** Xây dựng cây phân cấp để phân cụm dữ liệu.

2.5.4. Giảm chiều dữ liệu (Dimensionality Reduction):

Giảm chiều dữ liệu giúp giảm số lượng biến trong tập dữ liệu, từ đó giảm thiểu độ phức tạp và cải thiện hiệu suất mô hình. Scikit-learn hỗ trợ:

- **Phân tích thành phần chính (Principal Component Analysis - PCA):** Tìm kiếm các thành phần chính để biểu diễn dữ liệu.
- **Phân tích tuyến tính phân biệt (Linear Discriminant Analysis - LDA):** Tìm kiếm các thành phần phân biệt giữa các lớp.

2.5.5. Phát hiện bất thường (Anomaly Detection):

Phát hiện bất thường giúp xác định các mẫu dữ liệu không phù hợp hoặc hiếm gặp. Scikit-learn cung cấp các thuật toán như:

- **Máy vector hỗ trợ một lớp (One-Class SVM):** Phát hiện các điểm dữ liệu khác biệt so với phần còn lại.
- **Rừng cách ly (Isolation Forest):** Phát hiện bất thường bằng cách cô lập các điểm dữ liệu.

2.5.6. Tối ưu hóa mô hình và lựa chọn tham số (Model Selection and Hyperparameter Tuning):

Scikit-learn cung cấp các công cụ để lựa chọn mô hình và tối ưu hóa tham số như:

- **Tìm kiếm lưới (Grid Search):** Tìm kiếm qua không gian tham số để tìm tham số tối ưu.

- **Đánh giá chéo (Cross-Validation):** Đánh giá hiệu suất mô hình bằng cách chia dữ liệu thành nhiều tập con.

Nhờ vào các công cụ và thuật toán đa dạng, Scikit-learn trở thành một thư viện quan trọng trong việc phát triển và triển khai các mô hình học máy hiệu quả.

2.6. Kết luận

Scikit-learn là một thư viện mã nguồn mở mạnh mẽ dành cho Python, cung cấp một bộ công cụ toàn diện cho học máy, bao gồm các thuật toán phân loại, hồi quy, phân cụm và giảm chiều dữ liệu. Với giao diện thân thiện và tài liệu phong phú, Scikit-learn đã trở thành lựa chọn hàng đầu cho các nhà khoa học dữ liệu và lập trình viên khi phát triển và triển khai các mô hình học máy.

Ưu điểm của Scikit-learn:

- **Dễ sử dụng và tích hợp:** Scikit-learn cung cấp API nhất quán và dễ hiểu, cho phép người dùng nhanh chóng xây dựng và thử nghiệm các mô hình học máy. Thư viện này tích hợp tốt với các thư viện Python khác như NumPy, SciPy và Matplotlib, tạo điều kiện thuận lợi cho việc xử lý và trực quan hóa dữ liệu.
- **Đa dạng thuật toán:** Scikit-learn hỗ trợ nhiều thuật toán học máy phổ biến, từ các phương pháp đơn giản như hồi quy tuyến tính đến các mô hình phức tạp hơn như rừng ngẫu nhiên và máy vector hỗ trợ (SVM). Điều này giúp người dùng dễ dàng lựa chọn và áp dụng phương pháp phù hợp với bài toán cụ thể.
- **Cộng đồng hỗ trợ mạnh mẽ:** Với cộng đồng người dùng và phát triển rộng rãi, Scikit-learn liên tục được cập nhật và cải tiến. Người dùng có thể dễ dàng tìm kiếm tài liệu, hướng dẫn và giải đáp thắc mắc từ cộng đồng.

Nhược điểm cần lưu ý:

- **Hạn chế với học sâu:** Mặc dù Scikit-learn rất mạnh mẽ trong các tác vụ học máy truyền thống, nhưng nó không hỗ trợ các mô hình học sâu (deep learning) phức tạp như mạng nơ-ron sâu. Đối với các bài toán yêu cầu học sâu, người dùng có thể cần xem xét các thư viện khác như TensorFlow hoặc PyTorch.
- **Hiệu suất với dữ liệu lớn:** Scikit-learn chủ yếu hoạt động trong bộ nhớ (in-memory), do đó có thể gặp khó khăn khi xử lý các tập dữ liệu rất lớn. Trong trường hợp này, việc sử dụng các công cụ phân tán hoặc thư viện hỗ trợ xử lý dữ liệu lớn như Dask có thể là giải pháp thay thế.

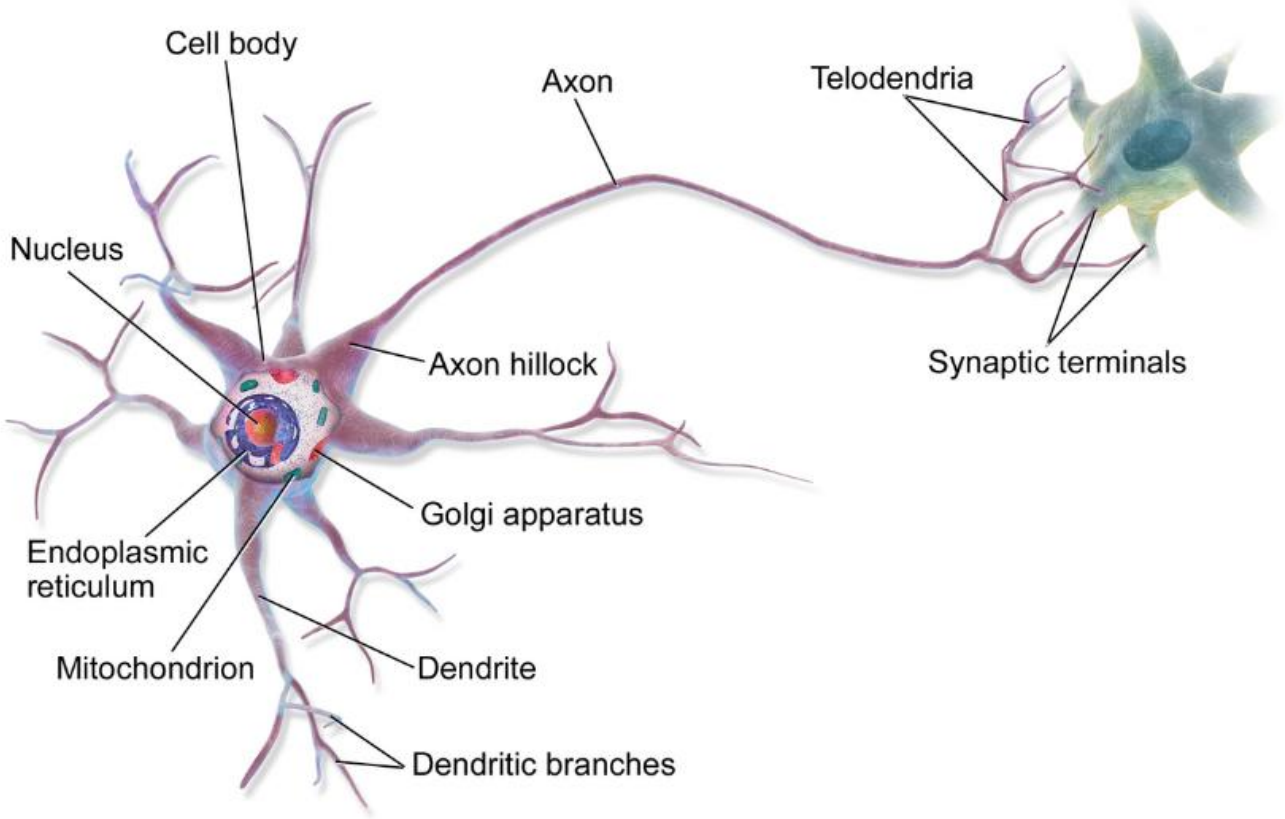
Kết luận:

Scikit-learn là một công cụ mạnh mẽ và linh hoạt cho học máy trong Python, phù hợp cho cả người mới bắt đầu và chuyên gia. Với các tính năng đa dạng và cộng đồng hỗ trợ mạnh mẽ, Scikit-learn tiếp tục là lựa chọn hàng đầu cho việc phát triển và triển khai các mô hình học máy hiệu quả.

CHƯƠNG 3: DỰ ĐOÁN UNG THƯ PHỔI

3.1. Sơ lược về ung thư phổi

Ung thư phổi xảy ra khi có sự xuất hiện của một khối u ác tính do sự tăng sinh tế bảo không thể kiểm soát trong các mô phổi. Nếu không được điều trị, nó có thể lây lan (di căn) ra ngoài phổi đến các mô hoặc bộ phận khác của cơ thể.



Tế bào ung thư

Các triệu chứng của ung thư phổi không tế bào nhỏ và ung thư phổi tế bào nhỏ về cơ bản là giống nhau, bao gồm:

- Ho kéo dài
- Ho có đờm hoặc ra máu
- Đau ngực, đặc biệt khi hít thở sâu, cười hoặc ho
- Khản tiếng
- Hụt hơi
- Khó thở
- Suy nhược cơ thể và mệt mỏi
- Chán ăn
- Sụt cân

Nguyên nhân:

- 90% do hút thuốc
- • Tiếp xúc với radon - một loại khí phóng xạ không mùi, không màu được tạo
- ra từ sự phân rã tự nhiên của uranium
- • Hít phải các chất độc hại khác(thạch tín, cadimi, crom...)

Chẩn đoán:

- Xét nghiệm hình ảnh: Một khối u bất thường có thể được nhìn thấy trên
- phim chụp Xquang, CT và PET/CT.
- Xét nghiệm tế bào đờm: Bạn sẽ ho để tạo ra đờm, sau đó bác sĩ sẽ lấy đờm để xét nghiệm thông qua kính hiển vi giúp xác định xem có tế bào ung thư hay không.
- Sinh thiết: giúp xác định xem các tế bào khối u có phải là ung thư hay không. Có thể lấy mẫu mô bằng cách: Nội soi phế quản, nội soi trung thất hoặc sử dụng kim.

Người ung thư phổi

3.2. Phương pháp chuẩn đoán

3.2.1. Đầu vào

- Với bài toán nhận biết bệnh nhân có bị bệnh ung thư phổi hay không thì đầu vào bài toán sẽ là 1 bức ảnh chụp x-quang vùng phổi của bệnh nhân. Bức ảnh này sẽ được đưa vào bên trong chương trình và chương trình sẽ xử lý dữ liệu. Bài toán sử dụng thuật toán CNN dùng để trích xuất ảnh đặc trưng, qua đó nhóm em sẽ có 2 loại dữ liệu để cho thuật toán training là hình ảnh chụp x-quang bệnh nhân bị ung thư phổi và hình ảnh chụp x-quang bênh nhân phổi bình thường.
- Các ảnh được đưa vào chương trình sẽ phải qua một bước tiền xử lý để đưa về phù hợp với mô hình đã huấn luyện.

3.2.2. Đầu ra

- Đầu ra sẽ là kết quả cho biết tấm phim chụp x-quang của người nào đó có bị mắc bệnh ung thư phổi hay không.

3.3. Cơ sở dữ liệu

3.3.1. Chương trình biểu mẫu

- Input: Là một Form các câu hỏi về chỉ số và thói quen và các số liệu liên quan đến ung thư phổi.
- Output : Kết quả nhận được chỉ đánh giá mức độ , khả năng mắc bệnh.
- Chương trình gồm chức năng nhận ảnh từ bên ngoài để nhận dạng.

3.3.2. Chương trình trợ lí ảo

- Input: Void
- Output: Đánh giá và lời khuyên của trợ lí ảo

3.4. Các thuật toán

Các thuật toán được sử dụng trong đoạn code này bao gồm:

3.4.1. Random Forest Classifier (Rừng ngẫu nhiên)

3.4.1.1. Giới thiệu về thuật toán

- Định nghĩa: Rừng Ngẫu Nhiên (Random Forest) là một thuật toán học máy thuộc nhóm học có giám sát (supervised learning)**,** được sử dụng cho cả bài toán phân loại (classification) và hồi quy (regression)**.** Nó được phát triển dựa trên phương pháp Bagging (Bootstrap Aggregating) và sử dụng nhiều cây quyết định **(**Decision Trees**)** để cải thiện độ chính xác và giảm overfitting.
- Nguyên lý hoạt động:

Thuật toán này hoạt động theo các bước sau:

+ Tạo nhiều tập dữ liệu con: Từ tập dữ liệu gốc, thuật toán chọn ngẫu nhiên nhiều tập con bằng phương pháp Bootstrap Sampling (lấy mẫu có hoàn lại).

+ Huấn luyện nhiều cây quyết định

- Với mỗi tập dữ liệu con, thuật toán huấn luyện một cây quyết định (Decision Tree) độc lập.
- Khi chọn mỗi nút trong cây, thuật toán không xem xét toàn bộ thuộc tính mà chỉ lấy một phần nhỏ được chọn ngẫu nhiên, giúp tạo ra sự đa dạng giữa các cây.

+ Dự đoán bằng mô hình Rừng Ngẫu Nhiên

- Đối với bài toán phân loại: Lấy kết quả dự đoán của từng cây và chọn nhãn xuất hiện nhiều nhất (Voting).
- Đối với bài toán hồi quy: Tính giá trị trung bình từ tất cả các cây (Averaging).

- Ưu điểm

+ Giảm Overfitting: Vì sử dụng nhiều cây quyết định nên mô hình có độ ổn định cao hơn so với một cây đơn lẻ.

+ Làm việc tốt với dữ liệu lớn: Random Forest có thể xử lý dữ liệu có nhiều thuộc tính và dạng dữ liệu khác nhau.
+ Tự động xử lý giá trị thiếu: Có thể ước lượng giá trị thiếu mà không cần loại bỏ dữ liệu.
+ Tính linh hoạt cao: Sử dụng được cho cả phân loại và hồi quy.

- Nhược điểm

+ Chi phí tính toán cao: Khi số lượng cây lớn, mô hình có thể chậm và tốn nhiều tài nguyên.
+ Khó giải thích: Do là tập hợp của nhiều cây, Random Forest không dễ diễn giải như một cây quyết định đơn lẻ.

3.4.1.2. Ứng dụng trong bài

Được sử dụng để xây dựng mô hình dự đoán nguy cơ mắc ung thư phổi dựa trên các đặc trưng như tuổi, hút thuốc, ho mãn tính, khó thở, đau ngực, tiền sử gia đình.
Random Forest là một tập hợp của nhiều cây quyết định (Decision Trees), giúp cải thiện độ chính xác và giảm overfitting.
Cách hoạt động:
Chia dữ liệu thành tập train-test (70/30).
Huấn luyện mô hình với 100 cây quyết định (n_estimators=100).
Dự đoán mức độ ung thư phổi dựa trên input từ người dùng.
Mức độ ung thư phổi được chia thành 4 nhóm (0: Không nguy cơ, 1: Nguy kịch, 2: Trung bình, 3: Thấp).

3.4.2. Xử lý hình ảnh với OpenCV để phân tích ảnh phổi

3.4.2.1. Giới thiệu

OpenCV (Open Source Computer Vision Library) là một thư viện mã nguồn mở mạnh mẽ dành cho xử lý ảnh (image processing) và thị giác máy tính (computer vision). Được phát triển bởi Intel**,** OpenCV hiện là một trong những thư viện phổ biến nhất cho các ứng dụng liên quan đến hình ảnh và video.

OpenCV hỗ trợ nhiều tính năng quan trọng như:
- Xử lý ảnh cơ bản: Chuyển đổi màu sắc, làm mờ, lọc ảnh, phát hiện cạnh,...
- Nhận diện đối tượng: Phát hiện khuôn mặt, vật thể, biển số xe,...
- Thị giác máy tính nâng cao: Phát hiện và theo dõi chuyển động, phân đoạn ảnh, ghép ảnh panorama,...
- Học máy & AI: Hỗ trợ nhận diện khuôn mặt, OCR (nhận diện ký tự quang học), mạng nơ-ron sâu (Deep Learning)...
- Xử lý video: Đọc, ghi video, theo dõi vật thể trong video,...

3.4.2.2. Ứng dụng trong bài

Khi người dùng tải lên ảnh X-quang phổi, OpenCV được sử dụng để:
Chuyển đổi ảnh sang grayscale (ảnh đen trắng).
Đếm số lượng pixel sáng và pixel tối.
Tính tỷ lệ pixel sáng/pixel tối để xác định nguy cơ mắc ung thư phổi.
Quy tắc phân loại nguy cơ ung thư phổi dựa trên tỷ lệ pixel sáng:

> 0.175 → Nguy cơ cao
0.15 - 0.175 → Có khả năng bệnh về phổi
< 0.15 → Không có khả năng bệnh

3.4.3. Xử lý ngôn ngữ tự nhiên (NLP) để nhận diện câu hỏi

3.4.3.1. Giới thiệu

- Định nghĩa: Keyword Matching (khớp từ khóa) là một kỹ thuật trong xử lý ngôn ngữ tự nhiên (NLP) và tìm kiếm thông tin (Information Retrieval), dùng để so sánh văn bản dựa trên sự xuất hiện của các từ khóa cụ thể. Nó thường được sử dụng trong tìm kiếm, chatbot, phân loại văn bản và lọc nội dung**.**
- Cách hoạt động**:**
+ Exact Match (Khớp chính xác): So sánh từ khóa với văn bản mà không có bất kỳ thay đổi nào.
+ Partial Match (Khớp một phần): Kiểm tra xem từ khóa có xuất hiện trong văn bản hay không, nhưng không cần chính xác hoàn toàn.
+ Fuzzy Matching (Khớp mờ): Dùng thuật toán như Levenshtein Distance để tìm các từ gần giống, giúp nhận diện lỗi chính tả hoặc các biến thể của từ.
- Ứng dụng:

+ Tìm kiếm văn bản (Search Engine)
+ Chatbot phản hồi theo từ khóa
+ Phân loại email (spam/non-spam)
+ Lọc nội dung trên mạng xã hội
+ Phân tích cảm xúc trong văn bản

3.4.3.2. Ứng dụng trong bài

Đoạn code trong /process sử dụng phương pháp so khớp từ khóa (Keyword Matching). Khi người dùng nhập văn bản, chatbot tìm kiếm từ khóa trong câu và trả về câu trả lời tương ứng.

Tổng kết:
Random Forest → Dự đoán ung thư phổi từ dữ liệu người dùng.
OpenCV → Xử lý ảnh X-quang để đánh giá nguy cơ ung thư phổi.
Keyword Matching → Chatbot nhận diện câu hỏi và phản hồi.

Kết luận

Trong đề tài "Dự báo ung thư phổi bằng Python", nhóm đã áp dụng các kiến thức đã học về Trí tuệ nhân tạo để xây dựng một hệ thống dự đoán nguy cơ mắc ung thư phổi dựa trên dữ liệu y tế. Trọng tâm của đề tài là việc triển khai thuật toán **Rừng ngẫu nhiên (Random Forest)** – một mô hình học máy mạnh mẽ và hiệu quả trong các bài toán phân loại.

Thông qua quá trình tiền xử lý dữ liệu, phân tích đặc trưng và huấn luyện mô hình, nhóm nhận thấy rằng thuật toán Random Forest có khả năng xử lý tốt các tập dữ liệu có nhiều đặc trưng, chống quá khớp (overfitting) và cho độ chính xác cao trong dự đoán. Kết quả thực nghiệm cho thấy mô hình đạt được độ chính xác ổn định và có tiềm năng ứng dụng trong hỗ trợ chẩn đoán ung thư phổi.

Mặc dù vậy, hiệu quả mô hình vẫn phụ thuộc nhiều vào chất lượng dữ liệu đầu vào. Trong các bước tiếp theo, nhóm có thể mở rộng quy mô dữ liệu, tối ưu tham số mô hình và thử nghiệm thêm các thuật toán khác như XGBoost hoặc mạng nơ-ron để cải thiện độ chính xác.

Đề tài không chỉ giúp nhóm hiểu rõ hơn về ứng dụng của thuật toán Random Forest mà còn củng cố kỹ năng lập trình Python, xử lý dữ liệu và triển khai các mô hình trí tuệ nhân tạo trong thực tế. Đây là bước khởi đầu quan trọng để áp dụng AI vào các vấn đề trong lĩnh vực chăm sóc sức khỏe và y học.