**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC ĐẠI NAM**

Logo, company name

Description automatically generated

**BÀI TẬP LỚN**

**TÊN HỌC PHẦN: TRÍ TUỆ NHÂN TẠO**

**ĐỀ TÀI: PHÂN TÍCH LOGIC AI TRONG “THUẬT TOÁN LẬP LỊCH THỜI KHÓA BIỂU”**

**Giảng viên hướng dẫn: Trần Đăng Công**

**Sinh viên thực hiện: Nguyễn Trung Anh**

**Nguyễn Doãn Bình An**

**Đặng Quốc Khánh**

**Hà Nội, năm 2025**

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC ĐẠI NAM**

Logo, company name

Description automatically generated

**BÀI TẬP LỚN**

**TÊN HỌC PHẦN: TRÍ TUỆ NHÂN TẠO**

**ĐỀ TÀI: XÂY DỰNG CHƯƠNG TRÌNH TẠO LỊCH HỌC TỰ ĐỘNG SỬ DỤNG THUẬT TOÁN DI TRUYỀN**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Mã Sinh Viên** | **Họ và Tên** | **Ngày Sinh** | **Điểm** | |
| **Bằng Số** | **Bằng Chữ** |
| **1** | **1771020051** | **Nguyễn Trung Anh** | **30/03/2005** |  |  |
| **2** | **1771020382** | **Đặng Quốc Khánh** | **29/09/2005** |  |  |
| **3** | **1771020004** | **Nguyễn Doãn Binh An** | **25/04/2005** |  |  |

**CÁN BỘ CHẤM THI 1 CÁN BỘ CHẤM THI 2**

**Hà Nội, năm 20225**

**LỜI NÓI ĐẦU**

- Trước khi bắt đầu em muốn gửi lời chúc tới các giảng viên đang đọc báo cáo của nhóm em. Đồng thời em muốn gửi lời cảm ơn đến thầy Trần Đăng Công đã dạy chúng em môn học Trí tuệ nhân tạo (AI) hết sức ý nghĩa và thú vị. Sau đây đề tài "Xây Dựng Chương Trình Tạo Lịch Học Tự Động Sử Dụng Thuật Toán Di Truyền" nhằm mục đích giải quyết bài toán lập lịch học cho các cơ sở giáo dục. Việc sắp xếp lịch học hợp lý đòi hỏi phải cân nhắc nhiều yếu tố như số lượng lớp học, giáo viên, phòng học, và các yêu cầu đặc thù khác. Bài toán này thuộc nhóm bài toán NP-hard, nghĩa là không thể tìm ra lời giải tối ưu bằng các phương pháp truyền thống trong thời gian hợp lý. Vì vậy, việc áp dụng thuật toán di truyền để tìm kiếm lời giải tối ưu hoặc gần tối ưu là một lựa chọn phù hợp.

- Trong khuôn khổ đề tài này, chương trình sẽ được xây dựng để tự động tạo ra các lịch học thỏa mãn các ràng buộc về thời gian, không gian và tài nguyên. Thuật toán di truyền sẽ được sử dụng để tạo ra các thế hệ lịch học mới, chọn lọc các cá thể tốt nhất, lai ghép và đột biến để cải thiện chất lượng của lời giải qua từng vòng lặp.

**MỤC LỤC**

[**CHƯƠNG 1. TỔNG QUAN VỀ TRÍ TUỆ NHÂN TẠO 8**](#_Toc193477624)

[**1.1. Giới thiệu về trí tuệ nhân tạo 8**](#_Toc193477625)

[***1.1.1. Khái niệm* 8**](#_Toc193477626)

[***1.1.2. Lịch sử hình thành* 8**](#_Toc193477627)

[***1.1.3. Các cột mốc quan trọng* 9**](#_Toc193477628)

[***1.1.4. Phân loại trí tuệ nhân tạo* 11**](#_Toc193477629)

[**1.2. Ưu và nhược điểm của AI. 12**](#_Toc193477630)

[**1.3. Ứng dụng của AI 13**](#_Toc193477631)

[**1.4. Công cụ và nền tảng AI phổ biến 17**](#_Toc193477632)

[***1.4.1. Thư Viện và Framework AI Phổ Biến* 17**](#_Toc193477633)

[***1.4.2. Nền tảng AI đám mây* 21**](#_Toc193477634)

[**1.5. Thách thức và hạn chế của AI 26**](#_Toc193477635)

[***1.5.1. Thách thức* 26**](#_Toc193477636)

[***1.5.2. Hạn Chế của AI* 27**](#_Toc193477637)

[**1.6. Xu hướng phát triển 27**](#_Toc193477638)

[***1.6.1. AI Tổng quát (AGI - Artificial General Intelligence)* 27**](#_Toc193477639)

[***1.6.2. AI kết hợp với điện toán lượng tử* 28**](#_Toc193477640)

[***1.6.3. AI và IoT (AIoT)* 28**](#_Toc193477641)

[***1.6.4. Ứng dụng AI trong sáng tạo nội dung (AI-Generated Content)* 29**](#_Toc193477642)

[**CHƯƠNG 2. PHÂN TÍCH LOGIC AI TRONG “THUẬT TOÁN LẬP LỊCH THỜI KHÓA BIỂU” 30**](#_Toc193477643)

[**2.1 Giới thiệu thuật toán di truyền (Genetic Algorithm - GA) 30**](#_Toc193477644)

[**2.2 Biểu diễn dữ liệu trong GA 30**](#_Toc193477645)

[**2.3 Đánh giá chất lượng cá thể (Hàm fitness) 30**](#_Toc193477646)

[**2.3.1 Yêu cầu cứng (ràng buộc bắt buộc) 30**](#_Toc193477647)

[**2.3.2 Yêu cầu mềm (tối ưu hóa) 31**](#_Toc193477648)

[**2.4 Quá trình tiến hóa trong GA 31**](#_Toc193477649)

[**2.4.1 Khởi tạo quần thể 31**](#_Toc193477650)

[**2.4.2 Chọn lọc 31**](#_Toc193477651)

[**2.4.3 Lai ghép (Crossover) 32**](#_Toc193477652)

[***2.4.4 Đột biến (Mutation)* 32**](#_Toc193477653)

[**2.5 Đánh giá và xuất kết quả 33**](#_Toc193477654)

[**2.6 Phân tích mã nguồn thuật toán GA 33**](#_Toc193477655)

[**Khai báo thư viện 33**](#_Toc193477656)

[**Lớp mô tả phòng học 34**](#_Toc193477657)

[**Lớp mô tả giáo viên 34**](#_Toc193477658)

[**Lớp mô tả nhóm sinh viên 34**](#_Toc193477659)

[**Lớp mô tả một lớp học cụ thể 34**](#_Toc193477660)

[**Lớp mô tả lịch học (cá thể trong thuật toán GA) 34**](#_Toc193477661)

[**Hàm thực thi thuật toán GA 36**](#_Toc193477662)

[**Vẽ biểu đồ đánh giá tiến trình tối ưu hóa 36**](#_Toc193477663)

[**Dữ liệu giả lập 36**](#_Toc193477664)

[**Chạy thuật toán GA 37**](#_Toc193477665)

[**Xuất kết quả ra file Excel 37**](#_Toc193477666)

[**CHƯƠNG 3. PHÂN TÍCH CODE 38**](#_Toc193477667)

[**3.1 Giải thích từng phần code 38**](#_Toc193477668)

[***3.1.1 Khai báo lớp và dữ liệu* 38**](#_Toc193477669)

[**3.2 Lớp Schedule và thuật toán GA 39**](#_Toc193477670)

[**3.3 Cách hoạt động của GA trong bài toán 40**](#_Toc193477671)

[**3.4 Đánh giá hiệu suất, ưu điểm, nhược điểm 40**](#_Toc193477672)

[**3.5 So sánh với phương pháp khác 40**](#_Toc193477673)

[**KẾT LUẬN 41**](#_Toc193477674)

[**DANH MỤC TÀI LIỆU THAM KHẢO 43**](#_Toc193477675)

**BẢNG CÁC TỪ VIẾT TẮT**

|  |  |  |
| --- | --- | --- |
| **STT** | **TỪ VIẾT TẮT** | **VIẾT ĐẦY ĐỦ** |
| **1** | **AI** | **Artificial intelligence** |
| **2** | **BI** | **Business Intelligence** |
| **3** | **AGI** | ***Artificial General Intelligence*** |
| **4** | **AIoT** | **Artificial Intelligence of Things** |
| **5** | **IoT** | **Internet of Things** |

**CHƯƠNG 1. TỔNG QUAN VỀ TRÍ TUỆ NHÂN TẠO**

## **1.1. Giới thiệu về trí tuệ nhân tạo**

### *1.1.1. Khái niệm*

- Trí tuệ nhân tạo (Artificial intelligence: AI): Đề cập đến khả năng của các hệ thống máy tính thực hiện các nhiệm vụ liên quan đến trí thông minh của con người, như học tập, suy luận, giải quyết vấn đề, nhận thức và đưa ra quyết định.

- Đây là một lĩnh vực nghiên cứu thuộc khoa học máy tính, tập trung phát triển và nghiên cứu các phương pháp cùng phần mềm giúp máy móc có khả năng nhận thức môi trường xung quanh, sử dụng học tập và trí tuệ để thực hiện hành động nhằm tối đa hóa khả năng đạt được các mục tiêu đã định. Những cỗ máy như vậy có thể được gọi là AI.

### *1.1.2. Lịch sử hình thành*

- Trí tuệ nhân tạo là đột phá công nghệ mới nhất, là ngành khoa học đang định hình lại xã hội của chúng ta. Đồng thời trí tuệ nhân tạo có tác động sâu sắc đến các ngành công nghiệp máy móc và công ty cung cấp năng lượng.

- Với sự phát triển nhanh chóng về kiến ​​thức và tiến bộ trong trí tuệ nhân tạo đang thúc đẩy mạnh mẽ nhu cầu về các dịch vụ kỹ thuật số mới để giúp khai thác công nghệ này với tiềm năng cao nhất. Cải thiện các sản phẩm đã có trên thị trường để tất cả chúng ta đưa vào sử dụng trong cuộc sống hàng ngày sẽ là cốt lõi cho tương lai của trí tuệ nhân tạo AI.

- Đa số trọng tâm của các nghiên cứu trí tuệ nhân tạo ban đầu được lấy từ cách tiếp cận bằng thực nghiệm của tâm lý học, và xem trọng cái gọi là "trí tuệ ngôn ngữ" - việc hiểu biết ngôn ngữ con người (xem Thử nghiệm Turing, Turing test).

- Các hướng nghiên cứu về trí thông minh nhân tạo không liên quan đến ngôn ngữ bao gồm ngành robotic và ngành thông minh tập thể (colective intelligence). Hai hướng tiếp cận này tập trung vào việc chủ động tác động lên môi trường hoặc việc đưa ra quyết định bằng đồng thuận về một vấn đề nào đó. Các hướng nghiên cứu này có nguồn gốc từ các mô hình thông minh xuất phát từ sinh học và chính trị học.

- Lý thuyết trí tuệ nhân tạo còn được rút ra từ các nghiên cứu về động vật, đặc biệt là côn trùng, do có thể dễ dàng được mô phỏng con trùng bằng robot (xem thêm sự sống nhân tạo (artificial life)) cũng như các động vật với nhận thức phức tạp hơn, ví dụ như loài khỉ, chúng có thể bắt chước con người trong nhiều trường hợp nhưng lại hạn chế về sự phát triển trong việc lập kế hoạch và nhận thức. Các nhà nghiên cứu về trí tuệ nhân tạo đưa ra kết luận rằng những loài động vật, có nhận thức đơn giản hơn con người, đáng ra có thể dễ dàng mô phỏng được. Tuy nhiên, đến giờ vẫn chưa có mô hình tính toán đủ tốt mô phỏng trí thông minh của động vật.

- Có những bài báo đưa ra khái niệm trí thông minh của máy (machine intelligence) như bài A Logical Calculus of the Ideas Immanent in Nervous Activity (Tính toán logic của các ý tưởng nội tại trong hoạt động thần kinh) (1943), do Warren McCulloch và Walter Pitts viết. On Computing Machinery and Intelligence (Về bộ máy tính toán và trí thông minh) (1950) được viết bởi Alan Turing và Man-Computer Symbiosis viết bởi J.C.R. Licklider. Xem thêm phần điều khiển học (cybernetics) và Thử thách Turing.

- Trước đó cũng đã có các bài báo phủ nhận khả năng tồn tại trí thông minh của máy dựa vào các lý do logic hay triết học như Minds, Machines and Gödel (1961) của John Lucas.

- Với sự phát triển của các kỹ thuật thực hành dựa trên các nghiên cứu về trí tuệ nhân tạo, những người ủng hộ ngành trí tuệ nhân tạo đã cho rằng phe chống đối ngành này đã liên tục thay đổi lập trường của họ trong các vấn đề như máy tính chơi cờ hay nhận dạng tiếng nói, mà trước đây chúng đã từng được coi là thông minh, để phủ nhận các thành tựu của ngành trí tuệ nhân tạo. Bởi vậy, Douglas Hofstadter, trong cuốn Gödel, Escher, Bach, đã chỉ ra rằng chính sự chuyển dịch đó đã định nghĩa trí thông minh là bất cứ việc gì mà con người làm được còn máy móc thì không.

- John von Neumann (trích dẫn trong E.T. Jaynes) đã thấy trước được điều này vào năm 1948 khi trả lời một lời bình luận tại một buổi diễn thuyết cho rằng may móc không thể suy nghĩ: "Bạn nhất quyết rằng có một điều gì đó mà máy móc không thể làm được. Nếu bạn nói cho tôi một cách chính xác đó là điều gì, thì tôi sẽ luôn luôn làm được một cái máy mà sẽ chỉ thực hiện được điều đó!". Von Neumann được cho là đã có ý nói đến luận đề Church-Turing khi khẳng định rằng bất kì một thủ tục có hiệu lực nào cũng có thể được mô phỏng bởi một máy tính (tổng quát) nào đó.

- Vào năm 1969 McCarthy và Hayes đã bắt đầu thảo luận về bài toán khung (frame problem) với bài luận của họ, Some Philosophical Problems from the Standpoint of Artificial Intelligence (Một số vấn đề triết học từ điểm khởi đầu của trí tuệ nhân tạo).

### *1.1.3. Các cột mốc quan trọng*

a) Thời kỳ sơ khai (Trước năm 1950):

- Ý tưởng về máy móc có trí tuệ. Mặc dù thuật ngữ "trí tuệ nhân tạo" chưa ra đời, nhưng các ý tưởng về máy móc có thể suy nghĩ đã xuất hiện từ lâu:

+ Thế kỷ 17-18: Nhà toán học Gottfried Wilhelm Leibniz và nhà triết học René Descartes đề cập đến việc máy móc có thể thực hiện tư duy.

+ Thế kỷ 19: Nhà logic học George Boole phát triển đại số Boolean – nền tảng cho logic máy tính.

+ Năm 1913: Bertrand Russell và Alfred North Whitehead xuất bản Principia Mathematica, hệ thống hóa logic toán học – bước đệm cho lập trình AI sau này.

+ Năm 1936: Alan Turing đề xuất Máy Turing, đặt nền tảng cho khái niệm máy tính hiện đại.

b) Giai đoạn khởi đầu (năm 1950 - 1960):

- Sự ra đời của AI

+ Năm 1950: Alan Turing công bố bài báo Computing Machinery and Intelligence, đề xuất Turing Test để đánh giá khả năng tư duy của máy tính.

+ Năm 1956: Hội nghị Dartmouth (do John McCarthy và Marvin Minsky tổ chức) chính thức đặt ra thuật ngữ Artificial Intelligence (AI), đánh dấu sự ra đời của ngành AI.

+ Cuối thập niên 1950 - đầu 1960: Newell & Simon phát triển chương trình Logic Theorist – hệ thống máy tính đầu tiên có thể chứng minh các định lý toán học. Frank Rosenblatt phát minh Perceptron – mô hình mạng nơ-ron đầu tiên.

c) Giai đoạn "mùa đông AI" (1970s - 1980s):

- Suy thoái do hạn chế công nghệ Sau những thành công ban đầu, AI gặp nhiều khó khăn do:

+Hạn chế về phần cứng: Máy tính thời đó quá yếu để xử lý AI phức tạp.

+ Thiếu dữ liệu: AI cần nhiều dữ liệu để học, nhưng thời đó chưa có Big Data.

+ Chi phí cao: Nghiên cứu AI rất tốn kém, khiến nhiều dự án bị cắt giảm ngân sách -> Giai đoạn này gọi là "Mùa đông AI" (AI Winter), do sự suy giảm hứng thú và đầu tư vào AI.

d) Giai đoạn hồi sinh (1990s - 2000s):

- AI ứng dụng thực tế Nhờ sự phát triển của máy tính mạnh hơn và dữ liệu lớn (Big Data), AI bắt đầu quay trở lại:

+ Năm 1997: Siêu máy tính Deep Blue của IBM đánh bại kỳ thủ cờ vua huyền thoại Garry Kasparov.

+ Năm 2000: AI được ứng dụng vào công cụ tìm kiếm, nhận dạng giọng nói, và hệ thống đề xuất.

e) Bùng nổ AI hiện đại (2010s - Nay):

- Machine Learning & Deep Learning Sự phát triển của học máy (Machine Learning) và học sâu (Deep Learning) đã đưa AI lên một tầm cao mới:

+ Năm 2011: IBM Watson đánh bại con người trong cuộc thi Jeopardy.

+ Năm 2012: Mạng nơ-ron sâu (Deep Neural Networks) của Google nhận diện ảnh tốt hơn con người (AlexNet).

+ Năm 2016: AlphaGo của DeepMind đánh bại kỳ thủ cờ vây số 1 thế giới Lee Sedol.

+ 2020s: AI phát triển mạnh mẽ với GPT, ChatGPT, DALL-E, Midjourney, Tesla AI (xe tự lái).

### *1.1.4. Phân loại trí tuệ nhân tạo*

a)Có 4 loại AI chính theo chức năng gồm: Reactive machines, limited memory, theory-of-mind và self-aware.

- Máy phản ứng (Reactive Machines):

+ Là loại AI cơ bản nhất, chỉ có thể phản ứng với tình huống hiện tại mà không có khả năng nhớ hoặc học từ kinh nghiệm.

+ Không có trí nhớ và không thể đưa ra quyết định dựa trên dữ liệu quá khứ.

+ Ví dụ: Deep Blue (máy tính chơi cờ vua của IBM) chỉ phân tích nước đi hiện tại và chọn phương án tốt nhất mà không nhớ các ván trước. AI nhận diện hình ảnh chỉ xác định đối tượng mà không lưu trữ thông tin về hình ảnh trước đó.

- Trí nhớ hạn chế (Limited Memory):

+ Có thể ghi nhớ dữ liệu quá khứ trong một khoảng thời gian ngắn để đưa ra quyết định tốt hơn. Đây là loại AI phổ biến hiện nay.

+ Ví dụ: Xe tự lái sử dụng dữ liệu từ cảm biến để theo dõi tốc độ, khoảng cách với xe khác và đưa ra quyết định lái xe an toàn. Chatbot và trợ lý ảo có thể nhớ cuộc trò chuyện trước đó trong cùng một phiên để phản hồi hợp lý hơn.

- Lý thuyết tâm trí (Theory of Mind AI):

+ Có thể hiểu và tương tác với con người như một thực thể có cảm xúc, niềm tin, mong muốn.

+ AI loại này sẽ có thể suy luận về ý định và trạng thái tinh thần của người khác.

+ Hiện tại, AI vẫn chưa đạt đến mức độ này, nhưng một số nghiên cứu đang hướng tới phát triển AI giao tiếp tự nhiên hơn.

+ Ví dụ tiềm năng: Robot xã hội có thể hiểu cảm xúc và phản ứng phù hợp trong giao tiếp con người. AI tâm lý có thể hỗ trợ điều trị tâm lý bằng cách đồng cảm với người dùng.

- AI tự nhận thức (Self-aware AI):

+ Là cấp độ cao nhất của AI, có khả năng tự nhận thức, hiểu về bản thân và có ý thức giống con người.

+ AI này có thể tự suy nghĩ, học hỏi, đặt mục tiêu và đưa ra quyết định mà không cần con người can thiệp.

+ Đây là chủ đề của nhiều bộ phim khoa học viễn tưởng, như Skynet (Terminator), Jarvis (Iron Man).

b) Phân loại theo khả năng thì được chia thành trí thông minh hạn hẹp (ANI), trí thông minh tổng quát (AGI), siêu trí tiệu (SGI).

- Trí tuệ nhân tạo hạn hẹp (ANI - Artificial Narrow Intelligence):

+ Còn gọi là AI yếu (Weak AI), chỉ có thể thực hiện một nhiệm vụ cụ thể và không có khả năng suy nghĩ như con người. Đây là loại AI phổ biến nhất hiện nay.

+ Ví dụ: Chatbot (ChatGPT, Google Bard, Siri, Alexa). Xe tự lái (Tesla Autopilot). Hệ thống đề xuất (Netflix, YouTube, Shopee). AI chơi game (AlphaGo, Stockfish).

- Trí tuệ nhân tạo tổng quát (AGI - Artificial General Intelligence):

+ Còn gọi là AI mạnh (Strong AI), có khả năng tư duy, học hỏi và thích nghi như con người. Có thể giải quyết nhiều vấn đề khác nhau, không chỉ giới hạn ở một nhiệm vụ cụ thể.

+ Nếu AGI phát triển, nó có thể thay thế con người trong nhiều lĩnh vực như nghiên cứu khoa học, y học, sáng tạo nghệ thuật...

+ Ví dụ tiềm năng:

Một AI có thể tự học bất kỳ kỹ năng nào như con người (lập trình, viết sách, sáng tác nhạc...).

Một trợ lý ảo có thể hiểu, phân tích cảm xúc và giao tiếp tự nhiên như con người.

- Siêu trí tuệ nhân tạo (ASI - Artificial Super Intelligence):

+ Mạnh hơn cả trí tuệ con người. Có khả năng suy nghĩ, sáng tạo, cảm xúc, tự nhận thức và đưa ra quyết định mà con người không thể hiểu hết.

+ Nếu ASI xuất hiện, nó có thể tự cải tiến và phát triển vượt xa trí tuệ con người, có khả năng kiểm soát nền văn minh nhân loại.

+ Ví dụ giả thuyết: Skynet trong Terminator (Kẻ Hủy Diệt). Jarvis / Vision trong Iron Man.

## **1.2. Ưu và nhược điểm của AI.**

- Ưu điểm của AI:

+ Năng suất vượt trội: Tự động hóa các tác vụ lặp đi lặp lại, giải phóng sức lao động con người. Xử lý thông tin với tốc độ và khối lượng khổng lồ, vượt xa khả năng con người. Hoạt động liên tục 24/7, không cần nghỉ ngơi.

+ Độ chính xác cao: Hạn chế tối đa lỗi do con người gây ra. Đưa ra quyết định dựa trên dữ liệu và thuật toán, ít bị ảnh hưởng bởi cảm xúc. Phân tích dữ liệu phức tạp để đưa ra kết quả chính xác hơn.

+ Khả năng sáng tạo: Hỗ trợ con người trong các lĩnh vực sáng tạo như nghệ thuật, âm nhạc, thiết kế. Tìm ra những giải pháp mới, đột phá cho các vấn đề phức tạp.

+ Cá nhân hóa trải nghiệm: Cung cấp dịch vụ, sản phẩm được cá nhân hóa dựa trên sở thích và hành vi của người dùng. Nâng cao trải nghiệm của khách hàng trong nhiều lĩnh vực như mua sắm, giải trí, giáo dục.

+ Giải quyết bài toán toàn cầu: Ứng dụng trong các lĩnh vực như y tế, môi trường, năng lượng để giải quyết những thách thức toàn cầu. Phân tích dữ liệu lớn để dự đoán và phòng ngừa rủi ro.

- Nhược điểm của AI:

+ Nguy cơ mất việc làm: Tự động hóa có thể thay thế con người trong nhiều lĩnh vực, dẫn đến thất nghiệp. Đòi hỏi người lao động phải thích ứng và nâng cao kỹ năng để đáp ứng nhu cầu mới.

+ Phân biệt đối xử: Thuật toán AI có thể bị sai lệch do dữ liệu đầu vào thiếu khách quan hoặc mang định kiến. Gây ra những quyết định không công bằng, ảnh hưởng đến các nhóm người yếu thế.

+ Vấn đề về đạo đức: Lo ngại về việc AI vượt khỏi tầm kiểm soát của con người. Sử dụng AI cho mục đích xấu như phát triển vũ khí tự động, thao túng thông tin.

## **1.3. Ứng dụng của AI**

a) Ứng dụng trong đời sống

- Trong ngành vận tải:

+ Trí tuệ nhân tạo được ứng dụng trên những phương tiện vận tải tự lái, điển hình là ô tô. Sự ứng dụng này góp phần mang lại lợi ích kinh tế cao hơn nhờ khả năng cắt giảm chi phí cũng như hạn chế những tai nạn nguy hiểm đến tính mạng.

+ Vào năm 2016, Otto, hãng phát triển xe tự lái thuộc Uber đã vận chuyển thành công 50.000 lon bia Budweisers bằng xe tự lái trên quãng đường dài 193 km. Theo dự đoán của công ty tư vấn công nghệ thông tin Gartner, trong tương lai, những chiếc xe có thể kết nối với nhau thông qua Wifi để đưa ra những lộ trình vận tải tốt nhất. xe vận tải không người lái

-Trong sản xuất:

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

- Trong y tế:

+ Ứng dụng tiêu biểu của trí tuệ nhân tạo trong lĩnh vực y tế là máy bay thiết bị bay không người lái được sử dụng trong những trường hợp cứu hộ khẩn cấp. Thiết bị bay không người lái có tốc độ nhanh hơn xe chuyên dụng đến 40% và vô cùng thích hợp để sử dụng ở những nơi có địa hình hiểm trở. thiết bị không người lái.

- Trong giáo dục:

+ Sự ra đời của trí tuệ nhân tạo giúp tạo ra những thay đổi lớn trong lĩnh vực giáo dục. Các hoạt động giáo dục như chấm điểm hay dạy kèm học sinh có thể được tự động hóa nhờ công nghệ AI. Nhiều trò chơi, phần mềm giáo dục ra đời đáp ứng nhu cầu cụ thể của từng học sinh, giúp học sinh cải thiện tình hình học tập theo tốc độ riêng của mình.

+ Trí tuệ nhân tạo còn có thể chỉ ra những vấn đề mà các khóa học cần phải cải thiện. AI còn có khả năng theo dõi sự tiến bộ của học sinh và thông báo đến giáo viên khi phát hiện ra vấn đề đối với kết quả học tập của học sinh. AI trong giáo dục.

+ Hơn nữa, sinh viên còn có thể học hỏi từ bất cứ nơi nào trên thế giới thông qua việc sử dụng những phần mềm có hỗ trợ AI. Công nghệ AI cũng cung cấp dữ liệu nhằm giúp sinh viên lựa chọn được những khóa học tốt nhất cho mình.

- Trong truyền thông:

+ Đối với lĩnh vực truyền thông, sự phát triển của trí tuệ nhân tạo góp phần làm thay đổi cách thức tiếp cận đối với khách hàng mục tiêu.

+ Nhờ những ưu điểm của công nghệ AI, các công ty có thể cung cấp quảng cáo vào đúng thời điểm, đúng khách hàng tiềm năng, dựa trên việc phân tích các đặc điểm về nhân khẩu học, thói quen hoạt động trực tuyến và những nội dung mà khách hàng thường xem trên quảng cáo. Tìm hiểu về sự phát triển trí tuệ nhân tạo trong cuộc sống.

- Trong ngành dịch vụ:

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

b) Ứng dụng vào xử lý ngôn ngữ tự nhiên:

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

- Các bước xử lý

+ Phân tích hình thái: Trong bước này từng từ sẽ được phân tích và các ký tự không phải chữ (như các dấu câu) sẽ được tách ra khỏi các từ. Trong tiếng Anh và nhiều ngôn ngữ khác, các từ được phân tách với nhau bằng dấu cách. Tuy nhiên trong tiếng Việt, dấu cách được dùng để phân tách các tiếng (âm tiết) chứ không phải từ. Cùng với các ngôn ngữ như tiếng Trung, tiếng Hàn, tiếng Nhật, phân tách từ trong tiếng Việt là một công việc không hề đơn giản.

+ Phân tích cú pháp: Dãy các từ sẽ được biến đổi thành các cấu trúc thể hiện sự liên kết giữa các từ này. Sẽ có những dãy từ bị loại do vi phạm các luật văn phạm.

+ Phân tích ngữ nghĩa: Thêm ngữ nghĩa vào các cấu trúc được tạo ra bởi bộ phân tích cú pháp. Tích hợp văn bản - Ngữ nghĩa của một câu riêng biệt có thể phụ thuộc vào những câu đứng trước, đồng thời nó cũng có thể ảnh hưởng đến các câu phía sau.

+ Phân tích thực nghĩa: Cấu trúc thể hiện điều được phát ngôn sẽ được thông dịch lại để xác định nó thật sự có nghĩa là gì.

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

- Các bài toán và ứng dụng:

+ Nhận dạng chữ viết: Có hai kiểu nhận dạng, thứ nhất là nhận dạng chữ in, ví dụ nhận dạng chữ trên sách giáo khoa rồi chuyển nó thành dạng văn bản điện tử như dưới định dạng doc của Microsoft Word chẳng hạn. Phức tạp hơn là nhận dạng chữ viết tay, có khó khăn bởi vì chữ viết tay không có khuôn dạng rõ ràng và thay đổi từ người này sang người khác. Với chương trình nhận dạng chữ viết in có thể chuyển hàng ngàn đầu sách trong thư viện thành văn bản điện tử trong thời gian ngắn. Nhận dạng chữ viết của con người có ứng dụng trong khoa học hình sự và bảo mật thông tin (nhận dạng chữ ký điện tử).

+ Nhận dạng tiếng nói: Nhận dạng tiếng nói rồi chuyển chúng thành văn bản tương ứng. Giúp thao tác của con người trên các thiết bị nhanh hơn và đơn giản hơn, chẳng hạn thay vì gõ một tài liệu nào đó bạn đọc nó lên và trình soạn thảo sẽ tự ghi nó ra. Đây cũng là bước đầu tiên cần phải thực hiện trong ước mơ thực hiện giao tiếp giữa con người với robot. Nhận dạng tiếng nói có khả năng trợ giúp người khiếm thị rất nhiều.

+ Tổng hợp tiếng nói: Từ một văn bản tự động tổng hợp thành tiếng nói. Thay vì phải tự đọc một cuốn sách hay nội dung một trang web, nó tự động đọc cho chúng ta. Giống như nhận dạng tiếng nói, tổng hợp tiếng nói là sự trợ giúp tốt cho người khiếm thị, nhưng ngược lại nó là bước cuối cùng trong giao tiếp giữa robot với người.

+ Dịch tự động (machine translate): Như tên gọi đây là chương trình dịch tự động từ ngôn ngữ này sang ngôn ngữ khác. Một phần mềm điển hình về tiếng Việt của chương trình này là Evtrans của Softex, dịch tự động từ tiếng Anh sang tiếng Việt và ngược lại, phần mềm từng được trang web vdict.com mua bản quyền, đây cũng là trang đầu tiên đưa ứng dụng này lên mạng. Tháng 10 năm 2008 có hai công ty tham gia vào lĩnh vực này cho ngôn ngữ tiếng Việt là công ty Lạc Việt (công ty phát hành từ điển Lạc Việt) và Google, một thời gian sau đó Xalo\_vn cũng đưa ra dịch vụ tương tự.

+ Tìm kiếm thông tin (information retrieval): Đặt câu hỏi và chương trình tự tìm ra nội dung phù hợp nhất. Thông tin ngày càng đầy lên theo cấp số nhân, đặc biệt với sự trợ giúp của internet việc tiếp cận thông tin trở lên dễ dàng hơn bao giờ hết. Việc khó khăn lúc này là tìm đúng nhất thông tin mình cần giữa bề bộn tri thức và đặc biệt thông tin đó phải đáng tin cậy. Các máy tìm kiếm dựa trên giao diện web như Google hay Yahoo hiện nay chỉ phân tích nội dung rất đơn giản dựa trên tần suất của từ khoá và thứ hạng của trang và một số tiêu chí đánh giá khác để đưa ra kết luận, kết quả là rất nhiều tìm kiếm không nhận được câu trả lời phù hợp, thậm chí bị dẫn tới một liên kết không liên quan gì do thủ thuật đánh lừa của các trang web nhằm giới thiệu sản phẩm (có tên tiếng Anh là SEO viết tắt của từ search engine optimization). Thực tế cho đến bây giờ chưa có máy tìm kiếm nào hiểu được ngôn ngữ tự nhiên của con người trừ trang www.ask.com được đánh giá là "hiểu" được những câu hỏi có cấu trúc ở dạng đơn giản nhất. Mới đây cộng đồng mạng đang xôn xao về trang Wolfram Alpha, được hứa hẹn là có khả năng hiểu ngôn ngữ tự nhiên của con người và đưa ra câu trả lời chính xác. Lĩnh vực này hứa hẹn tạo ra bước nhảy trong cách thức tiếp nhận tri thức của cả cộng đồng.

+ Tóm tắt văn bản: Từ một văn bản dài tóm tắt thành một văn bản ngắn hơn theo mong muốn nhưng vẫn chứa những nội dung thiết yếu nhất.

+ Khai phá dữ liệu (data mining) và phát hiện tri thức: Từ rất nhiều tài liệu khác nhau phát hiện ra tri thức mới. Thực tế để làm được điều này rất khó, nó gần như là mô phỏng quá trình học tập, khám phá khoa học của con người, đây là lĩnh vực đang trong giai đoạn đầu phát triển. Ở mức độ đơn giản khi kết hợp với máy tìm kiếm nó cho phép đặt câu hỏi để từ đó công cụ tự tìm ra câu trả lời dựa trên các thông tin trên web mặc cho việc trước đó có câu trả lời lưu trên web hay không (giống như trang Yahoo! hỏi và đáp, nơi chuyên đặt các câu hỏi để người khác trả lời), nói một cách nôm na là nó đã biết xử lý dữ liệu để trả lời câu hỏi của người sử dụng, thay vì máy móc đáp trả những gì chỉ có sẵn trong bộ nhớ.

+ Sửa lỗi chính tả: Là quá trình phát hiện và sửa các lỗi chính tả trong một đoạn văn bản. Phần mềm sửa lỗi chính tả thường được tích hợp trong các ứng dụng văn phòng như Microsoft Word hay Google Docs để hỗ trợ người dùng trong việc soạn thảo văn bản. Sửa lỗi chính tả có thể được áp dụng cho nhiều ngôn ngữ khác nhau, bao gồm tiếng Việt.

c) Ứng dụng vào Robot học

- Robot học (tiếng Anh: Robotics) là một ngành kỹ thuật bao gồm thiết kế, chế tạo, vận hành, và ứng dụng robot, cũng như các hệ thống máy tính để điều khiển, phản hồi tín hiệu cảm biến, và xử lý thông tin của chúng.

- Những công nghệ này liên hệ với các máy móc tự động dùng để thay thế con người trong những môi trường độc hại hoặc trong các quá trình sản xuất, hoặc bắt chước con người về hình thức, hành vi, hoặc/và nhận thức. Nhiều robot ngày nay được lấy cảm hứng từ các loài vật, còn gọi là robot phỏng sinh học.

## **1.4. Công cụ và nền tảng AI phổ biến**

### *1.4.1. Thư Viện và Framework AI Phổ Biến*

a) Khái niệm về thư viện

- Thư viện (Library) là một tập hợp các đoạn mã (code) được viết sẵn, giúp lập trình viên dễ dàng thực hiện các tác vụ nhất định mà không cần viết lại từ đầu. Nó giống như một bộ công cụ gồm các hàm, lớp, và mô-đun có thể tái sử dụng trong các chương trình khác nhau.

- Ví dụ:

+ NumPy: Cung cấp các hàm xử lý ma trận và tính toán số học.

+ Matplotlib: Vẽ biểu đồ và trực quan hóa dữ liệu.

+ Scikit-Learn: Hỗ trợ các thuật toán Machine Learning như phân loại, hồi quy, và clustering.

b) Khái niệm về Framework

- Framework là các đoạn code đã được viết sẵn, cấu thành nên một bộ khung và các thư viện lập trình được đóng gói. Chúng cung cấp các tính năng có sẵn như mô hình, API và các yếu tố khác để tối giản cho việc phát triển các ứng dụng web phong phú, năng động. Các framework giống như là chúng ta có khung nhà được làm sẵn nền móng cơ bản, bạn chỉ cần vào xây dựng và nội thất theo ý mình.

c) Một số Framework phổ biến

\* TensorFlow:

- Phát triển bởi: Google.

- Loại AI: Machine Learning (ML), Deep Learning (DL).

- Tính năng chính:

+ Xây dựng và huấn luyện các mô hình AI từ đơn giản đến phức tạp.

+ Hỗ trợ nhiều mô hình mạng nơ-ron sâu như CNN, RNN, và Transformer.

+ Tối ưu hiệu suất bằng GPU và TPU.

+ TensorFlow Lite giúp triển khai mô hình trên thiết bị di động.

+ TensorBoard hỗ trợ trực quan hóa các thông số và quá trình huấn luyện.

- Ưu điểm:

+ Khả năng xử lý dữ liệu lớn.

+ Hỗ trợ phân tán để tăng tốc độ huấn luyện.

+ Có hệ sinh thái mạnh mẽ, bao gồm TensorFlow Extended (TFX) và TensorFlow.js.

- Nhược điểm:

+ Khá phức tạp đối với người mới bắt đầu.

+ Mất nhiều thời gian để debug khi gặp lỗi.

- Ứng dụng:

+ Nhận diện hình ảnh (Image Recognition).

+ Xử lý ngôn ngữ tự nhiên (NLP).

+ Xe tự lái và robotics.

\* PyTorch:

- Phát triển bởi: Meta (Facebook).

- Loại AI: Deep Learning.

- Tính năng chính:

+ Dễ dàng xây dựng và tùy chỉnh các mô hình AI.

+ Hỗ trợ dynamic computation graph (đồ thị tính toán động), giúp dễ dàng debug.

+ Hỗ trợ GPU mạnh mẽ. PyTorch Lightning giúp đơn giản hóa code cho nghiên cứu.

- Ưu điểm:

+ Giao diện thân thiện, dễ đọc và dễ hiểu.

+ Thích hợp cho nghiên cứu và thử nghiệm các thuật toán mới.

+ Tích hợp tốt với NumPy và OpenCV.

- Nhược điểm:

+ Không có công cụ trực quan hóa mạnh như TensorBoard (dù có thể kết hợp TensorBoard với PyTorch).

+ Tối ưu hóa trên môi trường sản xuất không tốt bằng TensorFlow.

- Ứng dụng:

+ Nghiên cứu AI. Thị giác máy tính (Computer Vision).

+ Xử lý ngôn ngữ tự nhiên (NLP).

\* Keras:

- Phát triển bởi: Google.

- Loại AI: Deep Learning.

- Tính năng chính:

+ API cấp cao giúp dễ dàng xây dựng và huấn luyện mô hình AI.

+ Tích hợp hoàn toàn với TensorFlow.

+ Hỗ trợ xây dựng các mạng nơ-ron nhanh chóng bằng cách thêm các lớp (Layers).

- Ưu điểm:

+ Thân thiện với người mới học.

+ Tối ưu thời gian phát triển.

+ Hỗ trợ huấn luyện mô hình trên GPU và TPU.

- Nhược điểm:

+ Không phù hợp với các dự án AI quy mô lớn và phức tạp.

+ Ít linh hoạt hơn so với TensorFlow hoặc PyTorch.

- Ứng dụng:

+ Phân loại hình ảnh (Image Classification).

+ Chatbot và trợ lý ảo.

+ Phân tích và dự báo dữ liệu.

\* Scikit-Learn:

- Phát triển bởi: Cộng đồng mã nguồn mở.

- Loại AI: Machine Learning truyền thống.

- Tính năng chính:

+ Hỗ trợ các thuật toán Machine Learning như phân loại, hồi quy, và phân cụm.

+ Cung cấp các công cụ để tiền xử lý và trực quan hóa dữ liệu.

+ Tích hợp tốt với NumPy và Pandas. Ưu điểm: Giao diện dễ sử dụng, phù hợp cho người mới học.

+ Có nhiều thuật toán đã được tối ưu.

+ Thư viện đa dạng với các công cụ đánh giá mô hình.

- Nhược điểm:

+ Không phù hợp cho các bài toán Deep Learning.

+ Không hỗ trợ GPU để tăng tốc xử lý.

- Ứng dụng:

+ Phân tích dữ liệu và trực quan hóa.

+ Dự báo tài chính.

+ Phân loại khách hàng trong marketing.

### *1.4.2. Nền tảng AI đám mây*

a) Khái niệm

- AI đám mây (cluod AI): Sự kết hợp giữa điện toán đám mây và AI, được gọi là Artificial Intelligence Cloud hoặc Cloud AI, đại diện cho sự kết hợp đột phá của các công nghệ. Các tổ chức có thể kết hợp liền mạch các công cụ AI, thuật toán và dịch vụ đám mây vào hoạt động hàng ngày của mình bằng cách kết hợp AI với điện toán đám mây.

- Cloud AI cho phép các doanh nghiệp tận dụng toàn bộ tiềm năng của AI, bao gồm học máy, xử lý ngôn ngữ tự nhiên và thị giác máy tính. Việc tích hợp này mang lại nhiều lợi thế, giúp các doanh nghiệp có được lợi thế cạnh tranh.

b) Ứng dụng

- Các nhà phát triển có thể lưu trữ nền tảng AI mà không cần xây dựng và duy trì cơ sở hạ tầng riêng biệt bằng cách sử dụng các dịch vụ đám mây công cộng. Để kiểm tra và triển khai các ứng dụng AI, các nhà phát triển có thể sử dụng các mô hình và cài đặt được xác định trước. Hơn nữa, bằng cách vận hành chúng từ đám mây và sử dụng dữ liệu của bên thứ nhất do công ty sản xuất, các dịch vụ AI nói chung như chuyển giọng nói thành văn bản, phân tích và trực quan hóa không phải lúc nào cũng cần mô hình ML có thể được cải thiện tốt hơn.

- Một số ứng dụng dựa trên AI phổ biến trên đám mây bao gồm:

+ IoT: Các kiến ​​trúc và dịch vụ đám mây hỗ trợ Internet có thể lưu trữ và xử lý dữ liệu do nền tảng AI tạo ra trên các thiết bị IoT.

+ Chatbots: Chatbots là phần mềm dựa trên AI phổ biến, sử dụng xử lý ngôn ngữ tự nhiên để thực hiện các cuộc trò chuyện với người dùng, một lợi ích cho dịch vụ khách hàng đảm bảo cho sự hài lòng tức thời. Nền tảng đám mây lưu trữ và xử lý dữ liệu được thu thập bởi chatbot và dịch vụ đám mây kết nối chúng với các ứng dụng thích hợp để xử lý thêm. Dữ liệu khách hàng cũng được đưa trở lại ứng dụng chatbot nằm trên đám mây.

+ Business Intelligence: BI là một ứng dụng chính thống khác, nơi điện toán đám mây AI có thể thu thập dữ liệu về thị trường, đối tượng mục tiêu và đối thủ cạnh tranh của khách hàng. Đám mây một lần nữa tạo điều kiện thuận lợi cho việc lưu trữ và truyền dữ liệu trong khi AI chạy dữ liệu đó thông qua các mô hình phân tích dự đoán.

+ AI dưới dạng dịch vụ (AIaaS): Các nhà cung cấp đám mây công cộng hiện cung cấp dịch vụ gia công AI, cho phép các công ty thử nghiệm phần mềm và thuật toán ML mà không gặp rủi ro với cơ sở hạ tầng chính của họ. Họ có thể triển khai các ứng dụng AI có sẵn với chi phí chỉ bằng một phần chi phí của AI nội bộ với mức tiết kiệm CAPEX (chi phí vốn) đáng kể.

+ Điện toán đám mây nhận thức: Điện toán nhận thức là việc sử dụng các mô hình AI để tái tạo và mô phỏng quá trình suy nghĩ của con người trong các tình huống phức tạp. Những công ty như IBM và Google đã xây dựng nền tảng đám mây nhận thức để cung cấp những hiểu biết sâu sắc về nhận thức dưới dạng dịch vụ cho các doanh nghiệp và tạo điều kiện thuận lợi cho việc ứng dụng công nghệ này trong tài chính, bán lẻ, chăm sóc sức khỏe và các ngành khác.

c) Ưu và nhược điểm

\* Ưu điểm:

- Tiết kiệm chi phí:

+ Theo truyền thống, các trung tâm dữ liệu kinh doanh sử dụng các máy chủ đắt tiền với nhiều GPU để chạy các mô hình dựa trên ML. Chi phí phát triển, thử nghiệm và triển khai các mô hình này đã giảm đáng kể nhờ những tiến bộ về ảo hóa trong cả đám mây công cộng và riêng tư. Điều này đã giúp nhiều doanh nghiệp vừa và nhỏ có được một sân chơi bình đẳng.

- Năng suất:

+ Các thuật toán dựa trên AI đòi hỏi thời gian và nỗ lực đáng kể của quản trị viên trong việc xây dựng môi trường sản xuất và thử nghiệm, quản lý phần mềm cũng như cung cấp tài nguyên phần cứng cho hoạt động điện toán và lưu trữ. Đám mây lai được quản lý tập trung hoặc đám mây công cộng sẽ loại bỏ điều này, khiến nhân viên CNTT phải tập trung vào các nhiệm vụ không lặp lại.

- Tự động hóa:

+ Điện toán đám mây AI đang được tích hợp trực tiếp vào cơ sở hạ tầng để hỗ trợ tự động hóa các tác vụ lặp đi lặp lại và hợp lý hóa khối lượng công việc. Công nghệ AI có thể được sử dụng để giám sát, quản lý và tự phục hồi các thành phần đám mây công cộng và riêng lẻ trong môi trường đám mây lai.

- Phân tích:

+ Để có thêm thông tin chi tiết, dữ liệu được tìm thấy trong phần lớn khối lượng công việc trên đám mây phải được đánh giá. Việc khai thác dữ liệu theo thời gian thực cũng như việc tạo bảng điều khiển và phân tích gốc cho từng ứng dụng này được thực hiện đơn giản bằng các mô hình dựa trên AI.

- Quản lý dữ liệu:

+ Thông qua việc tạo và xử lý dữ liệu theo thời gian thực, trí tuệ nhân tạo (AI) tăng cường khối lượng công việc trên nền tảng đám mây trong chăm sóc khách hàng, tiếp thị, ERP và quản lý chuỗi cung ứng.

\*Nhược điểm:

- Chi phí cao:

+ Phát Sinh Chi Phí Lớn: Mặc dù có mô hình pay-as-you-go (trả tiền theo mức sử dụng), nhưng khi sử dụng với khối lượng lớn hoặc yêu cầu tài nguyên cao, chi phí có thể tăng đột biến.

+ Phí Dịch Vụ Bổ Sung: Các dịch vụ lưu trữ, xử lý dữ liệu và phân tích chuyên sâu thường tính phí riêng, làm tổng chi phí tăng lên đáng kể.

- Vấn đề kỹ thuật:

+ Mặc dù đúng là thông tin và dữ liệu trên đám mây có thể được truy cập mọi lúc, mọi nơi, nhưng đôi khi hệ thống này có thể gặp một số rối loạn chức năng nghiêm trọng. Bạn nên biết một thực tế là công nghệ này luôn dễ xảy ra sự cố và các vấn đề kỹ thuật khác. Ngay cả những nhà cung cấp dịch vụ đám mây tốt nhất cũng gặp phải rắc rối này, mặc dù luôn tuân thủ các tiêu chuẩn bảo trì cao. Bên cạnh đó, bạn sẽ cần một kết nối Internet rất tốt để có thể đăng nhập vào máy chủ mọi lúc. Bạn sẽ luôn bị mắc kẹt trong trường hợp mạng và sự cố kết nối.

- Tuân thủ bảo mật:

+ Vấn đề chính khác khi sử dụng đám mây là vấn đề tuân thủ bảo mật. Trước khi áp dụng công nghệ này, bạn nên biết rằng bạn sẽ giao tất cả thông tin nhạy cảm của công ty mình cho nhà cung cấp dịch vụ đám mây bên thứ ba. Điều này có thể khiến công ty của bạn gặp rủi ro. Do đó, bạn cần hoàn toàn đảm bảo rằng bạn chọn nhà cung cấp dịch vụ đáng tin cậy nhất, người sẽ giữ thông tin của bạn hoàn toàn an toàn.

- Rủi ro bị tấn công:

+ Lưu trữ thông tin trên đám mây có thể khiến công ty của bạn dễ bị tấn công và các mối đe dọa từ bên ngoài. Như bạn đã biết, không có gì trên Internet là hoàn toàn an toàn và do đó, luôn có khả năng ẩn dữ liệu nhạy cảm.

d) Ứng dụng

- Y tế và chăm sóc sức khỏe:

+ Chẩn đoán hình anh y khoa: Sử dụng AI để phân tích hình ảnh X-quang, MRI, CT Scan nhằm phát hiện sớm các bệnh lý như ung thư, tổn thương não, hoặc bệnh tim.

+ Phát Hiện Bệnh Dịch: Phân tích dữ liệu lớn để phát hiện và dự đoán các đợt bùng phát dịch bệnh.

+ Trợ lý y tế ao: Chatbot AI hỗ trợ bác sĩ và bệnh nhân trong việc tư vấn y tế.

+ Ví dụ: Google Health sử dụng Cloud AI để hỗ trợ bác sĩ đọc hình ảnh y khoa chính xác hơn.

- Thương mại điện tử:

+ Đề xuất sản phẩm: Phân tích hành vi người dùng để gợi ý sản phẩm phù hợp.

+ Chatbot tư vấn: Hỗ trợ khách hàng 24/7 bằng các trợ lý ảo thông minh.

+ Quản lý tồn kho: Dự báo nhu cầu và tối ưu hóa chuỗi cung ứng.

+ Ví dụ: Amazon và Lazada sử dụng Cloud AI để cá nhân hóa trải nghiệm mua sắm trực tuyến.

- Giao thông và vận tải:

+ Xe tự lái: Xử lý dữ liệu từ cảm biến và camera để điều khiển phương tiện tự động.

+ Dự báo giao thông: Phân tích lưu lượng xe và dự đoán các khu vực tắc nghẽn.

+ Quản lý đội xe: Tối ưu hóa tuyến đường và lịch trình vận chuyển.

+ Ví dụ: Tesla và Waymo sử dụng Cloud AI để huấn luyện các mô hình lái xe tự động.

- Giải trí và truyền thông:

+ Phân tích video và hình ảnh: Tự động nhận diện khuôn mặt, vật thể và khung cảnh trong video.

+ Tạo nội dung tự động: Tạo phụ đề, tóm tắt video hoặc đề xuất nội dung phù hợp cho người dùng.

+ Chống vi phạm bản quyền: Xác định nội dung vi phạm bản quyền bằng AI.

+ Ví dụ: YouTube sử dụng Cloud AI để phát hiện và chặn các video vi phạm bản quyền.

- Tài chính và ngân hàng:

+ Phát hiện gian lận: Phân tích giao dịch để phát hiện các hoạt động đáng ngờ.

+ Tư vấn đầu tư: Cung cấp các khuyến nghị đầu tư dựa trên phân tích dữ liệu tài chính.

+ Tự động hóa quy trình: Hỗ trợ xử lý các tác vụ như xét duyệt khoản vay hoặc bảo hiểm.

+ Ví dụ: JPMorgan Chase sử dụng Cloud AI để giám sát và phát hiện gian lận tài chính.

- Giáo dục:

+ Trợ giảng ảo: Cung cấp trợ lý ảo giúp học sinh và giáo viên giải đáp thắc mắc.

+ Phân tích học tập: Theo dõi tiến độ học tập và gợi ý nội dung học phù hợp.

+ Tạo đề thi và bài tập: Sinh đề thi tự động dựa trên nội dung bài giảng.

+ Ví dụ: Duolingo và Khan Academy sử dụng AI để cá nhân hóa trải nghiệm học tập.

- Sản xuất và công nghiệp:

+ Bảo trì dự đoán: Phát hiện sớm các sự cố máy móc thông qua phân tích dữ liệu cảm biến.

+ Tự động hóa quy trình sản xuất: Sử dụng robot và hệ thống AI để giám sát và tối ưu hóa dây chuyền sản xuất.

+ Kiểm tra chất lượng: Phát hiện lỗi sản phẩm bằng AI và thị giác máy tính.

+ Ví dụ: BMW và Siemens ứng dụng Cloud AI để giám sát và bảo trì máy móc trong nhà máy.

- Môi trường và năng lượng:

+ Dự báo thời tiết và thảm họa: Phân tích dữ liệu khí hậu để dự báo bão, lũ lụt hoặc hạn hán.

+ Tối ưu hóa tiêu thụ năng lượng: Điều chỉnh mức tiêu thụ điện dựa trên nhu cầu thực tế. Giám sát môi trường: Theo dõi mức độ ô nhiễm không khí, nước và đất.

+ Ví dụ: NASA sử dụng Cloud AI để theo dõi các biến đổi khí hậu từ dữ liệu vệ tinh.

## **1.5. Thách thức và hạn chế của AI**

### *1.5.1. Thách thức*

- Về kỹ thuật:

+ Dữ liệu không đầy đủ hoặc không chính xác: AI cần lượng lớn dữ liệu để huấn luyện, nhưng dữ liệu có thể bị thiếu, sai lệch hoặc không đại diện.

+ Thiếu khả năng giải thích: Nhiều mô hình AI (đặc biệt là Deep Learning) hoạt động như “hộp đen” và khó giải thích cách đưa ra quyết định.

+ Yêu cầu tài nguyên lớn: Huấn luyện và triển khai AI đòi hỏi hiệu suất tính toán cao và tài nguyên phần cứng mạnh.

+ Bảo mật và an toàn: AI có thể bị tấn công hoặc bị đánh lừa bằng các cuộc tấn công adversarial (tấn công đối nghịch).

- Về đạo đức và xã hội:

+ Thiên vị và phân biệt: AI có thể kế thừa các định kiến từ dữ liệu huấn luyện, dẫn đến kết quả không công bằng hoặc phân biệt đối xử.

+ Mất việc làm: Tự động hóa bằng AI có thể thay thế công việc của con người trong một số ngành nghề.

+ Quyền riêng tư: AI có thể xâm phạm quyền riêng tư khi thu thập và phân tích dữ liệu cá nhân.

+ Quản lý và giám sát: Việc xây dựng khung pháp lý và quy tắc đạo đức cho AI vẫn còn nhiều thách thức.

- Về kinh tế và triển khai

+ Chi phí cao: Phát triển, triển khai và duy trì hệ thống AI tốn kém, đặc biệt với các doanh nghiệp nhỏ và vừa.

+ Khó khăn trong tích hợp: Việc tích hợp AI vào các hệ thống hiện có có thể phức tạp và đòi hỏi kỹ thuật cao.

+ Tốc độ thay đổi nhanh: Công nghệ AI phát triển nhanh chóng, làm cho các hệ thống trở nên lỗi thời nếu không được cập nhật thường xuyên.

### *1.5.2. Hạn Chế của AI*

- Không có khả năng tư duy như con người: AI chỉ hoạt động dựa trên dữ liệu và thuật toán, không có khả năng tư duy sáng tạo hoặc cảm xúc như con người.

- Phụ thuộc vào dữ liệu: Nếu dữ liệu không đủ hoặc không chất lượng, AI sẽ không thể đưa ra kết quả chính xác.

- Khả năng thích nghi kém: AI khó xử lý các tình huống mới hoặc chưa từng gặp phải, đặc biệt trong môi trường thực tế.

- Thiếu nhận thức ngữ cảnh: AI có thể hiểu sai các ngữ cảnh phức tạp hoặc các tình huống đa nghĩa.

## **1.6. Xu hướng phát triển**

### *1.6.1. AI Tổng quát (AGI - Artificial General Intelligence)*

- Khái niệm: AI Tổng Quát (AGI) là AI có khả năng thực hiện bất kỳ nhiệm vụ trí tuệ nào mà con người có thể làm, bao gồm suy luận, sáng tạo, và giải quyết vấn đề trong các ngữ cảnh đa dạng.

- Khác với AI hiện tại: AI hiện nay chủ yếu là AI Hẹp (Narrow AI), chỉ giỏi trong một số tác vụ cụ thể như nhận diện hình ảnh hoặc dịch ngôn ngữ. Trong khi đó, AGI có khả năng tự học và thích nghi linh hoạt như con người.

- Ứng dụng tiềm năng:

+ Robot tự động thông minh có thể làm việc trong nhiều lĩnh vực.

+ Hỗ trợ ra quyết định chiến lược trong các lĩnh vực phức tạp.

+ Tạo ra các mô hình khoa học và dự đoán các hiện tượng tự nhiên.

- Ví dụ: DeepMind (thuộc Google) đang nghiên cứu AGI với các dự án như AlphaGo và AlphaFold.

### *1.6.2. AI kết hợp với điện toán lượng tử*

- Khái niệm: Điện toán lượng tử sử dụng các qubit (bit lượng tử) để thực hiện các phép tính song song với tốc độ vượt trội so với máy tính truyền thống. Khi kết hợp với AI, các thuật toán học máy có thể được tăng tốc đáng kể.

- Ưu điểm:

+ Giải quyết các bài toán tối ưu hóa phức tạp.

+ Mô phỏng các phản ứng hóa học và vật lý lượng tử.

+ Phát triển các mô hình AI mạnh mẽ hơn.

- Ứng dụng tiềm năng:

+ Dược phẩm: Tăng tốc nghiên cứu và phát triển thuốc mới.

+ Tài chính: Tối ưu hóa danh mục đầu tư và quản lý rủi ro.

+ Năng lượng: Tối ưu hóa các hệ thống năng lượng tái tạo.

- Ví dụ: IBM và Google đang phát triển các máy tính lượng tử và kết hợp chúng với AI để giải quyết các bài toán phức tạp.

### *1.6.3. AI và IoT (AIoT)*

- Khái niệm: AIoT (Artificial Intelligence of Things) là sự kết hợp giữa AI và Internet of Things (IoT), cho phép các thiết bị thông minh thu thập, phân tích và phản hồi dữ liệu trong thời gian thực.

- Ưu điểm:

+ Tối ưu hóa quy trình sản xuất và vận hành.

+ Giảm chi phí bảo trì nhờ bảo trì dự đoán.

+ Cải thiện trải nghiệm người dùng qua các thiết bị thông minh.

- Ứng dụng tiềm năng:

+ Nhà thông minh: Hệ thống AI điều chỉnh ánh sáng, nhiệt độ và an ninh dựa trên thói quen của người dùng.

+ Thành phố thông minh: Quản lý giao thông, năng lượng và rác thải hiệu quả hơn.

+ Chăm sóc sức khỏe: Giám sát bệnh nhân từ xa và phát hiện các dấu hiệu bất thường.

- Ví dụ: Amazon Alexa và Google Home sử dụng AIoT để điều khiển các thiết bị trong nhà.

### *1.6.4. Ứng dụng AI trong sáng tạo nội dung (AI-Generated Content)*

- Khái niệm: AI-Generated Content là việc sử dụng AI để tạo ra nội dung như văn bản, hình ảnh, video, âm thanh hoặc thậm chí cả mã nguồn. Các mô hình AI như GPT và DALL·E có thể tạo ra các nội dung sáng tạo chỉ từ mô tả đơn giản.

- Ưu điểm:

+ Tiết kiệm thời gian và chi phí trong sản xuất nội dung.

+ Cung cấp công cụ hỗ trợ cho các nhà sáng tạo.

+ Tạo ra nội dung đa dạng và phong phú.

- Ứng dụng tiềm năng:

+ Marketing và Quảng cáo: Tạo bài viết, hình ảnh và video quảng cáo.

+ Giải trí: Tạo âm nhạc, phim hoạt hình hoặc trò chơi điện tử.

+ Giáo dục: Tạo bài giảng, tài liệu học tập và mô phỏng trực quan.

+ Ví dụ: ChatGPT và DALL·E tạo ra văn bản và hình ảnh dựa trên yêu cầu của người dùng, trong khi Runway ML hỗ trợ chỉnh sửa video bằng AI.

**CHƯƠNG 2. PHÂN TÍCH LOGIC AI TRONG “THUẬT TOÁN LẬP LỊCH THỜI KHÓA BIỂU”**

## **2.1 Giới thiệu thuật toán di truyền (Genetic Algorithm - GA)**

Thuật toán di truyền (GA) là một phương pháp tối ưu hóa dựa trên nguyên lý chọn lọc tự nhiên và di truyền của Darwin. Trong bài toán lập lịch thời khóa biểu, GA được sử dụng để tìm ra lịch học tối ưu bằng cách tạo ra nhiều cá thể (lịch học ngẫu nhiên), đánh giá chất lượng của chúng, sau đó lai ghép và đột biến để cải thiện kết quả theo thời gian.

## **2.2 Biểu diễn dữ liệu trong GA**

GA làm việc với một tập hợp các cá thể (chromosome), mỗi cá thể biểu diễn một lịch học. Trong chương trình, mỗi cá thể được định nghĩa bởi lớp Schedule, trong đó chứa danh sách các lớp học (ClassSchedule), danh sách phòng học (Room), danh sách giáo viên (Teacher), và danh sách nhóm sinh viên (StudentGroup).

Mỗi lớp học bao gồm:

Môn học (subject)

Giáo viên (teacher)

Nhóm sinh viên (student\_group)

Thời lượng (duration)

Thiết bị yêu cầu (equipment)

Phòng học được chỉ định (room)

Thời gian học (time\_slot)

## **2.3 Đánh giá chất lượng cá thể (Hàm fitness)**

Hàm fitness() trong lớp Schedule có nhiệm vụ đánh giá mức độ tối ưu của một lịch học dựa trên các tiêu chí sau:

### 2.3.1 Yêu cầu cứng (ràng buộc bắt buộc)

**Tránh trùng phòng học**: Nếu hai lớp học cùng sử dụng một phòng vào cùng một thời điểm, điểm số bị trừ (-10 điểm).

**Tránh trùng lịch giáo viên**: Nếu một giáo viên có hai lớp học cùng thời điểm, điểm số bị trừ (-15 điểm).

**Tránh trùng lịch nhóm sinh viên**: Nếu một nhóm sinh viên bị xếp vào hai lớp học trùng thời gian, điểm số bị trừ (-10 điểm).

**Đảm bảo giáo viên rảnh vào thời gian giảng dạy**: Nếu giáo viên không rảnh trong thời gian xếp lịch, điểm số bị trừ (-5 điểm).

**Đảm bảo sức chứa của phòng học**: Nếu số lượng sinh viên vượt quá sức chứa của phòng học, điểm số bị trừ (-5 điểm).

### 2.3.2 Yêu cầu mềm (tối ưu hóa)

**Ưu tiên xếp lịch theo thời gian mong muốn của giáo viên**: Nếu lớp học được xếp vào thời gian giáo viên mong muốn, điểm số được cộng (+5 điểm).

Hàm fitness() tính tổng điểm dựa trên các tiêu chí trên để xác định mức độ tối ưu của lịch học.

## **2.4 Quá trình tiến hóa trong GA**

GA bao gồm ba bước chính:

### 2.4.1 Khởi tạo quần thể

Một tập hợp các lịch học ngẫu nhiên được tạo ra bằng phương thức assign\_random\_schedule(), trong đó mỗi lớp học được gán ngẫu nhiên vào một phòng học và một thời gian học.

### 2.4.2 Chọn lọc

Quần thể được sắp xếp theo giá trị fitness, và chỉ các cá thể tốt nhất được giữ lại để sinh ra thế hệ tiếp theo. Trong chương trình, population.sort(key=lambda x: x.fitness(), reverse=True) đảm bảo cá thể có fitness cao nhất được giữ lại.

Thuật toán di truyền sử dụng phương pháp chọn lọc để chọn ra các cá thể tốt nhất từ quần thể hiện tại nhằm tạo ra thế hệ mới có chất lượng cao hơn. Các phương pháp chọn lọc phổ biến bao gồm:

**Chọn lọc theo tỉ lệ thích nghi (Roulette Wheel Selection):** Xác suất chọn một cá thể tỉ lệ thuận với giá trị hàm thích nghi (fitness) của nó.

**Chọn lọc giải đấu (Tournament Selection):** Chọn một nhóm cá thể ngẫu nhiên, sau đó chọn cá thể có fitness cao nhất trong nhóm.

**Chọn lọc theo thứ hạng (Rank Selection):** Xếp hạng tất cả các cá thể theo giá trị fitness và chọn ra các cá thể tốt nhất theo một phân phối xác suất nhất định.

Trong bài toán lập lịch thời khóa biểu, phương pháp **Tournament Selection** thường được sử dụng vì nó giúp duy trì đa dạng quần thể đồng thời đảm bảo rằng các cá thể tốt có xác suất cao hơn được chọn.

### 2.4.3 Lai ghép (Crossover)

Hai cá thể tốt nhất được chọn để tạo ra cá thể con bằng cách kết hợp một phần lịch học của cá thể cha và cá thể mẹ:

point = len(self.classes) // 2

child\_classes = self.classes[:point] + other.classes[point:]

return Schedule(child\_classes, self.rooms, self.teachers, self.time\_slots)

**Lai ghép một điểm (Single-Point Crossover):** Chọn một điểm ngẫu nhiên trên chuỗi nhiễm sắc thể và hoán đổi phần sau của hai cá thể cha mẹ.

**Lai ghép hai điểm (Two-Point Crossover):** Chọn hai điểm ngẫu nhiên và hoán đổi phần giữa hai điểm đó.

**Lai ghép đồng nhất (Uniform Crossover):** Chọn từng gen từ một trong hai cha mẹ với xác suất nhất định.

Ví dụ trong lập lịch thời khóa biểu, mỗi cá thể đại diện cho một lịch học, việc lai ghép có thể diễn ra bằng cách hoán đổi một số lớp học giữa hai cá thể cha mẹ nhằm tạo ra lịch con mới.

### *2.4.4 Đột biến (Mutation)*

Một số cá thể sẽ bị thay đổi ngẫu nhiên với xác suất nhỏ để duy trì sự đa dạng trong quần thể:

if random.random() < 0.1:

random.choice(self.classes).time\_slot = random.choice(self.time\_slots)

Đột biến là quá trình thay đổi ngẫu nhiên một phần của cá thể để tăng tính đa dạng của quần thể và tránh rơi vào cực tiểu cục bộ. Một số phương pháp đột biến phổ biến:

**Đột biến hoán đổi (Swap Mutation):** Hoán đổi vị trí của hai lớp học trong lịch.

**Đột biến đảo ngược (Inverse Mutation):** Chọn một đoạn của chuỗi nhiễm sắc thể và đảo ngược thứ tự của nó.

**Đột biến thay thế (Replacement Mutation):** Thay thế một lớp học bằng một lớp học khác có cùng điều kiện nhưng khác về thời gian hoặc phòng học.

Ví dụ trong bài toán lập lịch, một lớp học có thể được di chuyển từ phòng A sang phòng B, hoặc thay đổi thời gian học để giảm xung đột.

## **2.5 Đánh giá và xuất kết quả**

Sau khi thuật toán GA chạy qua một số thế hệ, cá thể có điểm fitness cao nhất được chọn làm lịch học tốt nhất. Kết quả được vẽ thành biểu đồ để theo dõi quá trình tối ưu hóa:

plt.plot(fitness\_history)

plt.xlabel("Generations")

plt.ylabel("Fitness Score")

plt.title("Optimization Progress")

plt.show()

Cuối cùng, lịch học tối ưu được xuất ra file Excel:

df.to\_excel("schedule.xlsx", index=False, engine='openpyxl')

## **2.6** Phân tích mã nguồn thuật toán GA

Dưới đây là phần mã nguồn thực hiện thuật toán di truyền cho bài toán lập lịch thời khóa biểu:

Mã nguồn trên thực hiện các bước chính của thuật toán GA:

**Khởi tạo lịch học ngẫu nhiên** bằng cách gán phòng học, giáo viên và thời gian ngẫu nhiên cho mỗi lớp.

**Tính toán fitness** bằng cách kiểm tra xung đột phòng học và giáo viên.

**Thực hiện đột biến** bằng cách thay đổi thời gian học của một lớp học ngẫu nhiên.

# Khai báo thư viện

import random

import pandas as pd

import matplotlib.pyplot as plt

import openpyxl

# Lớp mô tả phòng học

class Room:

    def \_\_init\_\_(self, id, capacity, equipment):

        self.id = id

        self.capacity = capacity

        self.equipment = equipment

# Lớp mô tả giáo viên

class Teacher:

    def \_\_init\_\_(self, id, subjects, available\_times, preferred\_times):

        self.id = id

        self.subjects = subjects

        self.available\_times = available\_times  # Danh sách thời gian rảnh

        self.preferred\_times = preferred\_times  # Thời gian mong muốn

# Lớp mô tả nhóm sinh viên

class StudentGroup:

    def \_\_init\_\_(self, id, subjects):

        self.id = id

        self.subjects = subjects

# Lớp mô tả một lớp học cụ thể

class ClassSchedule:

    def \_\_init\_\_(self, subject, teacher, student\_group, duration, equipment):

        self.subject = subject

        self.teacher = teacher

        self.student\_group = student\_group

        self.duration = duration

        self.equipment = equipment

        self.room = None

        self.time\_slot = None

# Lớp mô tả lịch học (cá thể trong thuật toán GA)

class Schedule:

    def \_\_init\_\_(self, classes, rooms, teachers, time\_slots):

        self.classes = classes

        self.rooms = rooms

        self.teachers = teachers

        self.time\_slots = time\_slots

        self.assign\_random\_schedule()

    def assign\_random\_schedule(self):

        for cls in self.classes:

            cls.room = random.choice(self.rooms)

            cls.time\_slot = random.choice(self.time\_slots)

    def fitness(self):

        score = 0

        used\_slots = {}

        teacher\_slots = {}

        student\_slots = {}

        for cls in self.classes:

            key = (cls.room.id, cls.time\_slot)

            teacher\_key = (cls.teacher.id, cls.time\_slot)

            student\_key = (cls.student\_group.id, cls.time\_slot)

            # Kiểm tra trùng phòng học

            if key in used\_slots:

                score -= 10  # Phòng bị trùng lịch

            else:

                used\_slots[key] = True

            # Kiểm tra trùng lịch giáo viên

            if teacher\_key in teacher\_slots:

                score -= 15  # Giáo viên bị trùng lịch

            else:

                teacher\_slots[teacher\_key] = True

            # Kiểm tra trùng lịch sinh viên

            if student\_key in student\_slots:

                score -= 10  # Nhóm sinh viên bị trùng lịch

            else:

                student\_slots[student\_key] = True

            # Kiểm tra giáo viên có rảnh không

            if cls.time\_slot not in cls.teacher.available\_times:

                score -= 5  # Giáo viên không rảnh

            # Ưu tiên xếp lịch theo thời gian mong muốn

            if cls.time\_slot in cls.teacher.preferred\_times:

                score += 5  # Giáo viên được lịch mong muốn

            # Kiểm tra sức chứa phòng học

            if cls.room.capacity < len(cls.student\_group.subjects):

                score -= 5  # Phòng không đủ chỗ

        return score

    def crossover(self, other):

        point = len(self.classes) // 2

        child\_classes = self.classes[:point] + other.classes[point:]

        return Schedule(child\_classes, self.rooms, self.teachers, self.time\_slots)

    def mutate(self):

        if random.random() < 0.1:  # Xác suất đột biến

            random.choice(self.classes).time\_slot = random.choice(self.time\_slots)

# Hàm thực thi thuật toán GA

def genetic\_algorithm(classes, rooms, teachers, time\_slots, generations=100):

    population = [Schedule(classes, rooms, teachers, time\_slots) for \_ in range(10)]

    fitness\_history = []

    for \_ in range(generations):

        population.sort(key=lambda x: x.fitness(), reverse=True)

        fitness\_history.append(population[0].fitness())

        new\_population = [population[0]]  # Giữ cá thể tốt nhất

        while len(new\_population) < 10:

            p1, p2 = random.choices(population[:5], k=2)

            child = p1.crossover(p2)

            child.mutate()

            new\_population.append(child)

        population = new\_population

    plot\_fitness(fitness\_history)

    return population[0]  # Trả về lịch tốt nhất

# Vẽ biểu đồ đánh giá tiến trình tối ưu hóa

def plot\_fitness(fitness\_history):

    plt.plot(fitness\_history)

    plt.xlabel("Generations")

    plt.ylabel("Fitness Score")

    plt.title("Optimization Progress")

    plt.show()

# Dữ liệu giả lập

rooms = [Room("A101", 50, ["Projector"]), Room("B202", 30, ["Whiteboard"])]

teachers = [Teacher("T1", ["Math", "Physics"], ["Monday 8AM", "Tuesday 10AM"], ["Monday 8AM"]),

            Teacher("T2", ["English"], ["Monday 10AM", "Wednesday 8AM"], ["Monday 10AM"])]

students = [StudentGroup("S1", ["Math", "English"]), StudentGroup("S2", ["Physics", "English"])]

classes = [ClassSchedule("Math", teachers[0], students[0], 2, ["Projector"]),

           ClassSchedule("Physics", teachers[0], students[1], 2, []),

           ClassSchedule("English", teachers[1], students[0], 2, ["Whiteboard"])]

time\_slots = ["Monday 8AM", "Monday 10AM", "Tuesday 8AM", "Wednesday 8AM"]

# Chạy thuật toán GA

best\_schedule = genetic\_algorithm(classes, rooms, teachers, time\_slots)

# Xuất kết quả ra file Excel

def export\_to\_excel(schedule):

    data = []

    for cls in schedule.classes:

        data.append([cls.subject, cls.teacher.id, cls.student\_group.id, cls.room.id, cls.time\_slot])

    df = pd.DataFrame(data, columns=["Môn học", "Giáo viên", "Nhóm SV", "Phòng", "Thời gian"])

    df.to\_excel("schedule.xlsx", index=False, engine='openpyxl')

    print("Lịch học đã được xuất ra file schedule.xlsx")

export\_to\_excel(best\_schedule)

**CHƯƠNG 3. PHÂN TÍCH CODE**

## **3.1 Giải thích từng phần code**

### *3.1.1 Khai báo lớp và dữ liệu*

#### Lớp Room - Mô tả phòng học

class Room:

    def \_\_init\_\_(self, id, capacity, equipment):

        self.id = id

        self.capacity = capacity

        self.equipment = equipment

**id**: Mã phòng học (VD: "A101").

**capacity**: Số lượng sinh viên tối đa có thể chứa.

**equipment**: Danh sách thiết bị có trong phòng (VD: "Máy chiếu", "Bảng trắng").

**Vai trò**: Xác định xem lớp học có thể xếp vào phòng này hay không, dựa trên sức chứa và thiết bị yêu cầu.

#### Lớp Teacher - Mô tả giáo viên

class Teacher:

    def \_\_init\_\_(self, id, subjects, available\_times, preferred\_times):

        self.id = id

        self.subjects = subjects

        self.available\_times = available\_times  # Danh sách thời gian rảnh

        self.preferred\_times = preferred\_times  # Thời gian mong muốn

**id**: Mã giáo viên (VD: "GV01").

**subjects**: Danh sách môn giảng dạy.

**available\_times**: Thời gian giáo viên có thể dạy.

**preferred\_times**: Thời gian giáo viên ưu tiên.

**Vai trò**: Đảm bảo lớp học không bị xếp vào thời gian giáo viên không rảnh, và ưu tiên thời gian mong muốn của họ để tối ưu lịch học.

#### Lớp StudentGroup - Mô tả nhóm sinh viên

class StudentGroup:

    def \_\_init\_\_(self, id, subjects):

        self.id = id

        self.subjects = subjects

**id**: Mã nhóm sinh viên (VD: "S1").

**subjects**: Danh sách môn học nhóm đã đăng ký.

**Vai trò**: Giúp xác định môn học nào cần được sắp xếp vào lịch của nhóm này.

#### Lớp ClassSchedule - Mô tả một lớp học cụ thể

class ClassSchedule:

    def \_\_init\_\_(self, subject, teacher, student\_group, duration, equipment):

        self.subject = subject

        self.teacher = teacher

        self.student\_group = student\_group

        self.duration = duration

        self.equipment = equipment

        self.room = None

        self.time\_slot = None

**subject**: Môn học (VD: "Toán").

**teacher**: Giáo viên phụ trách.

**student\_group**: Nhóm sinh viên tham gia.

**duration**: Số tiết học cần thiết.

**equipment**: Thiết bị yêu cầu.

**room**: Phòng học được xếp (ban đầu là None).

**time\_slot**: Thời gian được xếp (ban đầu là None).

**Vai trò**: Đại diện cho một lớp học cụ thể, cần được xếp phòng, giáo viên, sinh viên, và thời gian phù hợp.

## 3.2 Lớp Schedule và thuật toán GA

**Khởi tạo**: Sinh ngẫu nhiên lịch học.

**Fitness Function**:

Tránh trùng phòng học.

Tránh giáo viên bị xếp lịch trùng.

Phòng phải đủ sức chứa.

Khích lệ giáo viên được xếp đúng thời gian mong muốn.

**Crossover**: Kết hợp hai lịch học tốt nhất.

**Mutation**: Ngẫu nhiên thay đổi một số thời gian học.

## 3.3 Cách hoạt động của GA trong bài toán

**Khởi tạo**: Tạo 10 cá thể (lịch học ngẫu nhiên).

**Chấm điểm**: Dựa trên hạn chế.

**Chọn lọc**: Giữ lại cá thể tốt nhất.

**Lai ghép**: Kết hợp giữa hai lịch học tốt nhất.

**Đột biến**: Nhằm tăng đa dạng giải pháp.

## 3.4 Đánh giá hiệu suất, ưu điểm, nhược điểm

**Ưu điểm**:

Tìm giải pháp khá tối ưu trong thời gian hợp lý.

Giải quyết bài toán NP-hard nhanh hơn phương pháp truyền thống.

**Nhược điểm**:

Có thể rơi vào các giải pháp cục bộ.

Tốn tài nguyên tính toán cao.

## 3.5 So sánh với phương pháp khác

**Thuật toán tham lam**: Tìm giải pháp nhanh nhưng không tối ưu.

**Lập trình ràng buộc**: Đạt được giải pháp tốt nhất nhưng tốn nhiều tài nguyên.

**KẾT LUẬN**

Ưu điểm, Nhược điểm và Hướng phát triển của Phân tích Logic AI trong “Thuật toán lập lịch thời khóa biểu”

Ưu điểm

Tự động hóa và tối ưu hóa:

Giảm thiểu công sức và thời gian so với lập lịch thủ công.

Đảm bảo phân bố hợp lý tài nguyên như giáo viên, phòng học, môn học.

Giải quyết ràng buộc phức tạp:

Đáp ứng nhiều điều kiện như số tiết tối đa, thời gian nghỉ, ưu tiên giáo viên.

Tối ưu hóa theo các tiêu chí như công bằng, giảm trùng lặp.

Khả năng mở rộng và tùy chỉnh:

Có thể áp dụng cho nhiều mô hình giáo dục khác nhau (trường học, đại học, đào tạo nội bộ).

Dễ dàng điều chỉnh để phù hợp với quy định riêng của từng tổ chức.

Hiệu suất cao:

Các thuật toán tối ưu (Genetic Algorithm, Constraint Satisfaction Problem, Machine Learning) có thể tìm ra lịch tốt nhất trong thời gian ngắn.

Nhược điểm

Khó khăn trong xử lý ràng buộc phức tạp:

Một số yêu cầu đặc biệt như ưu tiên giáo viên theo cấp bậc hoặc nhóm lớp có thể khó lập trình.

Tốn tài nguyên tính toán:

Các thuật toán tối ưu có thể yêu cầu phần cứng mạnh hoặc thời gian chạy lâu đối với bài toán lớn.

Khó kiểm soát tính linh hoạt:

Khi cần thay đổi lịch đột xuất (giáo viên nghỉ, sự kiện đột xuất), hệ thống có thể mất thời gian để tính toán lại.

Phụ thuộc vào dữ liệu đầu vào:

Nếu dữ liệu đầu vào không chính xác hoặc không đầy đủ, thuật toán có thể tạo ra lịch không hợp lý.

Hướng phát triển

Ứng dụng AI và học máy (Machine Learning):

Dùng AI để phân tích mô hình lịch sử lập lịch và đưa ra gợi ý thông minh.

Học thói quen giảng dạy của giáo viên để tạo lịch cá nhân hóa.

Tích hợp với hệ thống quản lý trường học:

Kết nối với hệ thống quản lý sinh viên (SIS) để tự động cập nhật lịch khi có thay đổi.

Đồng bộ với hệ thống điểm danh, quản lý nhân sự để tối ưu hóa lịch giảng dạy.

Cải thiện khả năng tương tác và điều chỉnh linh hoạt:

Xây dựng giao diện thân thiện giúp giáo viên và quản trị viên dễ dàng thay đổi lịch.

Cho phép điều chỉnh theo thời gian thực khi có phát sinh.

Phát triển thuật toán tối ưu hơn:

Kết hợp nhiều thuật toán (Hybrid Algorithm) để tăng tốc độ và độ chính xác.

Sử dụng thuật toán tiến hóa (Evolutionary Algorithm) để tìm giải pháp tốt nhất.

**DANH MỤC TÀI LIỆU THAM KHẢO**

1. Nguyễn Hồng Sơn (2007), *Giáo trình hệ thống Mạng máy tính CCNA* (Semester 1), NXB Lao động xã hội.
2. Phạm Quốc Hùng (2017), *Đề cương bài giảng Mạng máy tính*, Đại học SPKT Hưng Yên.
3. James F. Kurose and Keith W. Ross (2013), *Computer Networking: A top-down approach sixth Edition*, Pearson Education.