

# MỤC LỤC

<b>LỜI CẢM ƠN</b>	<b>1</b>
<b>DANH SÁCH HÌNH VẼ</b>	<b>2</b>
<b>DANH SÁCH BẢNG</b>	<b>3</b>
<b>I LỊCH LÀM VIỆC TẠI NƠI THỰC TẬP</b>	<b>4</b>
<b>II GIỚI THIỆU VỀ CƠ SỞ THỰC TẬP</b>	<b>6</b>
2.1 Tổng quan Viện Khoa Học và Công Nghệ Ứng Dụng . . . . .	6
2.2 Cơ cấu tổ chức . . . . .	6
2.2.1 Ban Lãnh đạo Viện . . . . .	7
2.2.2 Các đơn vị chức năng của Viện . . . . .	7
2.2.3 Nhân lực tại Viện . . . . .	8
2.3 Lĩnh vực hoạt động . . . . .	8
2.4 Sản phẩm tiêu biểu . . . . .	9
<b>III NỘI DUNG NHIỆM VỤ CHÍNH ĐƯỢC GIAO</b>	<b>11</b>
<b>IV NỘI DUNG CÁC CÔNG VIỆC VÀ KẾT QUẢ ĐẠT ĐƯỢC</b>	<b>14</b>
4.1 Tổng quan về trí tuệ nhân tạo và học sâu . . . . .	14
4.1.1 Khái niệm . . . . .	14
4.1.2 Lịch sử phát triển của AI . . . . .	15
4.1.3 Phân loại trí tuệ nhân tạo . . . . .	21
4.1.4 Ứng dụng của DL và AI . . . . .	24
4.2 Bài toán, mô hình đề xuất và thách thức huấn luyện . . . . .	29
4.2.1 Tổng quan về đề tài . . . . .	29
4.2.2 Cơ sở lý thuyết về mạng nơ-ron tích chập và đề xuất mô hình . . . . .	32
4.2.3 Một số thước đo mô hình phân loại . . . . .	45
4.2.4 Một số vấn đề và thách thức trong quá trình huấn luyện . . . . .	47
4.3 Huấn luyện, đánh giá và triển khai mô hình . . . . .	49
4.3.1 Dữ liệu và quy trình tiền xử lý . . . . .	49
4.3.2 Huấn luyện và đánh giá hiệu suất . . . . .	50

4.3.3	Triển khai giao diện hỗ trợ gán nhãn . . . . .	54
<b>V</b>	<b>KẾT LUẬN VỀ KẾT QUẢ ĐẠT ĐƯỢC QUA ĐỢT THỰC TẬP</b>	<b>57</b>
5.1	Mục tiêu đề ra và kết quả đạt được . . . . .	57
5.2	Kỹ năng và kiến thức đã học được . . . . .	57
5.3	Khó khăn gặp phải . . . . .	57
5.4	Đóng góp của đề tài . . . . .	58
<b>VI</b>	<b>PHẦN PHỤ LỤC</b>	<b>59</b>
6.1	Danh mục từ viết tắt . . . . .	59
6.2	Tài liệu tham khảo . . . . .	60
6.3	Phụ lục code . . . . .	62

## LỜI CẢM ƠN

Lời đầu tiên cho em xin gửi lời cảm ơn đến Ban Giám Hiệu, trường Đại học Công Nghệ Thông Tin và Truyền Thông cùng toàn thể các thầy cô giáo bộ môn, đã tận tình dạy dỗ và chỉ bảo giúp em có được những kiến thức, trang bị tốt nhất cho đợt thực tập nghề nghiệp này.

Em xin chân thành cảm ơn thầy TS. Đinh Xuân Lâm - Viện trưởng, Viện Khoa học và Ứng dụng Công nghệ, thầy TS. Trần Văn Khánh - Viện phó, Viện Khoa học và Ứng dụng Công Nghệ đã giúp đỡ, hướng dẫn, tạo điều kiện thuận lợi để em hoàn thành tốt các nhiệm vụ được giao đồng thời tích lũy được nhiều kinh nghiệm và kiến thức trong quá trình thực tập.

Do thời gian có hạn, trong quá trình nghiên cứu còn gặp nhiều khó khăn, em rất mong nhận được những sự đánh giá, nhận xét và góp ý từ các thầy cô trong hội đồng cùng các bạn sinh viên. Một lần nữa em xin chân thành cảm ơn.

Thái Nguyên, ngày 10 tháng 11 năm 2024

Sinh viên thực hiện

## DANH SÁCH HÌNH VẼ

2.1	Cơ cấu tổ chức của Viện . . . . .	7
4.1	DL và ML trong lĩnh vực AI . . . . .	14
4.2	Dấu mốc lịch sử của AI . . . . .	16
4.3	Một số loại AI phổ biến . . . . .	22
4.4	Ứng dụng của AI trong đời sống xã hội . . . . .	24
4.5	Cấu trúc mạng ANN cơ bản . . . . .	33
4.6	Cấu trúc mạng CNN cơ bản . . . . .	33
4.7	Hình ảnh được biểu diễn dưới dạng toán học . . . . .	34
4.8	Kiến trúc của mô hình MobileNetV2 . . . . .	36
4.9	Quá trình chia nhỏ các phép tích chập của Inception-V3 . . . . .	37
4.10	Quá trình chia nhỏ các phép tích chập của Inception-V3 . . . . .	38
4.11	Tích chập bất đối xứng trong Inception-V3 . . . . .	38
4.12	Bộ phân loại phụ trợ . . . . .	39
4.13	Kiến trúc mô hình Inception-V3 . . . . .	39
4.14	Kiến trúc ResNet101 . . . . .	40
4.15	Kiến trúc mô hình InceptionResNet-V2 . . . . .	42
4.16	Kiến trúc mô hình VGG16 . . . . .	43
4.17	Kiến trúc Xception . . . . .	44
4.18	Ma trận nhầm lẫn . . . . .	46
4.19	Tập dữ liệu ung thư cổ tử cung - Bệnh viện A, Thái Nguyên . . . . .	50
4.20	MobileNet-V2 CF . . . . .	53
4.21	Inception-V3 CF . . . . .	53
4.22	InceptionResNet-V2 CF . . . . .	53
4.23	ResNet101 CF . . . . .	53
4.24	VGG16 CF . . . . .	53
4.25	Xception CF . . . . .	53
4.26	Biểu đồ ma trận nhầm lẫn giữa các mô hình . . . . .	53
4.27	Công cụ Gradio . . . . .	54
4.28	Giao diện người dùng . . . . .	56

## **DANH SÁCH BẢNG**

1.1	Lịch làm việc tại nơi thực tập . . . . .	4
3.1	Lịch trình thực tập và các công việc cụ thể theo tuần . . . . .	11
4.1	Kết quả đánh giá hiệu suất mô hình. . . . .	52
6.1	Bảng danh mục từ viết tắt. . . . .	59

# I. LỊCH LÀM VIỆC TẠI NƠI THỰC TẬP

Bảng 1.1: Lịch làm việc tại nơi thực tập

Tuần	Nội dung công việc	Cán bộ hướng dẫn	Mức độ hoàn thành	Nhận xét của người hướng dẫn
1	Tìm hiểu về cơ cấu tổ chức, lĩnh vực nghiên cứu trọng tâm của Viện. Xây dựng đề cương thực tập	T.S Trần Văn Khánh		
2	Nghiên cứu và thu thập các tài liệu, các nghiên cứu, kết quả đã và đang thực hiện liên quan đến đề tài	T.S Trần Văn Khánh		
3	Phân tích các phương pháp, kỹ thuật tiền xử lý dữ liệu để xử lý dữ liệu cho bài toán. Áp dụng kỹ thuật trên bộ dữ liệu ung thư cổ tử cung	T.S Trần Văn Khánh		
4	Triển khai các mô hình Deep Learning (học sâu) cho tác vụ phân loại ung thư cổ tử cung trên tập dữ liệu	T.S Trần Văn Khánh		
5	Đánh giá kết quả, hiệu suất của các mô hình. Đưa ra các phương hướng để tiếp tục cải thiện hiệu suất của mô hình	T.S Trần Văn Khánh		
6	Tìm hiểu về các công cụ hỗ trợ triển khai nhanh các mô hình học sâu trên nền tảng website	T.S Trần Văn Khánh		

<b>Tuần</b>	<b>Nội dung công việc</b>	<b>Cán bộ hướng dẫn</b>	<b>Mức độ hoàn thành</b>	<b>Nhận xét của người hướng dẫn</b>
7	Xây dựng website gắn nhãn cho hình ảnh tế bào ung thư cổ tử cung. Tích hợp mô hình học sâu hỗ trợ dự báo và phát hiện trong quá trình gắn nhãn	T.S Trần Văn Khánh		
8	Tổng hợp nội dung, kết quả đạt được trong quá trình nghiên cứu, thực nghiệm. Viết báo cáo tổng kết thực tập nghề nghiệp	T.S Trần Văn Khánh		

## **II. GIỚI THIỆU VỀ CƠ SỞ THỰC TẬP**

### **2.1 Tổng quan Viện Khoa Học và Công Nghệ Ứng Dụng**

Trong bối cảnh đất nước đang trên đà đổi mới và phát triển mạnh mẽ, công cuộc công nghiệp hóa và hiện đại hóa được xem là nhiệm vụ hàng đầu, góp phần thúc đẩy sự phát triển kinh tế-xã hội. Các lĩnh vực khoa học, công nghệ và đổi mới sáng tạo không chỉ là những yếu tố quan trọng trong chiến lược phát triển bền vững mà còn đóng vai trò chủ chốt trong việc giải quyết các thách thức toàn cầu. Nhận thức rõ được tầm ảnh hưởng sâu rộng của khoa học và công nghệ đối với đời sống xã hội hiện đại, vào ngày 04/08/2023, trường Đại học Công nghệ Thông tin và Truyền thông Thái Nguyên đã quyết định thành lập Viện Khoa Học và Công Nghệ Ứng Dụng (Institute of Applied Science and Technology - IAST).

Viện Khoa Học và Công Nghệ Ứng Dụng được đặt tại tòa nhà C6.601 của trường, dưới sự lãnh đạo của TS. Đinh Xuân Lâm - Viện trưởng, cùng với TS. Trần Văn Khánh - Viện phó điều hành. Viện được hình thành với sứ mệnh nghiên cứu, phát triển và ứng dụng những công nghệ tiên tiến nhằm nâng cao chất lượng cuộc sống và thúc đẩy sự phát triển bền vững của cộng đồng. Đặc biệt, IAST sẽ chú trọng vào việc kết nối giữa lý thuyết và thực tiễn, tạo ra những sản phẩm công nghệ hữu ích phục vụ cho nhu cầu phát triển của các ngành công nghiệp và xã hội. Với đội ngũ cán bộ khoa học giàu kinh nghiệm và năng lực, Viện Khoa Học và Công Nghệ Ứng Dụng sẽ là cầu nối giữa các nhà nghiên cứu và doanh nghiệp, thúc đẩy chuyển giao công nghệ, hỗ trợ khởi nghiệp và phát triển các dự án nghiên cứu ứng dụng. Thông qua việc hợp tác với các tổ chức trong và ngoài nước, IAST không chỉ nâng cao khả năng nghiên cứu mà còn tăng cường khả năng cạnh tranh cho các sản phẩm công nghệ Việt Nam trên thị trường quốc tế.

Bằng những hoạt động nghiên cứu, đào tạo và chuyển giao công nghệ, Viện Khoa Học và Công Nghệ Ứng Dụng cam kết góp phần quan trọng vào việc phát triển nguồn nhân lực chất lượng cao, thúc đẩy sự đổi mới sáng tạo và phát triển bền vững, từ đó nâng cao vị thế của Việt Nam trên bản đồ khoa học và công nghệ toàn cầu.

### **2.2 Cơ cấu tổ chức**

Viện Khoa Học và Công Nghệ Ứng Dụng được thành lập nhằm thúc đẩy các hoạt động nghiên cứu, phát triển và chuyển giao công nghệ, đóng góp vào sự nghiệp phát triển khoa học và công nghệ tại Việt Nam. Cơ cấu tổ chức của Viện được thiết kế một cách hợp lý và khoa học,



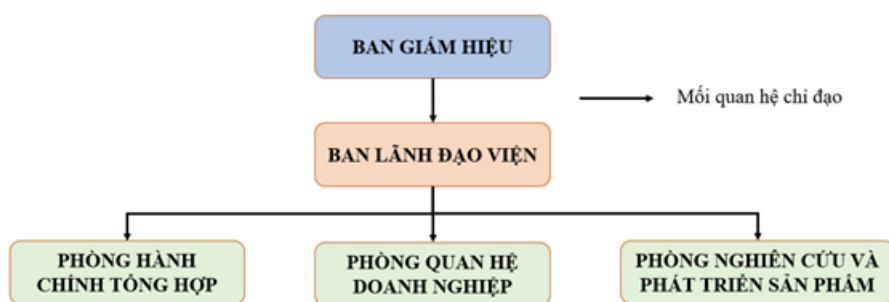
bao gồm ban lãnh đạo Viện và ba phòng chức năng chính, giúp tối ưu hóa hiệu quả công việc và nâng cao khả năng hợp tác giữa các bộ phận.

### 2.2.1 Ban Lãnh đạo Viện

Ban Lãnh đạo của Viện bao gồm 01 Viện trưởng và 02 phó Viện trưởng. Viện trưởng là người đại diện theo pháp luật, chịu trách nhiệm toàn diện về hoạt động của Viện, bao gồm việc thực hiện các quy định của pháp luật và quy định của trường Đại học Công Nghệ Thông Tin và Truyền Thông Thái Nguyên. Nhiệm vụ của Viện trưởng không chỉ giới hạn ở việc xây dựng và ban hành quy chế tổ chức mà còn bao gồm lãnh đạo, chỉ đạo thực hiện các chương trình công tác, kiểm tra việc thực hiện nhiệm vụ của các bộ phận, đồng thời đảm bảo an ninh chính trị và trật tự an toàn trong Viện. Phó Viện trưởng đóng vai trò là trợ lý cho Viện trưởng, giúp điều hành các lĩnh vực công tác được phân công. Họ cũng là cầu nối giữa Viện trưởng và các bộ phận chức năng, đảm bảo rằng tất cả các hoạt động diễn ra một cách suôn sẻ và hiệu quả.

### 2.2.2 Các đơn vị chức năng của Viện

Cơ cấu tổ chức của Viện (hình 2.1) còn bao gồm ba phòng chức năng chuyên môn, mỗi phòng có nhiệm vụ và chức năng riêng, nhằm đáp ứng yêu cầu công việc và thúc đẩy các hoạt động nghiên cứu:



Hình 2.1: Cơ cấu tổ chức của Viện

#### a) Phòng hành chính tổng hợp

Phòng này có nhiệm vụ giúp Viện trưởng triển khai công tác tổ chức, hành chính, kế hoạch và tài chính. Phòng hành chính tổng hợp thực hiện quản lý hồ sơ, theo dõi công tác thi đua khen thưởng, tổ chức tuyển dụng và đào tạo nhân lực cho Viện. Ngoài ra, phòng còn thực hiện các nhiệm vụ liên quan đến công tác văn thư, lưu trữ, và quản lý các tài sản của Viện. Việc xây dựng nội quy và quy chế hoạt động cũng là một phần quan trọng trong chức năng của phòng, nhằm đảm bảo rằng mọi hoạt động diễn ra theo quy định của pháp luật và nội bộ Viện.

### **b) Phòng quan hệ doanh nghiệp**

Phòng này tập trung vào việc xây dựng và phát triển mối quan hệ hợp tác giữa Viện với các cơ quan, tổ chức và doanh nghiệp trong lĩnh vực nghiên cứu khoa học và chuyển giao công nghệ. Nhiệm vụ chính của phòng bao gồm tổ chức các hoạt động hợp tác nghiên cứu, thuyết phục các doanh nghiệp tham gia vào các dự án nghiên cứu ứng dụng, và tìm kiếm các chuyên gia từ các tổ chức và doanh nghiệp để thỉnh giảng tại trường. Phòng cũng chủ động phối hợp với các đơn vị khác trong trường để triển khai các chương trình ươm tạo doanh nghiệp và khởi nghiệp, nhằm phát triển ý tưởng đổi mới sáng tạo trong cộng đồng sinh viên và giảng viên.

### **c) Phòng nghiên cứu và phát triển sản phẩm**

Phòng này chịu trách nhiệm thực hiện các hoạt động nghiên cứu, phát triển và ứng dụng công nghệ trong lĩnh vực công nghệ thông tin và truyền thông. Phòng tập trung vào nghiên cứu cải tiến sản phẩm, quy trình sản xuất, và phát triển các tính năng mới cho sản phẩm. Các hoạt động chính của phòng bao gồm nghiên cứu và phát triển công nghệ, xây dựng kế hoạch sử dụng hiệu quả các phòng thí nghiệm, và thực hiện các nhiệm vụ khác được Viện trưởng giao. Thông qua các hoạt động này, phòng sẽ đóng góp vào việc nâng cao chất lượng sản phẩm và dịch vụ, cũng như tạo ra giá trị gia tăng cho cộng đồng.

## **2.2.3 Nhân lực tại Viện**

Viện Khoa Học và Công Nghệ Ứng Dụng sử dụng đội ngũ viên chức và lao động hợp đồng theo đúng quy định của Nghị định số 08/2014/NĐ-CP và nghị định số 120/2020/NĐ-CP của chính phủ. Số lượng nhân lực được bố trí phù hợp với chức năng, nhiệm vụ của từng bộ phận, đảm bảo rằng mọi công việc đều được thực hiện hiệu quả và chuyên nghiệp. Việc thiết lập một cơ cấu tổ chức rõ ràng và hợp lý không chỉ giúp Viện Khoa Học và Công Nghệ Ứng Dụng hoạt động hiệu quả mà còn tạo điều kiện thuận lợi cho việc thúc đẩy các hoạt động nghiên cứu khoa học, chuyển giao công nghệ và ứng dụng các thành tựu khoa học vào thực tiễn đời sống. Viện cam kết không ngừng nâng cao chất lượng nghiên cứu, phát triển sản phẩm và hợp tác với các tổ chức, doanh nghiệp để góp phần vào sự phát triển bền vững của nền kinh tế và xã hội.

## **2.3 Lĩnh vực hoạt động**

Viện tập trung nghiên cứu khoa học và phát triển công nghệ trong các lĩnh vực mũi nhọn, bao gồm công nghệ thông tin, kỹ thuật và công nghệ số, kinh tế và quản trị số, cũng như truyền thông đa phương tiện. Từ nền tảng tri thức này, Viện cung cấp đa dạng dịch vụ khoa học và công

nghệ, góp phần thúc đẩy các lĩnh vực nói trên thông qua các giải pháp công nghệ tiên tiến và thực tiễn. Bên cạnh đó, Viện tham gia sản xuất, chuyển giao những sản phẩm là thành quả từ các công trình nghiên cứu và phát triển công nghệ, đảm bảo việc ứng dụng hiệu quả vào thực tiễn kinh tế - xã hội. Viện còn thực hiện các chương trình tập huấn nhằm nâng cao năng lực nghiên cứu khoa học, thúc đẩy đổi mới sáng tạo và tinh thần khởi nghiệp. Đồng thời, các chương trình này hỗ trợ chuyển giao công nghệ và tri thức liên quan, qua đó phát triển đội ngũ nhân lực chất lượng cao, có đủ kiến thức và kỹ năng để đáp ứng nhu cầu ngày càng cao của xã hội hiện đại.

## **2.4 Sản phẩm tiêu biểu**

Trong lĩnh vực khoa học, một sản phẩm tiêu biểu của Viện là VR360 của Hồ Ba Bể. Đây là công nghệ nhập vai cho phép người dùng khám phá vẻ đẹp tuyệt vời của một trong những kỳ quan thiên nhiên hùng vĩ nhất của Việt Nam ngay trong ngôi nhà của mình. Sử dụng công nghệ thực tế ảo tiên tiến, người dùng có thể tham gia vào một chuyến tham quan ảo, bao gồm sóng nước trong hồ, rừng cây tươi tốt và những biểu tượng đá vôi. Góc nhìn toàn cảnh 360 độ mang lại trải nghiệm nhập vai, giúp người dùng cảm nhận như họ đang thực sự có mặt tại hồ Ba Bể. Một số tính năng chính của trải nghiệm VR360:

- **Hình ảnh phân giải cao:** Trải nghiệm VR360 tự hào có hình ảnh phân giải cao, tái chế chi tiết phong cảnh của Hồ Ba Bể. Từ những làn sóng nhẹ nhàng trên bề mặt nước đến thảm thực vật xanh tươi của các khu rừng xung quanh, mọi yếu tố đều được tái sinh một cách tỉ mỉ để mang lại trải nghiệm chân thực và tuyệt vời.
- **Khám phá tương tác :** Người dùng có thể điều chỉnh các khu vực khác nhau của Hồ Ba Bể, khám phá các điểm thú vị với tốc độ của riêng họ. Tính tương tác của trải nghiệm VR360 cho phép một tiến trình cá nhân hóa, cho phép người dùng khám phá sâu hơn các cảnh của hồ mà họ thấy thú vị nhất.
- **Thông tin giáo dục :** Bên cạnh khung cảnh tuyệt đẹp, trải nghiệm VR360 cũng cung cấp nội dung giáo dục về tầm quan trọng sinh thái, hình thành địa chất và di sản văn hóa của Hồ Ba Bể. Tính năng này làm phong phú thêm trải nghiệm, mang lại cho người dùng kiến thức quý giá về hồ và môi trường xung quanh.
- **Khả năng tương thích :** Trải nghiệm VR360 được thiết kế để dễ dàng tiếp cận nhiều đối tượng. Dù sử dụng tai nghe VR để có trải nghiệm nhập vai hoàn toàn hay xem nội

dung 360 độ trên màn hình tiêu chuẩn, người dùng có thể thưởng thức vẻ đẹp của Hồ Ba Bể từ bất kỳ thiết bị nào.

Ngoài ra Viện đã và đang phát triển một số sản phẩm liên quan đến Artificial Intelligence - AI (trí tuệ nhân tạo), điển hình trong số đó là:

- Ứng dụng chatbot hỗ trợ người dùng giải đáp các câu hỏi liên quan đến pháp luật.
- Triển khai mô hình Deep Learning - DL (học sâu), hỗ trợ bác sĩ trong việc phát hiện sớm tế bào ung thư cổ tử cung.

Với sự phát triển không ngừng của công nghệ, trong tương lai Viện sẽ còn tận dụng và cung cấp nhiều sản phẩm nhằm đáp ứng được những nhu cầu của đời sống xã hội.

### III. NỘI DUNG NHIỆM VỤ CHÍNH ĐƯỢC GIAO

Bảng 3.1: Lịch trình thực tập và các công việc cụ thể theo tuần

Tuần	Công việc	Chi tiết công việc
1	Xây dựng đề cương thực tập	<ul style="list-style-type: none"> <li>- Tìm hiểu và chọn lựa đề tài nghiên cứu</li> <li>- Xây dựng đề cương chi tiết bám theo đề tài đã được lựa chọn</li> <li>- Lên kế hoạch, nội dung nghiên cứu thực hiện đề tài</li> </ul>
2	Thu thập và nghiên cứu tài liệu	<ul style="list-style-type: none"> <li>- Khảo sát, thu thập các công trình đã và đang nghiên cứu liên quan đến đề tài</li> <li>- Tổng hợp và phân tích các tài liệu liên quan (mô hình học sâu, bộ dữ liệu ung thư cổ tử cung, công cụ gán nhãn, v.v.)</li> </ul>
3	Phân tích, tiền xử lý dữ liệu	<ul style="list-style-type: none"> <li>- Phân tích bộ dữ liệu ung thư cổ tử cung của bệnh viện A, Thái Nguyên</li> <li>- Tiền xử lý dữ liệu: lọc các hình ảnh có kích thước quá lớn, ảnh lỗi</li> <li>- Áp dụng các kỹ thuật tăng cường chất lượng và sự phong phú của tập dữ liệu: làm nét, xoay, cắt, v.v.</li> <li>- Kỹ thuật phân chia dữ liệu nhằm đảm bảo tính nhất quán và đại diện của dữ liệu trong quá trình huấn luyện mô hình</li> </ul>

<b>Tuần</b>	<b>Công việc</b>	<b>Chi tiết công việc</b>
4	Triển khai các mô hình Deep Learning cho bài toán phân loại ung thư cổ tử cung	<ul style="list-style-type: none"> <li>- Điều chỉnh một số mô hình Deep Learning cho tác vụ phân loại ung thư cổ tử cung như: MobileNetV2, InceptionV3, ResNet101, VGG16, Xception.</li> <li>- Thực hiện huấn luyện các mô hình trên bộ dữ liệu</li> <li>- Cài đặt cấu hình, thử nghiệm các tham số cho mô hình và theo dõi các chỉ số hiệu suất của mô hình</li> </ul>
5	Đánh giá hiệu suất của các mô hình	<ul style="list-style-type: none"> <li>- Dựa vào các kết quả huấn luyện mô hình, đưa ra các nhận định về sự hiệu quả của mô hình trên tập dữ liệu, điểm mạnh và điểm yếu</li> <li>- Nghiên cứu đưa ra các giải pháp giúp mô hình cải thiện hơn nữa trên tập dữ liệu và tiếp tục huấn luyện mô hình</li> </ul>
6	Tìm hiểu công cụ triển khai nhanh các mô hình Deep Learning trên nền tảng website	<ul style="list-style-type: none"> <li>- Thực nghiệm một số công cụ cho phép triển khai nhanh các mô hình trí tuệ nhân tạo: Streamlit, Ngrok, Rapid</li> <li>- Triển khai và kiểm thử một vài mô hình học sâu trên các nền tảng này</li> </ul>

<b>Tuần</b>	<b>Công việc</b>	<b>Chi tiết công việc</b>
7	Xây dựng website gán nhãn cho hình ảnh tế bào ung thư cổ tử cung	<ul style="list-style-type: none"> <li>- Sử dụng OpenCV, Python, Rapid triển khai một ứng dụng website gán nhãn hình ảnh đơn giản.</li> <li>- Tích hợp các mô hình học sâu đã được huấn luyện vào website nhằm gợi ý, hỗ trợ đưa ra dự báo giúp giảm thiểu thời gian trong quá trình gán nhãn</li> <li>- Thực nghiệm việc triển khai mô hình theo nhiều cách như: mô hình có thể được tải lên trực tiếp từ một máy vật lý, triển khai mô hình thông qua API</li> </ul>
8	Viết báo cáo tổng kết cho đề tài	<ul style="list-style-type: none"> <li>- Tổng hợp lại các nội dung nghiên cứu, kết quả đã đạt được</li> <li>- Thuyết trình, demo sản phẩm trước Viện về các nội dung, kết quả đã thu thập được</li> <li>- Viết báo cáo tổng kết cho đề tài và đóng gói mã nguồn cho sản phẩm</li> </ul>

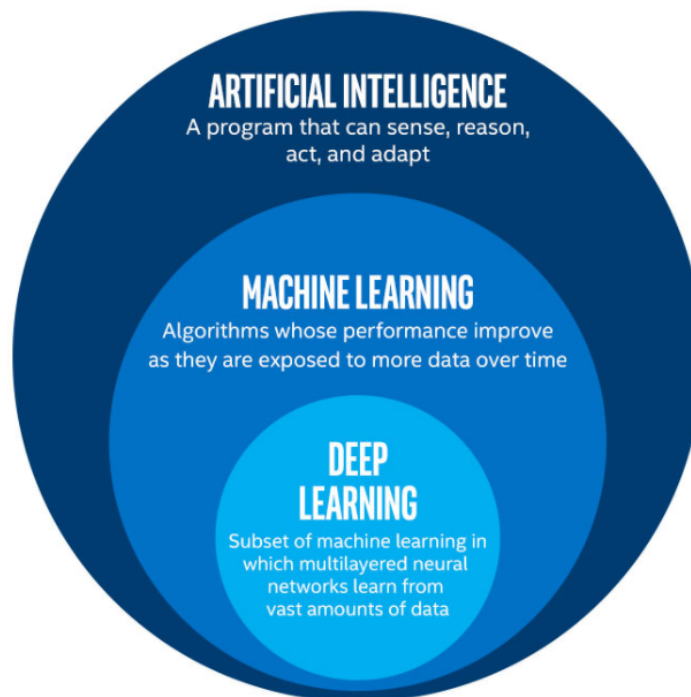
## IV. NỘI DUNG CÁC CÔNG VIỆC VÀ KẾT QUẢ ĐẠT ĐƯỢC

### 4.1 Tổng quan về trí tuệ nhân tạo và học sâu

#### 4.1.1 Khái niệm

Trong thế giới ngày nay, công nghệ đang phát triển rất nhanh chóng và chúng ta đang tiếp xúc với nhiều công nghệ mới khác nhau từng ngày. Ở đây, một trong những công nghệ bùng nổ của khoa học máy tính là trí tuệ nhân tạo, sẵn sàng tạo ra một cuộc cách mạng mới trên thế giới bằng cách tạo ra những cỗ máy thông minh. AI hiện đang ở khắp mọi nơi xung quanh chúng ta. Hiện tại, chúng đang hoạt động ở nhiều lĩnh vực từ nông nghiệp, công nghiệp, y tế, v.v., chẳng hạn như xe tự lái, vẽ tranh, sáng tạo nhạc, tương tác với người dùng, v.v.

AI là một trong những lĩnh vực hấp dẫn và phổ biến của khoa học máy tính, có phạm vi và ảnh hưởng rất lớn trong tương lai. Trí tuệ nhân tạo có xu hướng khiến máy móc hoạt động như con người. AI được tạo thành từ hai từ artificial (nhân tạo) và intelligence (trí thông minh), trong đó artificial có nghĩa là "do con người tạo ra", còn intelligence có nghĩa là "sức mạnh tư duy", do đó AI có nghĩa là "sức mạnh tư duy do con người tạo ra". Trí tuệ nhân tạo tồn tại với mong muốn tạo ra các cỗ máy móc có thể có các kỹ năng giống hệt con người như học tập, lý luận và giải quyết vấn đề.



Hình 4.1: DL và ML trong lĩnh vực AI



Với sự phát triển không ngừng của AI, một nhánh quan trọng và chuyên sâu đã nổi lên, đó là học sâu. Học sâu không chỉ là một phần của AI mà còn là nền tảng giúp AI phát triển mạnh mẽ, đây là một kỹ thuật học máy tiên tiến dựa trên việc học đại diện. Cách tiếp cận mạnh mẽ này cho phép máy móc tự động học các biểu diễn cấp cao từ dữ liệu. Do đó, các mô hình học sâu đạt được các kết quả đáng tin cậy trong các nhiệm vụ đầy thử thách, chẳng hạn như nhận dạng hình ảnh và xử lý ngôn ngữ tự nhiên. So với Machine Learning - ML (học máy), DL (hình 4.1) ra đời nhằm giải quyết những bài toán phức tạp hơn mà các phương pháp ML truyền thống không thể giải quyết được. Các thuật toán DL sử dụng mạng nơ-ron nhân tạo, một hệ thống máy tính học các tính năng cấp cao từ dữ liệu bằng cách tăng độ sâu (tức là số lớp) trong mạng. Mạng lưới thần kinh được lấy cảm hứng, một phần từ mạng lưới thần kinh sinh học, nơi các tế bào trong hầu hết các bộ não (bao gồm cả chúng ta) kết nối và hoạt động cùng nhau. Mỗi tế bào này trong một mạng lưới thần kinh được gọi là tế bào thần kinh.

Mạng nơ-ron học sâu cố gắng bắt chước bộ não con người thông qua sự kết hợp của dữ liệu đầu vào, trọng số và bias (tham số trong quá trình huấn luyện). Các yếu tố này hoạt động cùng nhau để nhận dạng, phân loại và mô tả chính xác các đối tượng trong dữ liệu. Mạng nơ-ron học sâu bao gồm nhiều lớp được kết nối với nhau, mỗi lớp được xây dựng dựa trên lớp trước đó để tinh chỉnh và tối ưu hóa dự đoán hoặc phân loại. Sự tiến triển của các tính toán thông qua mạng được gọi là lan truyền chuyển tiếp. Các lớp đầu vào và đầu ra là nơi mô hình DL thu nạp dữ liệu để xử lý và lớp đầu ra là nơi đưa ra dự đoán hoặc phân loại cuối cùng. Một quá trình khác được gọi là backpropagation (lan truyền ngược) là phương pháp sử dụng các thuật toán như Gradient Descent - GD (giảm dần độ dốc) để tính toán lỗi trong các dự đoán và sau đó điều chỉnh trọng số và độ lệch của hàm bằng cách di chuyển ngược qua các lớp trong quá trình huấn luyện. Cùng với feedforward (lan truyền về phía trước), backpropagation cho phép một mạng lưới thần kinh đưa ra dự đoán và khắc phục bất kỳ sai số nào để mô hình đạt độ chính xác cao nhất.

### **4.1.2 Lịch sử phát triển của AI**

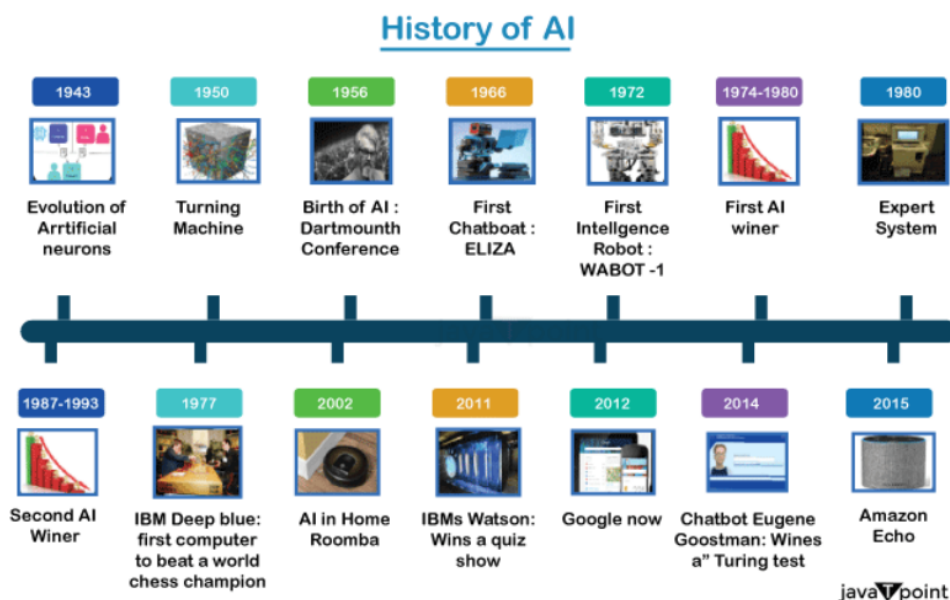
AI không phải là một công nghệ mới đối với các nhà nghiên cứu, công nghệ này đã có từ lâu, trải qua nhiều dốc mốc nổi bật trong lịch sử (hình 4.2), từ những ý tưởng ban đầu cho đến những bước đột phá hiện tại.

#### **a) Giai đoạn 1943-1952**

Từ năm 1943 đến năm 1952, đã có những tiến bộ đáng kể trong việc mở rộng trí tuệ nhân tạo. Trong suốt giai đoạn này, AI đã chuyển đổi từ một khái niệm đơn thuần thành các thí nghiệm

hữu hình và ứng dụng thực tế. Một số sự kiện chính đã xảy ra trong giai đoạn này:

- Năm 1943: Công trình đầu tiên hiện được công nhận là AI được thực hiện bởi Warren McCulloch và Walter Pitts vào năm 1943. Họ đã đề xuất một mô hình tế bào thần kinh nhân tạo.
- Năm 1949: Donald Hebb đã chứng minh một quy tắc cập nhật để sửa đổi cường độ kết nối giữa các tế bào thần kinh, quy tắc của ông hiện được gọi là học tập Hebbian.
- Năm 1950: Alan Turing là một nhà toán học người Anh và là người tiên phong trong lĩnh vực ML vào năm 1950. Alan Turing xuất bản "Máy tính và trí thông minh" trong đó ông đề xuất một bài kiểm tra. Bài kiểm tra có thể đánh giá khả năng của máy móc thể hiện sự thông minh tương đương với vị trí của con người, được gọi là bài kiểm tra Turing.
- Năm 1951: Marvin Minsky và Dean Edmonds đã tạo ra Artificial Neural Network - ANN (mạng trí tuệ nhân tạo) đầu tiên có tên là SNARC. Họ sử dụng 3.000 ống chân không để mô phỏng mạng lưới gồm 40 nơ-ron.



Hình 4.2: Dấu mốc lịch sử của AI

#### b) Giai đoạn 1952-1956

Trong khoảng thời gian từ năm 1952 đến năm 1956, AI bắt đầu nổi lên như một lĩnh vực nghiên cứu độc lập và được nhiều nhà khoa học tiên phong quan tâm. Đây là giai đoạn mà những người tiên phong trong lĩnh vực này, với tầm nhìn xa và tư duy đột phá, đã đặt những viên gạch

đầu tiên để xây dựng nền móng cho AI – một lĩnh vực sau này trở thành một phần không thể thiếu và mang tính cách mạng trong khoa học và công nghệ. Dưới đây là những sự kiện quan trọng đánh dấu sự khởi đầu của AI trong giai đoạn này:

- Năm 1952: Arthur Samuel là người tiên phong trong việc tạo ra chương trình chơi cờ, đánh dấu chương trình tự học chơi trò chơi đầu tiên trên thế giới.
- Năm 1955: Allen Newell và Herbert A. Simon đã tạo ra chương trình trí tuệ nhân tạo có tên là "Logic Theorist". Chương trình này đã chứng minh được 38 trong số 52 định lý toán học và tìm ra những cách chứng minh mới và tinh tế hơn cho một số định lý.
- Năm 1956: Thuật ngữ AI lần đầu tiên được nhà khoa học máy tính người Mỹ John McCarthy sử dụng tại hội nghị Dartmouth. Lần đầu tiên, AI được coi là một lĩnh vực học thuật.

#### **c) Giai đoạn 1956-1974**

Giai đoạn từ năm 1956 đến năm 1974 thường được gọi là "thời đại hoàng kim" của AI. Trong khoảng thời gian này, các nhà nghiên cứu và nhà phát triển AI tràn đầy nhiệt huyết đã đạt được những tiến bộ đáng kể trong lĩnh vực này. Sau đây là một số sự kiện đáng chú ý trong giai đoạn này:

- Năm 1958: Frank Rosenblatt đã giới thiệu perceptron, một trong những mạng nơ-ron nhân tạo đầu tiên có khả năng học từ dữ liệu. Phát minh này đã đặt nền tảng cho các mạng nơ-ron hiện đại. Đồng thời, John McCarthy đã phát triển ngôn ngữ lập trình Lisp, nhanh chóng được cộng đồng AI ưa chuộng và trở nên rất phổ biến trong số các nhà phát triển.
- Năm 1959: Arthur Samuel được ghi nhận là người đầu tiên đưa ra khái niệm "học máy" trong một bài báo quan trọng, trong đó ông đề xuất rằng máy tính có thể được lập trình và có hiệu suất vượt qua những người tạo ra chúng. Ngoài ra, Oliver Selfridge đã có những đóng góp đáng chú ý cho ML với ấn phẩm "Pandemonium A Paradigm for Learning" của ông. Công trình này đã phác thảo một mô hình có khả năng tự cải thiện, cho phép chúng khám phá các mẫu trong các sự kiện hiệu quả hơn.
- Năm 1964: Trong thời gian làm nghiên cứu sinh tiến sĩ tại MIT, Daniel Bobrow đã tạo ra STUDENT, một trong những chương trình đầu tiên về Natural Language Processing

- NLP (xử lý ngôn ngữ tự nhiên), với mục đích cụ thể là để giải quyết các bài toán đại số.
- Năm 1965: Dendral, Edward Feigenbaum, Bruce G. Buchanan, Joshua Lederberg và Carl Djerassi phát minh ra hệ thống chuyên gia hỗ trợ các nhà hóa học hữu cơ trong việc xác định các hợp chất hữu cơ không quen thuộc.
- Năm 1956: Các nhà nghiên cứu nhấn mạnh vào việc phát triển các thuật toán có thể giải quyết các vấn đề toán học. Joseph Weizenbaum đã tạo ra chatbot đầu tiên vào năm 1966, được đặt tên là ELIZA. Hơn nữa, Viện nghiên cứu Stanford đã tạo ra Shakey, robot thông minh di động đầu tiên kết hợp AI, Computer Vision - CV (thị giác máy tính) và NLP. Chúng có thể được coi là tiền thân của xe tự lái và máy bay không người lái ngày nay.
- Năm 1968: Arthur Bryson và Yu-Chi Ho đã phác thảo một thuật toán học tập được gọi là backpropagation, cho phép phát triển các mạng nơ-ron nhân tạo nhiều lớp. Điều này thể hiện một bước tiến đáng kể vượt ra ngoài perceptron và đặt nền tảng cho DL. Ngoài ra, Marvin Minsky và Seymour Papert đã biên soạn cuốn sách "Perceptrons", làm sáng tỏ những hạn chế của các mạng nơ-ron cơ bản. Ấn phẩm này đã dẫn đến sự suy giảm trong nghiên cứu mạng nơ-ron và sự hồi sinh trong nghiên cứu AI.
- Năm 1972: Robot hình người thông minh đầu tiên được chế tạo tại Nhật Bản có tên là WABOT-1.

#### **d) Mùa đông AI đầu tiên 1974-1980**

Mùa đông AI đầu tiên, diễn ra từ năm 1974 đến năm 1980 được biết đến là giai đoạn khó khăn đối AI. Trong thời gian này, nguồn tài trợ nghiên cứu đã giảm đáng kể và AI đối mặt với cảm giác thất vọng.

- Khoảng thời gian từ năm 1974 đến năm 1980 là khoảng thời gian mùa đông AI đầu tiên. Mùa đông AI ám chỉ khoảng thời gian mà các nhà khoa học máy tính phải đối mặt với tình trạng thiếu hụt nghiêm trọng nguồn tài trợ từ chính phủ cho các nghiên cứu AI.
- Trong mùa đông AI, sự quan tâm của công chúng đối với trí tuệ nhân tạo đã giảm xuống.

### **e) Giai đoạn 1980-1987**

Trong khoảng thời gian từ 1980 đến 1987, AI đã trải qua thời kỳ phục hưng và sức sống mới sau kỷ nguyên đầy thách thức của mùa đông AI đầu tiên. Sau đây là những sự kiện đáng chú ý trong khoảng thời gian này:

- Năm 1980: Hội nghị toàn quốc đầu tiên của hiệp hội AI tại Hoa Kỳ được tổ chức tại đại học Stanford.
- Năm 1980: Sau thời kỳ mùa đông của AI, trí tuệ nhân tạo đã trở lại với "Hệ thống chuyên gia". Các hệ thống chuyên gia được lập trình để mô phỏng khả năng ra quyết định của một chuyên gia con người. Ngoài ra, các máy Lisp Symbolics đã được đưa vào sử dụng thương mại, đánh dấu sự khởi đầu của sự hồi sinh của AI. Tuy nhiên, trong những năm tiếp theo, thị trường máy Lisp đã trải qua một sự suy thoái đáng kể.
- Năm 1981: Danny Hillis đã tạo ra các máy tính song song được thiết kế riêng cho AI và nhiều chức năng tính toán khác nhau, có kiến trúc tương tự như GPU hiện đại.
- Năm 1984: Marvin Minsky và Roger Schank đã giới thiệu cụm từ "AI winter" trong một cuộc họp của hiệp hội vì sự tiến bộ của trí tuệ nhân tạo. Họ cảnh báo thế giới kinh doanh rằng kỳ vọng thái quá về AI sẽ dẫn đến sự vỡ mộng và sự sụp đổ cuối cùng của ngành công nghiệp, điều này thực sự đã xảy ra ba năm sau đó.
- Năm 1985: Judea Pearl giới thiệu phương pháp phân tích nhân quả mạng Bayes, trình bày các phương pháp thống kê để mã hóa sự không chắc chắn trong hệ thống máy tính.

### **f) Mùa đông AI thứ hai 1987-1993**

- Khoảng thời gian từ năm 1987 đến năm 1993 là khoảng thời gian AI winter thứ hai.
- Một lần nữa, các nhà đầu tư và chính phủ ngừng tài trợ cho nghiên cứu AI vì chi phí cao nhưng kết quả không hiệu quả.

### **g) Giai đoạn 1993-2011**

Từ năm 1993 đến năm 2011, AI đã có những bước tiến đáng kể, đặc biệt là trong việc phát triển các chương trình máy tính thông minh. Trong thời đại này, các chuyên gia AI đã chuyển trọng tâm từ việc cố gắng bắt kịp trí thông minh của con người sang việc tạo ra phần mềm thực dụng, khéo léo phù hợp với các nhiệm vụ cụ thể. Một số sự kiện đáng chú ý trong khoảng thời gian này:

- Năm 1997: Deep Blue của IBM đã đạt được một cột mốc lịch sử khi đánh bại nhà vô địch cờ vua thế giới Gary Kasparov, đánh dấu lần đầu tiên một máy tính chiến thắng một nhà vô địch cờ vua thế giới đương nhiệm. Hơn nữa, Sepp Hochreiter và Jurgen Schmidhuber đã giới thiệu mạng nơ-ron hồi quy Long Short Term Memory (LSTM), cách mạng hóa khả năng xử lý toàn bộ chuỗi dữ liệu như giọng nói hoặc video.
- Năm 2006: AI xuất hiện trong thế giới kinh doanh cho đến năm 2006. Các công ty như Facebook, Twitter và Netflix cũng bắt đầu sử dụng AI.
- Năm 2009: Rajat Raina, Anand Madhavan và Andrew Ng đã phát hành bài báo có tựa đề "Sử dụng bộ xử lý đồ họa cho học sâu không giám sát mở rộng", giới thiệu khái niệm sử dụng GPU để đào tạo mạng nơ-ron mở rộng.
- Năm 2011: Jurgen Schmidhuber, Dan Claudiu Cire, Ueli Meier và Jonathan Masci đã tạo ra Convolutional Neural Network - CNN (mạng nơ-ron tích chập) đầu tiên đạt được hiệu suất "siêu phàm" khi trở thành người chiến thắng trong cuộc thi nhận dạng biển báo giao thông của Đức. Ngoài ra, Apple đã ra mắt Siri, một trợ lý cá nhân được kích hoạt bằng giọng nói có khả năng tạo ra phản hồi và thực hiện hành động để đáp lại các lệnh thoại.

#### **h) Giai đoạn 2011 đến nay**

Từ năm 2011 đến thời điểm hiện tại, những tiến bộ đáng kể đã diễn ra trong lĩnh vực trí tuệ nhân tạo. Những thành tựu này có thể là nhờ sự kết hợp của DL, ứng dụng dữ liệu mở rộng và quá trình tìm kiếm Artificial General Intelligence - AGI (trí tuệ nhân tạo tổng quát):

- Năm 2011: Watson của IBM đã giành chiến thắng trong Jeopardy, một chương trình đố vui mà chúng phải giải quyết các câu hỏi phức tạp cũng như câu đố. Watson đã chứng minh rằng chúng có thể hiểu ngôn ngữ tự nhiên và có thể giải quyết các câu hỏi khó một cách nhanh chóng.
- Năm 2012: Google ra mắt tính năng ứng dụng Android là "Google Now", có thể cung cấp thông tin cho người dùng dưới dạng dự đoán. Ngoài ra, Geoffrey Hinton, Sutskever và Alex Krizhevsky đã trình bày một cấu trúc CNN sâu giành chiến thắng trong thử thách ImageNet, làm bùng nổ sự phát triển của nghiên cứu và ứng dụng trong lĩnh vực học sâu.

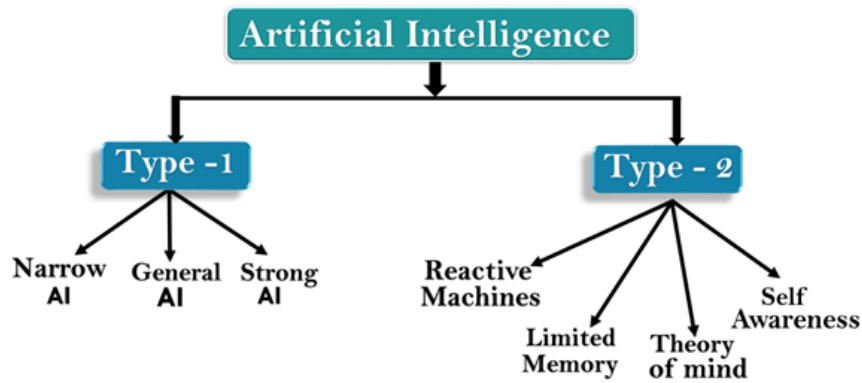
- Năm 2013: Hệ thống Tianhe-2 của Trung Quốc đã đạt được một kỳ tích đáng chú ý khi tăng gấp đôi tốc độ của các siêu máy tính hàng đầu thế giới để đạt 33.86 petaflop. Chúng vẫn giữ được vị thế là hệ thống nhanh nhất thế giới trong ba lần liên tiếp. Hơn nữa DeepMind đã công bố học tăng cường, một CNN có được các kỹ năng thông qua quá trình học lặp đi lặp lại và phần thưởng, cuối cùng vượt qua các chuyên gia con người trong việc chơi trò chơi. Ngoài ra nhà nghiên cứu Tomas Mikolov của Google và nhóm của ông đã giới thiệu Word2vec, một công cụ được thiết kế để tự động phân biệt các kết nối ngữ nghĩa giữa các từ.
- Năm 2014: Chatbot "Eugene Goostman" đã giành chiến thắng trong cuộc thi "Bài kiểm tra Turing" khét tiếng. Trong khi Ian Goodfellow và nhóm của ông tiên phong trong mạng Generative Adversarial Network - GAN (mạng nơ-ron tạo sinh), một loại khuôn khổ học máy được sử dụng để tạo hình ảnh, thay đổi hình ảnh và tạo deepfake, và Diederik Kingma và Max Welling đã giới thiệu bộ mã hóa tự động biến thiên (VAE) để tạo hình ảnh, video và văn bản. Ngoài ra, Facebook đã thiết kế hệ thống nhận dạng khuôn mặt học sâu deepface, có khả năng nhận dạng khuôn mặt người trong hình ảnh kỹ thuật số với độ chính xác gần như tương đương với khả năng của con người.
- Năm 2016: AlphaGo của DeepMind đã giành chiến thắng trước kỳ thủ cờ vây nổi tiếng Lee Sedol tại Seoul, Hàn Quốc, gợi lại trận đấu cờ vua giữa Kasparov và Deep Blue gần hai thập kỷ trước. Trong khi đó, Uber đã khởi xướng một chương trình thí điểm cho xe tự lái ở Pittsburgh, phục vụ cho một nhóm người dùng hạn chế.
- Năm 2022: Vào tháng 11, OpenAI đã ra mắt ChatGPT, cung cấp giao diện hướng trò chuyện cho GPT-3.5 LLM.

Ngoài ra còn có rất nhiều cột mốc và ứng dụng nổi bật khác nữa, hiện nay AI đã phát triển đến một mức độ đáng kinh ngạc. Khái niệm về DL, Big Data (dữ liệu lớn), Data Science (khoa học dữ liệu) vẫn đang là xu hướng bùng nổ. Tương lai của AI sẽ còn rất phát triển và bùng nổ hơn nữa.

#### **4.1.3 Phân loại trí tuệ nhân tạo**

Trí tuệ nhân tạo có thể được chia thành nhiều loại khác nhau, chủ yếu có hai loại chính: dựa trên khả năng và dựa trên chức năng của AI. Hình 4.3 biểu diễn một số loại AI.

##### **a) AI loại 1 - Dựa trên khả năng**



Hình 4.3: Một số loại AI phổ biến

### **Narrow AI (AI yếu và hẹp)**

- AI hẹp là loại AI có khả năng thực hiện một nhiệm vụ chuyên dùng với trí thông minh, phổ biến nhất và hiện có trong thế giới trí tuệ nhân tạo là AI hẹp.
- AI hẹp không thể thực hiện vượt quá phạm vi hoặc giới hạn của chúng vì chỉ được đào tạo cho một nhiệm vụ cụ thể. Do đó, chúng cũng được gọi là AI yếu. AI hẹp có thể thất bại theo những cách không thể đoán trước nếu vượt quá giới hạn.
- Một số ví dụ về AI hẹp là chơi cờ vua, gợi ý mua hàng trên trang thương mại điện tử, xe tự lái, nhận dạng giọng nói và nhận dạng hình ảnh.

### **General AI (AI chung)**

- AI nói chung là một loại trí thông minh có thể thực hiện bất kỳ nhiệm vụ trí tuệ nào một cách hiệu quả như con người.
- Ý tưởng đằng sau AI nói chung là tạo ra một hệ thống có thể thông minh hơn và tự suy nghĩ như con người.
- Hiện tại, chưa có hệ thống nào có thể trở thành AI nói chung và có thể thực hiện bất kỳ nhiệm vụ nào hoàn hảo như con người.
- Các nhà nghiên cứu trên toàn thế giới hiện đang tập trung vào việc phát triển máy móc có AI tổng quát.
- Các hệ thống AI nói chung vẫn đang trong quá trình nghiên cứu và cần rất nhiều sự nỗ lực cũng như thời gian để phát triển các hệ thống như vậy.



## **Strong AI (siêu AI)**

- Siêu AI là một cấp độ trí tuệ của hệ thống mà ở đó máy móc có thể vượt qua trí thông minh của con người và có thể thực hiện bất kỳ nhiệm vụ nào tốt hơn con người.
- Một số đặc điểm chính của AI mạnh bao gồm khả năng suy nghĩ, lý luận, giải quyết câu đố, đưa ra phán đoán, lập kế hoạch, học hỏi và giao tiếp một cách độc lập.
- Siêu AI vẫn là một khái niệm giả định về trí tuệ nhân tạo. Phát triển các hệ thống như vậy trong thực tế vẫn là nhiệm vụ khó khăn.

## **b) AI loại 2 - Dựa trên chức năng**

### **Reactive Machines (Máy móc phản ứng)**

- Máy móc phản ứng thuần túy là loại trí tuệ nhân tạo cơ bản nhất.
- Những hệ thống AI như vậy không lưu trữ ký ức hoặc kinh nghiệm trong quá khứ để thực hiện các hành động trong tương lai.
- Những cỗ máy này chỉ tập trung vào các tình huống hiện tại và phản ứng theo cách tốt nhất có thể.
- AlphaGo của Google cũng là một ví dụ về máy móc phản ứng.

### **Limited Memory (Máy móc có bộ nhớ hạn chế)**

- Máy có bộ nhớ hạn chế có thể lưu trữ những trải nghiệm trong quá khứ hoặc một số dữ liệu trong một khoảng thời gian ngắn.
- Những máy này chỉ có thể sử dụng dữ liệu đã lưu trữ trong một khoảng thời gian giới hạn.
- Xe tự lái là một trong những ví dụ tốt nhất về hệ thống bộ nhớ giới hạn. Những chiếc xe này có thể lưu trữ tốc độ gần đây của những chiếc xe gần đó, khoảng cách của những chiếc xe khác, giới hạn tốc độ và các thông tin khác để điều hướng trên đường.

### **Theory of Mind (Lý thuyết về tri thức)**

- Lý thuyết về tri thức AI tức là phải hiểu được cảm xúc của con người, niềm tin của con người và có thể tương tác xã hội như con người.

- Loại máy AI này vẫn chưa được phát triển, nhưng các nhà nghiên cứu đang nỗ lực và cải tiến rất nhiều để phát triển các máy AI như vậy.

### Soft Awareness (tự nhận thức)

- Tự nhận thức AI là tương lai của trí tuệ nhân tạo. Những cỗ máy này sẽ cực kỳ thông minh và sẽ có ý thức, tình cảm và nhận thức bản thân riêng.
- Những cỗ máy này sẽ thông minh hơn trí óc con người.
- Trí tuệ nhân tạo tự nhận thức vẫn chưa tồn tại trong thực tế và đây chỉ là một khái niệm mang tính giả thuyết.

## 4.1.4 Ứng dụng của DL và AI

Học sâu nói riêng và trí tuệ nhân tạo nói chung có rất nhiều ứng dụng trong xã hội ngày nay (hình 4.4). Chúng đang trở nên thiết yếu trong cuộc sống, vì chúng có thể giải quyết các vấn đề phức tạp một cách hiệu quả trong nhiều ngành như công nghiệp, y tế, giải trí, tài chính, giáo dục, v.v. AI đang làm cho cuộc sống hàng ngày của chúng ta thoải mái và nhanh chóng hơn.



Hình 4.4: Ứng dụng của AI trong đời sống xã hội

### a) AI trong thiên văn học

- Nhận dạng đối tượng thiên thể tự động: Hệ thống AI có thể tự động nhận dạng và phân loại các đối tượng thiên thể trong hình ảnh thiên văn, hỗ trợ khám phá các ngôi sao,

thiên hà và các hiện tượng vũ trụ khác. Nói một cách đơn giản, AI có thể phát hiện và phân loại mọi thứ trong không gian bằng cách xem ảnh.

- Khám phá hành tinh: AI giúp các nhà thiên văn học tìm ra các hành tinh bên ngoài hệ mặt trời của chúng ta bằng cách xem xét nhiều dữ liệu. Chúng có thể nhận thấy những thay đổi nhỏ trong ánh sáng từ các ngôi sao, cho chúng ta biết có thể có các hành tinh xung quanh chúng.
- Phân tích thông tin không gian: AI đóng vai trò quan trọng trong nghiên cứu không gian. Hỗ trợ các nhà khoa học bằng cách kiểm tra cẩn thận lượng lớn dữ liệu phức tạp thu thập được từ các quan sát không gian. Điều này giúp các nhà thiên văn học khám phá ra các mô hình tinh vi, hiện tượng bất thường và các kết nối mà nếu không có AI, có thể rất khó để nhận thấy. Về cơ bản, AI hoạt động như một trợ lý chuyên dụng, sàng lọc qua đồng dữ liệu vũ trụ để tìm ra những cây kim kiến thức có giá trị.
- Theo dõi các sự kiện không gian theo thời gian thực: Các công cụ hỗ trợ AI có thể liên tục theo dõi bầu trời đêm, tìm kiếm các sự kiện bất ngờ như sao nổ hoặc các vụ nổ tia gamma mạnh. Điều này cho phép các nhà khoa học nhanh chóng nghiên cứu các sự kiện này chỉ tiết hơn khi chúng xảy ra.
- Làm cho kính thiên văn thông minh hơn: AI giống như bộ não của kính thiên văn. Giúp chúng hoạt động tốt hơn bằng cách thay đổi cài đặt của chúng khi đang di chuyển. Ví dụ, nếu thời tiết nhiều mây hoặc nếu các nhà khoa học muốn nghiên cứu một thứ cụ thể nào đó trong không gian, AI có thể điều chỉnh kính thiên văn để có được kết quả tốt nhất. Giống như có một chiếc kính thiên văn có thể suy nghĩ và thích ứng với tình huống.

#### **b) AI trong chăm sóc sức khỏe**

- Giúp bác sĩ nhìn rõ hơn bên trong cơ thể: AI giống như một trợ thủ đắc lực cho bác sĩ khi họ xem hình ảnh bên trong cơ thể bệnh nhân, như chụp X-quang hoặc chụp MRI. AI sử dụng các thuật toán thông minh để tìm ra những vấn đề như khối u hoặc xương gãy rất chính xác. Điều này có nghĩa là bác sĩ có thể tìm ra những gì đang diễn ra nhanh hơn và chính xác hơn, điều này rất tốt cho bệnh nhân và để chẩn đoán tốt hơn.
- Phát hiện sớm các vấn đề sức khỏe: AI hoạt động như một bác sĩ sức khỏe. Chúng xem xét thông tin sức khỏe để tìm hiểu xem chúng ta có thể mắc một số bệnh nhất định

trong tương lai hay không. Khi thấy nguy cơ cao, bác sĩ có thể can thiệp sớm để giúp duy trì sức khỏe. Điều này thực sự quan trọng đối với các tình trạng như tiểu đường và các vấn đề về tim vì phát hiện chúng vào thời điểm này có nghĩa là điều trị tốt hơn và ít rắc rối hơn cho bệnh nhân.

- Kế hoạch điều trị cá nhân hóa: AI xem xét thông tin sức khỏe chẳng hạn như gen, những bệnh đã mắc phải trước đây và cách phản ứng với các phương pháp điều trị. Sau đó, chúng lập một kế hoạch đặc biệt dành riêng cho từng người. Điều này có nghĩa là phương pháp điều trị sẽ trở nên hiệu quả hơn và không gây ra nhiều vấn đề.
- Quản lý chức năng và nguồn lực của bệnh viện: AI hoạt động như một người quản lý bệnh viện. Giúp ích cho những việc như khi nào bệnh nhân đến, nơi để các nguồn lực như bác sĩ và vật tư, và cách đảm bảo mọi thứ diễn ra tốt đẹp. Thậm chí có thể đoán trước được có bao nhiêu bệnh nhân có thể đến, do đó bệnh viện sử dụng nhân viên và nguồn lực của mình theo cách tốt nhất có thể.

#### **c) AI trong trò chơi**

- Nhân vật trò chơi thông minh: AI giống như bộ não đằng sau các nhân vật trò chơi không do người chơi điều khiển. Họ làm cho những nhân vật này, hành động giống người thật hoặc kẻ thù thông minh hơn. Họ có thể học hỏi từ những gì người chơi làm và thay đổi hành vi của họ, điều này làm cho trò chơi trở nên thú vị và sống động hơn.
- Tạo thế giới trò chơi bằng AI: AI có thể tự tạo ra các phần của trò chơi điện tử. Chúng có thể tạo ra các cấp độ, bản đồ và địa điểm để khám phá mà không cần con người phải tự tay tạo ra. Điều này có nghĩa là trò chơi có thể có thế giới lớn hơn và thú vị hơn vì AI thực hiện rất nhiều công việc, giống như một người xây dựng thế giới trò chơi.
- Làm cho trò chơi có cảm giác chân thực: AI giúp làm cho trò chơi hoạt động giống với thế giới thực hơn. Chúng tạo ra đồ họa trông giống hệt những thứ chúng ta nhìn thấy và chúng làm cho cách mọi thứ di chuyển trong trò chơi có cảm giác chân thực, giống như trong cuộc sống thực. Chúng thậm chí còn đoán người chơi có thể làm gì tiếp theo để trò chơi trông mượt mà và tự nhiên.

#### **d) AI trong tài chính**

- Nhận dạng và phòng ngừa gian lận: AI luôn theo dõi các giao dịch ngân hàng. Khi chúng thấy điều gì đó đáng ngờ, chúng sẽ báo động và giúp ngân hàng ngăn chặn

những kẻ xấu ăn cắp tiền. Điều này xảy ra rất nhanh, mà không cần mọi người phải kiểm tra mọi giao dịch.

- **Giao dịch tự động:** AI giúp các nhà giao dịch làm việc tự động. AI sử dụng nhiều thuật toán khác nhau để nhanh chóng mua và bán cổ phiếu trong khi phân tích tất cả thông tin thị trường. Điều này thúc đẩy các chiến lược giao dịch, giúp đầu tư hiệu quả.
- **Kiểm soát rủi ro:** AI giúp kiểm tra nhiều dữ liệu để kiểm tra mức độ rủi ro của một thứ gì đó, chẳng hạn như cho vay hoặc đầu tư. Chúng xem xét những thứ như liệu ai đó có thể trả nợ hay mức độ an toàn của khoản đầu tư. Điều này giúp các ngân hàng và công ty đầu tư đưa ra những lựa chọn thông minh hơn để họ không bị mất tiền và có thể giúp những người khác tiết kiệm và tăng tiền của họ.

#### **e) AI trong bảo mật dữ liệu**

- **Dự đoán mối đe dọa:** AI xem xét các rắc rối trong quá khứ và theo dõi các mối nguy hiểm mới đang xuất hiện. Bằng cách này, AI có thể dự đoán những điều tồi tệ có thể xảy ra trong tương lai, như vi phạm bảo mật hoặc tấn công mạng. Theo cách này, các công ty có thể chuẩn bị trước để bảo vệ dữ liệu quan trọng của mình, giống như việc xây dựng một pháo đài vững chắc trước bất kỳ cuộc tấn công nào xảy ra.
- **Phản hồi an toàn tự động:** AI hoạt động như một người bảo vệ kỹ thuật số có thể phản hồi khi có sự cố. Nếu thấy có điều gì đó không ổn xảy ra, chẳng hạn như một cuộc tấn công mạng, AI có thể tự động hành động. Chúng hoàn toàn có thể cô lập phần đang bị tấn công. Theo cách này, AI giữ an toàn cho các kho dữ liệu quan trọng trong thế giới kỹ thuật số.

#### **f) AI trong truyền thông xã hội**

- **Gợi ý thông minh:** AI đóng vai trò như một hướng dẫn viên trên phương tiện truyền thông xã hội. Chúng theo dõi những gì người dùng thích và làm, sau đó gợi ý những nội dung tương tự, như bài đăng, video hoặc quảng cáo.
- **Trợ lý ảo và chatbot:** Chatbot AI và trợ lý ảo hoạt động như những người trợ giúp kỹ thuật số trên phương tiện truyền thông xã hội. Chúng phản hồi nhanh và có thể trò chuyện với con người. Chúng trả lời các câu hỏi, chia sẻ thông tin và thậm chí giúp giải quyết các vấn đề.

- **Phân tích tình cảm:** AI có thể tìm ra cảm xúc của mọi người trên mạng xã hội. Chúng xem xét tất cả các bình luận và bài đăng và quyết định xem đó là loại thông điệp vui vẻ, buồn hay yêu thích. Điều này giúp các công ty hiểu được mọi người nghĩ gì để họ có thể đưa ra các phương án theo cách đúng đắn. Giống như có một thước đo tâm trạng cho internet để các doanh nghiệp có thể làm cho khách hàng của họ hài lòng hơn.
- **Phân tích xu hướng:** AI theo dõi tất cả các cuộc trò chuyện và những gì đang phổ biến hiện nay. Điều này giúp các công ty và người dùng bình thường hiểu được mọi người đang nghĩ gì và nói gì. AI hoạt động như một phóng viên tin tức truyền thông xã hội giúp khách hàng luôn cập nhật những gì đang thịnh hành và mọi người đang bàn tán về điều gì.

#### **g) AI trong nông nghiệp**

- **Nông nghiệp tự động:** AI điều khiển một số máy móc như máy kéo và máy bay không người lái. Những máy móc này có thể tự gieo hạt, nhổ cỏ dại và phun thuốc lên cây trồng. Chúng làm rất tốt và chính xác theo nhu cầu.
- **Nông nghiệp thông minh:** AI giúp nông nghiệp trở nên siêu hiệu quả. Giúp nông dân sử dụng đúng lượng phân bón và thuốc trừ sâu, không quá nhiều và không quá ít. Điều này có nghĩa là ít chất thải hơn và cây trồng phát triển tốt hơn.

#### **h) AI trong giáo dục**

- **Sáng tạo nội dung giáo dục:** AI đóng vai trò là trợ lý giảng dạy cho các nhà giáo dục. Giúp họ tạo ra những thứ như bài kiểm tra, kế hoạch bài học và tài liệu học tập. Điều này giúp việc giảng dạy dễ dàng và tốt hơn vì các giáo viên có nhiều thời gian hơn cho học sinh và tài liệu chất lượng hơn.
- **Trợ lý học tập ảo** là những AI sẵn sàng giải đáp mọi câu hỏi, giải thích kiến thức và hỗ trợ học sinh bất kỳ lúc nào, dù ngày hay đêm. Điều này giúp việc học trở nên dễ dàng và thú vị hơn, vì học sinh luôn có "người đồng hành" để nhờ giúp đỡ khi gặp khó khăn. Bên cạnh đó, AI còn giảm bớt áp lực cho giáo viên, khi có thể xử lý các câu hỏi phổ biến, từ đó cho phép giáo viên dành nhiều thời gian hơn cho việc giảng dạy cá nhân hóa.

- Lộ trình học tập tùy chỉnh: AI tìm ra điểm mạnh và điểm yếu của học sinh, sau đó AI cung cấp cho học sinh những kiến thức, nội dung phù hợp để học và cách học tốt nhất. Điều này giúp việc học trở nên dễ dàng và thú vị hơn.

## 4.2 Bài toán, mô hình đề xuất và thách thức huấn luyện

### 4.2.1 Tổng quan về đề tài

#### a) Giới thiệu

Ung thư cổ tử cung là một trong những loại ung thư phổ biến nhất ở phụ nữ trên toàn cầu và là nguyên nhân hàng đầu gây tử vong ở nhiều quốc gia, đặc biệt xảy ra tại các khu vực có điều kiện y tế hạn chế. Việc phát hiện sớm và điều trị kịp thời có thể giảm thiểu đáng kể nguy cơ tử vong, giúp nâng cao chất lượng sống và kéo dài tuổi thọ cho người bệnh. Tuy nhiên, các quy trình chẩn đoán hình ảnh và xét nghiệm hiện tại đòi hỏi sự can thiệp của các chuyên gia và mất nhiều thời gian để gán nhãn, xử lý và phân tích hình ảnh bệnh lý.

Đề tài "Xây dựng mô-đun gán nhãn dữ liệu tích hợp hỗ trợ của mô hình AI" nhằm hỗ trợ các chuyên gia y tế trong việc gán nhãn các hình ảnh y tế liên quan đến ung thư cổ tử cung một cách nhanh chóng và chính xác hơn. Với sự hỗ trợ của mô hình AI, mô-đun sẽ có khả năng tự động phân tích và nhận diện các đặc điểm đáng ngờ trong hình ảnh, giúp đẩy nhanh quá trình sàng lọc, đồng thời đảm bảo độ chính xác trong việc xác định các dấu hiệu bệnh lý.

#### b) Mục tiêu

Mục tiêu của đề tài tập trung vào một số khía cạnh sau:

- Tự động hóa quy trình gán nhãn: Giảm bớt thời gian và công sức cho các bác sĩ chuyên gia trong việc phân tích hình ảnh y tế.
- Nâng cao hiệu suất chẩn đoán: Sử dụng AI để cải thiện độ chính xác trong phát hiện các đặc điểm bệnh lý liên quan đến ung thư cổ tử cung.
- Xây dựng công cụ trực quan: Tích hợp mô hình AI vào một giao diện đơn giản, dễ sử dụng, tạo điều kiện thuận lợi cho các bác sĩ, chuyên gia có thể tương tác, gán nhãn và đánh giá hình ảnh một cách hiệu quả.

#### c) Tính cấp thiết

Với tốc độ phát triển không ngừng của công nghệ AI, việc tích hợp các mô hình AI vào quy trình gán nhãn và chẩn đoán ung thư cổ tử cung có tiềm năng lớn trong việc giảm tải công việc

cho các chuyên gia y tế. Không chỉ giúp tăng tốc độ xử lý hình ảnh, mà AI còn giúp nâng cao độ chính xác trong phát hiện sớm các tổn thương tiền ung thư. Đặc biệt, các công cụ AI có thể tạo ra những giải pháp hỗ trợ từ xa, giúp các khu vực không có đủ bác sĩ chuyên môn vẫn có thể tiến hành sàng lọc ban đầu.

Việc lựa chọn đề tài "Xây dựng mô-đun gán nhãn dữ liệu tích hợp hỗ trợ của mô hình AI" trong bối cảnh này là cực kỳ cấp thiết, nhằm tạo ra công cụ hỗ trợ các bác sĩ và nhân viên y tế trong việc chẩn đoán và gán nhãn các hình ảnh liên quan đến ung thư cổ tử cung một cách hiệu quả và chính xác hơn. Việc áp dụng AI không chỉ làm tăng hiệu quả của quy trình chẩn đoán mà còn mở ra tiềm năng lớn trong việc cung cấp các dịch vụ y tế chất lượng cho các khu vực khó khăn, giảm thiểu tỷ lệ tử vong và nâng cao chất lượng sống cho bệnh nhân.

#### **d) Phạm vi và giới hạn**

Phạm vi của đề tài bao gồm việc xây dựng một mô-đun gán nhãn dữ liệu tích hợp mô hình AI để hỗ trợ trong quá trình chẩn đoán ung thư cổ tử cung. Mô-đun này sẽ tập trung vào các nhiệm vụ chính như: gán nhãn tự động các hình ảnh y tế liên quan, phát hiện và phân loại các đặc điểm bệnh lý có khả năng chỉ ra dấu hiệu của ung thư cổ tử cung. Đề tài tập trung vào dữ liệu hình ảnh từ các phương pháp chẩn đoán phổ biến là xét nghiệm tế bào, nhằm tạo ra một công cụ hữu ích cho các chuyên gia trong lĩnh vực y tế. Giới hạn của đề tài bao gồm các yếu tố sau:

- **Dữ liệu hạn chế:** Mô-đun chỉ được triển khai và thử nghiệm trên các tập dữ liệu nhất định liên quan đến ung thư cổ tử cung, do vậy không thể khái quát hết tất cả các dạng biểu hiện của bệnh. Việc gán nhãn dữ liệu dựa trên các hình ảnh sẵn có, không thể áp dụng cho các loại hình ảnh khác ngoài phạm vi này.
- **Giới hạn về độ chính xác:** Mô hình AI phụ thuộc vào chất lượng và khối lượng dữ liệu huấn luyện. Do đó, mô-đun có thể chưa đạt được độ chính xác cao trong mọi trường hợp và có thể yêu cầu hiệu chỉnh thêm khi triển khai trên các tập dữ liệu mới hoặc trong các môi trường thực tế khác nhau.
- **Không thể thay thế hoàn toàn chuyên gia y tế:** Mô-đun AI chỉ đóng vai trò hỗ trợ gán nhãn và sàng lọc ban đầu, không thay thế được quá trình đánh giá và chẩn đoán cuối cùng của các chuyên gia y tế. Các quyết định lâm sàng cuối cùng vẫn cần được thực hiện bởi các bác sĩ có chuyên môn.
- **Hạn chế và khả năng mở rộng:** Do mô hình AI được huấn luyện cho một loại ung thư



cụ thể là ung thư cổ tử cung, nên việc áp dụng mô-đun này cho các loại ung thư khác hoặc các bệnh lý khác có thể không hiệu quả và yêu cầu tái huấn luyện mô hình.

#### **e) Cách tiếp cận và công nghệ sử dụng**

##### **Cách tiếp cận dựa trên mô hình học sâu**

- Sử dụng các mô hình DL trong việc xử lý hình ảnh, đặc biệt là các mô hình CNN để phân tích và nhận diện các đặc điểm bất thường trên ảnh y tế.
- Các mô hình như MobileNetV2, InceptionV3, InceptionResNetv2, ResNet101, Xception và VGG16 được sử dụng để tự động phát hiện các hình ảnh có khả năng chứa dấu hiệu bệnh lý.

##### **Tiền xử lý và gán nhãn dữ liệu**

- Tiền xử lý hình ảnh bao gồm các kỹ thuật như chuẩn hóa, tăng cường dữ liệu, nhằm cải thiện chất lượng dữ liệu, giúp mô hình học sâu đạt hiệu suất cao hơn trong giai đoạn huấn luyện.
- Sử dụng phương pháp gán nhãn bán tự động, trong đó mô hình AI hỗ trợ gợi ý các hình ảnh có khả năng bị tổn thương và các chuyên gia sẽ xác nhận hoặc điều chỉnh lại nhãn. Cách tiếp cận này giúp tiết kiệm thời gian và tăng độ chính xác của quá trình gán nhãn.

##### **Sử dụng thư viện và nền tảng hỗ trợ AI**

- Keras và Tensorflow: Đây là hai thư viện học sâu mạnh mẽ, cung cấp các công cụ cần thiết cho việc xây dựng, huấn luyện và triển khai các mô hình AI. Tensorflow và Keras hỗ trợ các mô hình CNN và dễ dàng tích hợp vào ứng dụng.
- OpenCV: Thư viện xử lý ảnh được sử dụng để thao tác với các hình ảnh y tế, thực hiện các bước tiền xử lý, phân tích và gán nhãn. OpenCV giúp cải thiện khả năng tương tác và tăng tốc độ xử lý của mô-đun.

##### **Triển khai mô-đun và phát triển giao diện người dùng**

- Sử dụng công cụ Gradio để xây dựng một giao diện người dùng cơ bản, gắn gửi giúp các chuyên gia y tế có thể tương tác trực tiếp với mô-đun gán nhãn, kiểm tra và chỉnh sửa các nhãn được AI gợi ý.

- Giao diện cho phép dễ dàng tải lên các hình ảnh y tế, hiển thị kết quả phân tích từ AI và thực hiện điều chỉnh cần thiết trước khi lưu lại.

### **Đánh giá và tối ưu hóa mô hình**

- Áp dụng các kỹ thuật đánh giá độ chính xác của mô hình bao gồm các chỉ số F1-score, recall, precision, accuracy để đảm bảo mô hình có khả năng phát hiện chính xác các dấu hiệu bệnh lý.
- Tối ưu hóa mô hình bằng cách điều chỉnh tham số và áp dụng các phương pháp giảm thiểu nhiễu, giúp mô hình đạt hiệu suất cao trong điều kiện thực tế.

## **4.2.2 Cơ sở lý thuyết về mạng nơ-ron tích chập và đề xuất mô hình**

### **a) Mạng nơ-ron tích chập**

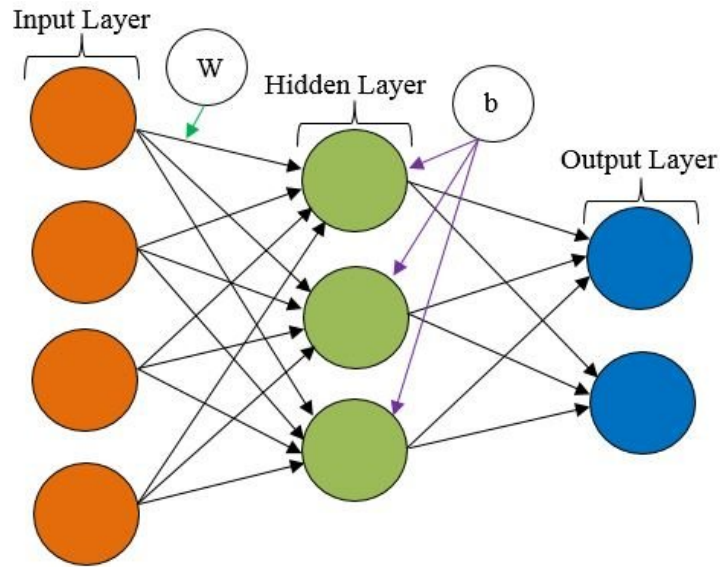
#### **Khái niệm**

Mạng nơ-ron ra đời đã thúc đẩy quá trình phát triển của ngành CV. Hiện tại có rất nhiều các kiến trúc mạng nơ-ron khác nhau và các kiến trúc mới vẫn đang tiếp tục được khám phá ngày qua ngày. Nhưng ít ai biết rằng đằng sau những khám phá đó là một tiến trình khoa học lâu dài và bền bỉ trong gần 20 năm. Với sự kết hợp đồng bộ của phát triển kiến trúc mạng, khả năng tính toán của máy tính và các phương pháp tối ưu hóa. Mạng nơ-ron tích chập là một loại kiến trúc mạng sâu thường được sử dụng trong thị giác máy tính. CNN được sử dụng trong nhiều tập dữ liệu khác nhau như hình ảnh, âm thanh và văn bản. Các loại mạng CNN khác nhau được sử dụng cho các mục đích khác nhau. Tích chập là một khái niệm trong xử lý tín hiệu số nhằm biến đổi thông tin đầu vào thông qua một phép tích chập với bộ lọc để trả về đầu ra là một tín hiệu mới. Tín hiệu này sẽ làm giảm những đặc trưng mà bộ lọc không quan tâm và chỉ giữ lại những đặc trưng chính.

#### **Cấu trúc chung của mạng nơ-ron tích chập**

Trong một ANN có ba loại lớp chính được mô tả trong hình 4.5:

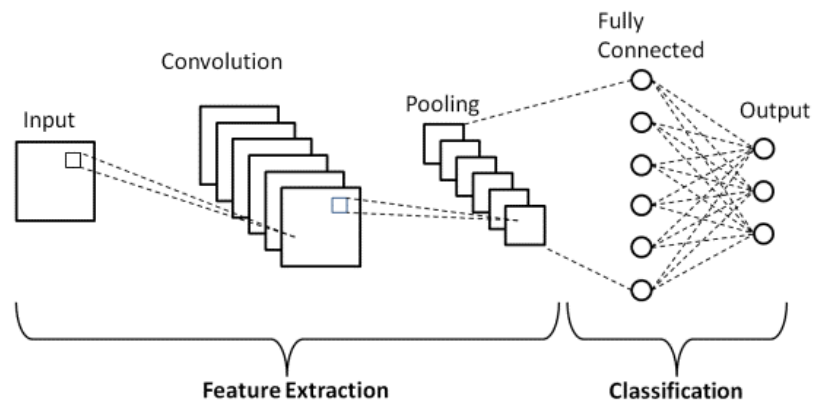
- Input Layer (lớp đầu vào): Đây là lớp mà chúng ta đưa dữ liệu đầu vào trong mô hình. Số lượng nơ-ron của lớp này bằng với tổng số đặc điểm trong dữ liệu của chúng ta (số pixel trong hình ảnh).
- Hidden Layer (lớp ẩn): Dữ liệu từ lớp đầu vào được đưa vào các lớp ẩn. Có thể có nhiều lớp ẩn tùy thuộc vào mô hình và kích thước dữ liệu của chúng ta. Mỗi lớp ẩn có thể



Hình 4.5: Cấu trúc mạng ANN cơ bản

có số lượng nơ-ron khác nhau, thường lớn hơn số lượng các đặc trưng. Đầu ra từ mỗi lớp được tính bằng cách nhân ma trận của đầu ra của lớp trước đó với các trọng số có thể học được của lớp đó và sau đó thêm các độ lệch theo sau là các hàm kích hoạt giúp mạng trở nên phi tuyến tính.

- Output Layer (lớp đầu ra): Đầu ra từ lớp ẩn sau đó được đưa vào một hàm Logistic như Sigmoid hoặc Softmax để chuyển đổi đầu ra của mỗi lớp thành xác suất của từng lớp.



Hình 4.6: Cấu trúc mạng CNN cơ bản

Dữ liệu được đưa vào mô hình và đầu ra từ mỗi lớp được lấy từ bước trước đó được gọi là feedforward, sau đó chúng ta tính toán lỗi bằng hàm lỗi, một số hàm lỗi phổ biến là cross-entropy, squared loss error (lỗi mất mát bình phương), v.v. Sau đó chúng ta truyền ngược vào mô hình bằng cách tính đạo hàm. Bước này gọi là backpropagation, về cơ bản cách này giúp

giảm thiểu sai số của mô hình trong giai đoạn huấn luyện. Mạng CNN (hình 4.6) là phiên bản mở rộng của ANN chủ yếu được dùng để trích xuất đặc trưng từ tập dữ liệu ma trận dạng lưới. CNN bao gồm nhiều lớp như lớp đầu vào, lớp tích chập, lớp gộp và các lớp kết nối đầy đủ.

Lớp tích chập (Convolutional Layer) áp dụng các bộ lọc cho hình ảnh đầu vào để trích xuất các đặc điểm, lớp gộp (Pooling Layer) lấy các mẫu hình ảnh quan trọng để giảm tính toán sau đó sử dụng lớp kết nối đầy đủ (Fully Connected) để đưa ra dự đoán cuối cùng. Mạng học các bộ lọc và tối ưu thông qua Backpropagation và Gradient Descent. Hình ảnh trong thực tế được biểu diễn dưới dạng toán học, là một khối hộp có chiều dài, chiều rộng và chiều cao (là số kênh màu của hình ảnh - thông thường là màu đỏ, xanh lá, xanh lam) kí hiệu là RGB (hình 4.7).



Hình 4.7: Hình ảnh được biểu diễn dưới dạng toán học

Một kiến trúc CNN hoàn chỉnh cũng có thể được gọi là convnets. Convnets là một chuỗi các lớp và mỗi lớp biến đổi một khối lượng đầu vào thành một đầu ra khác thông qua một hàm khả vi. Giả sử với hình ảnh có kích thước là  $32 \times 32 \times 3$ , quá trình tích chập được mô tả như sau:

- **Input Layer:** Đây là lớp mà chúng ta đưa dữ liệu đầu vào cho mô hình của mình. Trong CNN nói chung, dữ liệu đầu vào sẽ là một hình ảnh hoặc một chuỗi hình ảnh. Lớp này chứa dữ liệu đầu vào với chiều rộng là 32, chiều cao là 32 và chiều sâu là 3.
- **Convolutional Layer:** Đây là lớp được sử dụng để trích xuất đặc điểm từ tập dữ liệu đầu vào. Chúng áp dụng một tập hợp các bộ lọc có thể học được gọi là kernel cho các hình ảnh đầu vào. Các bộ lọc (kernel) là các ma trận nhỏ hơn thường có hình dạng  $2 \times 2$ ,  $3 \times 3$  hoặc  $5 \times 5$ . Chúng trượt trên dữ liệu hình ảnh đầu vào và tính tích vô hướng giữa các giá trị trong bộ lọc và pixel trong các vùng ảnh tương ứng. Ví dụ ta sử dụng 12 bộ lọc cho lớp này, chúng ta sẽ thu được đầu ra có kích thước  $32 \times 32 \times 12$ .

- **Activation Layer (lớp kích hoạt):** Bằng cách thêm một hàm kích hoạt vào đầu ra của lớp trước đó, các lớp kích hoạt sẽ làm cho mạng có tính chất phi tuyến tính. Chúng áp dụng một hàm kích hoạt cho từng phần tử ở đầu ra của lớp tích chập.
- **Pooling Layer (lớp gộp):** Lớp này được chèn vào với chức năng chính là làm giảm kích thước của đặc trưng, giúp tính toán nhanh hơn, giảm bộ nhớ và cũng ngăn ngừa tình trạng Overfitting (quá khớp). Hai loại lớp Pooling phổ biến nhất là Max Pooling và Average Pooling. Nếu sử dụng Max Pooling với bộ lọc là  $2 \times 2$  và bước nhảy là 2, ta thu được kết quả có kích thước là  $16 \times 16 \times 12$ .
- **Flatten Layer (lớp làm phẳng):** Các bản đồ đặc trưng sẽ được dàn phẳng thành một vector một chiều để di chuyển vào Fully Connected Layer.
- **Fully Connected:** Lấy dữ liệu từ Flatten Layers sau đó tính toán phù hợp cho các tác vụ phân loại và hồi quy.
- **Output Layer:** Đầu ra từ Fully Connected được đưa vào một hàm Logistic như Sigmoid hoặc Softmax, hàm này chuyển đổi đầu ra của mỗi lớp thành điểm xác suất của từng lớp.

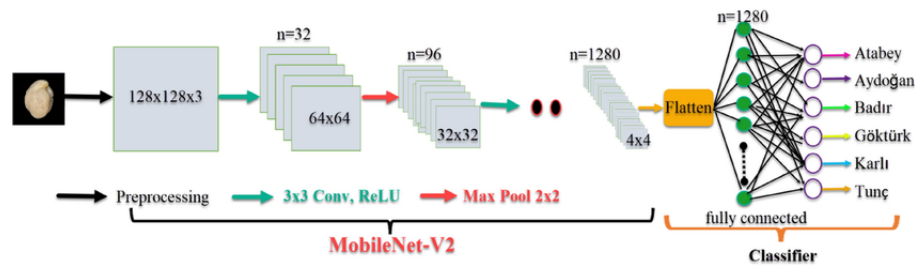
## **b. Mô hình đề xuất**

Trong nghiên cứu này, 6 mô hình mạng nơ-ron tích chập phổ biến được sử dụng là MobileNet-V2, Inception-V3, InceptionResNet-V2, ResNet101, VGG16 và Xception. Mỗi mô hình này đã được phát triển và tối ưu hóa với các kiến trúc và đặc điểm riêng biệt nhằm cải thiện khả năng trích xuất đặc trưng và hiệu quả tính toán trong xử lý ảnh.

### **MobileNet-V2**

MobileNet-V2 là một mạng nơ-ron tích chập nhẹ, được thiết kế riêng cho các ứng dụng thị giác di động và nhúng. Các nhà nghiên cứu của Google đã phát triển chúng như một sự cải tiến so với mô hình MobileNet ban đầu. Một khía cạnh đáng chú ý khác của mô hình này là khả năng tạo ra sự cân bằng tốt giữa kích thước mô hình và độ chính xác, khiến chúng trở nên lý tưởng cho các thiết bị có hạn chế về tài nguyên.

Kiến trúc của MobileNet-V2 (hình 4.8) kết hợp một số tính năng chính góp phần vào hiệu quả và hiệu suất của chúng trong các tác vụ phân loại hình ảnh. Các tính năng này bao gồm tích chập có thể tách theo chiều sâu, tích chập đảo chiều, nút cổ chai, nút cổ chai tuyến tính, khối nén và kích hoạt. Mỗi tính năng này đóng vai trò quan trọng trong việc giảm độ phức tạp tính



Hình 4.8: Kiến trúc của mô hình MobileNetV2

toán của mô hình trong khi vẫn duy trì độ chính xác cao. Việc sử dụng MobileNet-V2 để phân loại hình ảnh mang lại một số lợi thế:

- Đầu tiên, kiến trúc nhẹ cho phép triển khai hiệu quả trên các thiết bị di động và những có tài nguyên hạn hẹp.
- Thứ hai, kiến trúc MobileNetV2 đạt được độ chính xác cạnh tranh so với các mô hình lớn hơn và tốn kém hơn về mặt tính toán.
- Cuối cùng, kích thước nhỏ của mô hình cho phép thời gian suy luận nhanh hơn, phù hợp với các ứng dụng thời gian thực.

Kiến trúc của MobileNet-V2 bao gồm một loạt các lớp tích chập, tiếp theo là các tích chập có thể tách theo chiều sâu, các phần tích chập đảo chiều, thiết kế nút thắt, các khối nén và kích hoạt. Các thành phần này hoạt động cùng nhau để giảm số lượng tham số và phép tính cần thiết trong khi vẫn duy trì khả năng nắm bắt các tính năng phức tạp của mô hình.

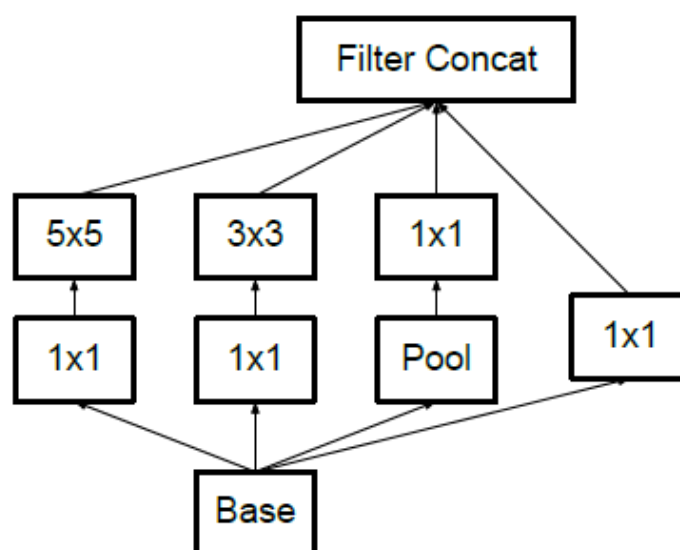
- Tích chập có thể tách theo chiều sâu: Tích chập phân tách theo chiều sâu là một kỹ thuật được sử dụng trong MobileNet-V2 để giảm chi phí tính toán của tích chập. Chúng tách tích chập chuẩn thành hai phép toán riêng biệt: tích chập theo chiều sâu và tích chập theo điểm. Sự tách biệt này làm giảm đáng kể số phép tính cần thiết, giúp mô hình hiệu quả hơn.
- Inverted Residuals (tích chập đảo chiều): Là một thành phần chính của kiến trúc, giúp cải thiện độ chính xác của mô hình. Chúng tạo ra các cấu trúc nút thắt để mở rộng số kênh trước khi áp dụng các phép tích chập để tách theo chiều sâu. Sự mở rộng này cho phép mô hình nắm bắt các tính năng phức tạp hơn và tăng cường sức mạnh biểu diễn của chúng.
- Bottleneck Design (nút thắt cổ chai): Làm giảm thêm chi phí tính toán bằng cách sử dụng phép tích chập 1x1 để giảm số kênh trước khi áp dụng phép tích chập tách biệt

theo chiều sâu. Lựa chọn thiết kế này giúp duy trì sự cân bằng tốt giữa kích thước mô hình và độ chính xác.

- Linear Bottleneck (nút thắt tuyến tính): Giải quyết vấn đề mất thông tin trong quá trình nút thắt bằng cách sử dụng các kích hoạt tuyến tính thay vì các kích hoạt phi tuyến tính, mô hình bảo toàn nhiều thông tin hơn và cải thiện khả năng nắm bắt các chi tiết từ dữ liệu.
- Squeeze-and-Excitation (khối nén và kích hoạt): Các khối nén và hàm kích hoạt được thêm vào nhằm hiệu chỉnh lại các phản hồi tính năng theo kênh một cách phù hợp, cho phép mô hình tập trung vào các tính năng nhiều thông tin hơn và loại bỏ các tính năng ít liên quan hơn.

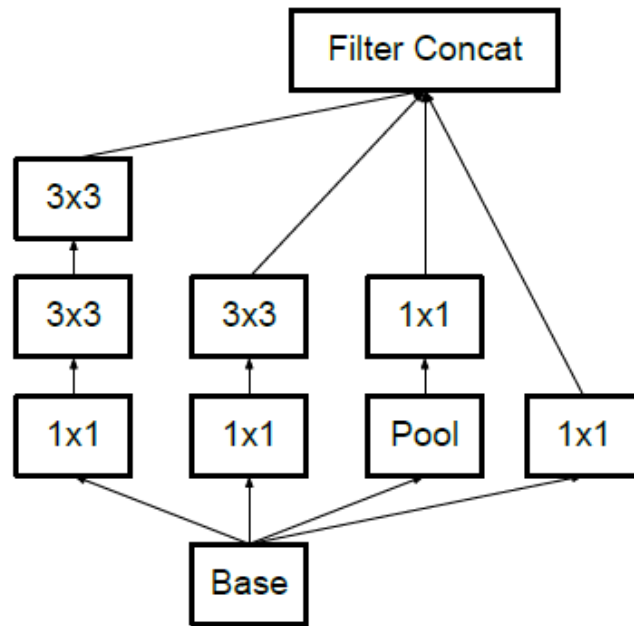
### Inception-V3

Mô hình Inception-V3 được phát hành vào năm 2015, có tổng cộng 42 lớp và tỷ lệ lỗi thấp hơn so với các phiên bản trước đó. Một số điểm nổi bật của mô hình Inception-V3 là:



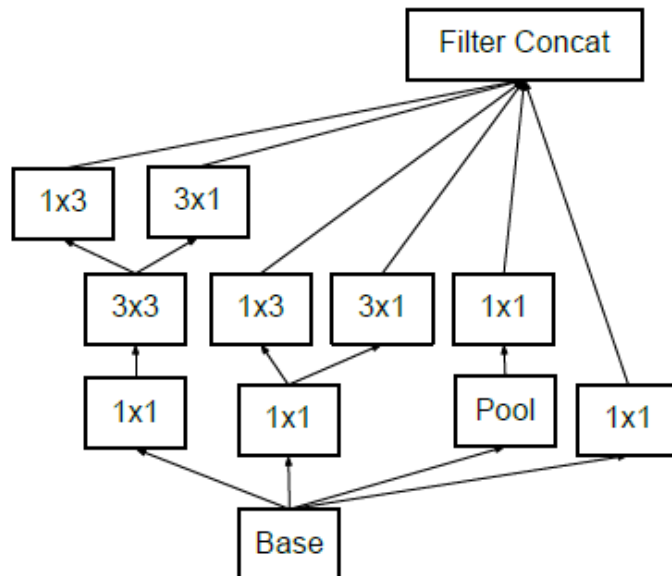
Hình 4.9: Quá trình chia nhỏ các phép tích chập của Inception-V3

- Factorization into Smaller Convolutions (phân tích thành các tích chập nhỏ): Đây là một trong những lợi thế chính của mô hình giúp giảm kích thước tích chập, các phép tích chập được phân tích thành các phép tích chập nhỏ hơn nữa. Do số lượng tham số giảm nên chi phí tính toán cũng giảm. Giả sử có một lớp tích chập 5x5 như hình 4.9. Để giảm chi phí tính toán, lớp tích chập 5x5 được thay thế bằng hai lớp tích chập là 3x3 như hình 4.10.



Hình 4.10: Quá trình chia nhỏ các phép tích chập của Inception-V3

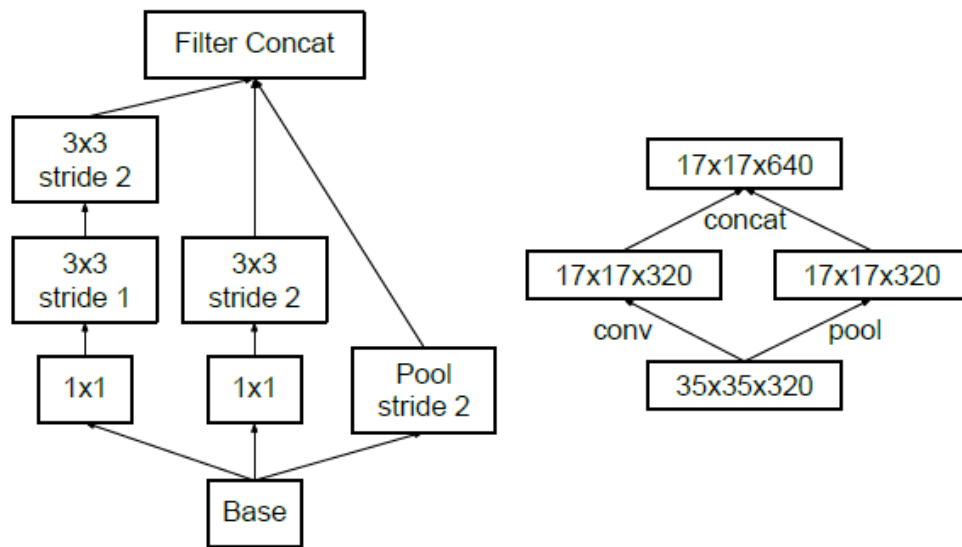
- Spatial Factorization into Asymmetric Convolutions (tích chập bất đối xứng): Mặc dù các phép tích chập lớn hơn đã được phân tích thành các phép tích chập nhỏ hơn, nhưng ta vẫn mong muốn phân tích nhỏ hơn nữa. Một giải pháp để làm được điều đó là sử dụng tích chập bất đối xứng. Tích chập bất đối xứng có dạng  $n \times 1$ . Ở đây họ đã thay thế lớp tích chập  $3 \times 3$  bằng phép tích chập  $1 \times 3$  và theo sau là  $3 \times 1$ . Sau khi áp dụng hai kỹ thuật tối ưu ban đầu, cấu trúc tích chập hình 4.11 có dạng:



Hình 4.11: Tích chập bất đối xứng trong Inception-V3

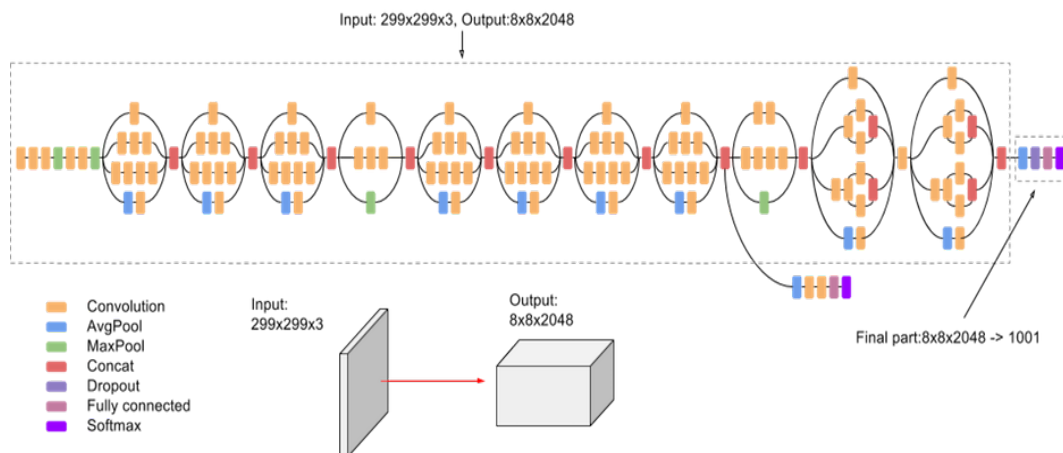


- Utility of Auxiliary Classifiers (bộ phân loại phụ trợ): Mục tiêu của việc sử dụng bộ phân loại phụ trợ là cải thiện sự hội tụ của các mạng nơ-ron sâu. Chủ yếu được sử dụng để giải quyết các vấn đề về hiện tượng mất gradient trong mô hình. Các bộ phân loại hoạt động như một bộ điều chỉnh trong kiến trúc mô hình.
- Efficient Grid Size Reduction (giảm kích thước lưới): Để giảm kích thước, Inception-V3 đã tăng kích thước kích hoạt của bộ lọc (hình 4.12). Điều này được thực hiện bằng cách sử dụng hai khối tích chập song song và sau đó ghép lại.



Hình 4.12: Bộ phân loại phụ trợ

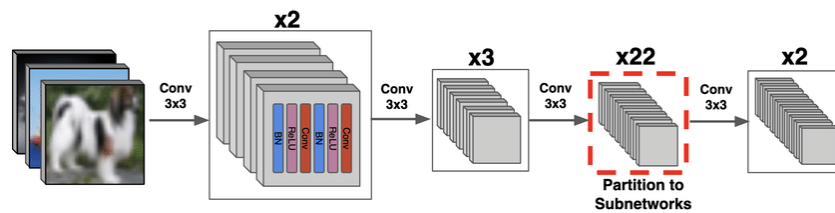
Sau khi thực hiện tất cả các tối ưu hóa, mô hình Inception-V3 có cấu trúc tổng quan như hình 4.13.



Hình 4.13: Kiến trúc mô hình Inception-V3

## ResNet101

ResNet101 (hình 4.14) là một trong những mạng nơ-ron sâu thuộc họ Residual Networks (ResNet), nổi tiếng với khả năng huấn luyện các mạng rất sâu mà không gặp vấn đề mất mát gradient. ResNet được giới thiệu bởi nhóm nghiên cứu của Microsoft trong bài báo "Deep Residual Learning for Image Recognition" và đã đạt thành tích cao trong nhiều cuộc thi về CV. Kiến trúc ResNet101 có 101 lớp (layers) và được xây dựng dựa trên cơ chế kết nối dư (Residual Connections), giúp duy trì hiệu suất tốt khi độ sâu của mạng tăng lên. Kiến trúc tổng quan của ResNet101 bao gồm các thành phần như:



Hình 4.14: Kiến trúc ResNet101

- Convolutional Layer: Thực hiện phép tích chập để trích xuất các đặc trưng từ ảnh đầu vào. Một lớp tích chập kích thước 7x7 với 64 kênh, bước nhảy là 2.
- Residual Block: Đây là yếu tố chủ chốt của ResNet, giúp truyền tải thông tin giữa các lớp để giảm thiểu sự mất mát gradient. Có bốn giai đoạn trong khối Residual, giai đoạn đầu tiên có 3 khối Residual mỗi khối có 3 lớp tích chập, giai đoạn thứ hai gồm 4 khối Residual mỗi khối có 3 lớp tích chập, giai đoạn thứ ba gồm 23 khối Residual mỗi khối có 3 lớp tích chập, ở giai đoạn cuối cùng gồm 3 khối Residual mỗi khối có 3 lớp tích chập.
- Pooling Layer: Giảm kích thước của các đặc trưng, giúp giảm số lượng tham số và tăng tốc độ tính toán. Sau các khối Residual, một lớp Pooling trung bình được áp dụng để giảm kích thước xuống 1x1.
- Fully Connected: Kết nối các đặc trưng đã học từ các lớp trước và đưa ra dự đoán cuối cùng.

Trong các mạng nơ-ron truyền thống, khi độ sâu của mạng tăng lên, việc truyền tải thông tin từ các lớp đầu vào đến các lớp cuối cùng trở nên khó khăn, gây ra vấn đề Vanishing Gradient (mất mát gradient). Để giải quyết vấn đề này, ResNet đã giới thiệu Residual Connections. Mỗi

khối Residual thực hiện phép biến đổi thông tin từ đầu vào  $x$  thành đầu ra  $F(x)$  để tạo thành đầu ra cuối cùng của khối  $y = F(x) + x$ . Điều này giúp bảo toàn các đặc trưng từ các lớp trước đó và cho phép mạng huấn luyện các lớp sâu mà không gặp phải tình trạng mất mát thông tin. Với các kiến trúc sâu như ResNet101, nhóm nghiên cứu đã giới thiệu khối Bottleneck để giảm số lượng tính toán nhưng vẫn giữ lại được thông tin cần thiết. Khối Bottleneck bao gồm ba lớp tích chập: tích chập  $1 \times 1$ , tích chập  $3 \times 3$  và tích chập  $1 \times 1$ .

ResNet101 là một mô hình CNN mạnh mẽ, phổ biến và đáng tin cậy trong các bài toán thị giác máy tính. Với cơ chế Residual, ResNet101 đã khắc phục được những hạn chế của các mạng nơ-ron sâu truyền thống, mở ra khả năng huấn luyện các mạng nơ-ron rất sâu mà không gặp vấn đề mất mát gradient.

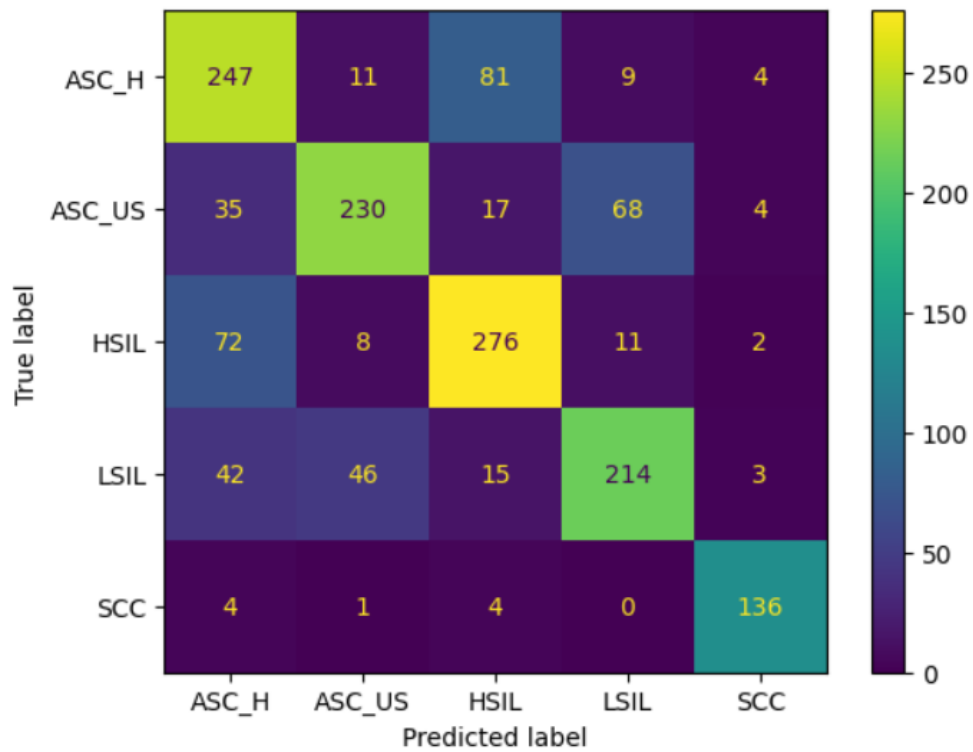
### **InceptionResNet-V2**

InceptionResNet-V2 (hình 4.15) là một trong những mạng nơ-ron tích chập tiên tiến, kết hợp các ý tưởng từ kiến trúc mạng Inception và ResNet để tối ưu hóa hiệu quả và độ chính xác trong các tác vụ nhận diện ảnh. Google phát triển mạng này với mục tiêu tăng cường hiệu suất và độ chính xác của các mô hình Inception trước đó (như InceptionV3 và V4) bằng cách tích hợp các khối Residual của ResNet. Kiến trúc tổng quan của InceptionResNet-V2 gồm ba phần chính:

- Stem (phần gốc): Khởi tạo mạng với các lớp tích chập cơ bản, tương tự các kiến trúc CNN khác nhưng được điều chỉnh để phù hợp với các khối Inception và Residual.
- InceptionResNet Blocks: Thành phần chính của mạng, gồm nhiều khối InceptionResNet được sắp xếp thành ba nhóm chính: InceptionResNet-A, InceptionResNet-B và InceptionResNet-C. Mỗi khối có cấu trúc Inception được kết hợp với lớp Residual để giảm thiểu vấn đề GD và tăng khả năng học đặc trưng.
- Reduction Blocks: Các khối giảm chiều giúp giảm độ phân giải của các đặc trưng trong mạng mà không làm mất thông tin quan trọng.

Stem là tầng đầu tiên của mạng, nơi tiếp nhận các ảnh đầu vào và thực hiện các phép biến đổi cơ bản (chủ yếu là Convolutional và Pooling) để chuyển ảnh thành các đặc trưng cơ bản trước khi đưa vào các khối InceptionResNet. InceptionResNet-V2 bao gồm ba loại khối chính, mỗi khối thực hiện các phép tích chập và tổng hợp các thông tin từ nhiều bộ lọc với các kích thước khác nhau.

- InceptionResNet-A: Tích hợp các phép tích chập với kích thước  $1 \times 1$  và  $3 \times 3$ , sau đó



Hình 4.15: Kiến trúc mô hình InceptionResNet-V2

tổng hợp thông tin và sử dụng kết nối Residual (lớp nối tắt) để cải thiện độ chính xác của đặc trưng.

- InceptionResNet-B: Sử dụng các tích chập  $1 \times 1$  và  $7 \times 1$ ,  $1 \times 7$  để mở rộng phạm vi thông tin thu nhận, giúp mạng dễ dàng nhận dạng các mẫu lớn hơn trong ảnh.
- InceptionResNet-C: Dùng các tích chập  $1 \times 1$  và  $3 \times 3$  để tăng cường khả năng trích xuất đặc trưng.

Giống các kiến trúc Inception khác, InceptionResNet-V2 cũng sử dụng các khối giảm chiều (Reduction) để giảm độ phân giải của đặc trưng nhưng làm tăng số lượng kênh. Mạng có hai khối giảm chiều chính:

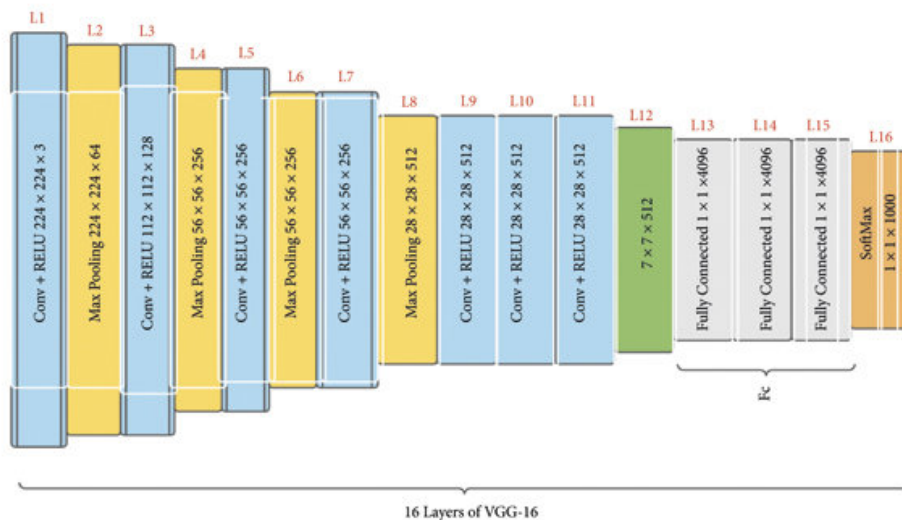
- Reduction-A: Giảm độ phân giải từ  $35 \times 35$  xuống  $17 \times 17$ .
- Reduction-B: Giảm độ phân giải từ  $17 \times 17$  xuống  $8 \times 8$ .

InceptionResNet-V2 ứng dụng cơ chế Residual từ ResNet, giúp khắc phục vấn đề mất gradient và tránh Overfitting, đặc biệt khi mạng sâu hơn. Các khối InceptionResNet có một lớp Shortcut Connection nối thẳng đầu vào đến đầu ra thông qua một phép cộng. Nhờ đó, mạng

vẫn giữ lại được các đặc trưng quan trọng từ lớp trước, giúp tối ưu hiệu suất học. Inception-ResNetV2 là một kiến trúc CNN tối ưu cho các bài toán phức tạp về thị giác máy tính. Việc kết hợp Inception và Residual giúp cải thiện đáng kể hiệu suất mô hình so với các kiến trúc trước, đặc biệt khi làm việc trên các tập dữ liệu lớn và đa dạng.

## VGG16

VGG16 (hình 4.16) là một trong những kiến trúc mạng nơ-ron tích chập (CNN) nổi tiếng, được phát triển bởi nhóm nghiên cứu tại đại học Oxford và được giới thiệu lần đầu trong cuộc thi ImageNet Large Scale Visual Recognition Challenge (ILSVRC) năm 2014. Mạng này đặc biệt được chú ý nhờ kiến trúc đơn giản nhưng hiệu quả, với độ sâu 16 lớp, trong đó hầu hết các lớp tích chập đều có kích thước nhỏ, chỉ là  $3 \times 3$ . Cấu trúc của VGG16 giúp mô hình có khả năng học được các đặc trưng phức tạp mà không cần các lớp tích chập lớn, tập trung vào việc sử dụng nhiều lớp với các bộ lọc nhỏ để tăng độ sâu.



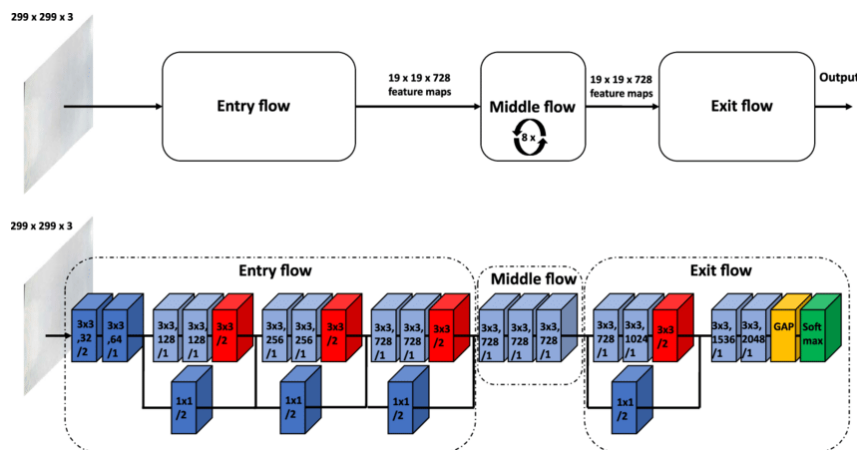
Hình 4.16: Kiến trúc mô hình VGG16

Mỗi lớp tích chập trong VGG16 có kích thước bộ lọc là  $3 \times 3$  và stride là 1, nhằm mục đích thu được nhiều đặc trưng mà không làm mất quá nhiều thông tin không gian của hình ảnh. Lý do chọn kích thước này là để giữ cho số lượng tham số thấp nhưng vẫn đảm bảo mạng có khả năng nắm bắt các đặc trưng nhỏ. Lớp Max Pooling (gộp cực đại) có kích thước  $2 \times 2$  với stride là 2, được đặt sau mỗi hai hoặc ba lớp tích chập để giảm kích thước không gian của đặc trưng, giúp mạng trích xuất các đặc trưng quan trọng đồng thời giảm số lượng tham số và chi phí tính toán. VGG16 có ba lớp kết nối đầy đủ: Lớp thứ nhất và thứ hai có 4096 đơn vị (nơ-ron). Lớp cuối cùng là lớp phân loại, có số lượng đơn vị bằng số lượng lớp cần phân loại, tùy thuộc vào

tác vụ đầu ra. Trong các ứng dụng phân loại, lớp đầu ra thường sử dụng hàm kích hoạt Softmax để dự đoán xác suất cho từng lớp. VGG16 là một trong những kiến trúc CNN tiêu chuẩn, đã và đang được sử dụng trong nhiều ứng dụng về thị giác máy tính. Kiến trúc đơn giản nhưng hiệu quả của VGG16 đã giúp mô hình này trở thành lựa chọn phổ biến cho các bài toán liên quan đến ảnh, đặc biệt là khi có nguồn tài nguyên tính toán mạnh.

## Xception

Xception (Extreme Inception) là một kiến trúc mạng nơ-ron tích chập (CNN) được phát triển bởi François Chollet vào năm 2017 (hình 4.17), nhằm cải tiến kiến trúc Inception của Google. Xception dựa trên ý tưởng của Inception nhưng sử dụng một kỹ thuật mới gọi là tích chập chiều sâu phân tách (Depthwise Separable Convolution), giúp tăng hiệu suất và độ chính xác của mô hình. Xception có một vài đặc điểm nổi bật:



Hình 4.17: Kiến trúc Xception

- Sử dụng tích chập chiều sâu phân tách: Đây là điểm cốt lõi của Xception, giúp giảm số lượng tham số và chi phí tính toán so với tích chập thông thường.
- Tối ưu hóa luồng dữ liệu (flow): Kiến trúc Xception tách việc trích xuất đặc trưng và kết hợp đặc trưng, giúp tăng khả năng học được các đặc trưng phức tạp.
- Độ sâu lớn và nhiều lớp: Xception có nhiều lớp hơn so với các kiến trúc truyền thống, đặc biệt là trong việc sử dụng các khối tích chập để trích xuất đặc trưng.

Xception có tổng cộng 36 lớp tích chập được chia thành 3 khối lớn:

- Entry Flow (dòng vào): Phần đầu của mạng, nơi xử lý các đặc trưng cơ bản từ ảnh đầu vào. Entry Flow bắt đầu với một số lớp tích chập thường để học các đặc trưng cơ bản.

Sau đó, nó sử dụng tích chập chiều sâu phân tách với các bộ lọc lớn hơn, giúp mô hình học được các đặc trưng phức tạp mà không tăng quá nhiều số lượng tham số. Mỗi khối trong Entry Flow kết hợp với lớp gộp (Max Pooling) và một kết nối dư để giữ lại thông tin gốc khi lan truyền.

- Middle Flow (dòng giữa): Middle Flow là phần trọng tâm của mạng, bao gồm 8 khối tích chập chiều sâu phân tách. Mỗi khối tích chập được thiết kế với kết nối dư, giúp mạng có thể truyền thông tin từ các lớp trước đó, giảm tình trạng mất mát thông tin khi độ sâu của mạng tăng lên.
- Exit Flow (dòng ra): Phần cuối cùng của mạng, tập trung các đặc trưng trừu tượng nhất để chuẩn bị cho giai đoạn phân loại. Phần này sử dụng các lớp tích chập chiều sâu phân tách với nhiều bộ lọc hơn, giúp mạng học các đặc trưng tổng quát nhất. Kết thúc bằng lớp gộp toàn cục (Global Average Pooling) và một lớp kết nối đầy đủ (Fully Connected) với hàm Softmax để dự đoán.

Xception là một bước tiến trong việc tối ưu hóa CNN với cách tiếp cận độc đáo thông qua tích chập chiều sâu phân tách. Mặc dù có một số hạn chế khi làm việc với các tập dữ liệu nhỏ, Xception vẫn được sử dụng rộng rãi nhờ hiệu suất cao và khả năng trích xuất đặc trưng mạnh mẽ, đặc biệt trong các ứng dụng phân loại và nhận dạng ảnh.

### 4.2.3 Một số thước đo mô hình phân loại

Đánh giá mô hình là một khâu quan trọng trong quá trình xây dựng một mô hình. Giúp xác định được chất lượng và hiệu suất của mô hình, qua đó lựa chọn ra được mô hình tốt nhất để sử dụng và triển khai trong thực tế. Một số thước đo cho bài toán phân loại như: Precision (độ chuẩn xác), Recall (độ phủ), Confusion Matrix (ma trận nhầm lẫn).v.v.

#### a) Accuracy

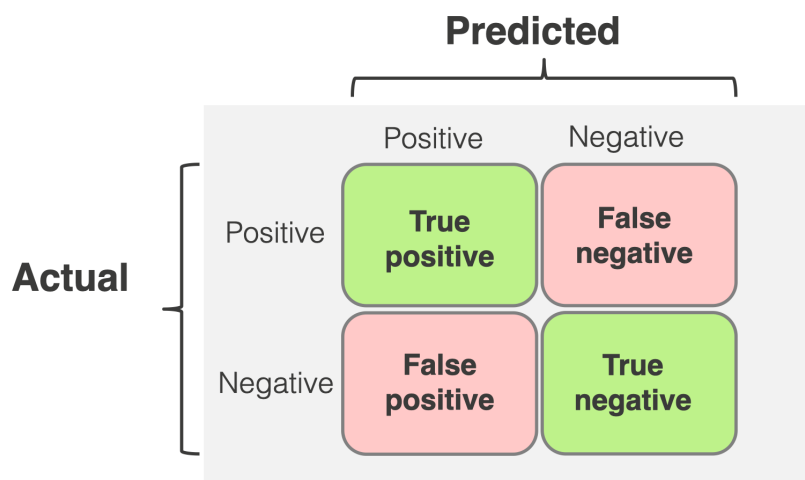
Khi xây dựng một mô hình phân loại chúng ta sẽ muốn biết một cách khái quát tỷ lệ các trường hợp được dự báo đúng trên tổng số các trường hợp là bao nhiêu. Tỷ lệ này được gọi là độ chính xác. Độ chính xác càng cao thì mô hình càng tốt. Độ chính xác được tính bằng công thức sau:

$$\text{acc} = \frac{S}{T}$$

Trong đó  $S$  là tổng số điểm dữ liệu được dự đoán là đúng trong tập kiểm tra và  $T$  là tổng số điểm dữ liệu trên tập kiểm tra.

## b) Confusion Matrix

Nhược điểm của Accuracy là chỉ cho ta biết độ chính xác khi dự báo của mô hình, nhưng không thể hiện mô hình đang dự đoán sai như thế nào. Vì vậy chúng ta cần một phương pháp đánh giá khác được gọi là Confusion Matrix (ma trận nhầm lẫn). Confusion Matrix (hình 4.18) là một ma trận thể hiện số lượng điểm dữ liệu thuộc vào một class và được dự đoán thuộc vào class. Confusion Matrix cung cấp thêm thông tin về tỉ lệ phân lớp đúng giữa các lớp, hay giúp phát hiện các lớp có tỉ lệ phân lớp nhầm cao nhờ vào các khái niệm True (False) Positive (Negative).



Hình 4.18: Ma trận nhầm lẫn

Trong đó:

- True positive (TP): Là số lượng điểm của lớp positive được phân loại đúng là positive.
- True negative (TN): Là số lượng điểm của lớp negative được phân loại đúng là negative.
- False positive (FP): Là số lượng điểm của lớp negative bị phân loại nhầm thành positive.
- False negative (FN): Là số lượng điểm của lớp positive bị phân loại nhầm thành negative.

## c) Recall



Recall (Độ phủ) đo lường tỷ lệ dự báo chính xác các trường hợp dương tính trên toàn bộ các mẫu thuộc nhóm dương tính. Độ phủ được tính bởi:

$$\text{recall} = \frac{TP}{TP + FN}$$

Để tính được độ phủ thì cần phải biết trước nhãn của dữ liệu. Do đó độ phủ có thể được dùng để đánh giá trên tập huấn luyện và thẩm định khi chúng ta đã biết trước nhãn.

#### **d) Precision**

Độ chính xác hay còn gọi là Precision dùng để xác định các trường hợp là đúng trong các trường hợp được dự báo là dương tính.

$$\text{precision} = \frac{TP}{TP + FP}$$

#### **e) F1 Score**

F1 Score là trung bình điều hòa giữa độ chuẩn xác và độ phủ, là chỉ số đại diện trong việc đánh giá tỷ lệ dự báo đúng của các trường hợp mẫu dương tính.

$$\text{f1 score} = \frac{2}{\text{precision}^{-1} + \text{recall}^{-1}}$$

F1 Score luôn nằm trong khoảng của độ chuẩn xác và độ phủ. Do đó khi có sự mất cân bằng giữa độ phủ và độ chuẩn xác thì chúng ta sẽ cân bằng được hai giá trị này.

### **4.2.4 Một số vấn đề và thách thức trong quá trình huấn luyện**

#### **a) Chất lượng và số lượng dữ liệu**

Sự thành công của các mô hình học sâu phụ thuộc rất nhiều vào chất lượng và số lượng dữ liệu đào tạo. Nếu dữ liệu được sử dụng để đào tạo không đủ hoặc chứa lỗi, có thể ảnh hưởng tiêu cực đến hiệu suất của mô hình. Do đó, việc có dữ liệu sạch và phong phú là điều cần thiết để có kết quả học sâu tối ưu. Để tạo ra một tập dữ liệu đa dạng, cần thu thập nhiều thông tin. Sử dụng các kỹ thuật như tăng cường dữ liệu để làm phong phú dữ liệu. Làm sạch và chuẩn hóa dữ liệu thông qua quá trình xử lý trước để loại bỏ lỗi và sự không nhất quán. Tối ưu hóa tập dữ liệu hơn nữa bằng cách áp dụng các phương pháp như học chuyển giao và học bán giám sát, đặc biệt là khi xử lý dữ liệu hạn chế, để cải thiện hiệu suất mô hình.

#### **b) Độ phức tạp của mô hình**

Các mô hình học sâu, do bản chất phức tạp của chúng, thường bị quá khớp. Điều này có nghĩa là chúng vượt trội trên dữ liệu mà chúng được đào tạo nhưng lại gặp khó khăn với dữ liệu mới, chưa được biết đến. Quá khớp xảy ra khi mô hình về cơ bản ghi nhớ dữ liệu đào tạo thay vì học cách khái quát hóa từ dữ liệu đó, dẫn đến giảm hiệu suất trong các ứng dụng thực tế. Các phương pháp chính quy hóa như Dropout và chính quy hóa L1/L2 là những công cụ có giá trị để ngăn chặn tình trạng quá khớp trong các mô hình học máy. Chúng thực hiện điều này bằng cách thêm các hình phạt vào độ phức tạp của mô hình hoặc vô hiệu hóa ngẫu nhiên một số nơ-ron. Các kỹ thuật giám sát và dừng sớm giúp xác định thời điểm mô hình bắt đầu học từ dữ liệu nhiễu, cho phép chúng ta dừng đào tạo và duy trì mô hình hoạt động tốt hơn.

### **c) Thời gian đào tạo**

Các mô hình học sâu đòi hỏi sức mạnh tính toán đáng kể và thời gian đào tạo kéo dài, gây ra trở ngại đáng kể cho các cá nhân và doanh nghiệp nhỏ. Nhu cầu về phần cứng đắt tiền và đầu tư thời gian kéo dài có thể hạn chế khả năng tiếp cận công nghệ tiên tiến này, cản trở việc áp dụng rộng rãi trong các thực thể hạn chế về nguồn lực. Sử dụng các dịch vụ dựa trên đám mây hoặc phần cứng chuyên dụng (ví dụ: GPU hoặc TPU) để tăng tốc quá trình đào tạo. Sử dụng đào tạo phân tán để phân phối khối lượng công việc trên nhiều thiết bị. Lượng tử hóa và cắt tỉa mô hình có thể giảm kích thước và thời gian đào tạo của mô hình mà không làm giảm đáng kể độ chính xác.

### **d) Điều chỉnh siêu tham số**

Việc lựa chọn siêu tham số tối ưu, như tốc độ học và kích thước lô, có thể khó khăn và tốn thời gian. Đây là khía cạnh quan trọng của việc đào tạo các mô hình học sâu. Sử dụng các công cụ điều chỉnh siêu tham số tự động như tìm kiếm lưới hoặc tối ưu hóa Bayesian để khám phá và tìm ra bộ siêu tham số tối ưu một cách có hệ thống. Các mô hình được đào tạo trước với các siêu tham số đã thiết lập có thể là điểm khởi đầu.

### **e) Mất gradient**

Trong mạng nơ-ron sâu, gradient có thể trở nên quá nhỏ (biến mất) hoặc quá lớn (bùng nổ) trong quá trình đào tạo, khiến việc cập nhật trọng số của mô hình trở nên khó khăn. Các hàm kích hoạt như ReLU ngăn chặn vấn đề gradient biến mất, cho phép học hiệu quả trong các mạng nơ-ron sâu. Các kỹ thuật cắt gradient và khởi tạo trọng số thích hợp bảo vệ chống lại gradient bùng nổ, đảm bảo đào tạo ổn định và hiệu quả các mạng này bằng cách hạn chế các bản cập nhật trọng số quá lớn.

### **f) Khả năng giải thích hạn chế**

Các mô hình học sâu thường được gọi là “hộp đen” vì rất khó để diễn giải cách chúng đưa ra quyết định. Để hiểu cách một mô hình đưa ra quyết định, chúng ta có thể sử dụng các phương pháp như bản đồ chú ý, làm nổi bật các phần quan trọng của đầu vào và hình ảnh hóa tính năng, cho thấy những gì mô hình học được. Sự lan truyền liên quan theo từng lớp theo dõi luồng thông tin. Các mô hình đơn giản hơn như cây quyết định hoặc tập hợp có thể làm cho các mô hình phức tạp dễ diễn giải hơn mà không làm giảm độ chính xác.

#### **g) Giới hạn phần cứng**

Các mô hình học sâu có thể tốn nhiều bộ nhớ và tính toán, khiến chúng khó chạy trên các thiết bị có nguồn lực hạn chế. Nâng cao hiệu quả mô hình bằng cách hợp lý hóa thiết kế và áp dụng lượng tử hóa để giảm kích thước. Sử dụng điện toán biên để xử lý trên thiết bị, giảm sự phụ thuộc vào dịch vụ đám mây. Phương pháp này cải thiện hiệu suất và đảm bảo quyền riêng tư của dữ liệu đồng thời giảm thiểu tài nguyên tính toán và độ trễ.

#### **h) Dữ liệu không cân bằng**

Trong các tập dữ liệu thực tế, các lớp mất cân bằng có thể dẫn đến các mô hình hoạt động tốt trên lớp chiếm đa số nhưng lại kém trên lớp chiếm thiểu số. Để giải quyết các tập dữ liệu mất cân bằng, có thể áp dụng các phương pháp như lấy mẫu quá mức (tăng mẫu lớp thiểu số), lấy mẫu dưới mức (giảm mẫu lớp đa số) và tạo dữ liệu tổng hợp. Có thể sử dụng mất mát tiêu điểm, một hàm mất mát chuyên biệt, để đưa thêm trọng số cho lớp thiểu số trong quá trình đào tạo, giúp mô hình học tốt hơn từ lớp đó.

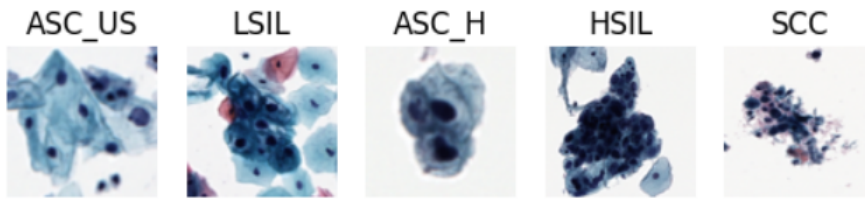
### **4.3 Huấn luyện, đánh giá và triển khai mô hình**

#### **4.3.1 Dữ liệu và quy trình tiền xử lý**

##### **a) Tập dữ liệu**

Trong nghiên cứu này, dữ liệu được sử dụng là bộ hình ảnh ung thư cổ tử cung từ bệnh viện A, Thái Nguyên, do các bác sĩ chuyên gia tế bào học thu thập và gán nhãn thủ công. Bộ dữ liệu bao gồm 22.434 hình ảnh với kích thước đa dạng, phân thành 5 loại tổn thương tế bào chính: ASC\_H, ASC\_US, HSIL, LSIL và SCC. Trong đó, SCC là loại tế bào dương tính với ung thư, còn các loại khác là tế bào tiền ung thư có khả năng tiến triển thành ung thư trong tương lai. Hình 4.19 mô tả về 5 loại tế bào có trong tập dữ liệu.

Các hình ảnh tế bào trong nghiên cứu này được thu thập thông qua phương pháp Pap smear, do các bác sĩ chuyên khoa thực hiện. Dịch lấy từ cổ tử cung được phết lên lam kính, sau đó được



Hình 4.19: Tập dữ liệu ung thư cổ tử cung - Bệnh viện A, Thái Nguyên

nhuộm màu và quan sát dưới kính hiển vi. Thông qua quá trình này, các bác sĩ có thể quan sát chi tiết cấu trúc tế bào và khoanh vùng những khu vực có dấu hiệu tổn thương. Từ đó, các vùng tế bào bất thường sẽ được cắt ra để phục vụ cho quá trình phân tích và gán nhãn.

#### **b) Quy trình tiền xử lý dữ liệu**

Để chuẩn bị dữ liệu cho mô hình, quá trình tiền xử lý đã được thực hiện nhằm tăng chất lượng và tính đại diện của dữ liệu. Một số các bước đã được tiến hành xử lý cho tập dữ liệu:

- Loại bỏ các ảnh không phù hợp: Những ảnh có kích thước quá lớn, ảnh mờ, ảnh bị lỗi đã được loại bỏ khỏi tập dữ liệu nhằm giảm tải cho quá trình tiền xử lý và tránh làm nhiễu mô hình.
- Tăng cường dữ liệu: Nhằm tăng tính đa dạng của tập dữ liệu, các kỹ thuật tăng cường dữ liệu được áp dụng như xoay ảnh, phóng to, thu nhỏ, lật ảnh và dịch chuyển ảnh.
- Chuẩn hóa dữ liệu: Tất cả các ảnh sau khi được xử lý đều được đưa về cùng kích thước là 224x224 và chuẩn hóa màu sắc nhằm đảm bảo tính đồng nhất, giúp mô hình học hiệu quả hơn.

Để tạo sự chuẩn xác và tính công bằng trong huấn luyện và đánh giá, bộ dữ liệu được phân chia thành 3 tập nhỏ: tập huấn luyện, tập kiểm định, tập kiểm tra với tỷ lệ lần lượt là 70%, 15% và 15%. Tập huấn luyện được sử dụng để dạy mô hình nhận diện các đặc trưng của dữ liệu; tập kiểm định giúp điều chỉnh siêu tham số và ngăn ngừa quá khớp; còn tập kiểm tra cuối cùng sẽ đánh giá hiệu suất thực tế của mô hình trên dữ liệu chưa từng thấy.

### **4.3.2 Huấn luyện và đánh giá hiệu suất**

#### **a) Tinh chỉnh và huấn luyện mô hình**

Tất cả 6 mô hình là MobileNet-V2, Inception-V3, InceptionResNet-V2, ResNet101, VGG16 và Xception là những mô hình đã được huấn luyện trên một tập dữ liệu hình ảnh cực lớn và đa dạng là ImageNet, Nhằm tận dụng lợi thế từ các đặc trưng đã được học, các mô hình này sẽ

được áp dụng phương pháp tinh chỉnh (fine-tune) để điều chỉnh lại trên tập dữ liệu ung thư cổ tử cung đặc thù. Việc sử dụng các mô hình đã được huấn luyện sẵn cho phép khai thác tri thức hình ảnh tổng quát từ ImageNet, giúp các mô hình dễ dàng nhận diện được các đặc điểm chung của hình ảnh. Sau đó, khi tinh chỉnh trên tập dữ liệu chuyên biệt về ung thư cổ tử cung, các mô hình có thể học thêm những đặc điểm chi tiết, đặc trưng riêng biệt của tế bào ung thư, từ đó cải thiện hiệu suất trong việc phân loại các loại tổn thương tế bào cổ tử cung một cách chính xác.

Để tối ưu hóa hiệu suất mô hình, quá trình điều chỉnh cả tham số và siêu tham số được sử dụng. Với Learning Rate (tốc độ học) ban đầu là  $1e-4$ , Batch Size (kích thước lô) bằng 64, Epochs (số lượt huấn luyện) là 100 kết hợp với một số lớp Dropout (tỷ lệ nơ-ron bị bỏ qua trong quá trình huấn luyện) khoảng 20%. Bên cạnh việc điều chỉnh các siêu tham số, Early Stopping (dừng sớm) và Learning Rate Scheduler (lập lịch tốc độ học) cũng được áp dụng để cải thiện quá trình huấn luyện.

- **Early Stopping:** Kỹ thuật này được sử dụng để ngăn chặn việc huấn luyện quá lâu dẫn đến Overfitting. Early Stopping sẽ theo dõi hiệu suất mô hình trên tập kiểm định. Nếu hiệu suất không cải thiện trong một số epoch liên tiếp (số lượng do người dùng chỉ định), quá trình huấn luyện sẽ dừng lại sớm, giúp tiết kiệm tài nguyên và tránh làm mô hình trở nên phức tạp không cần thiết.
- **Learning Rate Scheduler:** Đây là phương pháp thay đổi Learning Rate một cách linh hoạt trong quá trình huấn luyện để đạt hiệu suất tối ưu. Thông thường, Learning Rate sẽ giảm dần sau một số epoch để mô hình học tập tinh chỉnh tốt hơn, ổn định ở vùng lân cận của cực tiểu toàn cục.

## b) Đánh giá hiệu suất mô hình

Để đánh giá hiệu quả của các mô hình trong bài toán phân loại tế bào ung thư cổ tử cung, nhiều chỉ số và phương pháp đánh giá khác nhau đã được sử dụng. Kết quả đánh giá được thể hiện ở bảng 4.1.

Name	Accuracy	Precision	Recall	F1 score
MobileNet-V2	65.51	68.84	69.05	68.41
InceptionV3	<b>69.93</b>	<b>73.24</b>	<b>72.25</b>	<b>72.57</b>

InceptionResNetV2	<b>71.62</b>	<b>74.36</b>	<b>74.12</b>	<b>74.04</b>
VGG16	61.94	65.22	65.15	64.65
ResNet101	58.83	61.08	61.91	60.41
Xception	65.58	67.74	68.75	67.18

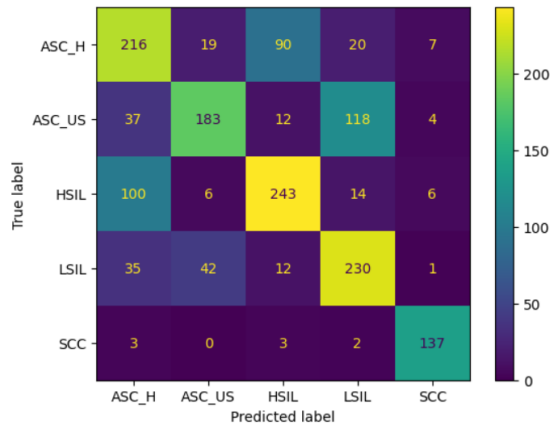
Bảng 4.1: Kết quả đánh giá hiệu suất mô hình.

Mô hình InceptionResNet-V2 đạt hiệu suất cao nhất trong số tất cả các mô hình, với accuracy (71.62%), precision (74.36%), recall (74.12%) và F1-score (74.04%). Đây là mô hình cân bằng tốt nhất về các chỉ số đánh giá cho thấy hiệu suất vượt trội trong bài toán này. Inception-V3 cũng có hiệu suất gần tương đương với InceptionResNet-V2, với các chỉ số accuracy (69.93%), precision (73.24%), recall (72.25%) và F1-score (72.57%). Điều này cho thấy Inception-V3 là một mô hình tiềm năng cho tác vụ này nhờ khả năng duy trì độ chính xác cao.

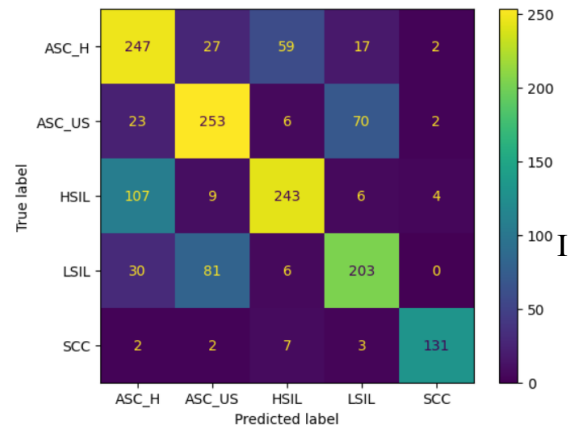
Xception và MobileNet-V2 có hiệu suất trung bình, với độ chính xác lần lượt là 65.58% và 65.51%. Dù không nổi bật như InceptionResNet-V2 hay Inception-V3, hai mô hình này vẫn đạt được độ chính xác và F1-score khá cao, cho thấy khả năng phân loại nhất định. VGG16 và ResNet101 là hai mô hình có hiệu suất thấp nhất. VGG16 có accuracy là 61.94%, trong khi ResNet101 chỉ đạt accuracy 58.83%, cùng với các chỉ số precision và recall thấp. Điều này có thể do cấu trúc của hai mô hình này không tối ưu cho bài toán phân loại tế bào ung thư cổ tử cung hoặc do chúng chưa phù hợp với đặc điểm của bộ dữ liệu.

Tiếp theo, quan sát vào Confusion Matrix giữa các mô hình (hình 4.26), điểm chung của cả 6 mô hình là đều nhận diện rất tốt trên nhãn SCC. Mô hình InceptionResNet-V2 và Inception-V3 cho thấy hiệu suất vượt trội khi hầu như có sự nhận diện tương đối chính xác trên cả 5 nhãn mà không gây ra những sự chênh lệch, nhầm lẫn nhiều. Đối với MobileNet-V2 mô hình nhận diện nhầm lẫn nhiều giữa các nhãn như ASC\_H, ASC\_US, HSIL, trong khi đó Xception cũng gặp phải hiện tượng này nhưng thậm chí còn nhầm lẫn nhiều trên cả nhãn LSIL. ResNet101 và VGG16 với những chỉ số hiệu suất accuracy, f1-score, recall, precision thấp, phần nào được thể hiện qua ma trận nhầm lẫn này, khi chúng hoạt động khá tệ trên bộ dữ liệu kiểm tra.

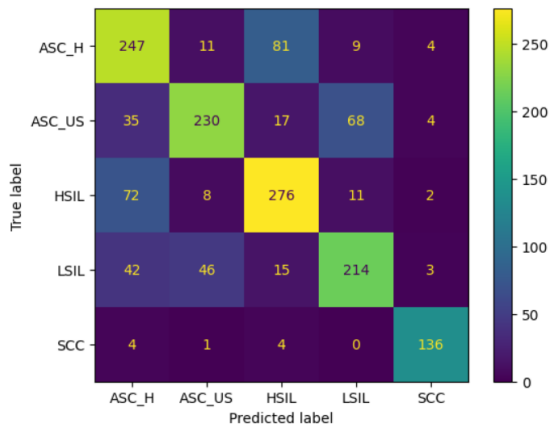
Từ những nhận định và phân tích chi tiết về hiệu suất các mô hình, InceptionResNet-V2 nổi bật là mô hình có độ tin cậy và độ chính xác cao nhất. Do đó, mô hình này được chọn để tích hợp vào hệ thống gán nhãn bán tự động, nhằm tối ưu hóa độ chính xác và hiệu quả trong quá trình sử dụng.



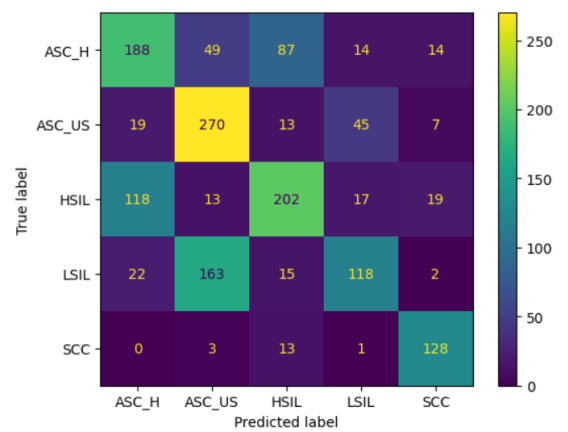
Hình 4.20: MobileNet-V2 CF



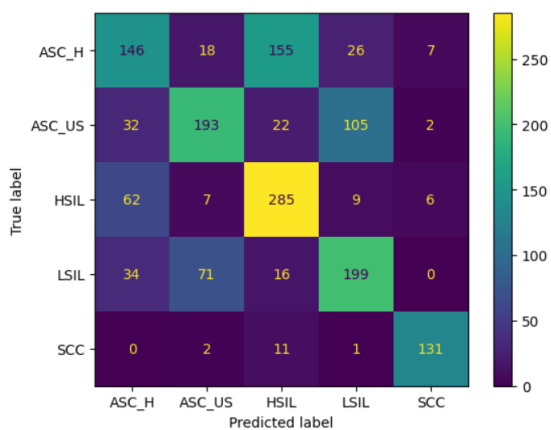
Hình 4.21: Inception-V3 CF



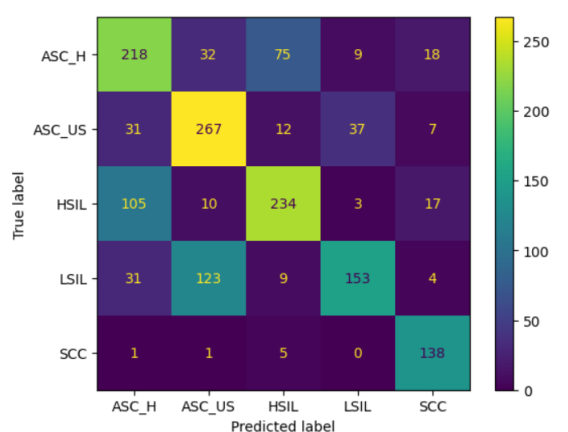
Hình 4.22: InceptionResNet-V2 CF



Hình 4.23: ResNet101 CF



Hình 4.24: VGG16 CF



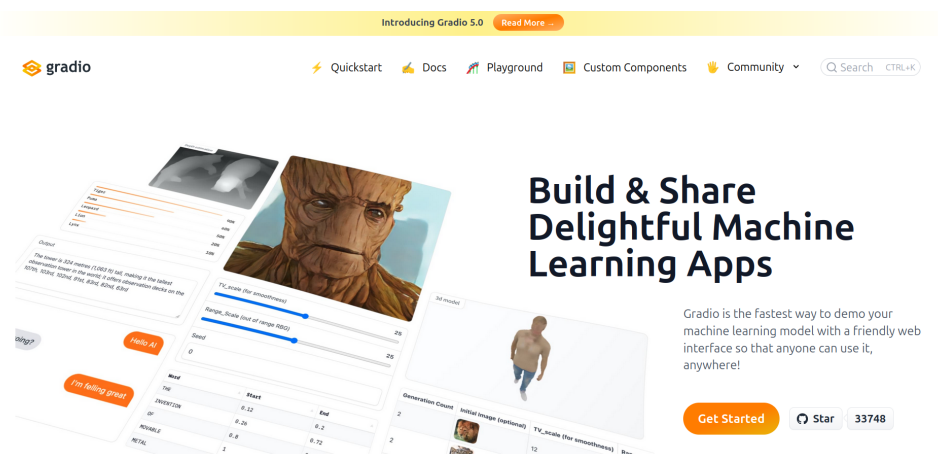
Hình 4.25: Xception CF

Hình 4.26: Biểu đồ ma trận nhầm lẫn giữa các mô hình

### 4.3.3 Triển khai giao diện hỗ trợ gán nhãn

#### a) Công cụ Gradio

Gradio (hình 4.27) là một gói Python nguồn mở cho phép nhanh chóng xây dựng một bản demo hoặc ứng dụng web cho mô hình học máy, API hoặc bất kỳ hàm Python tùy ý nào. Sau đó có thể chia sẻ ứng dụng này chỉ trong vài giây bằng các tính năng chia sẻ tích hợp. Để giúp người dùng dễ dàng tiếp cận và thực hiện quá trình gán nhãn, công cụ Gradio đã được sử dụng để triển khai một giao diện trực quan và dễ sử dụng. Công cụ này cho phép người dùng thao tác trực tiếp với các hình ảnh, gán nhãn một cách nhanh chóng và thuận tiện.



Hình 4.27: Công cụ Gradio

#### b) Triển khai mô hình

Sử dụng hai cách thức để triển khai mô hình lên web: triển khai từ local (cục bộ) và sử dụng API. Sự kết hợp giữa hai phương pháp nhằm đáp ứng nhu cầu sử dụng cụ thể của ứng dụng, linh hoạt và dễ dàng mở rộng.

##### Triển khai từ local (cục bộ)

- Quá trình triển khai: Mô hình được chạy trực tiếp trên máy tính cục bộ hoặc máy chủ cá nhân, nơi đã cài đặt công cụ Gradio để tạo giao diện web. Người dùng truy cập giao diện thông qua một địa chỉ nội bộ hoặc đường dẫn IP, có thể là trong mạng nội bộ hoặc qua kết nối mạng.
- Ưu điểm: Việc triển khai từ local giúp kiểm soát toàn bộ quy trình, dễ dàng chỉnh sửa mô hình và giao diện. Vì không cần phụ thuộc vào máy chủ bên ngoài, độ trễ trong việc xử lý mô hình thấp, phù hợp cho thử nghiệm hoặc môi trường có giới hạn về người dùng.



- **Nhược điểm:** Máy tính cục bộ cần phải luôn hoạt động và kết nối internet để duy trì truy cập, có thể không khả thi cho nhiều người dùng cùng lúc hoặc những ứng dụng yêu cầu độ bền bỉ cao. Khả năng mở rộng và bảo trì cũng gặp khó khăn nếu lượng truy cập lớn.

### **Triển khai thông qua API**

- **Quá trình triển khai:** Trong phương pháp này, mô hình được triển khai trên công cụ Litserve, mô hình sẽ được cung cấp qua một API RESTful hoặc GraphQL. Người dùng từ bất kỳ thiết bị nào có kết nối mạng đều có thể gửi yêu cầu đến API, nhận về kết quả từ mô hình.
- **Ưu điểm:** API giúp tăng khả năng mở rộng vì có thể phục vụ nhiều người dùng cùng lúc từ nhiều địa điểm khác nhau. Phương pháp này phù hợp với các ứng dụng cần hoạt động liên tục và có thể điều chỉnh tài nguyên để đáp ứng lưu lượng truy cập cao.
- **Nhược điểm:** Việc triển khai qua API đòi hỏi các bước bảo mật dữ liệu và quản lý truy cập. Ngoài ra, quá trình gọi API có thể làm tăng độ trễ vì phải xử lý qua mạng, và chi phí duy trì máy chủ hoặc dịch vụ đám mây cũng cần được cân nhắc.

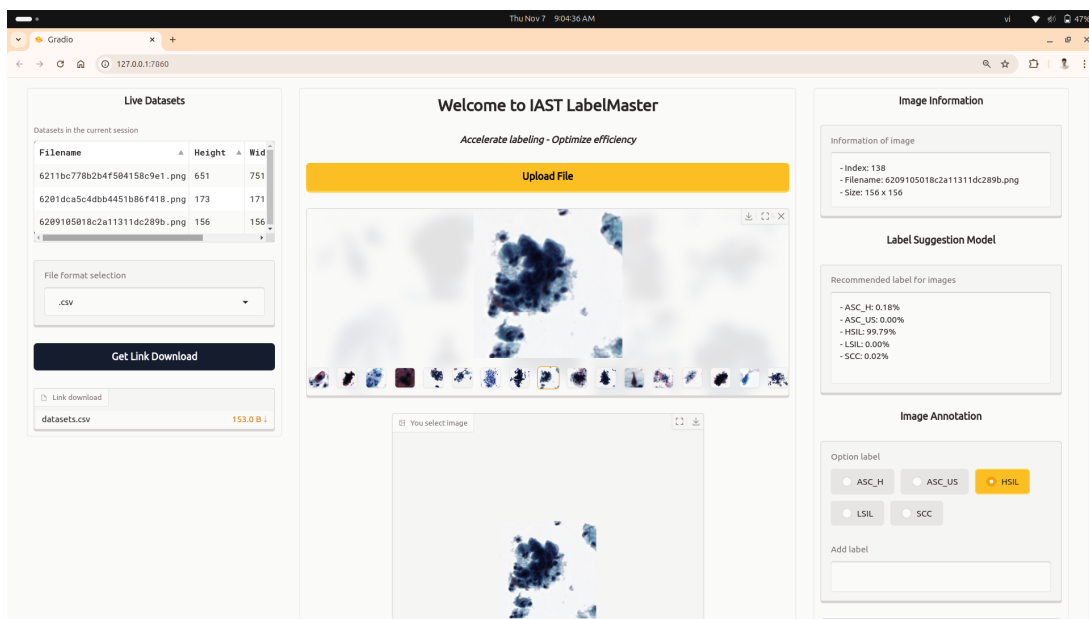
### **c) Xây dựng giao diện người dùng**

#### **Chức năng chính**

- **Tải hình ảnh:** Cho phép người dùng tải lên hình ảnh từ máy tính để thực hiện gán nhãn hoặc phân tích.
- **Gợi ý nhãn:** Đưa ra các gợi ý nhãn cho hình ảnh giúp hỗ trợ người dùng đưa ra quyết định cuối cùng.
- **Chỉnh sửa nhãn:** Người dùng có thể điều chỉnh hoặc thay đổi nhãn cho từng hình ảnh để đảm bảo độ chính xác cao nhất.
- **Xem trước kết quả:** Cho phép xem trước nhãn đã gán trên ảnh để người dùng dễ dàng kiểm tra và xác nhận.
- **Lưu kết quả:** Lưu lại ảnh và các nhãn đã gán vào một số định dạng tệp như: csv và txt, dễ dàng tải xuống để sử dụng trong các ứng dụng khác.

## Minh hoạ giao diện người dùng

Hình 4.28 mô tả giao diện khi người dùng truy cập vào trang web. Sau khi truy cập thành công, người dùng dễ dàng có thể tải các tệp hoặc thư mục ảnh từ upload file, hình ảnh hoặc thư mục sau khi được tải lên sẽ hiển thị ở một dạng lưới. Khi người dùng chọn một ảnh bất kì trong lưới hình ảnh, mô-đun sẽ tự động đưa ra các thông tin về hình ảnh trong mục information image như: kích thước ảnh, tên ảnh, kích thước tệp. Phần quan trọng nhất của mô-đun này, là mô hình sẽ đưa ra những gợi ý về tỷ lệ cho mỗi nhãn trong mục label suggestion model. Mục option label cho phép người dùng có thể điều chỉnh hoặc thay đổi nhãn cho từng hình ảnh hoặc thêm một nhãn mới trong phần add label.



Hình 4.28: Giao diện người dùng

Khi các thông tin được lưu lại bởi người dùng, mục live datasets cho phép người dùng xem trước nhãn cũng như thông tin về những ảnh đã được lưu lại có thể dễ dàng kiểm tra và xác nhận. Và cuối cùng người dùng có thể tải xuống tệp dữ liệu đã được gán nhãn thông qua get link download.

## **V. KẾT LUẬN VỀ KẾT QUẢ ĐẠT ĐƯỢC QUA ĐỢT THỰC TẬP**

### **5.1 Mục tiêu đề ra và kết quả đạt được**

- Mục tiêu của đợt thực tập là xây dựng một mô đun gắn nhãn tích hợp mô hình AI nhằm hỗ trợ người dùng gắn nhãn dữ liệu một cách nhanh chóng và chính xác, đồng thời tạo ra một giao diện web trực quan với Gradio.
- Kết quả đạt được là hệ thống gắn nhãn tự động đã được hoàn thiện, có khả năng nhận diện và gợi ý nhãn phù hợp cho các đối tượng trong hình ảnh, giúp người dùng tiết kiệm thời gian và công sức so với phương pháp gắn nhãn thủ công. Hệ thống hoạt động hiệu quả trên giao diện web triển khai qua Gradio, cho phép người dùng thao tác dễ dàng và trực quan.

### **5.2 Kỹ năng và kiến thức đã học được**

- Phát triển và huấn luyện thành công 6 mô hình mạng CNN nổi bật với tập dữ liệu ung thư cổ tử cung. Bên cạnh đó nắm được các kỹ thuật tiền xử lý dữ liệu, tinh chỉnh siêu tham số và các phép đo hiệu suất cho bài toán phân loại.
- Sử dụng thành thạo các thư viện hỗ trợ việc huấn luyện các mô hình học sâu: Keras và Tensorflow.
- Nắm bắt được các phép đo để đánh giá hiệu suất mô hình, từ đó đưa ra phương án để cải thiện, tối ưu hóa mô hình.
- Trong quá trình thực tập, đã phát triển kỹ năng sử dụng Gradio để xây dựng giao diện người dùng cho ứng dụng AI, bao gồm cách tích hợp mô hình AI vào hệ thống gắn nhãn và tối ưu hóa quy trình để giảm độ trễ.

### **5.3 Khó khăn gặp phải**

- Khó khăn đầu tiên gặp phải là hiệu suất của mô hình chưa đạt mức tối ưu, do nhiều yếu tố như chất lượng dữ liệu còn hạn chế và thời gian thực nghiệm chưa đủ. Điều này dẫn đến việc mô hình chưa có được các yếu tố cần thiết để cải thiện khả năng học hỏi và đưa ra dự đoán chính xác nhất.

- Việc tối ưu hóa thời gian phản hồi của mô hình AI trên giao diện Gradio, đặc biệt khi phải xử lý lượng lớn hình ảnh cũng như mô hình có kích thước nặng, bên cạnh đó việc xây dựng giao diện trực quan và dễ sử dụng cũng là một thử thách lớn.

## **5.4 Đóng góp của đề tài**

- Mô đun gán nhãn tự động tích hợp AI này đã giúp giảm thiểu thời gian gán nhãn dữ liệu và tăng độ chính xác trong quá trình gán nhãn nhờ khả năng nhận diện thông minh từ mô hình AI. Điều này không chỉ nâng cao hiệu quả làm việc mà còn giúp chuẩn hóa quá trình gán nhãn trong nhóm, hỗ trợ việc xử lý dữ liệu lớn dễ dàng hơn.
- Giao diện triển khai qua Gradio mang lại trải nghiệm người dùng tốt, giúp người dùng nhanh chóng tải hình ảnh, xem gợi ý nhãn từ AI, và lưu kết quả dễ dàng.

## VI. PHẦN PHỤ LỤC

### 6.1 Danh mục từ viết tắt

Bảng 6.1: Bảng danh mục từ viết tắt.

STT	Kí hiệu chữ viết tắt	Chữ viết tắt đầy đủ
1	AI	Artificial Intelligence
2	ML	Machine Learning
3	DL	Deep Learning
4	ANN	Artificial Neural Network
5	CNN	Convolutional Neural Network
6	GD	Gradient Descent
7	CV	Computer Vision
8	NLP	Natural Language Processing
9	IAST	Institute of Applied Science and Technology
10	CF	Confusion Matrix

## 6.2 Tài liệu tham khảo

1. Dong, Na and Zhao, Li and Wu, Chun-Ho and Chang, Jian-Fang. "Inception v3 based cervical cell classification combined with artificially extracted features." *ASC*, vol. 93, 2020, pp. 106311. Elsevier.
2. Mathivanan, Sandeep Kumar, et al. "Enhancing cervical cancer detection and robust classification through a fusion of deep learning models." *Scientific Reports*, vol. 14, no. 1, 2024, pp. 10812. Nature Publishing Group UK London.
3. Arbyn, Marc, et al. "Estimates of incidence and mortality of cervical cancer in 2018: a worldwide analysis." *The Lancet Global Health*, vol. 8, no. 2, 2020, pp. e191–e203. Elsevier.
4. Kessler, Theresa A. "Cervical cancer: prevention and early detection." In *Seminars in Oncology Nursing*, vol. 33, no. 2, 2017, pp. 172–183. Elsevier.
5. Ali, Md Mamun, et al. "Machine learning-based statistical analysis for early stage detection of cervical cancer." *Computers in Biology and Medicine*, vol. 139, 2021, pp. 104985. Elsevier.
6. Kaushik, Manoj, et al. "Cytokine gene variants and socio-demographic characteristics as predictors of cervical cancer: A machine learning approach." *Computers in Biology and Medicine*, vol. 134, 2021, pp. 104559. Elsevier.
7. Rahaman, Md Mamunur, et al. "DeepCervix: A deep learning-based framework for the classification of cervical cells using hybrid deep feature fusion techniques." *Computers in Biology and Medicine*, 2021. Elsevier.
8. Litjens, Geert, et al. "A survey on deep learning in medical image analysis." *Medical Image Analysis*, vol. 42, 2017, pp. 60–88. Elsevier.
9. Lee, Kisuk, et al. "Superhuman accuracy on the SNEMI3D connectomics challenge." *arXiv preprint arXiv:1706.00120*, 2017.
10. Chollet, François. "Xception: Deep learning with depthwise separable convolutions." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1251–1258.

11. Szegedy, Christian, et al. "Rethinking the inception architecture for computer vision." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2818–2826.
12. Huang, Gao, et al. "Densely connected convolutional networks." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4700–4708.
13. Tan, Mingxing, and Le, Quoc. "EfficientNet: Rethinking model scaling for convolutional neural networks." *ICML*, 2019, pp. 6105–6114. PMLR.
14. Sandler, Mark, et al. "MobileNetV2: Inverted residuals and linear bottlenecks." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4510–4520.
15. Xie, Saining, et al. "Aggregated residual transformations for deep neural networks." *CVPR*, 2017, pp. 1492–1500.
16. Russakovsky, Olga, et al. "ImageNet large scale visual recognition challenge." *International Journal of Computer Vision*, vol. 115, 2015, pp. 211–252. Springer.
17. He, Kaiming, et al. "Deep residual learning for image recognition." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.
18. Simonyan, Karen, and Zisserman, Andrew. "Very deep convolutional networks for large-scale image recognition." *arXiv preprint arXiv:1409.1556*, 2014.
19. Mohammed, Mohanad, et al. "A stacking ensemble deep learning approach to cancer type classification based on TCGA data." *Scientific Reports*, vol. 11, no. 1, 2021, pp. 15626. Nature Publishing Group UK London.
20. Ganaie, Mudasir A, et al. "Ensemble deep learning: A review." *Engineering Applications of AI*, vol. 115, 2022, pp. 105151. Elsevier.
21. Sagi, Omer, and Rokach, Lior. "Ensemble learning: A survey." *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 8, no. 4, 2018, pp. e1249. Wiley Online Library.

22. Kandel, Ibrahim, et al. "Comparing stacking ensemble techniques to improve musculoskeletal fracture image classification." *Journal of Imaging*, vol. 7, no. 6, 2021, pp. 100. MDPI.
23. Yang, Yun, et al. "Two-stage selective ensemble of CNN via deep tree training for medical image classification." *IEEE Transactions on Cybernetics*, vol. 52, no. 9, 2021, pp. 9194–9207. IEEE.
24. Lever, Jake, et al. "Classification evaluation." *Nature*, 2016. Nature Research.
25. Pramanik, Rishav, et al. "A fuzzy distance-based ensemble of deep models for cervical cancer detection." *Computer Methods and Programs in Biomedicine*, vol. 219, 2022, pp. 106776. Elsevier.

### 6.3 Phụ lục code

```
1 import pandas as pd
2 import json
3 import tensorflow as tf
4 import cv2
5 import os
6 import gradio as gr
7 from PIL import Image
8 import numpy as np
9
10 def process_upload_file(root_folder):
11     """
12     Define function return list image then user upload folder
13     Args: path_folder
14     Results: [image1.jpg, image2.jpg,.v.v]
15     """
16
17     img = []
18     for i in root_folder:
19         img.append(i)
20
21     return img
22
```



```

23 def information(img_path):
24     """
25     Define a function to show index, name, height and width fog images
    , after user
26     selects any image in galery
27         - Args: image_path
28         - Result: [0, image1.jpg, 200, 300]
29     """
30
31     img_name = img_path.value['image']['path']
32     img = cv2.imread(img_name)
33
34     h, w, _ = img.shape
35
36     return img_path.index, img_name, h, w
37
38 def suggestion_label(img_path):
39     """
40     Define a function to predict label for image, after user selects
    any image in gallery
41         - Args: image_path
42         - Result: [20%, 30%, 40%, v.v.]
43     """
44
45     model = tf.keras.models.load_model("path_to_my_model")
46     img = img_path.value["image"]["path"]
47     img = cv2.imread(img)
48     img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
49     img_resize = cv2.resize(img, (224, 224))
50     img_array = np.array(img_resize).astype("float32") / 255.0
51     img_batch = np.expand_dims(img_array, axis=0)
52     results = model.predict(img_batch)
53
54     return (
55         f"- ASC_H: {results[0][0]*100:.2f}%\n"
56         f"- ASC_US: {results[0][1]*100:.2f}%\n"
57         f"- HSIL: {results[0][2]*100:.2f}%\n"
58         f"- LSIL: {results[0][3]*100:.2f}%\n"
59         f"- SCC: {results[0][4]*100:.2f}%\n"

```

```

60     )
61
62 def process_image(img_path: gr.SelectData):
63     """
64     Define a function to process two functions information and
65     suggestion_label simultaneously
66     """
67     index, img_name, h, w = information(img_path)
68     label = suggestion_label(img_path)
69
70     return f"- Index: {index}\n- Filename: {img_name.split("/")[-1]}\n
71 - Size: {w} x {h}", label, img_name
72
73 def add_option_label(new_label):
74     """
75     Define a function to add label if user wants extend label
76     """
77     add = gr.Radio(choices=[new_label])
78     return add
79
80 def show_and_update_dataframe(img_path, label, df_current):
81     """
82     Define a function show and update dataframe
83     """
84
85     filename = img_path.split("/")[-1]
86     img = cv2.imread(img_path)
87     h, w, _ = img.shape
88
89     new_row = pd.DataFrame(data=[[filename, h, w, label]], columns=["
90 Filename", "Height", "Width", "Label"])
91
92     df_state = pd.concat([df_current, new_row], ignore_index=True)
93
94     return df_state, df_state
95
96 def download_dataframe(df_current, file_format):

```

```

96     """
97     Define a function to download dataframe, user can download file
98     """
99     if file_format == ".csv":
100         path = "datasets.csv"
101         df_current.to_csv(path, index=False)
102     elif file_format == ".txt":
103         path = "datasets.txt"
104         df_current.to_csv(path, index=False)
105     else:
106         pass
107
108     return path
109
110 """
111 Setup layouts for upload file state
112 """
113 upload_file = gr.UploadButton(label="Upload File",
114                               file_count="directory",
115                               size="lg",
116                               variant="stop")
117
118 """
119 Setup layouts for show all image in galery
120 """
121 show_file = gr.Gallery(label="Show Image",
122                        show_label=True,
123                        columns=5,
124                        rows=2,
125                        object_fit="fill",
126                        height=300,
127                        interactive=False,
128                        allow_preview=True)
129
130 """
131 Setup layouts for show information image state
132 """
133 show_info_file = gr.Textbox(lines=3, show_label=True, label="
    Information of image")

```

```

134
135 """
136 Setup layouts for show suggestion label model state
137 """
138 show_suggest_model = gr.Textbox(lines=5, show_label=True, label="
139     Recommended label for images")
140
141 """
142 Setup layouts for show select label state
143 """
144 select_label = gr.Radio(choices=["ASC_H", "ASC_US", "HSIL", "LSIL", "
145     SCC"], label="Option label", show_label=True, interactive=True)
146
147 """
148 Setup layouts for add label state
149 """
150 add_label = gr.Textbox(label="Add label", show_label=True)
151
152 """
153 Setup layouts for show images, after user select image to gallery
154 """
155 show_single_image = gr.Image(label="You select image",
156     show_label=True,
157     image_mode="RGB",
158     type="filepath",
159     height=500,
160     width=500,
161     interactive=False)
162
163 """
164 Setup layouts for dataframe live
165 """
166 show_dataframe = gr.Dataframe(label="Datasets in the current session",
167     show_label=True, interactive=True)
168
169 """
170 Setup layouts for select file format
171 """
172 show_format_file = gr.Dropdown([".csv", ".txt", ".json"],

```

```

170         label="File format selection",
171         show_label=True,
172         interactive=True)
173
174 with gr.Blocks(theme="citrus", fill_height=True, fill_width=True) as
    demo:
175     """
176     Setup application with name is demo
177     """
178
179     df_state = gr.State(value=pd.DataFrame(columns=["Filename", "
Height", "Width", "Label"]))
180
181     with gr.Row():
182         with gr.Column(scale=1, variant="panel"):
183             gr.Markdown("<h3 style='text-align: center;'>Live Datasets
</h3>")
184             show_dataframe.render()
185             show_format_file.render()
186             download = gr.Button(value="Get Link Download", variant="
huggingface")
187             download_file = gr.File(label="Link download", type="
filepath", interactive=False)
188
189             with gr.Column(scale=2, variant="panel"):
190                 gr.Markdown("<h1 style='text-align: center;'>Welcome to
IAST LabelMaster </h1>")
191                 gr.Markdown("<h4 style='text-align: center;'><i>Accelerate
labeling - Optimize efficiency</i></h4>")
192                 upload_file.render()
193                 show_file.render()
194                 show_single_image.render()
195
196             with gr.Column(scale=1, variant="panel"):
197                 gr.Markdown("<h3 style='text-align: center;'>Image
Information</h3>")
198                 show_info_file.render()
199                 gr.Markdown("<h3 style='text-align: center;'>Label
Suggestion Model</h3>")

```

```

200         show_suggest_model.render()
201         gr.Markdown("<h3 style='text-align: center;*>Image
Annotation</h3*>")
202         select_label.render()
203         add_label.render()
204         save = gr.Button(variant="huggingface", value="Save")
205
206
207     """
208     Process events when user interacts
209     """
210
211     upload_file.upload(fn=process_upload_file, inputs=upload_file,
outputs=show_file)
212
213     show_file.select(fn=process_image, inputs=None, outputs=[
show_info_file, show_suggest_model, show_single_image])
214
215     add_label.submit(fn=add_option_label, inputs=add_label, outputs=
select_label)
216
217     save.click(fn=show_and_update_dataframe, inputs=[show_single_image
, select_label, df_state], outputs=[show_dataframe, df_state])
218     download.click(fn=download_dataframe, inputs=[df_state,
show_format_file], outputs=download_file)
219
220 if __name__ == "__main__":
221     demo.queue()
222     demo.launch()

```