ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

KHOA HỆ THỐNG THÔNG TIN

BÁO CÁO BÀI BÁO KHOA HỌC

Đề tài:

PHÂN LỚP HÌNH ẢNH MÔ HỌC UNG THƯ VÚ

Giảng viên: ThS. Nguyễn Hồ Duy Trí

Lớp: IS252.M22

Môn học: Khai thác dữ liệu

Nhóm 09:

Phan Mai Kiều Anh	–	19521209
Nguyễn Thị Hồng Yến	–	19521130
Hồ Thị Thanh Vân	–	19522507
Nguyễn Đức Quang	–	19522094

TP. Hồ Chí Minh, ngày 02 tháng 6 năm 2022

LỜI CẢM ƠN

Đầu tiên, nhóm chúng em xin gửi lời cảm ơn đến trường Đại học Công Nghệ Thông Tin – Đại học Quốc gia Thành phố Hồ Chí Minh và Khoa Hệ Thống Thông Tin đã tạo điều kiện, môi trường cho chúng em có cơ hội nghiên cứu và làm việc trong suốt thời gian học tập tại đây.

Tiếp theo, nhóm xin gửi lời cảm ơn chân thành đến Thầy Nguyễn Hồ Duy Trí. Thầy đã đưa ra một đề tài mới lạ và lần đầu tiên chúng em thực hiện. Đây chính là một cơ hội đặc biệt để chúng em có thể tiếp cận với các bài báo khoa học trong và ngoài nước. Thông qua việc đọc và viết báo cáo sau khi đọc, chúng em có thêm nhận thức về những kiến thức liên quan đến nghiên cứu khoa học. Cùng với đó, Thầy đã đưa ra những đề xuất và định hướng trong suốt quá trình thực hiện đề tài. Thầy cũng đã tạo điều kiện thuận lợi để chúng em thực hiện báo cáo bằng cách giải đáp các thắc mắc cho chúng em ngay tại lớp cũng như các kênh thông tin liên lạc khác.

Mặc dù chúng em luôn cố gắng chu toàn mọi công đoạn thực hiện đề tài nhưng vẫn không thể tránh khỏi sai sót, khuyết điểm, những vấn đề tồn đọng chưa thể giải quyết. Vậy nên, chúng em rất mong nhận được nhận xét từ Thầy, lời nhận xét của Thầy chính là động lực phát triển giúp chúng em có thể hoàn thiện bản thân trong tương lai.

TP. Hồ Chí Minh, tháng 6 năm 2022

Nhóm 09 thực hiện

NHẬN XÉT CỦA GIẢNG VIÊN

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

MỤC LỤC

DANH MỤC CÁC KÝ HIỆU, CÁC CHỮ VIẾT TẮT

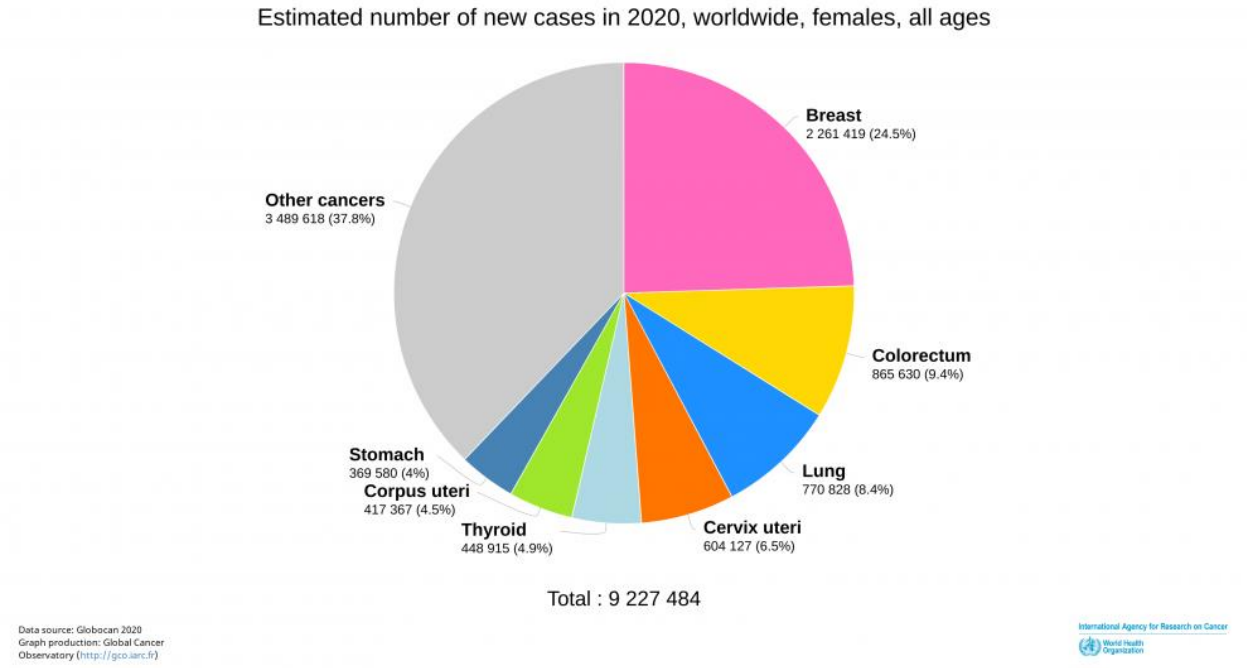
STT	Kí hiệu, chữ viết tắt	Giải thích
1	CNN	Convolutional Neural Network (Mạng nơ ron tích chập)
2	Overfitting	Quá luyện dữ liệu
3	CAD	Computer – Aided Design
4	P	Condition positive
5	N	Condition negative
6	TP	True positive
7	TN	True negative
8	FP	False positive
9	FN	False negative
10	CNTT	Công nghệ thông tin
11	Conv	Convolution layer
12	FC	Fully Connected Layer
13	VGG	Visual Geometry Group

DANH MỤC HÌNH ẢNH

MỞ ĐẦU

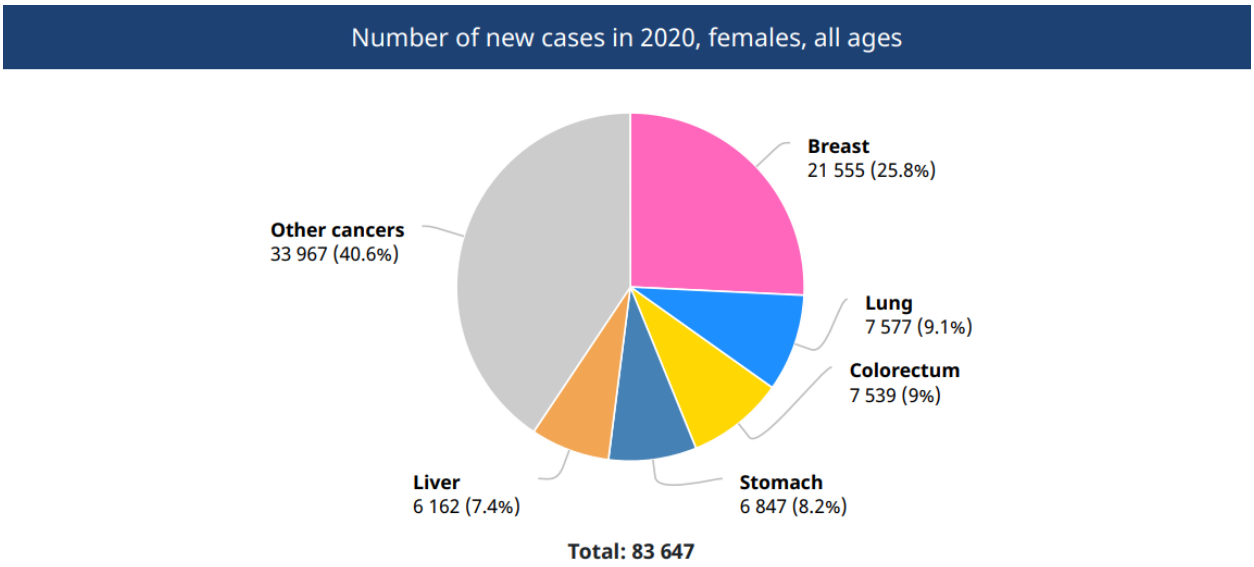
1. Lý do chọn đề tài

Hiện nay, ung thư vú là bệnh ung thư hàng đầu ở phụ nữ, nó cũng là nguyên nhân hàng đầu gây tử vong ở phụ nữ, cả ở các nước phát triển và đang phát triển. Tỷ lệ mắc bệnh ung thư vú đang gia tăng ở các nước đang phát triển do tuổi thọ tăng, tốc độ đô thị hóa tăng và thay đổi lối sống. Theo báo cáo ghi nhận ung thư toàn cầu Globocan 2020, trong số ung thư ở nữ, số người mới mắc ung thư vú đứng hàng thứ nhất, với 2.261.419 người, chiếm 24.5%.



Hình 1 Số ca bệnh ung thư được phát hiện ở nữ giới năm 2020 [1]

Tại Việt Nam, số ca mắc mới ung thư vú cũng đứng hàng thứ nhất với 21.555 người, chiếm 25,8%.



Hình 2 Phát hiện các bệnh ung thư ở phụ nữ năm 2020 [2]

Ngày nay, công nghệ thông tin (CNTT) đang dần chứng tỏ tầm ảnh hưởng rất lớn đến mọi mặt của đời sống xã hội. Đối với hoạt động của ngành y tế, có thể thấy rằng, CNTT ngày càng đóng vai trò quan trọng, không chỉ "bà đỡ" cho quá trình cải cách hành chính trong công tác quản lý, điều hành của cơ quan quản lý mà còn "đỡ đầu" cho việc triển khai và ứng dụng thành công các kỹ thuật cao trong công tác khám chữa bệnh như chụp cắt lớp, mổ nội soi… rồi trong các công tác giảng dạy, đào tạo, giám sát dịch bệnh, nghiên cứu phát triển thuốc…Với độ chính xác cao, tiết kiệm được tối đa ngân sách, CNTT cũng được ứng dụng trong bài toán phân lớp, điển hình là phân lớp mô bệnh học trong lĩnh vực y tế. Nói đến phân lớp ta cũng biết đến nhiều thuật toán như: Random Forest, Naive Bayes, Support Vector Machine hay Linear Classifiers… nhưng điểm yếu của chúng là thường áp dụng cho các bài toán về số liệu, khi áp dụng cho hình ảnh thì còn nhiều hạn chế khi phải sử dụng một mẫu dữ liệu có nhiều điểm dữ liệu, suy ra phải xử lý một lượng dữ liệu khổng lồ, mà việc coi loại dữ liệu ảnh như những loại khác thì không khả thi bởi nó không thể nêu ra được sự tương quan về không gian của các đối tượng trong ảnh, mất đi các đặc trưng quan trọng dẫn đến kết quả không chính xác.

Nổi bật lên dạo gần đây, Convolutional Neural Network (CNNs – Mạng nơ-ron tích chập) là một trong những mô hình tiên tiến. Nó thường được sử dụng nhiều trong các bài toán nhận dạng các vật trong ảnh. Giúp cho chúng ta xây dựng được những hệ thống thông minh với độ chính xác cao.

Vì thế trong nghiên cứu này nhóm chúng em lựa chọn chủ đề ***Phân lớp hình ảnh*** sử dụng phương pháp **Training from scratch** và **Transfer learning**. Trong đó, nhóm dựa trên bài báo khoa học **"Breast cancer histology images classification: Training from scratch or transfer learning?"** bài báo được chấp nhận vào ngày 15 tháng 10 năm 2018 và được đăng tải trên tạp chí ICT Express như một mục tiêu nghiên cứu, tiến hành tìm hiểu sâu về công trình nghiên cứu của nhóm tác giả nhằm nắm được kiến thức và có cái nhìn sâu sắc về mạng nơ-ron tích chập nói chung và phân lớp hình ảnh với hai phương pháp Training from scratch và Tranfer learning nói riêng.

2. Mục tiêu của đề tài

Trong nghiên cứu này, nhóm chúng em tập trung vào các vấn đề sau:

- Tìm hiểu về bài toán phân lớp
- Hiểu về nguyên lý hoạt động của mạng tích chập
- Xây dựng bộ dữ liệu dựa trên nguồn và mô tả của tác giả
- Phân tích khả năng của mô hình học đào tạo sâu so với mô hình đào tạo từ đầu
- So sánh khả năng của mô hình học đào tạo sâu so với mô hình đào tạo từ đầu
- Xác định loại mạng nào được đào tạo sẵn hoạt động tốt hơn.
- Kiểm tra ảnh hưởng của của kích thước dữ liệu trong hiệu suất của các mạng.
- Phân tích, đánh giá thực nghiệm của từng kiến trúc phân lớp trên bộ dữ liệu đã xây dựng
- Đưa ra kết luận của nhóm

3. Đối tượng nghiên cứu

Mô ảnh bệnh ung thư vú được phân thành hai lớp: lành tính và ác tính.

4. Phạm vi nghiên cứu

- Tìm hiểu về bài toán phân lớp.
- Tìm hiểu về mạng tích chập.
- Tìm hiểu về kĩ thuật học chuyển tiếp Transfer learning.
- Xây dựng bộ dữ liệu theo nguồn và mô tả của tác giả.
- Thực nghiệm từng mô hình phân lớp trên bộ dữ liệu đã được xây dựng.
- Kiểm tra dữ liệu với từng mô hình.
- Đánh giá chọn ra mô hình tốt nhất.
- Phân tích và so sánh kết quả thực nghiệm so với lý thuyết và so với kết quả gốc trong bài báo.
- Kết luận, đưa ra hướng nghiên cứu tiếp theo.

TỔNG QUAN

1. Tóm lược đề tài

Ngày nay, thế giới y học đã phát triển mạnh mẽ, nó cũng ảnh hưởng rất nhiều trong việc mang lại nhận thức cho mọi người về ung thư vú. Ung thư vú là một trong những nguyên nhân hàng đầu gây ra tỷ lệ tử vong cao ở phụ nữ. BRCA1, BRCA, béo phì, dùng thuốc tránh thai, chu kỳ kinh nguyệt không đều, tiếp xúc nhiều với xạ trị và hormone estrogen là một số yếu tố nguy cơ cao gây ung thư vú. Những yếu tố này góp phần gây ra đột biến trong các tế bào và dẫn đến sự phát triển không thể kiểm soát của các tế bào.

Hình 3 Các yếu tố nguy cơ của ung thư vú [3]

Những triệu chứng đầu tiên xảy ra trong ung thư vú có thể đe dọa đến tính mạng nếu không được chẩn đoán ở giai đoạn đầu... Kích ứng da, đỏ đau và sưng là một số triệu chứng khác ở giai đoạn ban đầu như xói mòn núm vú hoặc chảy nước đột ngột từ núm vú. Do đó, việc phát hiện sớm là điều cần thiết để ngăn ngừa và điều trị.

Hình 4 Triệu chứng và dấu hiệu cảnh báo của ung thư vú [4]

Phân tích dữ liệu thu được từ các kỹ thuật được sử dụng trong sàng lọc và theo dõi ung thư vú là rất phức tạp do các biểu hiện lâm sàng tương tự của các loại ung thư khác nhau. Phân tích dữ liệu là một quá trình tốn nhiều thời gian và công sức nhưng cũng là một bước rất quan trọng để cung cấp sự chẩn đoán rõ ràng. Vì vậy, nó đã trở nên cần thiết để tự động hóa một số nhiệm vụ trong quy trình chẩn đoán để giảm gánh nặng cho bác sĩ và các nhà nghiên cứu bệnh học. Trong bối cảnh này, các kỹ thuật học máy nổi lên như một mô hình tối ưu cung cấp các giải pháp đáng tin cậy và có thể thực hiện một số nhiệm vụ chẩn đoán một cách tự động và thông minh.

Giữa vô vàn các kĩ thuật học máy khác nhau thì CNN đang được chú ý do các ưu điểm vượt trội của nó. Tuy nhiên, đào tạo đầy đủ (đào tạo từ đầu) của CNN thì không dễ dàng vì CNN đòi hỏi nhiều dữ liệu đào tạo và các đơn vị xử lý đồ họa hiệu suất cao (GPU) để xử lý việc đào tạo với bộ dữ liệu lớn đó là một quá trình tốn thời gian. Bên cạnh đó, quá

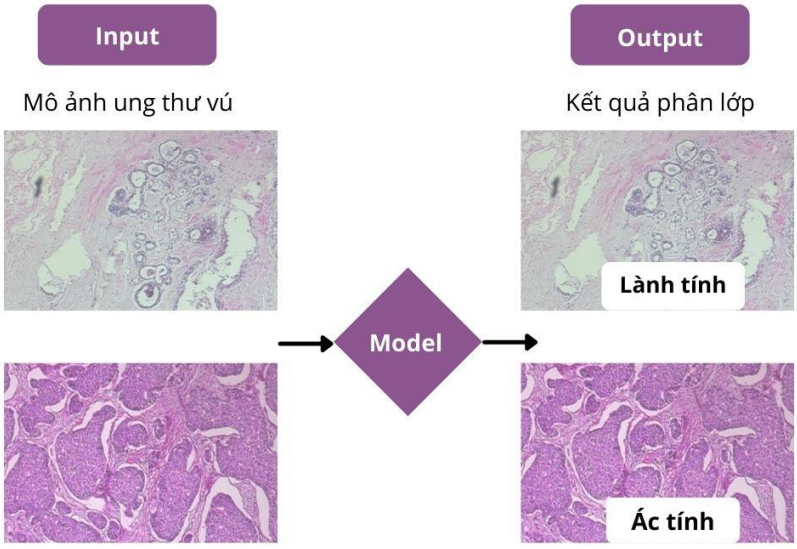
trình đào tạo trong CNN rất phức tạp và đòi hỏi phải điều chỉnh liên tục các thông số để đảm bảo việc học được tương đương giữa tất cả các lớp do các vấn đề hội tụ và overfitting.

Do vậy nhóm tác giả đã đề xuất mô hình học chuyển giao thay thế cho mô hình đào tạo từ đầu. Trong học chuyển giao, mạng lưới đào tạo một nhiệm vụ được tinh chỉnh và áp dụng cho một nhiệm vụ khác nhưng có liên quan tới nhiệm vụ trước đó. Nhóm tác giả đặt ra một câu hỏi: Liệu rằng một mạng được đào tạo đầy đủ hay mạng được tinh chỉnh trước có hoạt động tốt cho việc phân loại độc lập phóng đại các hình ảnh mô bệnh học ung thư vú hay không? Để trả lời câu hỏi này, hai thí nghiệm đã được thực hiện: (a) Phân loại độc lập phóng đại các hình ảnh mô bệnh học liên quan đến ung thư vú thu được từ tập dữ liệu BreakHis, (b) Nghiên cứu hiệu suất cho ba lần phân tách dữ liệu thử nghiệm: train - test. Đối với mỗi bộ thử nghiệm, đem hiệu suất của hai mô hình so sánh với nhau nhằm phân tích khả năng hoạt động của mô hình học tập chuyển giao so với mạng được đào tạo đầy đủ để áp dụng phân loại ung thư vú là lành tính hay ác tính.

2. Những giải pháp khoa học được thực hiện:

2.1. Framework cơ bản của bài toán

Trong nghiên cứu các phương pháp, nhóm chúng em tiếp cận phương pháp phân lớp đối tượng và các bước để xây dựng bộ phân lớp. Dữ liệu tranning đầu vào sẽ bao gồm các bức ảnh đã được gán nhãn với số lượng ảnh trong từng danh mục phải bằng nhau để quá trình trainning đạt kết quả tốt nhất. Sau đó, các dữ liệu được phân tách và được rút trích đặc trưng. Mục đích của việc trainning là để các lớp học cách nhận diện, cuối cùng cần đánh giá các lớp sau khi trainning bằng cách đưa ra các dự đoán về nhãn của các ảnh trong tập test, đầu ra sẽ thu được các ảnh gán nhãn tương đồng với loại đối tượng.



Hình 5 Mô tả bài toán phân lớp hình ảnh mô học ung thư vú

2.2. Tiền xử lý dữ liệu

Tiền xử lý dữ liệu đóng vai trò quan trọng trong việc giải quyết các vấn đề liên quan đến kích thước không cân bằng và giới hạn dữ liệu trong đó tăng cường ảnh là kĩ thuật phóng to dữ liệu để giảm bớt vấn đề về kích thước và dữ liệu bị giới hạn. Dữ liệu thu thập ở dạng hình ảnh, sau đó được lấy ra theo tỉ lệ. Cuối cùng là gán nhãn và chú thích cho các ảnh có được.

2.3. Rút trích đặc trưng

Rút trích đặc trưng (Trích chọn đặc trưng) là quá trình chọn lọc một tập con chứa các thuộc tính liên quan để sử dụng trong quá trình xây dựng mô hình. Trong đó, một bộ dữ liệu thô ban đầu được chia và giảm thành các nhóm dễ quản lý hơn nhằm giảm số lượng các đặc trưng trong bộ dữ liệu bằng cách tạo các đặc trưng mới từ các đặc trưng hiện có. Nhóm đặc trưng được rút trích có thể tóm tắt hầu hết các thông tin có trong đặc trưng ban đầu. Trích xuất đặc trưng được ứng dụng trong mô hình túi từ, xử lý ảnh, bộ tự mã hóa.

2.4. Phân loại đặc trưng thành các lớp

Phân loại ảnh là phân loại một tập hợp các hình ảnh theo các danh mục, ở đó thuật toán xem xét và dán nhãn cho hình ảnh từ một tập danh mục được xác định và đào tạo từ trước. Với mỗi tập hình ảnh, mỗi hình ảnh mô tả thuật toán sẽ quan sát toàn bộ dữ liệu và dựa trên hình dạng, màu sắc để hình thành giả thuyết liên quan đến nội dung hình ảnh. Kết quả thu được là tập dữ liệu ban đầu, hình ảnh đã được phân loại một cách tự động.

Các kỹ thuật phân loại ảnh

- Phân loại có giám sát: Thuật toán được huấn luyện trên một tập hình ảnh đã được dán nhãn. Từ dữ liệu mẫu này, thuật toán có thể trích xuất dưới dạng dữ liệu quan trọng để đưa vào xử lý. Một số phương pháp phân loại phổ biến: Support Vector Machines, Decision Tree, K Nearest Neightbors. Mạng nơ ron thường được sử dụng để phân loại hình ảnh có giám sát gồm: AlexNet, ResNet, DenseNet và Inception.
    - Phân loại nhãn đơn
    - Phân loại đa nhãn
- Phân loại không giám sát: Phương pháp liên quan đến bước trích xuất đặc điểm với các thông tin chi tiết về hình ảnh. Được xử lý bằng các phương pháp phân cụm tham số (Gausian Mixture Models) và phi tham số (K-means).
    - Phân loại video
    - Phân loại 3D

Cách hoạt động: máy tính xử lý dưới dạng ảnh pixel, hình ảnh chỉ là dạng ma trận và kích thước của ma trận phụ thuộc vào độ phân giải hình ảnh. Xử lý hình ảnh là tiến hành phân tích dữ liệu toán học, các thuật toán chia nhỏ dữ liệu thành một tập hợp các đặc điểm. Quá trình trích xuất đặc điểm là phân loại hình ảnh. Một dữ liệu tốt cần đảm bảo bộ dữ liệu phải cân bằng và chất lượng hình ảnh.

CÁC NGHIÊN CỨU LIÊN QUAN

1. Tiền xử lý dữ liệu

1.1. Khái niệm và các kĩ thuật tiền xử lý dữ liệu

1.1.1. Khái niệm

Ngày nay, cơ sở dữ liệu trong thế giới thực rất dễ bị ảnh hưởng bởi nhiễu, thiếu và không nhất quán do kích thước rất lớn của chúng (thường là vài gigabyte trở lên) và nguồn gốc dữ liệu có thể đến từ nhiều nguồn không đồng nhất. Bộ dữ liệu có chất lượng thấp sẽ dẫn đến kết quả khai thác dữ liệu chất lượng thấp. [5] Vậy nên, tiền xử lý dữ liệu là điều cần thiết nhằm xử lý dữ liệu thô/gốc (raw/original data) để cải thiện chất lượng dữ liệu (data quality) và kết quả của khai thác dữ liệu. [6]

1.1.2. Chất lượng dữ liệu

Chất lượng dữ liệu (data quality) cần đảm bảo các yếu tố:

- Tính chính xác (accuracy): giá trị dữ liệu đúng, không có sai sót trong quá trình thu thập, nhập hay chuyển đổi dữ liệu.
- Tính hiện hành (currency/timeliness): giá trị dữ liệu không bị lỗi thời.
- Tính toàn vẹn (completeness): tất cả các giá trị của một thuộc tính đều được ghi nhận.
- Tính nhất quán (consistency): tất cả giá trị dữ liệu đều được biểu diễn như nhau trong mọi trường hợp. [6]

1.1.3. Các kỹ thuật tiền xử lý dữ liệu

1.1.3.1. Làm sạch dữ liệu

- Xử lý dữ liệu bị thiếu

Dữ liệu không ở trạng thái sẵn có khi cần dùng có thể xảy ra do nguyên nhân chủ quan từ con người. Một số giải pháp: bỏ qua, xử lý thủ công, xử lý tự động bằng cách dùng

các giá trị thay thế (trị phổ biến, hằng số toàn cục, trung bình toàn cục…) Ngoài ra, có thể ngăn chặn dữ liệu bị thiếu bằng cách cải thiện quy trình thu thập và nhập liệu.

- Nhận diện phần tử biên và giảm nhiễu

Là kỹ thuật nhận biết và loại bỏ những dữ liệu không theo đặc trưng chung của tập dữ liệu. Một số giải pháp để nhận biết phần tử biên: dựa trên phân bố thống kê (statistical distribution – based), khoảng cách (distance-based), mật độ (density-based), độ lệch (deviation-based). Từ kết quả nhận biết, tiếp tục thực hiện giảm nhiễu cho dữ liệu thông qua:

  - Bining: làm mịn giá trị dữ liệu thông qua các giá trị xung quanh nó.
  - Hồi quy (regression): thường dùng hồi quy tuyến tính để tìm mối quan hệ giữa các thuộc tính (hoặc biến), từ đó một số thuộc tính được dùng để dự đoán thuộc tính khác.
  - Phân tích cụm (cluster analysis): các giá trị tương tự nhau sẽ được gom thành cụm trực quan, các giá trị không nằm trong các cụm sẽ được xem là nhiễu.
- Xử lý dữ liệu không nhất quán

Dữ liệu không nhất quán là dữ liệu được ghi nhận theo nhiều cách khác nhau cho cùng một đối tượng. Một số giải pháp: phát hiện dựa trên phân tích, ràng buộc dữ liệu, điều chỉnh thủ công hoặc tự động.

1.1.3.2. Tích hợp dữ liệu

Là kỹ thuật tích hợp dữ liệu từ các nguồn khác nhau vào kho dữ liệu. Kỹ thuật này yêu cầu đảm bảo tính tương đương của thông tin giữa các nguồn, nhằm hỗ trợ việc giảm thiểu dư thừa dữ liệu, phát hiện dữ liệu không nhất quán. Từ đó, nâng cao hiệu quả khai thác dữ liệu về độ chính xác và thời gian xử lý.

1.1.3.3. Biến đổi dữ liệu

Biến đổi dữ liệu để thích hợp với kỹ thuật khai thác dữ liệu:

- Làm trơn (mượt, mịn) dữ liệu (smoothing) → giảm nhiễu.
- Kết hợp dữ liệu (aggregation) → thu giảm dữ liệu.
- Tổng quát hoá (generalization) → thu giảm dữ liệu.
- Chuẩn hoá (normalization) → dữ liệu thuộc một miền giá trị định trước.
- Xây dựng thuộc tính/đặc trưng (attribute/feature construction).

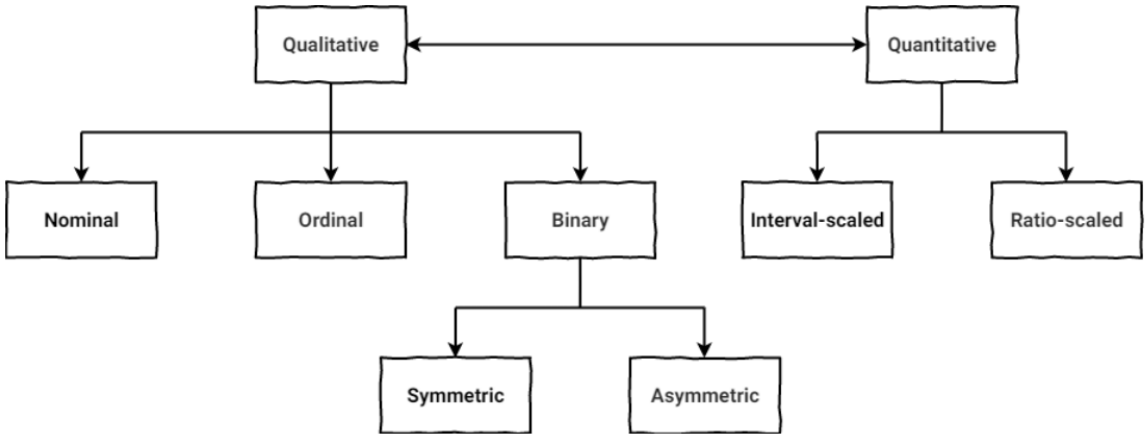
1.1.3.4. Thu giảm dữ liệu

Là kỹ thuật thu giảm kích thước dữ liệu (giảm số phần tử) sao cho đảm bảo tính đúng dữ liệu nhằm tối ưu quá trình khai thác dữ liệu, bằng cách tổng hợp dữ liệu, lựa chọn thuộc tính thích hợp, thu giảm chiều, thu giảm lượng, rời rạc hóa, tạo cây phân cấp ý niệm.

1.1.3.5. Rời rạc hóa dữ liệu

Là kỹ thuật làm giảm số lượng giá trị của một thuộc tính liên tục bằng cách chia miền giá trị của thuộc tính thành các khoảng (binning). Các nhãn được gán cho các khoảng này và được dùng thay thế giá trị thực của thuộc tính.

1.1.3.6. Tạo cây phân cấp ý niệm



Hình 6 Cây phân cấp ý niệm về phân loại thuộc tính dữ liệu [6]

Kỹ thuật này hỗ trợ khai thác dữ liệu ở mức trừu tượng, trong đó dạng rời rạc hoá dữ liệu có độ hữu dụng cao. Mục đích dùng để thu giảm dữ liệu bằng việc thu thập và thay thế các ý niệm cấp thấp bằng các ý niệm cấp cao.

1.2. Tiền xử lý dữ liệu được sử dụng trong đồ án

Tăng cường dữ liệu (data augmentation) được sử dụng để làm giảm việc mất cân bằng và bị giới hạn của kích thước dữ liệu. Một số kỹ thuật tăng cường dữ liệu phổ biến như lật, cắt, phóng đại, xoay góc, nội suy, dịch và chèn nhiễu đã được thực hiện cho các nghiên cứu trước đó. Tuy nhiên, các phương pháp được sử dụng cho hình ảnh ảnh tự nhiên sẽ không hữu hiệu đối với hình ảnh y khoa. Hơn nữa, cường độ (intensities) đóng vai trò quan trọng trong các phương thức liên quan tới hình ảnh y khoa. Do đó, các cách tiếp cận nên được lựa chọn dựa trên bộ dữ liệu của nghiên cứu. Vì hình ảnh mô bệnh học có đối xứng xoay và phản chiếu, nên có khả năng loại bỏ một số đặc tính vốn có khỏi hình ảnh nếu các kỹ thuật nâng cao được sử dụng. Do đó, đối với quá trình thực hiện full training và transfer learning, chỉ có xoay góc được sử dụng để tăng cường dữ liệu. Các phép quay được thực hiện hướng về tâm của ảnh với số đo góc quay lần lượt là 90°, 180° và 270°.

2. Các phương pháp rút trích đặc trưng

2.1. Đặc trưng ảnh

2.1.1 Đặc trưng màu sắc

Màu sắc là một đặc trưng nổi bật và được sử dụng phổ biến nhất trong tìm kiếm ảnh theo nội dung. Mỗi một điểm ảnh (thông tin màu sắc) có thể được biểu diễn như một điểm trong không gian màu sắc ba chiều. Các không gian màu sắc thường dùng là: RGB, Munsell, CIE, HSV. [7]

2.1.1.1. Color histogram

Là histogram biểu diễn bằng một hàm rời rạc:

$$h(r_k) = n_k.$$

Trong đó: r_k là mức xám thứ k; n_k là số lượng điểm ảnh có mức xám r_k. [8]

Thông thường histogram được chuẩn hóa như sau:

$$h(r_k) = \frac{n_k}{n}.$$

Với n là tổng số pixel trong ảnh. [8]

Ưu điểm của color histogram:

- Phản ánh phân bố màu sắc trong ảnh
- Bất biến với phép quay ảnh (không làm méo)
- Bất biến với phép dịch ảnh

Nhược điểm của color histogram:

- Không phản ánh tính không gian
- Nhạy với phép thay đổi ánh sáng

2.1.1.2. Color moments

Nếu coi giá trị mức xám tại mỗi điểm ảnh trong ảnh là biến ngẫu nhiên thì histogram của ảnh chính là hàm mật độ phân bố xác suất của biến ngẫu nhiên được xác định bởi các giá trị sau:

- Mean (giá trị kỳ vọng, giá trị trung bình).
- Standard deviation (độ lệch của các điểm so với giá trị trung bình).
- Skewness (độ lệch phân bố).

⇨ Giá trị moments các cấp.

Giá trị trung bình (mean): $E_i = \sum_N^{j=1} \frac{1}{N} p_{ij}$

Moment cấp 2: độ lệch chuẩn (standard deviation): $\sigma_i = \sqrt{\left(\frac{1}{N}\sum_N^{j=1}(p_{ij} - E_i)^2\right)}$

Moment cấp 3: độ lệch phân bố (Skewness): $S_i = \sqrt[3]{\left(\frac{1}{N}\sum_N^{j=1}(p_{ij} - E_i)^3\right)}$

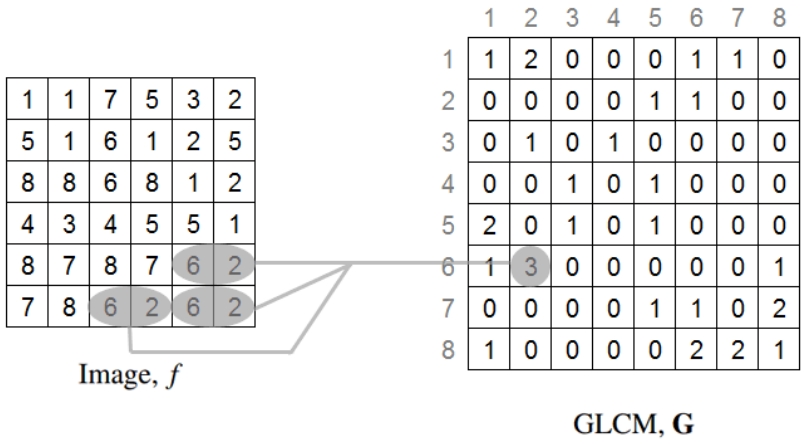
Trong đó: p_{ij} là giá trị của kênh màu i tại pixel có vị trí j trong ảnh. [8]

2.1.2. Đặc trưng về kết cấu

Kết cấu ảnh thể hiện sự sắp xếp về mặt không gian của các giá trị độ chói (ảnh đa mức xám), màu sắc (ảnh màu). Kết cấu ảnh được tạo từ các phần tử kết cấu gọi là texel. Có hai loại kết cấu ảnh là kết cấu tự nhiên và kết cấu nhân tạo.

Với hai cách tiếp cận:

- Tiếp cận cấu trúc: thường áp dụng cho phân tích các kết cấu nhân tạo. Ví dụ như ảnh được tạo thành từ các phần tử kết cấu (texel) hay các mẫu (pattern), phân tích tương quan giữa các texel và các pattern.
- Tiếp cận thống kê: là tính toán các giá trị moment các cấp, hoặc ma trận đồng hiện (Co-occurrence matrix). [8]



Hình 7 Ví dụ về ma trận đồng xuất hiện [9]

2.1.3. Đặc trưng về hình dạng

Hình dạng của một ảnh hay một vùng là một đặc trưng quan trong trong việc xác định và phân biệt ảnh trong nhận dạng mẫu. Mục tiêu chính của biểu diễn hình dạng trong nhận dạng mẫu là đo thuộc tính hình học của một đối tượng được dùng trong phân lớp, so sánh và nhận dạng đối tượng. [7]

Cách thực hiện:

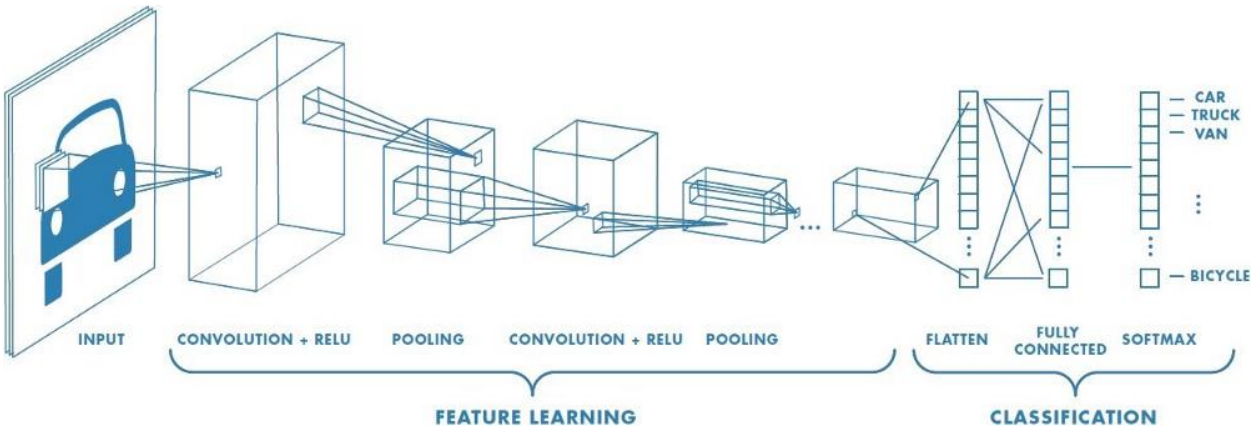
- Bước 1: Phát hiện các thuộc tính hình dạng (biên cạnh)
  - Phương pháp gradients.
  - Phương pháp laplacian.
- Bước 2: Mã hóa các đường biên
  - Chaincode.

2.2. Đặc trưng học sâu

2.2.1. Mạng thần kinh tích chập

Mạng thần kinh tích chập (ConvNet/CNN) là một loại mạng Nơ-ron được áp dụng phổ biến nhất trong các lĩnh vực như nhận dạng và phân loại hình ảnh. Nó có thể lấy hình ảnh đầu vào và gán một bộ trọng số cho các đặc trưng/đối tượng khác nhau trong hình ảnh và có thể phân biệt được từng đặc trưng/đối tượng này với nhau. Tiền xử lý trong CNN được yêu cầu ít hơn nhiều so với các thuật toán phân loại khác. Với các phương thức sơ khai, các bộ lọc được thiết kế thủ công, thì mạng thần kinh tích chập có khả năng tự học để chọn ra các bộ lọc/đặc trưng tối ưu nhất.

Kiến trúc CNN tương tự như mô hình kết nối các nơ-ron trong bộ não con người và nó được lấy cảm hứng từ hệ thống vỏ thị giác trong bộ não (visual cortex).



Hình 8 Kiến trúc mạng CNN [10]

Mạng thần kinh tích chập cấu thành bởi các lớp:

a. Lớp tích chập (Convolution Layer – Conv)
b. Lớp gộp (Pooling Layer – Pool)
c. Lớp kết nối đầy đủ (Fully Connected Layer – FC)

Mạng thần kinh tích chập có vai trò giảm chiều hình ảnh thành một dạng dễ xử lý hơn mà không đánh mất đi các đặc trưng quan trọng của hình ảnh để có thể dự đoán tốt (Good Prediction). [11]

2.2.2. Lớp tích chập

Mục tiêu của phép tính tích chập là lớp đầu tiên trích xuất các đặc trưng từ hình ảnh đầu, không nhất thiết chỉ giới hạn trong một lớp tích chập. Đa số, lớp tích chập đầu tiên chịu trách nhiệm nắm bắt các đặc trưng cấp thấp như màu sắc (colors), hướng dốc (gradient orientation) ... Với các lớp tích chập được thêm vào sau, mô hình được thiết kế để nắm bắt các đặc trưng cấp cao hơn, do đó mà mạng có thể học được những thông tin, những hình ảnh như cách mà con người hiểu về hình ảnh.

Input đầu vào là một bức ảnh được biểu diễn bởi ma trận pixel với kích thước: [H x W x D] với W là chiều rộng, H là chiều cao, D là độ sâu hay có thể hiểu là số lớp màu của ảnh. Ảnh đầu vào được cho qua một bộ lọc chạy dọc bức ảnh.

Kích thước bộ lọc (Filter) là một tham số quan trọng của lớp tích chập, nó tỷ lệ thuận với tham số cần học tại mỗi lớp tích chập và là tham số quyết định receptive field của lớp này. Kích thước phổ biến của bộ filter là 3x3 hoặc 5x5 vì kích thước nhỏ thì rút trích đặc trưng có tính cục bộ cao, phát hiện được các đặc trưng nhỏ, bắt được những phần cơ bản của ảnh và với các số lẻ sẽ xác định tâm một điểm ở tầng phía trước. Sau đó áp dụng phép tích vô hướng để tính toán, cho ra một giá trị duy nhất. Đầu ra của phép tích chập là một tập các giá trị ảnh được gọi là mạng đặc trưng. Với bộ lọc (Filter) khác nhau kết quả sẽ cho ra khác nhau.

Receptive field là vùng mà một nơ-ron có thể nhìn thấy để đưa ra quyết định hay là phần trên bức ảnh để tính tích chập với filter.

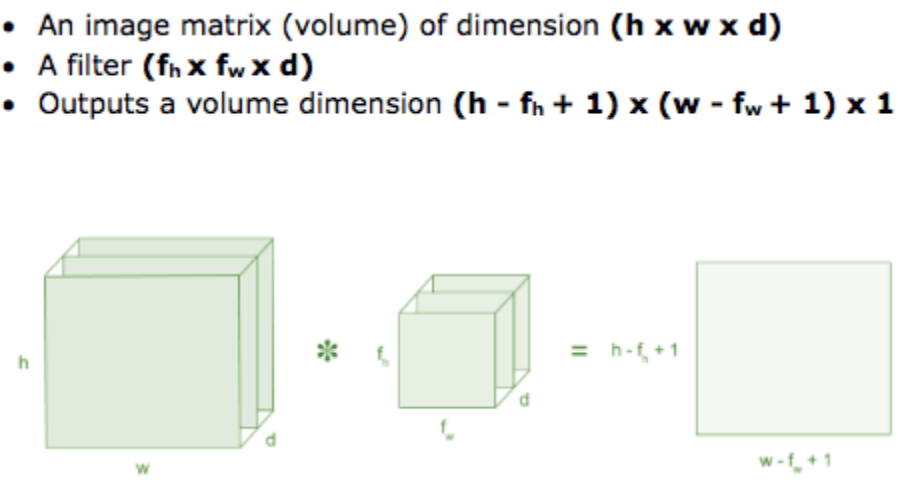
Feature map là ma trận đầu ra của quá trình tích chập trước đó. Mỗi giá trị ở feature map được tính bằng tổng của tích các phần tử tương ứng của bộ filter và receptive field trên ảnh. Để tính được tất cả các giá trị của feature map thì filter phải trượt từ trái sang phải, từ trên xuống dưới với bước nhảy stride.

Kích thước đầu ra của ảnh với mỗi layer được tính như sau:

$$\left(\frac{W - F + 2P}{S} + 1\right) \times \left(\frac{H - F + 2P}{S} + 1\right) \times K$$

Trong đó:

- W, H: Chiều rộng và chiều cao.
- F: Kích thước bộ lọc.
- S: Giá trị Stride.
- P: Số lượng zero-padding thêm vào viền ảnh.
- K: Số lượng bộ lọc.



Hình 9 Mô phỏng mạng nơ ron tích chập [11]

2.2.3. Các siêu tham số của tích chập

Moder Hyperparameter là một cấu hình nằm ngoài mô hình và giá trị của nó không thể được ước tính từ dữ liệu. Nó được sử dụng trong quá trình huấn luyện, giúp mô hình tìm được các parameters hợp lý nhất, thường được chọn thủ công bởi những người tham gia trong việc huấn luyện mô hình, chúng có thể được định nghĩa dựa trên một vài chiến lược heuristics (bằng kinh nghiệm).

Stride là độ trượt, hiểu đơn giản là khoảng cách dịch chuyển của bộ lọc sau mỗi lần tính. Ví dụ Stride = 3 thì sau khi tính xong tại 1 vùng ảnh, nó sẽ dịch sang phải 3 pixel, tương tự với dịch xuống dưới.

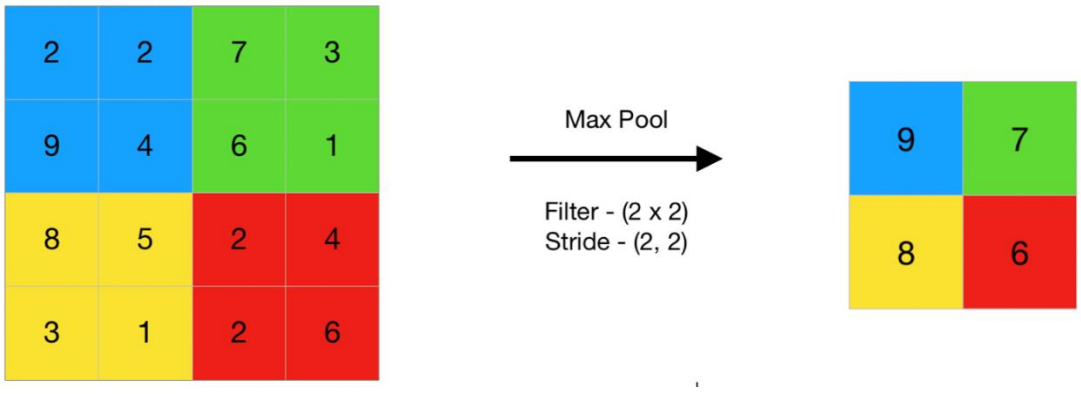
Zero-Padding là việc thêm các giá trị 0 ở xung quanh biên ảnh, để đảm bảo phép tích chập được thực hiện đủ trên toàn ảnh.

Batch size là số lượng dữ liệu Mini-Batch Gradient Descent sử dụng trong 1 lần để cập nhật tham số. Batch size càng lớn thì càng tận dụng được tính toán vectorization. Nếu batch size = 1 mô hình hội tụ nhanh hơn, tuy nhiên hàm mất mát dao động quanh minimum chứ không hội tụ về nó được, cũng không tận dụng được tính toán vectorization.

2.2.4. Lớp gộp

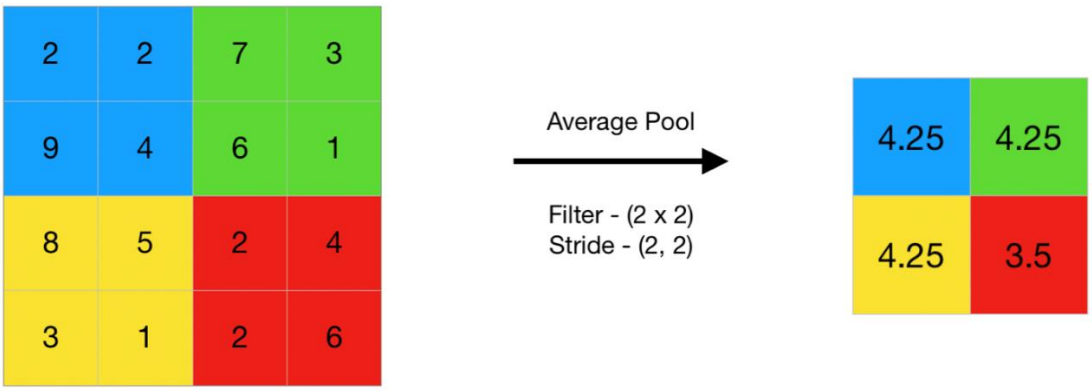
Pooling Layer có chức năng làm giảm chiều không gian của đầu vào và giảm độ phức tạp tính tạo của model, ngoài ra Pooling Layer còn giúp kiểm soát hiện tượng Overffiting. Có nhiều loại Pool Layer như: L2 pooling, Max pooling và Average pooling và Sum pooling. Trong đó, Max pooling và Average pooling là những dạng pooling phổ biến nhất

- Max pooling
    - Trả về giá trị lớn nhất hoặc lấy tổng trung bình từ phần hình ảnh được bao phủ bởi bộ lọc.
    - Loại bỏ các nguồn nhiễu và thực hiện khử nhiễu song song với giảm kích thước.
    - Bảo toàn các đặc trưng đã phát hiện
    - Được sử dụng thường xuyên



Hình 10 Max pooling trong CNN [12]

- Average pooling
    - Trả về giá trị trung bình của tất cả giá trị từ phần hình ảnh được bao phủ bởi bộ lọc.
    - Giảm kích thước feature map
    - Được sử dụng trong mạng LeNet



Hình 11 Average pooling trong CNN [12]

2.2.5. Hàm kích hoạt

Hàm kích hoạt là những hàm phi tuyến tính được áp dụng vào đầu ra của lớp tích chập trong tầng ẩn của một mô hình mạng, và được sử dụng làm input data cho tầng tiếp theo. Nếu không có hàm phi kích hoạt phi tuyến tính, khả năng dự đoán của lớp tích chập sẽ bị giới hạn và giảm đi rất nhiều, sự kết hợp của các hàm giữa các tầng ẩn là để giúp mô hình học được các quan hệ phi tuyến tính phức tạp tiềm ẩn trong dữ liệu. Một số các hàm kích hoạt phổ biến sau là: Sigmoid, Relu, Softmax.

- Sigmoid: nhận đầu vào là một số thực và chuyển thành một giá trị trong khoảng (0;1). Đầu vào là số thực âm rất nhỏ sẽ cho đầu ra tiệm cận với 0, nếu đầu vào là một số thực dương lớn sẽ cho đầu ra là một số tiệm cận với 1.

$$sigmoid(x) = \frac{e^x}{1 + e^x}$$

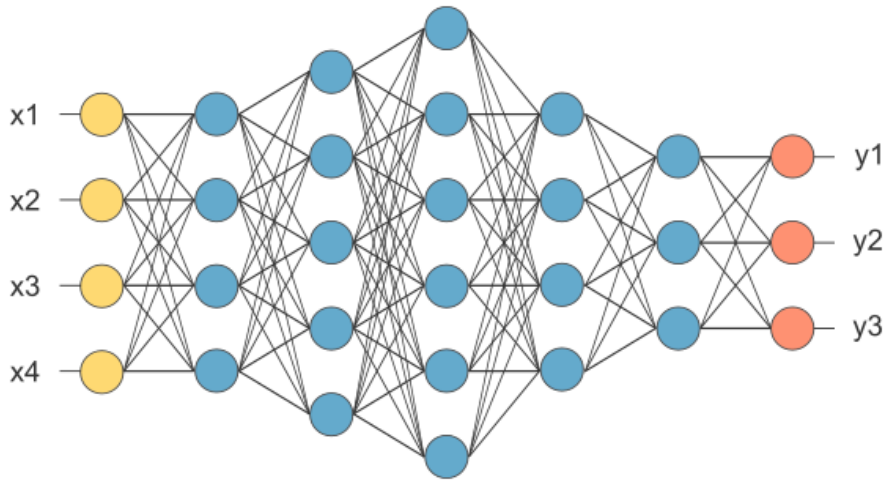
Hàm Sigmoid là một hàm liên tục và đạo hàm của nó cũng khá đơn giản, cho ra kết quả đẹp, dẫn đến việc áp dụng hàm vào mô hình mạng dễ dàng trong việc xây dựng vô hình và cập nhật tham số dựa trên back-propagation.

- Relu: Rectified linear unit (ReLU) là một hàm kích hoạt được sử dụng trên tất cả các thành phần. Mục đích của nó là tăng tính phi tuyến tính cho mạng. Những biến thể khác của ReLU có thể kể đến là PreLU, Noisy ReLU, Leaky ReLU, ELUs. Trong quá trình sử dụng cần lưu ý đến vấn đề learning rate và theo dõi dead unit
- Softmax: softmax có thể được coi là một hàm logistic tổng quát lấy đầu vào là một vector chứa các giá trị $x \epsilon \mathbb{R}^n$ và cho ra là một vector gồm các xác suất $p \epsilon \mathbb{R}^n$ thông qua một hàm softmax ở cuối kiến trúc mạng. Nó được định nghĩa như sau:

$$p = \begin{pmatrix} p_1 \\ \vdots \\ p_n \end{pmatrix} với\ p_i = \frac{e^{x_i}}{\sum_{j=1}^{n} e^{x_i}}$$

2.2.6. Lớp kết nối đầy đủ

Đầu vào lớp này là ảnh được dàn phẳng thành một vector thay vì mảng nhiều chiều như trước. Mỗi một nơ-ron của lớp này sẽ liên kết với mọi nơ-ron của lớp khác. Vector đầu ra đã được làm phẳng sẽ được đưa vào một mạng nơ-ron suy luận tiến (feedforward) và phương pháp truyền ngược (backpropagation) được áp dụng cho quá trình huấn luyện. Qua một loạt lần lặp, mô hình có thể phân biệt giữa các đặc trưng cốt lõi và các đặc trưng không quan trọng trong hình ảnh. Tại layer cuối cùng sẽ sử dụng softmax để phân loại đối tượng dựa vào vector đặc trưng đã được tính toán trước đó.



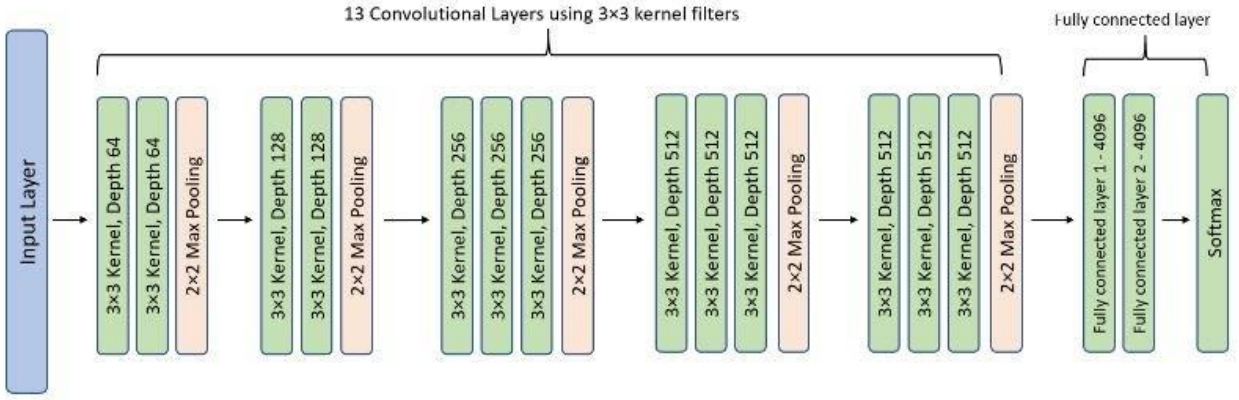
Hình 12 Lớp kết nối đầy đủ [13]

3. VGGNET

VGGNets là tên viết tắt của Visual Geometry Group là kiến trúc mạng lưới thần kinh tích chập với nhiều lớp. Kiến trúc VGG là cơ sở của các mô hình nhận dạng đối tượng, được phát triển dựa trên nhiều tác vụ và bộ dữ liệu ngoài ImageNet, hiện tại VGG là một trong những kiến trúc nhận dạng hình ảnh phổ biến nhất. Kiến trúc VGG bao gồm các khối, trong đó mỗi khối bao gồm các lớp tích chập và Max Pooling. VGG có nhiều biến thể và 2 loại phổ biến là VGG16 và VGG19 trong đó 16, 19 lần lượt là số lớp trong mỗi lớp. Trong mạng lưới thần kinh tích chập, khi số lượng lớp tăng lên thì khả năng của mô hình sẽ phù hợp với các chức năng phức tạp hơn. [14] Mô hình VGG hay VGGNets có kích thước chuẩn hóa đầu vào của hình ảnh là 224 x 224, lớp tiền xử lý dữ liệu lấy hình ảnh RGB với các giá trị pixel trong phạm vi 0 - 255, các hình ảnh đầu vào sau khi tiền xử lý truyền qua lớp trọng số, các hình ảnh đào tạo được truyền qua các lớp chập (convolution).

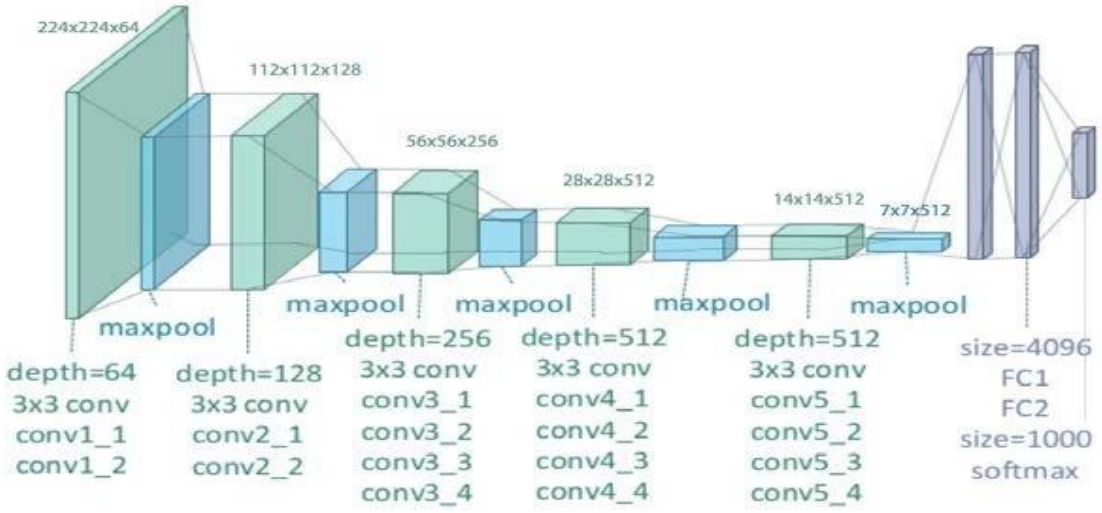
Mô hình VGG16 gồm 16 lớp, đây là mô hình mạng lưới thần kinh kết hợp được đề xuất bởi A. Zisserman và K. Simonyan từ đại học Oxford, họ đã xuất bản mô hình trong bài nghiên cứu có tiêu đề: “Very Deep Convoluation Networks for Large-Scale Image Recognition”. Mô hình đạt được độ chính xác thử nghiệm ~92.7% trong ImageNet, đây là mô hình phổ biến nhất được gửi tới ILSVRC-2014. Có tổng cộng 13 lớp tích chập và 3 lớp

fully connected và 5 lớp gộp trong kiến trúc VGG16. VGG có các bộ lọc nhỏ 3*3 với độ sâu nhiều hơn thay vì có các bộ lọc lớn như 7 x 7 của ZFNet. [15]



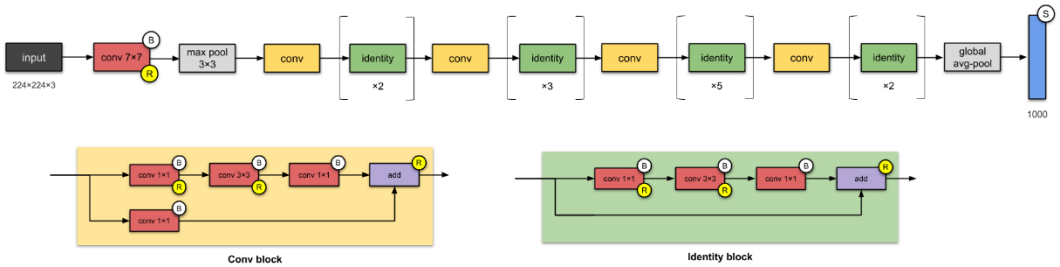
Hình 13 Kiến trúc mạng VGG16 [16]

Khác với VGG16 thì VGG19 bao gồm 16 lớp tích chập với 3 lớp fully connected và 5 pooling layers; giống với VGG16 với 2 lớp tích chập và 1 lớp Maxpool ở hai block đầu tiên nhưng VGG19 có sự thay đổi trong kiến trúc mạng ở block thứ 3, thứ 4 và thứ 5. Chính nhờ điểm cải thiện này, mà mạng VGG19 học sâu hơn, nói cách khác mạng học được nhiều đặc trưng của ảnh hơn.



Hình 14 Kiến trúc mạng VGG19 [17]

4. ResNet50

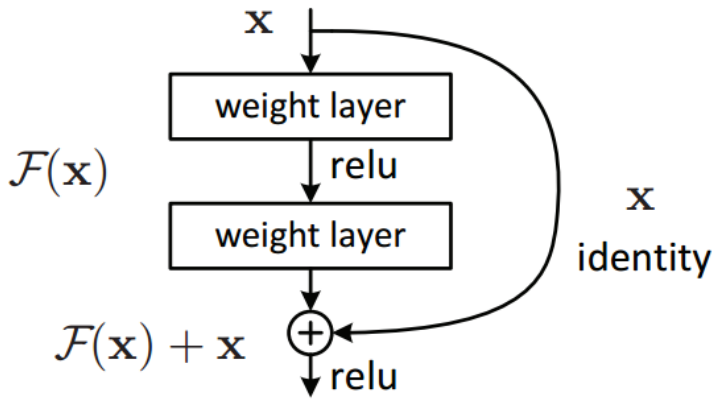


Hình 15 Kiến trúc ResNet bao gồm 2 khối đặc trưng là khối tích chập (Conv Block) và khối xác định (Identity Block) [18]

ResNet là tên viết tắt của Residual Network được phát triển bởi Microsoft năm 2015 với bài báo "Deep residual learning for image recognition", ResNet chiến thắng cuộc thi ImageNet ILSVRC năm 2015 với tỷ lệ lỗi là 3.57%. Ngoài ra nó còn đứng đầu tiên trong cuộc thi ILSVRC và COCO 2015 với ImageNet Detection, ImageNet localization, Coco detection và Coco segmentation. Là mạng được thiết kế để làm việc với hàng trăm hàng nghìn lớp tích chập. ResNet có cấu trúc gần giống VGG với nhiều lớp xếp chồng làm cho mô hình này học sâu hơn.

Một mạng noron có khả năng làm xấp xỉ mọi hàm với dữ liệu huấn luyện được cung cấp, tuy nhiên xấp xỉ tốt dữ liệu không phải là mục tiêu then chốt mà cần mô hình có khả năng tổng quát hóa dữ liệu. Những kiến trúc trước đây thường cải tiến độ chính xác nhờ gia tăng chiều sâu của mạng CNN. Tuy nhiên độ sâu mạng không chỉ đơn giản là xếp chồng các lớp lại với nhau, rất khó để huấn luyện vì vấn đề vanishing gradient bởi độ dốc truyền ngược trở lại các lớp trước đó, phép nhân lặp đi lặp lại có thể làm cho độ dốc cực nhỏ. Làm cho hiệu suất mạng bị bão hòa hoặc giảm hiệu quả.

Các nhà nghiên cứu đã giải quyết vấn đề này trên ResNet bằng cách sử dụng kết nối tắt. Các kết nối tắt (skip connection) giúp giữ thông tin không bị mất bằng cách kết nối đồng nhất để xuyên qua một hay nhiều lớp. Một khối như vậy gọi là một Residual Block.



Hình 16 Residual Block [18]

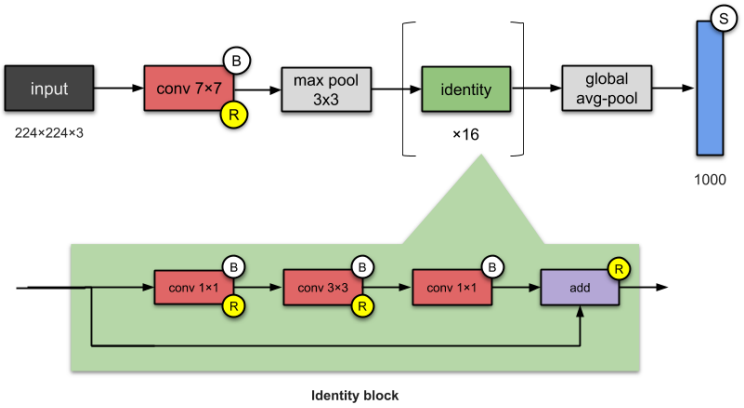
Ý tưởng của Residual Block là Feedfoward x(input) qua một số lớp conv-max-conv, thu được F(x) sau đó thêm x vào H(x) = F(x) + x. Mô hình sẽ dễ học hơn khi thêm đặc trưng từ layer trước vào, việc này chống lại việc đạo hàm bằng 0. Với H(x) là giá trị dự đoán, F(x) là giá trị thật (nhãn), muốn H(x) bằng hoặc xấp xỉ F(x). Việc F(x) có được từ x như sau:

$$\mathrm{X} -> \ \mathrm{weight1} -> \mathrm{ReLU} -> \ \mathrm{weight2}$$

Giá trị H(x) có được bằng cách:

$$F(x) + x \rightarrow ReLU \text{ [18]}$$

So với VGGNet, ResNet ít phức tạp hơn vì chúng có ít bộ lọc hơn. Mặc dù có kiến trúc khối kế thừa lại từ GoogleNet nhưng ResNet lại dễ tóm tắt và triển khai hơn rất nhiều vì kiến trúc cơ sở của nó chỉ gồm các khối tích chập và khối xác định. Ta có thể đơn giản hóa kiến trúc của ResNet-50 như hình bên dưới:



Hình 17 Kiến trúc tóm tắt của mạng ResNet-50 [18]

5. Phân lớp

5.1. Bài toán phân lớp

Bài toán phân lớp là quá trình phân lớp một đối tượng dữ liệu vào một hay nhiều lớp đã cho trước nhờ một mô hình phân lớp.

- Mô hình này được xây dựng dựa trên môt tập dữ liệu được xây dựng trước đó có dán nhãn (tập huấn luyện).
- Quá trình phân lớp là quá trình gán nhãn cho đối tượng dữ liệu.

Như vậy, nhiệm vụ của bài toán phân lớp là cần tìm một mô hình phân lớp để khi có dữ liệu mới thì có thể xác định được dữ liệu đó thuộc vào phân lớp nào.

Ví dụ về các vấn đề phân lớp bao gồm:

- Đưa ra một ví dụ, hãy phân lớp xem nó có phải là thư rác hay không.
- Cho một ký tự viết tay, hãy phân nó thành một trong những ký tự đã biết.

Từ góc độ mô hình hóa, phân lớp yêu cầu một tập dữ liệu đào tạo với nhiều mẫu về đầu vào và đầu ra để học hỏi. Một mô hình sẽ sử dụng tập dữ liệu huấn luyện và sẽ tính toán cách ánh xạ các mẫu tốt nhất về dữ liệu đầu vào, vào các nhãn lớp cụ thể. Như vậy, tập dữ liệu huấn luyện phải đủ lớn để đại diện cho vấn đề và có nhiều mẫu về mỗi nhãn lớp. [19]

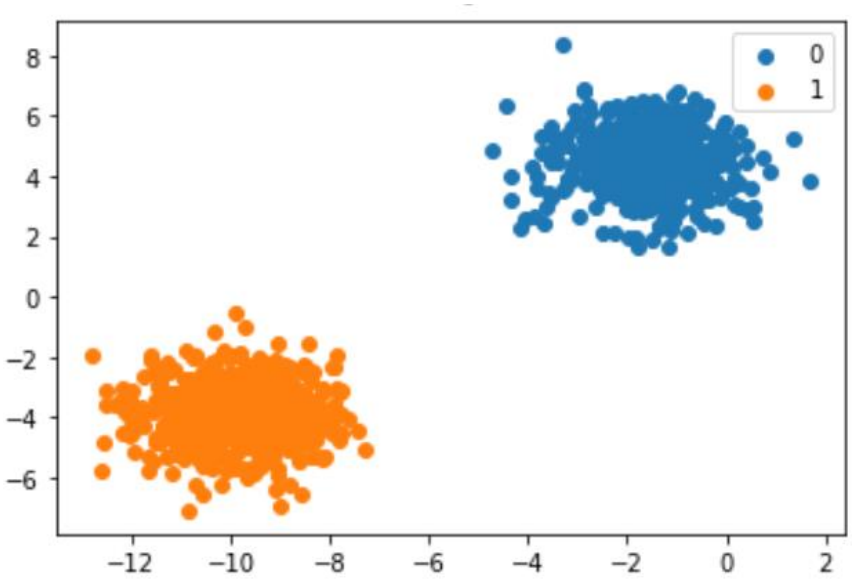
Có nhiều loại thuật toán phân lớp khác nhau để mô hình hóa các bài toán mô hình dự báo phân lớp. Một số bài toán phân lớp dữ liệu như: Phân lớp nhị phân (binary classification), phân lớp đa lớp (multiclass classification), phân loại đa nhãn, phân loại không cân bằng…

4.1.1. Phân lớp nhị phân:

Là bài toán gán nhãn dữ liệu cho đối tượng vào 2 nhãn lớp khác nhau dựa vào việc dữ liệu đó có hay không có đặc trưng (feature) của bộ phân lớp.

Thông thường, các nhiệm vụ phân loại nhị phân liên quan đến một lớp là trạng thái bình thường và một lớp khác là trạng thái bất thường. Lớp cho trạng thái bình thường được gán nhãn lớp 0 và lớp có trạng thái bất thường được gán nhãn lớp 1 [19].

- Một số bài toán phân lớp điển hình như:
    - Xét nghiệm bệnh dương tính hay âm tính
    - Dự đoán giá vàng tăng hay giảm
    - Phân loại thư điện tử là thư bình thường hay thư rác



Hình 18 Đồ thị phân tán của phân lớp nhị phân [20]

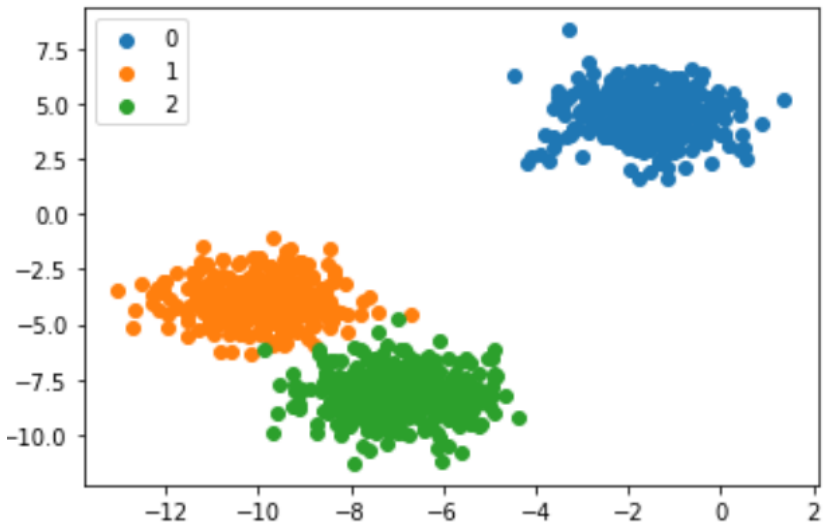
- Các thuật toán phổ biến có thể được sử dụng để phân lớp nhị phân bao gồm: Hồi quy logistics, K láng giềng gần nhất, Cây quyết định, Support vector machine, Naïve Bayes

4.1.2. Phân lớp đa lớp:

Ở bài toán này, dữ liệu được phân chia thành nhiều nhóm khác nhau dựa vào các nhãn cho trước. Không giống như phân lớp nhị phân, phân lớp nhiều lớp không có khái niệm về kết quả bình thường và bất thường. Thay vào đó, các mẫu được phân loại là thuộc về một trong một loạt các lớp đã biết.

Số lượng nhãn lớp có thể rất lớn đối với một số bài toán. Ví dụ: một mô hình có thể dự đoán một bức ảnh thuộc về một trong số hàng nghìn hoặc hàng chục nghìn khuôn mặt trong hệ thống nhận dạng khuôn mặt. Một số ngữ cảnh của bài toán có thể là:

- Phân loại phương tiện giao thông
- Phân loại động vật là con gì
- Phân loại đồ vật gì hay món ăn gì

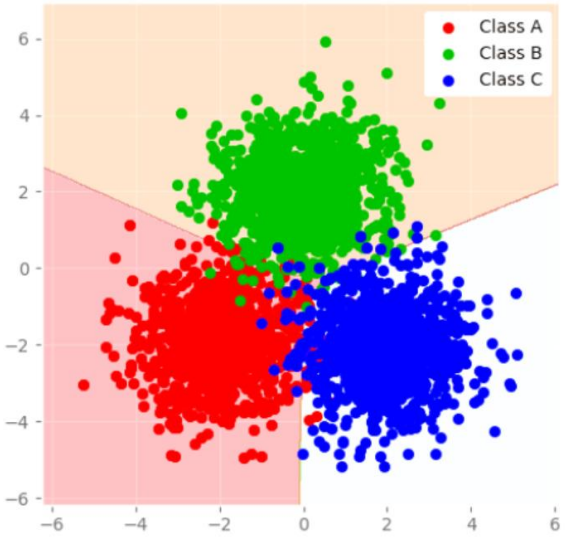


Hình 19 Đồ thị phân tán của phân lớp đa lớp [20]

- Các thuật toán phổ biến có thể được sử dụng để phân lớp nhị phân bao gồm: K láng giềng gần nhất, Cây quyết định, Support vector machine, Naïve Bayes

4.1.3. Phân loại đa nhãn:

Phân lớp đa nhãn cũng liên quan đến nhiệm vụ phân lớp dựa trên nhiều nhãn khác nhau, nhưng một hoặc nhiều nhãn khác nhau có thể được sử dụng để tiên đoán cho mỗi một mẫu.



Hình 20 Đồ thị phân tán của phân loại đa nhãn [21]

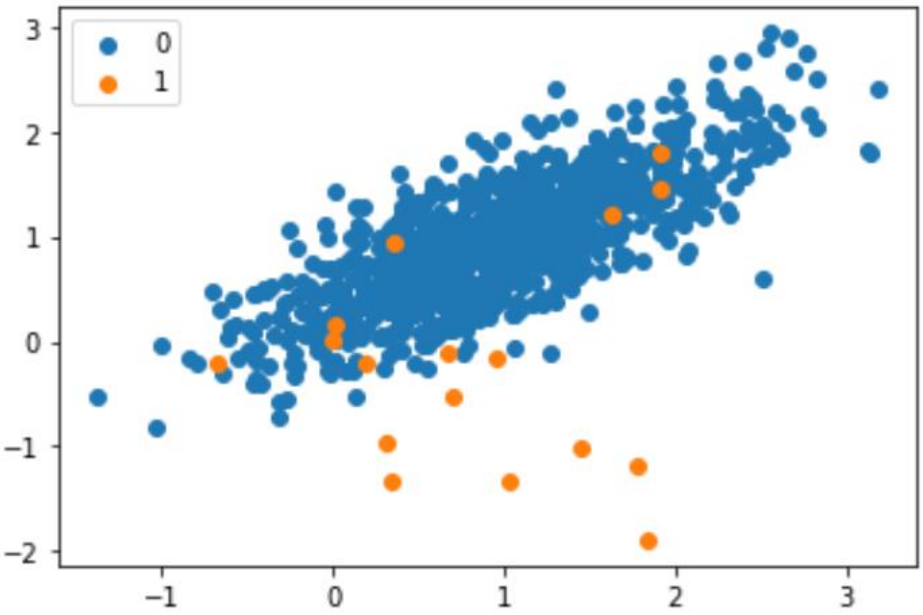
Các phương pháp phổ biến thường được dùng cho phân lớp đa nhãn là: Rừng ngẫu nhiên (Logistic regression), Cây quyết định, Phương pháp thúc đẩy gradient

4.1.4. Phân loại không cân bằng:

Phân lớp không cân bằng đề cập đến các nhiệm vụ phân lớp trong đó số lượng trong mỗi lớp được phân phối không đồng đều [19]. Về cơ bản nó cũng là một dạng phân lớp nhị phân trong đó phần lớn các ví dụ trong tập dữ liệu huấn luyện thuộc về lớp bình thường và một số ít các ví dụ thuộc lớp bất thường.

- Một số ví dụ thực tiễn về phân lớp cân bằng là:
    - Phát hiện lừa đảo

- Phát hiện tin giả
- Chuẩn đoán bệnh trong y khoa



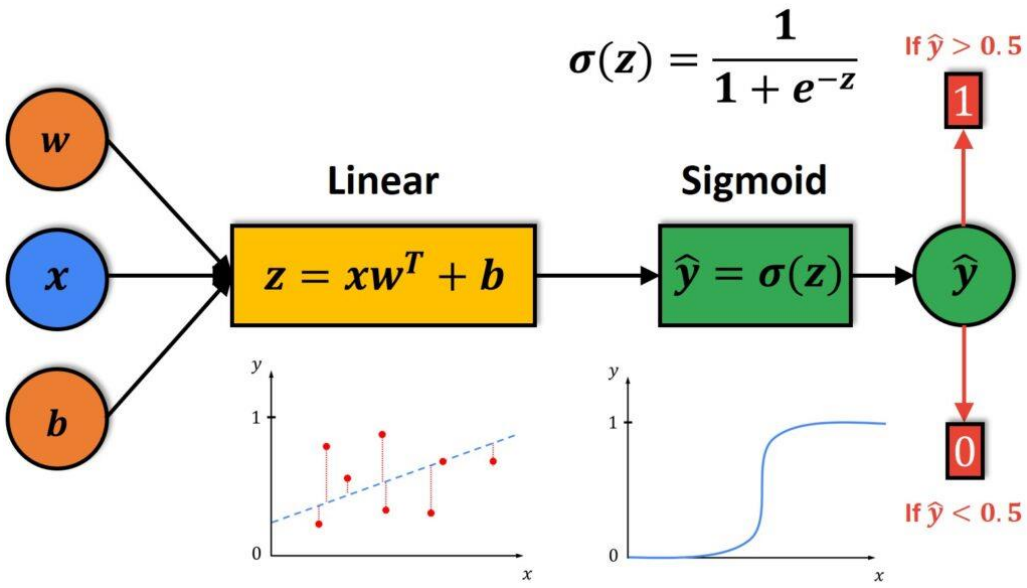
Hình 21 Đồ thị phân tán của phân loại không cân bằng [20]

- Các thuật toán thường dùng trong bài toán này là: Hồi quy logistic chi phí nhạy (cost-sensitive logistic regression), Cây quyết định chi phí nhạy (cost-sensitive decision tree), Support vector machine chi phí nhạy (cost sensitive SVM)

5.2. Thuật toán hồi quy Logistic

Hồi quy logistic là một mô hình hồi quy nhằm dự đoán giá trị đầu ra rời rạc (discrete target variable) y ứng với một véc-tơ đầu vào x. Việc này tương đương với chuyện phân loại các đầu vào x vào các nhóm y tương ứng.

Trong thực tế, hồi quy logistic không chỉ ứng dụng trong dự đoán nhãn mà còn dùng để dự đoán xác suất xảy ra của một vấn đề cụ thể. Trong dự báo thời tiết, hồi quy logistic không chỉ dự đoán một ngày trong tương lai có mưa hay không, mà còn dự đoán được xác suất xảy ra mưa. Tương tự, hồi quy logistic có thể được sử dụng để dự đoán khả năng bệnh nhân mắc một bệnh cụ thể với các triệu chứng nhất định, đó là lý do tại sao nó rất phổ biến trong lĩnh vực y học [22] và được nhóm tác giả sử dụng trong paper này.

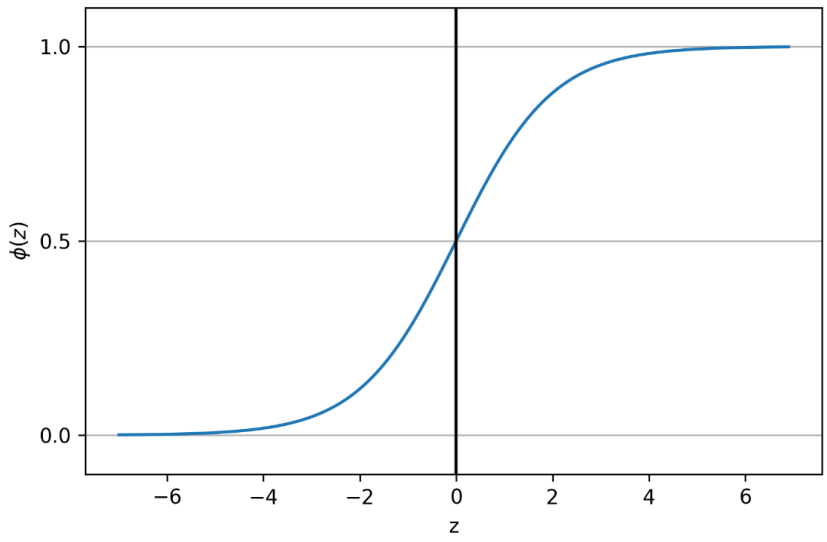


Hình 22 Minh họa mô hình hồi quy Logistic 23 [23]

Cũng giống như hồi quy tuyến tính giả định rằng dữ liệu tuân theo một hàm tuyến tính, hồi quy Logistic mô hình hóa dữ liệu bằng cách sử dụng hàm sigmoid. Nguyên lý hoạt động của hồi quy logistic là đưa đầu ra của mô hình hồi quy tuyến tính đi qua một hàm kích hoạt sigmoid để tất cả giá trị output của hàm giả định sẽ nằm trong khoảng [0,1] và xem giá trị này chính là xác suất biến cố input thuộc một lớp xác định nào đó trong bộ dữ liệu huấn luyện.

Hàm sigmoid được định nghĩa như sau:

$$S(x) = \frac{1}{1+e^{-x}}$$



Hình 24 Đồ thị hàm sigmoid

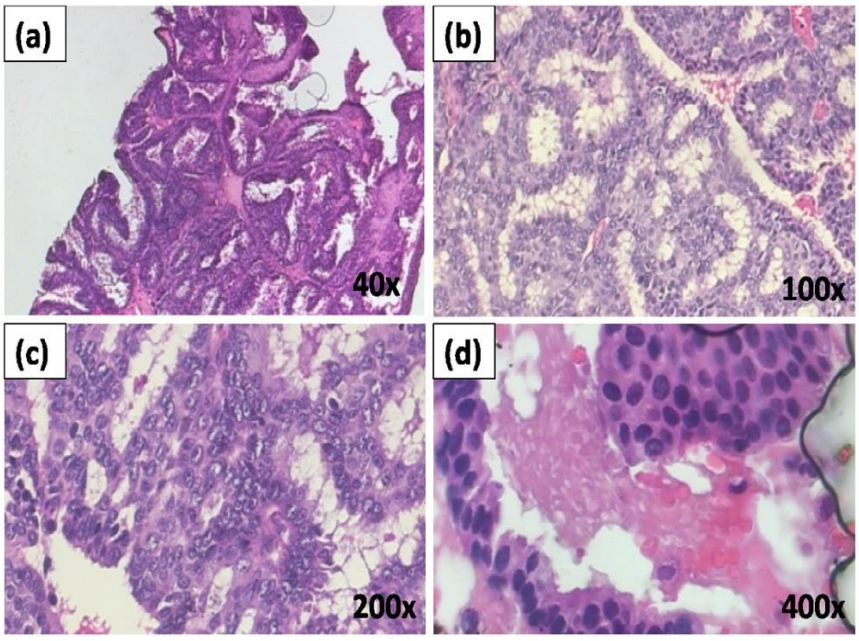
- Ưu điểm của Hồi quy Logistic:
    - Đơn giản, dễ thực hiện và hiệu quả
    - Không đòi hỏi sức mạnh tính toán cao
    - Dễ diễn giải, được sử dụng rộng rãi bởi các nhà phân tích dữ liệu và nhà khoa học
    - Không yêu cầu mở rộng các tính năng
    - Hồi quy logistic cung cấp điểm xác suất cho các quan sát
- Nhược điểm Hồi quy Logistic:
    - Hồi quy logistic không thể xử lý một số lượng lớn các tính năng / biến phân loại.
    - Nó dễ bị tràn bị quá mức.
    - Không thể giải quyết vấn đề phi tuyến tính với hồi quy logistic
    - Không hoạt động tốt với các biến độc lập không tương quan với biến mục tiêu và rất giống nhau hoặc tương quan với nhau.

BỘ DỮ LIỆU BREAK-HIS

1. Tổng quan về dữ liệu

Từ trước đến nay, các bộ dữ liệu về chủ đề y tế được biết đến là một giới hạn cho giới nghiên cứu. Lý do của nó chính là các thủ tục thu thập dữ liệu quá phức tạp và tốn kém. Do đó, rất khó để có thể tìm kiếm và thu thập một tập dữ liệu có đầy đủ các yêu cầu liên quan và được đầu tư trên quy mô lớn.

Trong bài báo này, nhóm tác giả đã sử dụng bộ dữ liệu BreakHis có sẵn và công khai được thu thập, xây dựng với sự hợp tác của Phòng thí nghiệm (P&D), Parana, Brazil. Bộ dữ liệu chứa tổng cộng 7909 mẫu hình ảnh chụp mô ung thư vú được thu thập từ 82 bệnh nhân ở bốn mức độ phóng đại khác nhau (40x, 100x, 200x, 400x). Các mẫu trong bộ dữ liệu được chia thành hai nhóm chính: lành tính và ác tính. Trong đó nhóm lành tính và ác tính bao gồm 2480 và 5429 mẫu tương ứng.



Hình 25 Hình ảnh mô bệnh ung thư vú từ Bộ dữ liệu BreakHis của một bệnh nhân bị ung thư biểu mô nhú (Ác tính) với bốn mức độ phóng đại (a) 40x, (b) 100x (c) 200x và (d) 400x.

2. Xử lý dữ liệu

2.1. Tăng cường dữ liệu

Tăng cường dữ liệu là một cách tiếp cận được sử dụng trong các mô hình sâu để mở rộng bộ dữ liệu để giảm bớt vấn đề kích thước dữ liệu hạn chế. Trong các nghiên cứu trước đó, chúng ta biết đến rất nhiều kỹ thuật tăng cường dữ liệu phổ biến như: lật, cắt xén, mở rộng, xoay dịch hình ảnh một cách ngẫu nhiên, thêm nhiễu… Nhưng tất cả các phương pháp tăng cường được sử dụng cho hình ảnh tự nhiên thì không phù hợp cho tệp dữ liệu hình ảnh y tế vì nó có thể làm mất đi một số tính chất vốn có của hình ảnh.

Đối với cả hai phương pháp đào tạo đầy đủ và học chuyển giao, nhóm tác giả đã sử dụng biện pháp xoay hình ảnh làm kỹ thuật tăng cường dữ liệu. Hình ảnh được thực hiện xoay trung tâm của chúng với ba góc: 90◦, 180◦ và 270◦.

```
2.1   Đi tất cả các thư mục nằm trong 3 tệp 90 -10; 80 - 20; 70 -30
for s in ["traintest73","traintest82","traintest91"]:
      for t in ["test", "train"]:
            for c in ["benign", "malignant"]:
print("./"+s+"/"+t+"/"+c+"/*")
print(len(glob.glob("./"+s+"/"+t+"/"+c+"/*")))
```

```
2.2    Xoáy ảnh và gán tên
for s in ["traintest73","traintest82","traintest91"]:
  for t in ["test", "train"]:
    for c in ["benign", "malignant"]:
      for i in glob.glob("./"+s+"/"+t+"/"+c+"/*"):
        n1 = str("."+i.split(".")[-2])+"_90."+str(i.split(".")[-1])
        n2 = str("."+i.split(".")[-2])+"_180."+str(i.split(".")[-1])
        n3 = str("."+i.split(".")[-2])+"_270."+str(i.split(".")[-1])

        image = cv2.imread(i)

# grab the dimensions of the image and calculate the center of the
# image
        (h, w) = image.shape[:2]
        (cX, cY) = (w // 2, h // 2)
# rotate our image by 90 degrees around the center of the image
        M = cv2.getRotationMatrix2D((cX, cY), 90, 1.0)
        rotated90 = cv2.warpAffine(image, M, (w, h))
```

```
        # rotate our image by 180 degrees around the image
        M = cv2.getRotationMatrix2D((cX, cY), 180, 1.0)
        rotated180 = cv2.warpAffine(image, M, (w, h))

        # rotate our image by 270 degrees around the image
        M = cv2.getRotationMatrix2D((cX, cY), 270, 1.0)
        rotated270 = cv2.warpAffine(image, M, (w, h))

        cv2.imwrite(n1, rotated90)
        cv2.imwrite(n2, rotated180)
        cv2.imwrite(n3, rotated270)
```

2.2. Phân loại độc lập phóng đại - Magnification independent classification

Yếu tố phóng đại được biết tới với một vai trò quan trọng trong việc phân tích hình ảnh mô học. Các hình ảnh mô học bao gồm sự đa dạng của các mô, nhưng việc phân tích các mô này trở nên phức tạp khi chúng ở độ phóng đại thấp.

Trong bộ dữ liệu này, hình ảnh với nhiều các yếu tố phóng đại cũng đã được nhóm tác giả xem xét và sử dụng các phân loại khác nhau cho hệ số phóng đại cụ thể (40x, 100x, 200x, 400x).

2.3. Phân chia dữ liệu

Chia dữ liệu thành các tập train-test là một thực tế phổ biến trong các mạng thần kinh được sử dụng để phân tích hiệu suất. Để xác định ảnh hưởng của việc đào tạo - kiểm tra kích thước dữ liệu đối với hiệu suất của các mạng, việc phân chia tỉ lệ cho bộ train-test được nhóm tác giả sử dụng như sau: (90% - 10%, 80% - 20% và 70% - 30%).

Dựa trên bộ dữ liệu đã được nhóm tác giả tăng cường và phóng đại gồm có hai nhóm chính: lành tính và ác tính. Trong đó nhóm lành tính và ác tính bao gồm 2480 và 5429 mẫu tương ứng. Nhóm chúng em tiến hành phân chia bộ dữ liệu train-test như sau:

a. Bước 1: Tạo lần lượt các thư mục cấp 1 traintest91, traintest82, traintest73.

```
!mkdir traintest91
!mkdir traintest82
!mkdir traintest73
```

b. Bước 2: Với mỗi thư mục tạo thư mục cấp 2 là train và test.

```
!mkdir train
!mkdir test
```

c. Bước 3: Trong mỗi thư mục 2, tạo thư mục cấp 3 là benign và malignant: nhằm lưu các ảnh/ gán nhãn thuộc tệp lành tính và ác tính.

```
!mkdir benign
!mkdir malignant
```

Tổng thể quá trình 1,2,3 như sau:

```
!mkdir traintest91
%cd traintest91
!mkdir train
%cd train
!mkdir benign
!mkdir malignant
%cd ../
!mkdir test
%cd test
!mkdir benign
!mkdir malignant
```

d. Bước 4:

Dữ liệu từ nguồn ban đầu có tỉ lệ số ảnh lành tính và ác tính (2480: 5429) không giống nhau, nhóm tác giả đã sử dụng tỉ lệ hai nhãn là 1:1 tức 2480 ảnh cho mỗi nhãn lành tính, ác tính.

- Tạo thư mục Malignant random để lấy ngẫu nhiên 2480 ảnh vào

```
!mkdir malignant_random
```

- Random ảnh vào thư mục

```
n,  t=0, time.time()
pathsource = './malignant_full/*'
files = sorted(glob.glob(pathsource))
file_count = len(files)
```

```
random_files = random.sample (range(0, file_count), file_count
)
t = time.time() - t
limit = 2480
start = 0
for i in random_files:
  if (start < limit):
    shutil.copy (files[random_files[i]], "./malignant_random")
  else:
    break
  start = start + 1
  print (i)
  print ("\n")
```

e. Bước 5: Sau đó random ảnh vào các thư mục cấp 3 theo tỉ lệ và nhãn.

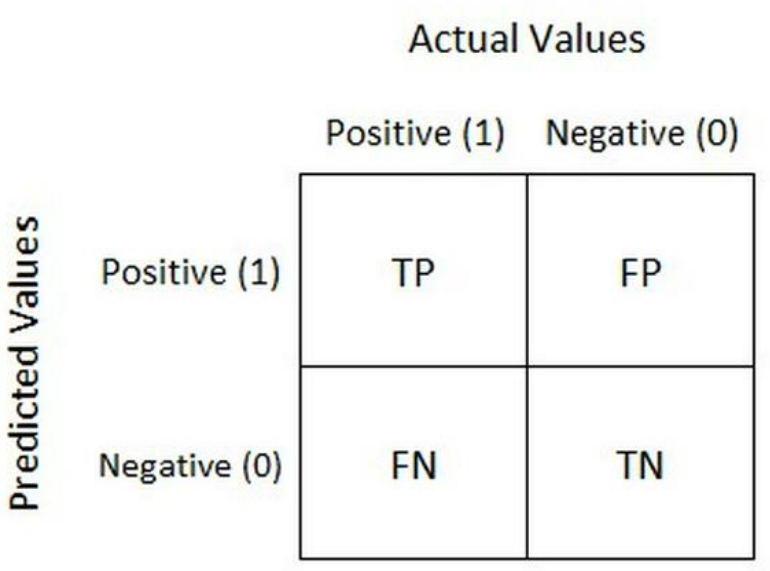
```
#Train/Malignant 70%
n,  t=0, time.time()
pathsource = './malignant_random/*'
files = sorted(glob.glob(pathsource))
file_count = len(files)
random_files = random.sample (range(0, file_count), file_count
)
t = time.time() - t
limit = file_count * 0.7
start = 0
for i in random_files:
  if (start < limit):
    shutil.copy (files[random_files[i]], "/content/drive/MyDri
ve/HK6-Nam3/HK6-
HeHoTroQuyetDinh/breast/Dataset/traintest73/train/malignant")
  else:
    break
  start = start + 1
  print (i)
  print ("\n")
```

THỰC NGHIỆM – ĐÁNH GIÁ

1. Độ đo

1.1. Ma trận Confusion

Ma trận nhầm lẫn là một phương pháp đánh giá kết quả của những bài toán phân loại với việc xem xét cả những chỉ số về độ chính xác và độ bao quát của các dự đoán cho từng lớp. Một ma trận nhầm lẫn gồm 4 chỉ số sau đối với mỗi lớp phân loại:



Hình 26 Ma trận nhầm lẫn [24]

1.2. True/False – Positive/Negative

- **Condition positive (P)**: Tổng số dữ liệu có nhãn là 1.
- **Condition Negative (N)**: Tổng số dữ liệu có nhãn là 0.
- **True positive (TP)**: Điểm dữ liệu được dự đoán là 1 và có nhãn là 1.
- **True negative (TN)**: Điểm dữ liệu được dự đoán là 0 và có nhãn là 0.
- **False positive (FP)**: Điểm dữ liệu được dự đoán là 1 và có nhãn là 0.
- **False negative (FN)**: Điểm dữ liệu được dự đoán là 0 và có nhãn là 1. [25]

1.3. Accuracy

Accuracy là độ chính xác của mô hình, được tính bằng tỉ lệ giữa số điểm được dự đoán đúng trên tổng số điểm dữ liệu kiểm thử.

$$\textbf{Accuracy} = \frac{TP+TN}{P+N} = \frac{TP+TN}{TP+FP+TN+FN}$$

1.4. Precision

Precision hay Positive Predictive Value (PPV): Tỉ lệ nhãn 1 đoán đúng. Trong tất cả các dự đoán Positive được đưa ra, bao nhiêu dự đoán là chính xác? Chỉ số này được tính theo công thức:

$$\textbf{Precision} = \frac{TP}{TP+FP}$$

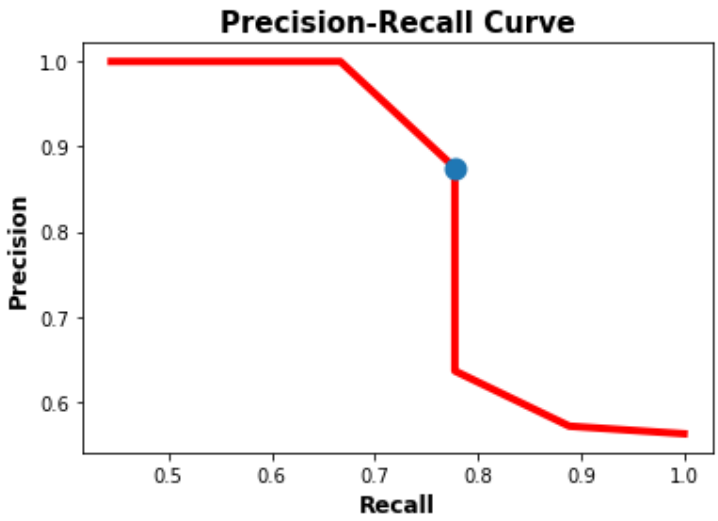
1.5. Recall

Sensitivity, Recall, Hit Rate, Or True Positive Rate (TPR): Độ nhạy - Tỷ lệ nhãn 1 thực. Trong tất cả các trường hợp Positive, bao nhiêu trường hợp đã được dự đoán chính xác? Chỉ số này được tính theo công thức:

$$\textbf{Recall} = \frac{TP}{TP+FN}$$

1.6. APS (Average Precision Score)

Tương tự như ROC curve, chúng ta cũng có thể đánh giá mô hình dựa trên việc thay đổi một ngưỡng và quan sát giá trị của Precision và Recall. Khái niệm Area Under the Curve (AUC) cũng được định nghĩa tương tự. Với Precision-Recall Curve, AUC còn có một tên khác là **Average precision (AP). APS** chính là phần diện tích phía dưới đường Precision-Recall Curve.



Hình 27 Average Precision Score [26]

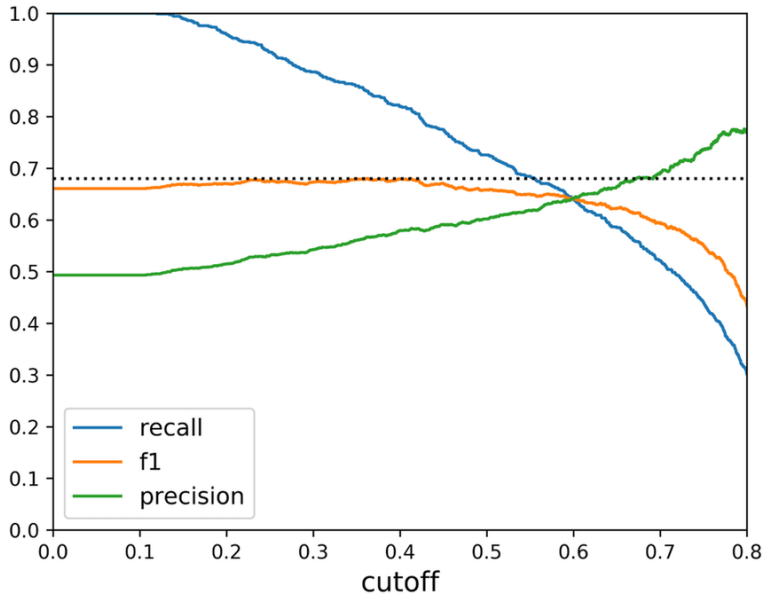
1.6. F1 – score

Chỉ có Precision hay chỉ có Recall thì không đánh giá được chất lượng mô hình.

- Chỉ dùng Precision, mô hình chỉ đưa ra dự đoán cho một điểm mà nó chắc chắn nhất. Khi đó Precision = 1, tuy nhiên ta không thể nói là mô hình này tốt.
- Chỉ dùng Recall, nếu mô hình dự đoán tất cả các điểm đều là positive. Khi đó Recall = 1, tuy nhiên ta cũng không thể nói đây là mô hình tốt

F1 score - Điểm F1: Điểm F1 là một trung bình hài hòa Precision và Recall**.**

$$\mathbf{F1} = \frac{2 * \boldsymbol{Precision} * \boldsymbol{Recall}}{\boldsymbol{Precision} + \boldsymbol{Recall}}$$



Hình 28 F1 – score [27]

1.7. Specificity

Specificity là độ đặc hiệu. Nó biểu diễn tỷ lệ phân loại chính xác các mẫu 0 trên tất cả các mẫu 0, được tính theo công thức:

$$\textbf{Specificity} = \frac{TN}{TN+FP}$$

1.8. FPR

False Positive Rate/Fall-out biểu diễn tỷ lệ gắn nhãn sai các mẫu 0 thành 1 trên tất cả các mẫu 0, được tính theo công thức:

$$\textbf{FPR} = 1 - \textbf{Specificity} = \frac{FP}{TN+FP}$$

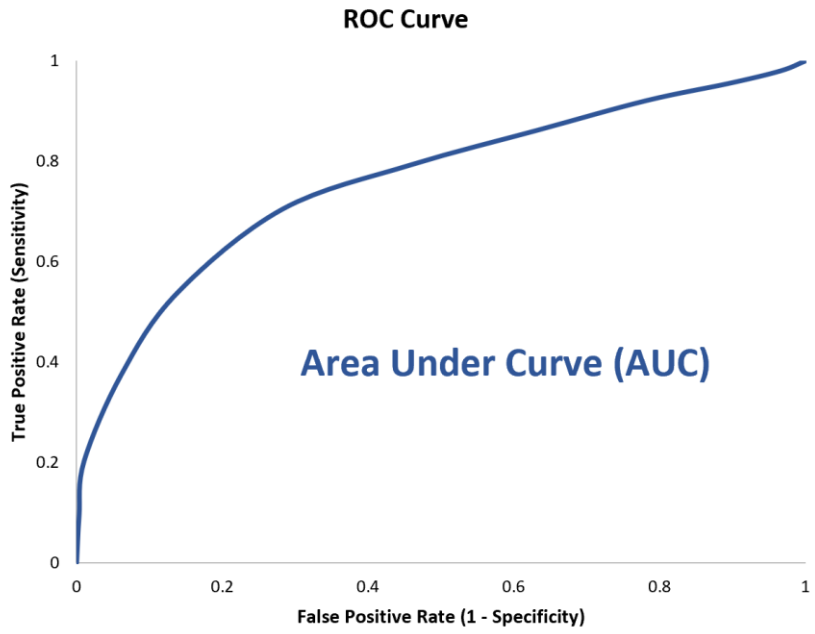
1.9. AUC - ROC

AUC - ROC là một phương pháp tính toán hiệu suất của một mô hình phân loại theo các ngưỡng phân loại khác nhau. Giả sử với bài toán phân loại nhị phân (2 lớp) sử dụng hồi quy logistic (logistic regression), việc chọn các ngưỡng phân loại **[0…1]** khác nhau sẽ ảnh hưởng đến khả năng phân loại của mô hình và ta cần tính toán được mức độ ảnh hưởng của các ngưỡng. AUC là từ viết tắt của Area Under The Curve còn ROC viết tắt của Receiver Operating Characteristics.

ROC là một đường cong biểu diễn xác suất và AUC biểu diễn mức độ phân loại của mô hình. AUC-ROC còn được biết đến dưới cái tên AUROC (Area Under The Receiver Operating Characteristics).

Chỉ số AUC càng cao thì mô hình càng chính xác trong việc phân loại các lớp.

Đường cong ROC biểu diễn các cặp chỉ số (TPR, FPR) tại mỗi ngưỡng với TPR là trục tục và FPR là trục hoành.



Hình 29 ROC Curve [28]

2. Phân tích kết quả thực nghiệm

2.1. Theo bài báo

Table 1
Performance Analysis for Histopathological Image Classification using Fine-tuned Pre-trained Network (VGG16, VGG19, and ResNet50).

Classifier	Training–Testing Data Splitting	Class Type	Precision	Recall	F_1 Score	Accuracy	AUC	APS
VGG 16 + LR	90%–10%	B	0.93	0.92	0.93	**92.60%**	**95.65%**	**95.95%**
		M	0.93	0.93	0.93			
		Avg/Total	**0.93**	**0.93**	**0.93**			
	80%–20%	B	0.92	0.93	0.92	92.20%	93.95%	93.89%
		M	0.93	0.92	0.92			
		Avg/Total	0.92	0.92	0.92			
	70%–30%	B	0.92	0.92	0.92	91.73%	93.49%	93.29%
		M	0.92	0.91	0.92			
		Avg/Total	0.92	0.92	0.92			
VGG 19 + LR	90%–10%	B	0.88	0.93	0.90	90.00%	91.85%	91.27%
		M	0.93	0.88	0.90			
		Avg/Total	0.90	0.90	0.90			
	80%–20%	B	0.89	0.90	0.89	89.50%	91.76%	91.13%
		M	0.90	0.89	0.90			
		Avg/Total	0.90	0.90	0.90			
	70%–30%	B	0.90	0.91	0.91	90.40%	91.14%	90.38%
		M	0.90	0.90	0.90			
		Avg/Total	0.90	0.90	0.90			
ResNet50 + LR	90%–10%	B	0.77	0.81	0.79	79.40%	79.39%	82.03%
		M	0.81	0.78	0.80			
		Avg/Total	0.79	0.79	0.79			
	80%–20%	B	0.80	0.78	0.79	78.90%	79.23%	80.56%
		M	0.81	0.78	0.80			
		Avg/Total	0.79	0.79	0.79			
	70%–30%	B	0.78	0.81	0.79	78.73%	79.12%	79.09%
		M	0.80	0.77	0.78			
		Avg/Total	0.79	0.79	0.79			

Bảng 1: Phân tích hiệu suất cho phân loại hình ảnh sử dụng Fine-tuned

Từ bảng 1 có thể quan sát thấy rằng mạng VGG16 được đào tạo sẵn vượt trội so với ResNet50 trong khi hiệu suất của VGG16 và VGG19 là tương đương. Trong học chuyển giao, ResNet50 đã cung cấp kết quả không đạt yêu cầu qua bộ dữ liệu BreakHis. Overfitting là lý do duy nhất cho sự do dung lượng quá lớn của mạng. Đóng băng nhiều lớp hơn trong mạng có thể là một giải pháp để giảm dung lượng hiệu quả của mạng và quá mức. Giải pháp này không được tính đến trong công việc này do hạn chế về không gian, nhưng nhóm tác giả đảm bảo rằng giải pháp sẽ được xem xét trong phiên bản mở rộng của bài báo này. Tuy nhiên, trong đánh giá hiệu suất cho mạng được đào tạo đầy đủ, ResNet50 cho thấy một hiệu suất tốt hơn mạng VGG16 và VGG19.

Table 2
Performance Analysis for Histopathological Image Classification using Full-trained Network (VGG16, VGG19, and ResNet50).

Classifier	Training–Testing Data Splitting	Class Type	Precision	Recall	F_1 Score	Accuracy	AUC	APS
VGG 16	90%–10%	B	0.63	0.64	0.64	64.40%	**75.00%**	**76.79%**
		M	0.66	0.64	0.65			
		Avg/Total	0.64	0.64	0.64			
	80%–20%	B	0.63	0.63	0.63	63.20%	66.77%	65.23%
		M	0.63	0.64	0.63			
		Avg/Total	0.63	0.63	0.63			
	70%–30%	B	0.66	0.60	0.63	63.93%	75.18%	73.23%
		M	0.62	0.68	0.65			
		Avg/Total	0.64	0.64	0.64			
VGG 19	90%–10%	B	0.50	0.87	0.64	51.80%	60.40%	59.88%
		M	0.60	0.19	0.29			
		Avg/Total	0.55	0.52	0.46			
	80%–20%	B	0.57	0.76	0.65	59.70%	65.61%	62.00%
		M	0.65	0.43	0.52			
		Avg/Total	0.61	0.60	0.59			
	70%–30%	B	0.81	0.02	0.04	50.27%	61.34%	55.44%
		M	0.50	0.99	0.66			
		Avg/Total	0.66	0.50	0.35			
ResNet50	90%–10%	B	0.77	0.78	0.79	78.80%	82.71%	79.34%
		M	0.80	0.78	0.79			
		Avg/Total	0.79	0.79	0.79			
	80%–20%	B	0.79	0.80	0.79	79.30%	82.88%	79.67%
		M	0.80	0.78	0.79			
		Avg/Total	0.79	0.79	0.79			
	70%–30%	B	0.70	0.82	0.80	**79.93%**	84.35%	80.76%
		M	0.81	0.78	0.79			
		Avg/Total	**0.80**	**0.80**	**0.80**			

Bảng 2: Phân tích hiệu suất cho phân lớp hình ảnh sử dụng Full - trained

Từ Bảng 2, người ta đã phát hiện ra rằng VGG16 và VGG 19, cả hai mạng đều thiên vị một lớp cụ thể. Giá trị thu hồi là rất cao đối với một lớp cụ thể và đồng thời rất thấp đối với các lớp khác cả VGG16 và VGG19 được đào tạo đầy đủ, nhưng ResNet50 rất nhạy với cả hai lớp sẽ hỗ trợ hoạt động tốt hơn so với VGG16 và VGG19.

Từ Bảng 1 và 2, người ta đã quan sát thấy rằng sự suy giảm hiệu suất của VGG16, VGG19 và mạng ResNet50 tinh chỉnh là không đáng kể đối với việc giảm kích thước dữ liệu đào tạo. Cả ba mạng đều hoạt động tốt khi dữ liệu đào tạo chiếm 90% tổng số dữ liệu. VGG19 thực hiện tốt việc phân tách dữ liệu đào tạo - thử nghiệm 80%-20%. Tuy nhiên, hiệu suất của mạng ResNet50 và VGG16 cho toàn bộ phân tách gần như tương tự nhau. Độ lệch của VGG19 so với xu hướng thông thường là do độ nhạy hơn của nó đối với lành tính và ác tính trong quá trình phân tách 90%- 10% và 70%-30%. Tóm lại, nó đã được chứng minh rằng việc sử dụng phương pháp học chuyển giao dẫn đến một hiệu suất đáng chú ý

đối với CNN và đào tạo đầy đủ về phương thức hình ảnh mô bệnh học, ngay cả khi bộ dữ liệu đào tạo bị hạn chế về kích thước.

Hơn nữa, để nghiên cứu hiệu suất của mô hình với kích thước của dữ liệu train – test, ba phân chia dữ liệu train – test khác nhau (90%-10%, 80%-20% và 70%-30%) được sử dụng vì kích thước của bộ dữ liệu có ảnh hưởng đáng kể đến hiệu quả hoạt động của CNN. Trong bối cảnh này, phân tích ROC và AUC được sử dụng để so sánh hiệu suất của tất cả các mạng, như minh họa hình bên dưới.

Trong hình, đường cong ROC và AUC của mạng được đào tạo trước và được đào tạo đầy đủ được so sánh trong tập dữ liệu 90%-10%. Người ta thấy rằng, VGG16 được đào tạo trước (AUC-95,65%) và VGG19 (AUC-91,85%) vượt trội so với VGG16 được đào tạo đầy đủ (AUC- 75,00%) và VGG19 (AUC-60,40%) với một số lượng đáng kể trong khi ResNet50 được đào tạo trước (AUC-79,39%) thấp hơn so với ResNet50 được đào tạo đầy đủ (AUC-82,71%) nhưng với một lượng nhỏ.

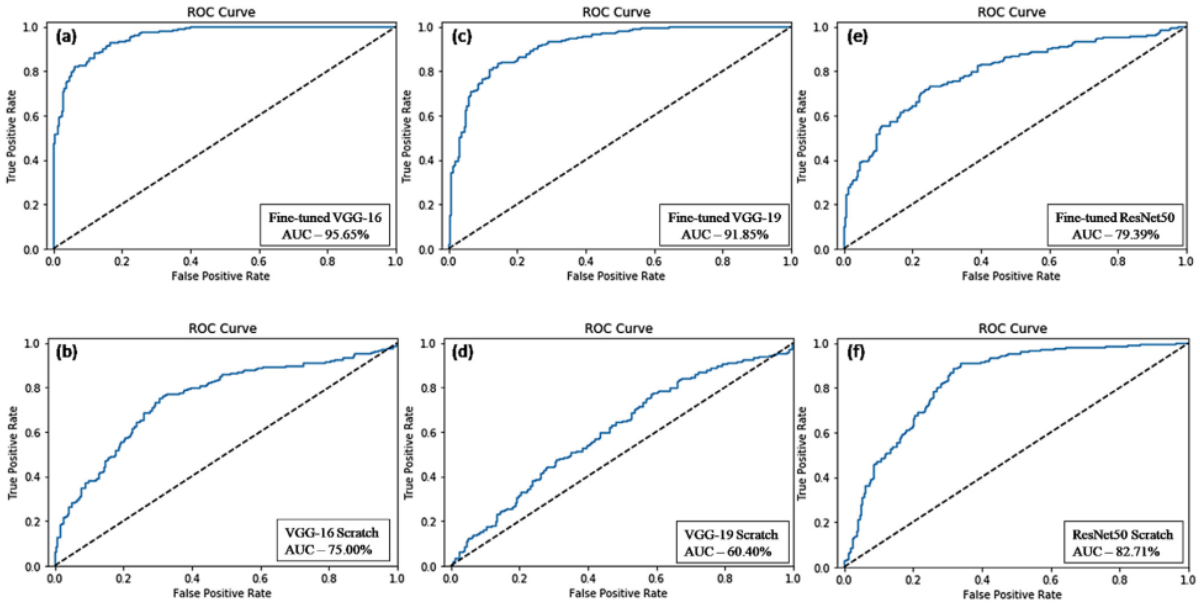


Fig. 2. ROC analysis for breast cancer classification with 90%–10% training and testing set splitting in (a) Fine-tuned pre-trained VGG16, (b), Fine-tuned pre-trained VGG19, (c) Fine-tuned pre-trained ResNet50, (d) Fully-trained VGG16, (e) Fully-trained VGG19, and (f) Fully-trained ResNet50.

Hình 30: Phân tích ROC cho phân lớp 90% - 10%

Xu hướng tương tự được quan sát thấy đối với 2 phân chia còn lại của dữ liệu train – test được hiển thị trong hình.

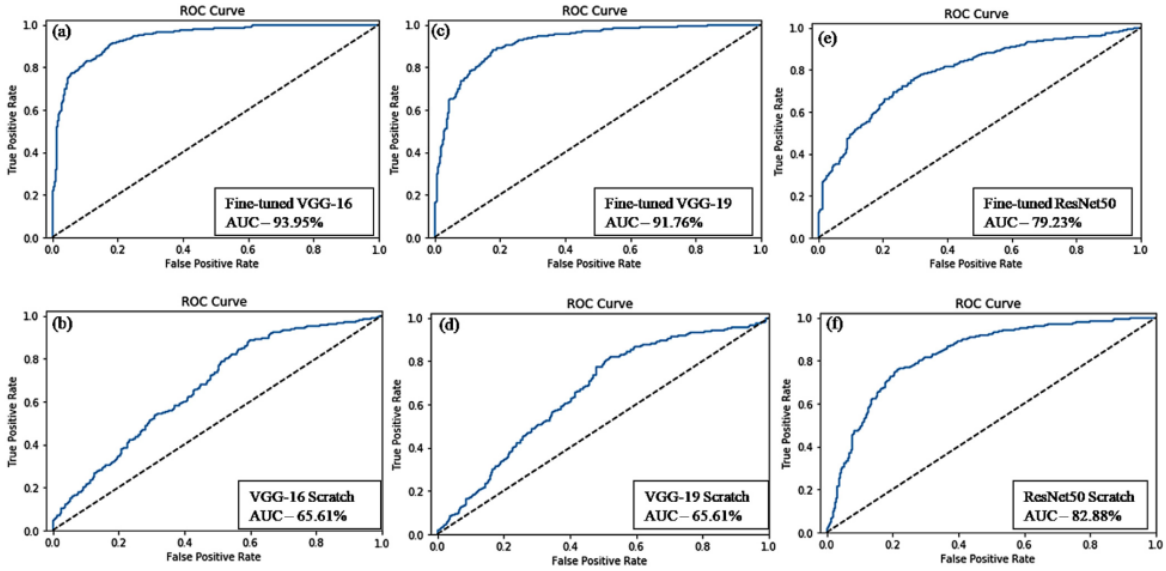


Fig. 3. ROC analysis for breast cancer classification with 80%–20% training and testing set splitting in (a) Fine-tuned pre-trained VGG16, (b), Fine-tuned pre-trained VGG19, (c) Fine-tuned pre-trained ResNet50, (d) Fully-trained VGG16, (e) Fully-trained VGG19, and (f) Fully-trained ResNet50.

Hình 31: Phân tích ROC cho phân lớp 80% - 20%

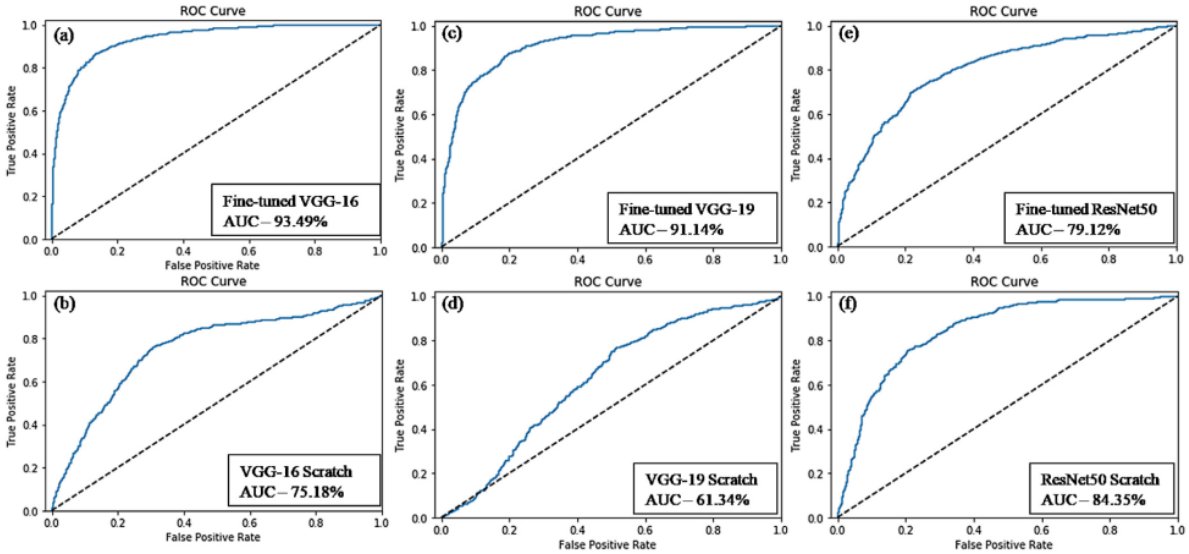


Fig. 4. ROC analysis for breast cancer classification with 70%–30% training and testing set splitting in (a) Fine-tuned pre-trained VGG16, (b), Fine-tuned pre-trained VGG19, (c) Fine-tuned pre-trained ResNet50, (d) Fully-trained VGG16, (e) Fully-trained VGG19, and (f) Fully-trained ResNet50.

Hình 32: Phân tích ROC cho phân lớp 70% - 30%

2.2. Theo nhóm thực hiện

Trong phạm vi nghiên cứu còn gặp nhiều giới hạn về máy chạy thực nghiệm cũng như thời gian còn hạn chế, nên nhóm chúng em đã thực hiện quy trình huấn luyện dựa trên những bước thực nghiệm của tác giả. Đầu tiên là với 3 bộ phân tách dữ liệu 90%-10%, 80%-20%, 70%-30% đi kèm với ba kiến trúc VGG16, VGG19, ResNet50. Theo như trong bài báo không đề cập đến số lần chạy epoch và batchsize, và nhóm chúng em đã gửi mail đến tác giả để hỏi một số vấn đề, tác giả phản hồi đã sử dụng phương pháp ModelCheckpoint và EarlyStopping của thư viện Keras, trong đó ModelCheckpoint lưu mô hình bằng cách theo dõi độ chính xác của mô hình bằng cách chuyển val_acc cho ModelCheckpoint, mô hình được lưu nếu val_acc ở epoch hiện tại lớn hơn epoch trước đó. EarlyStoping sẽ ngừng việc huấn luyện mô hình khi các tham số không tăng, tác giả đã đặt mô hình ngừng huấn luyện nếu giá trị accuracy không tăng trong 20 epoch.

Đối với transfer learning để huấn luyện, toàn bộ mạng đào tạo trước (VGG16, VGG19, ResNet50) sử dụng tạo feature và các đặc trưng trích xuất được sử dụng để phân loại hồi quy Logictis. Ngoài ra tác giả còn sử dụng phân tích ROC và AUC để so sánh hiệu suất của tất cả các mạng. Do đó đối với bộ dữ liệu BreakHis chưa được chia traning – testing theo các tỉ lệ mà chỉ có các ảnh được phóng đại theo các chỉ số 40x, 100x, 200x, 400x, nhóm chúng em đã tự thực hiện phân chia dữ liệu theo như tác giả đề cập trong bài báo và việc phân chia được thực hiện trên Colab và sử dụng thư viện Keras. Trước khi huấn luyện mô hình, nhóm thực hiện tăng cường dữ liệu bằng cách xoay ảnh theo các góc đã nêu. Đối với huấn luyện mô hình Full Trained với các kiến trúc mạng, nhóm thực nghiệm với 1 epoch với batch size 30.

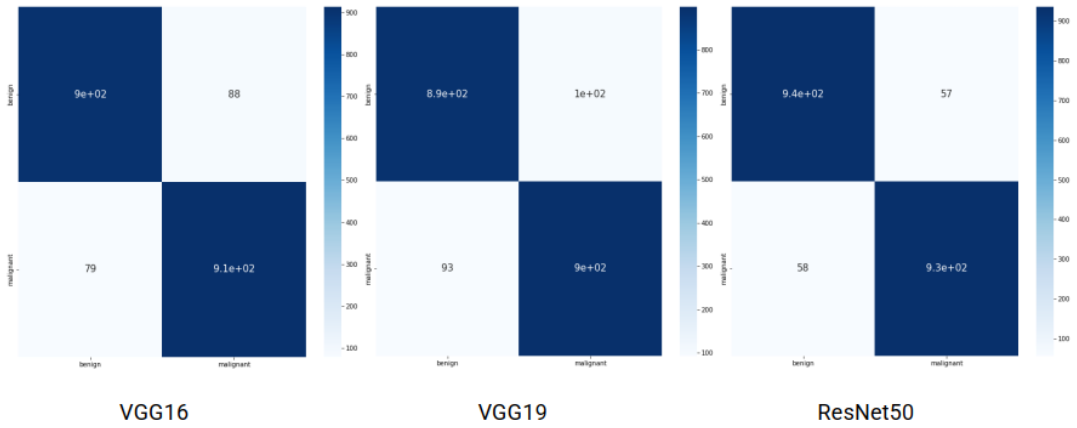
2.2.1 Transfer learning

Phân loại	Dữ liệu Train Test	Loại Lớp	Precision	Recall	F1 score	Accuracy	AUC
VGG 16 + LR	90%–10%	B	0.92	0.91	0.92	91.58%	97.60%
		M	0.91	0.92	0.92		
		Avg/Total	0.92	0.92	0.92		
	80%–20%	B	0.92	0.91	0.91	91.25%	97.00%
		M	0.91	0.92	0.91		

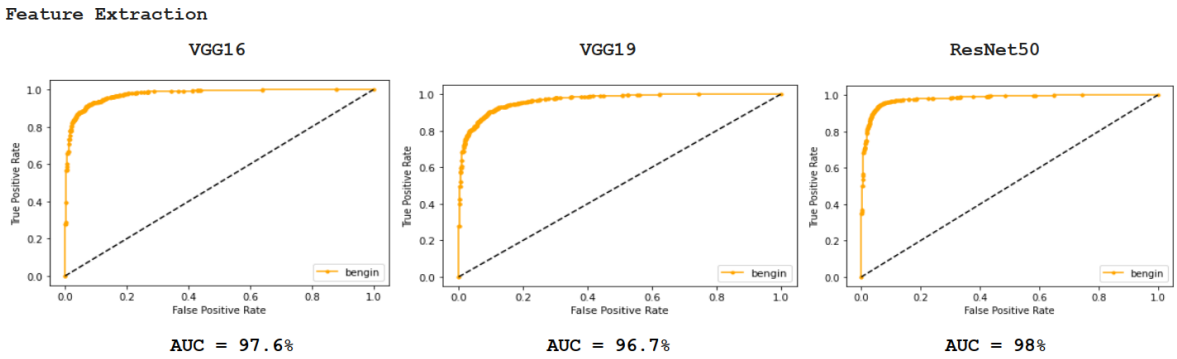
		Avg/Total	0.91	0.91	0.91		
	70%–30%	B	0.92	0.90	0.91		
		M	0.91	0.92	0.91	90.05%	97.00%
		Avg/Total	0.91	0.91	0.91		
	90%–10%	B	0.91	0.90	0.90		
		M	0.90	0.91	0.90	90.27%	
		Avg/Total	0.90	0.90	0.90		96.70%
	80%–20%	B	0.91	0.90	0.90		
VGG 19 + LR		M	0.90	0.91	0.90	90.50%	96.40%
		Avg/Total	0.91	0.90	0.90		
	70%–30%	B	0.91	0.90	0.91		
		M	0.91	0.91	0.91	90.76%	96.50%
		Avg/Total	0.91	0.91	0.91		
	90%–10%	B	**0.94**	**0.94**	**0.94**		
		M	**0.94**	**0.94**	**0.94**	**94.20%**	**98.00%**
		Avg/Total	**0.94**	**0.94**	**0.94**		
	80%–20%	B	0.93	0.92	0.92		
ResNet50 + LR		M	0.92	0.93	0.92	92.21%	97.70%
		Avg/Total	0.92	0.92	0.92		
	70%–30%	B	0.92	0.93	0.92		
		M	0.93	0.92	0.92	92.47%	97.7%
		Avg/Total	0.92	0.92	0.92		

Bảng 3: Phân tích hiệu suất phân loại hình ảnh mô bệnh học bằng cách sử dụng Mạng được đào tạo trước tinh chỉnh (VGG16, VGG19 và ResNet50

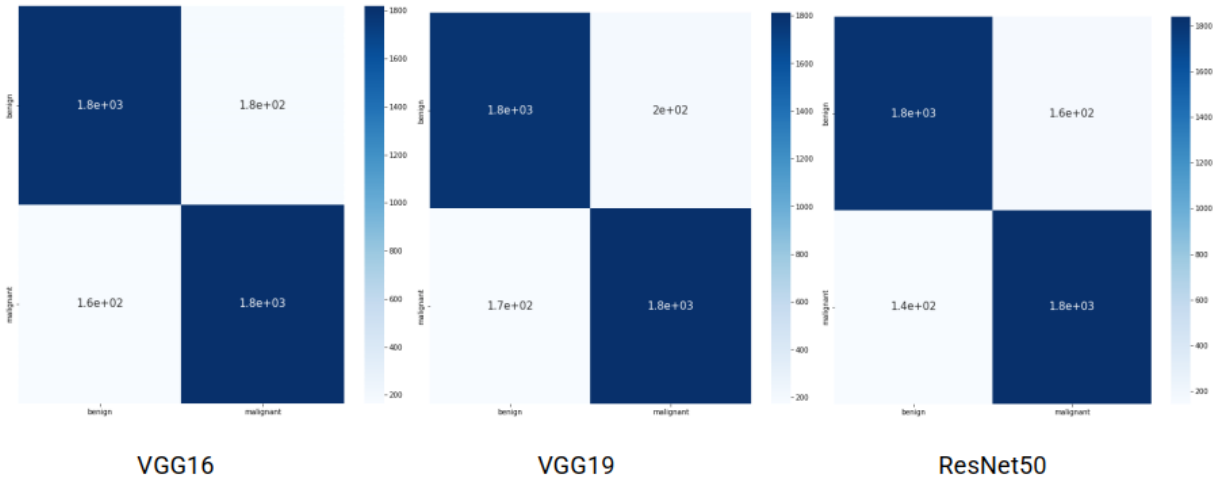
Theo kết quả chạy thực nghiệm cho thấy, ResNet50 vượt trội nhất so với 3 kiến trúc với độ chính xác khá cao (94.20%) trên tập dữ liệu 90%-10% với AUC đo được là 98%. Theo bài báo của tác giả thì trong Fine-tuned Pre-trained, VGG16 với tập dữ liệu train-test 90 - 10% có hiệu suất cao nhất. Mặc dù nhóm train cùng một kiến trúc, cùng tập dữ liệu với tác giả nhưng do ảnh chúng em lấy ngẫu nhiên, các layer có thể khác nhau và sử dụng thư viện không giống nhau nên có phần kết quả khác với nhóm tác giả của bài báo.



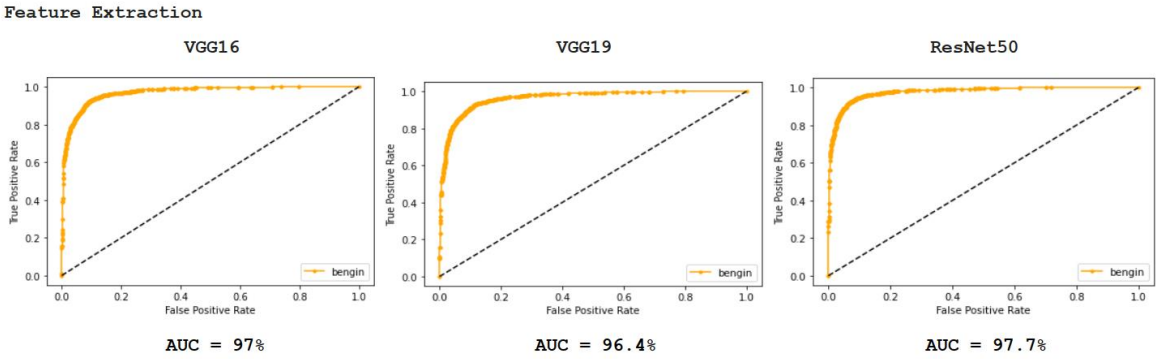
Hình 33: Ma trận nhầm lẫn của tập dữ liệu train-test 90 -10%



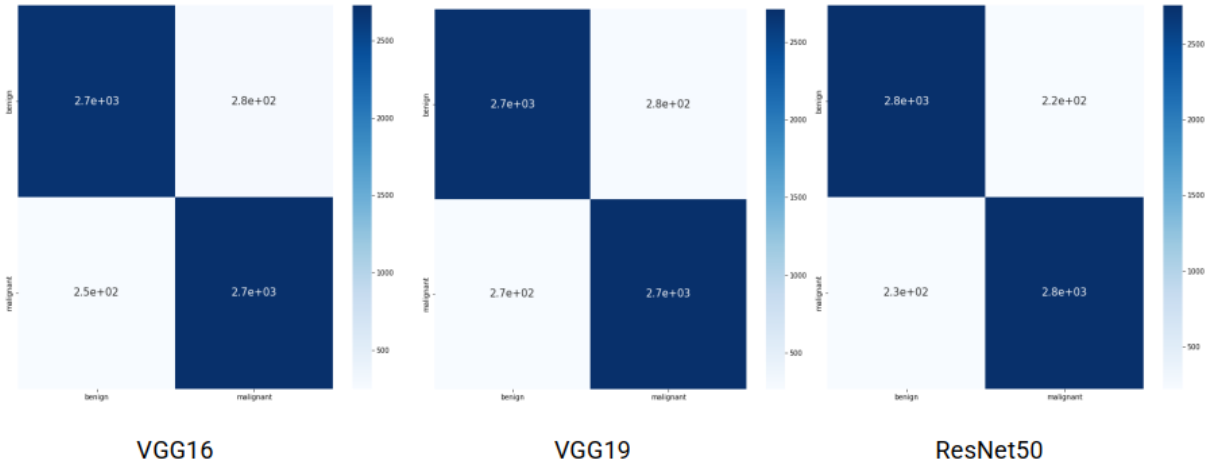
Hình 34: Biểu đồ ROC-AUC của tập dữ liệu train-test 90 – 10%



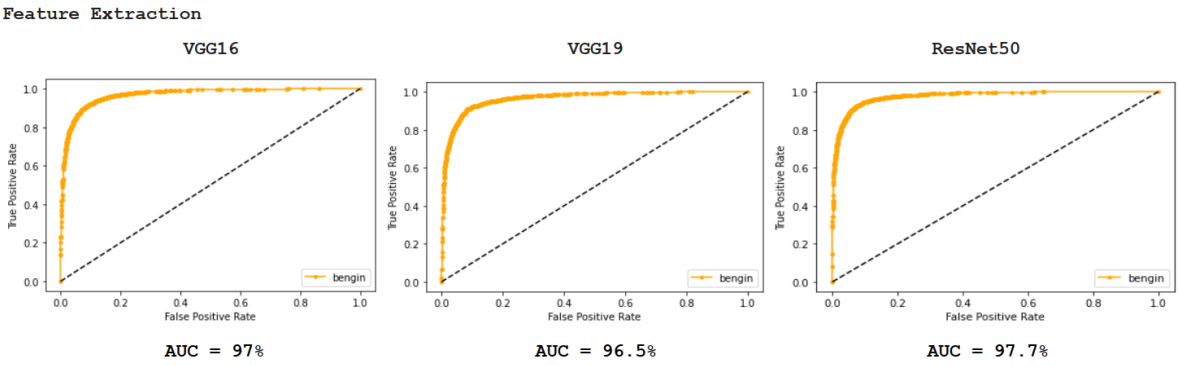
Hình 35: Ma trận nhầm lẫn của tập dữ liệu train-test 80 - 20%



Hình 36: Biểu đồ ROC-AUC của tập dữ liệu train-test 80 – 20%



Hình 37 Ma trận nhầm lẫn của tập dữ liệu train-test 70 – 30%



Hình 38 Biểu đồ ROC-AUC của tập dữ liệu train-test 70 – 30%

Cả 9 ma trận trên đều có phần in đậm nằm trên đường chéo chính, nó cho thấy hiệu suất của đào tạo tinh chỉnh là khá chính xác. Về phần biểu đồ ROC-AUC, AUC trung bình của tất cả các mô hình thuộc Transfer Learrning là khoảng 97%, một con số cao cho thấy độ chính xác mà mô hình học được.

2.2.2 Fully trained

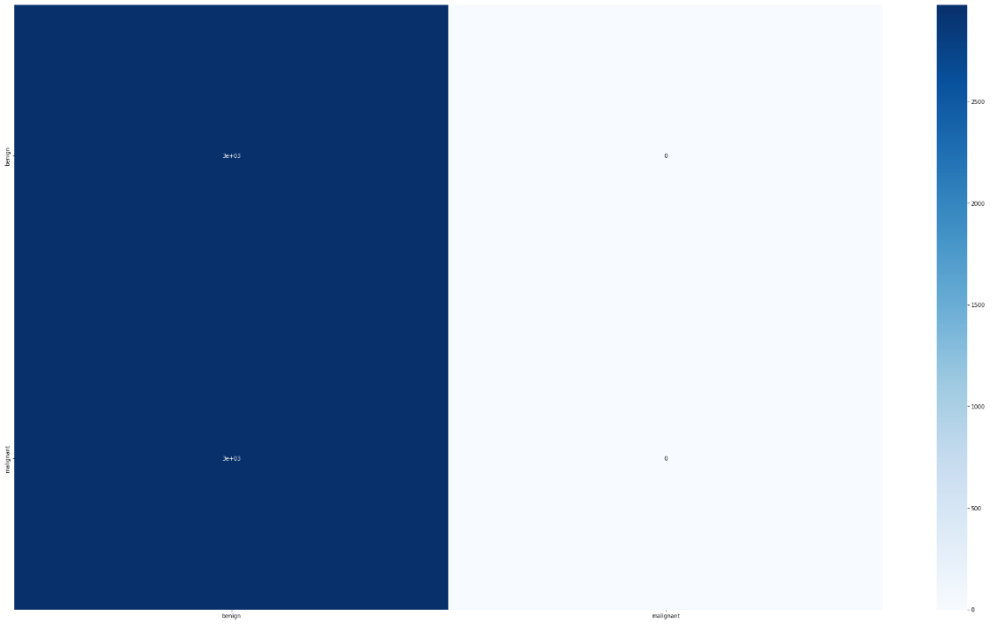
Tiếp theo là Full-trained với 3 bộ phân tách dữ liệu 90%-10%, 80%-20%, 70%-30% và đi kèm với ba kiến trúc VGG16, VGG19, ResNet50. Khi thực hiện huấn luyện mô hình, thời gian huấn luyện cho 1 kiến trúc mạng rơi vào 15 tiếng đòi hỏi thời gian train rất lâu, nên chúng em chỉ train 1 epoch.

Loại	Dữ liệu	Precison	Recall	F1 score	Accuracy	AUC
VGG16	90%-10%	0.24	0.49	0.32	49.48%	50.00%
	80%-20%	0.23	0.47	0.31	49.80%	50.00%
	70%-30%	0.25	0.5	0.33	49.55%	50.00%
VGG19	90%-10%	0.24	0.48	0.31	50.00%	50.00%
	80%-20%	0.24	0.48	0.32	49.89%	49.80%
	70%-30%	0.26	0.52	0.34	49.79%	49.20%
ResNet50	90%-10%	0.73	0.82	0.75	75.51%	48.60%
	80%-20%	0.71	0.85	0.76	75.60%	49.20%
	70%-30%	**0.73**	**0.85**	**0.77**	**76.15%**	**49.80%**

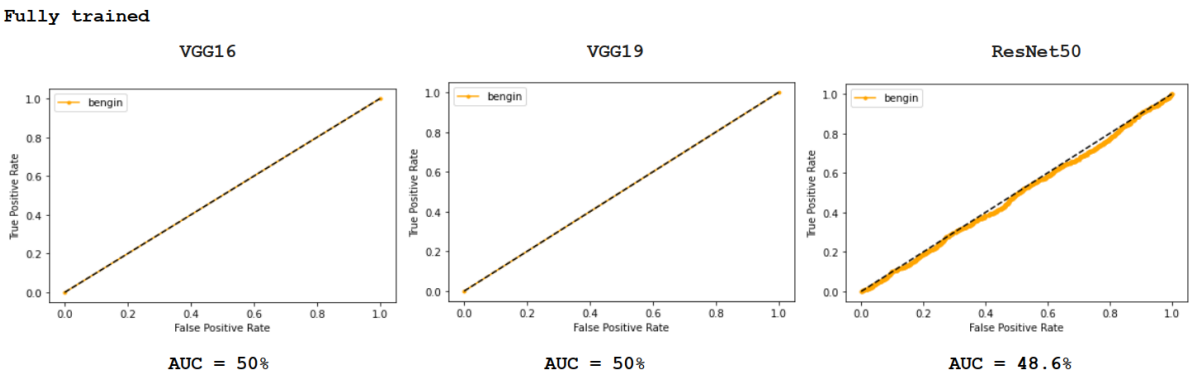
Bảng 4 Phân tích hiệu suất phân loại hình ảnh mô bệnh học bằng cách sử dụng Mạng được đào tạo đầy đủ (VGG16, VGG19 và ResNet50)

Dựa theo số liệu thì có thể thấy rằng ở Full-trained, ResNet50 vẫn tiếp tục vượt mặt cả 2 kiến trúc VGG16, VGG19 với độ chính xác ở tập dữ liệu train-test 70 – 30% là 76.15% với AUC đo được là 49.8% trong khi VGG16, VGG19 chỉ nằm ở mức 49%.

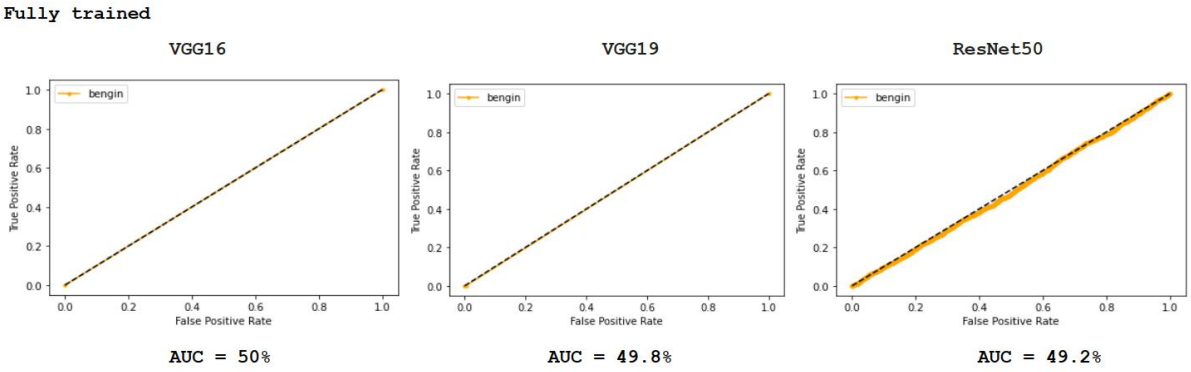
Ma trận nhầm lẫn của các kiến trúc và tập dữ liệu ở Full-trained đa phần nghiêng về 1 lớp. Điều này một phần là do mô hình chưa có đủ thời gian để học.



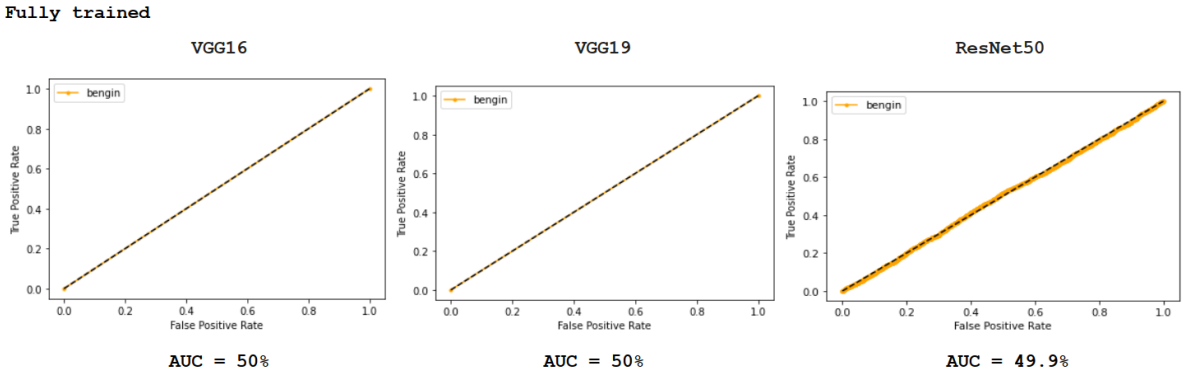
Hình 39 Ma trận nhầm lẫn Fully trained



Hình 40 Biểu đồ ROC-AUC của tập dữ liệu train-test 90 – 10%



Hình 41 Biểu đồ ROC-AUC của tập dữ liệu train-test 80 – 20%



Hình 42 Biểu đồ ROC-AUC của tập dữ liệu train-test 70 -30%

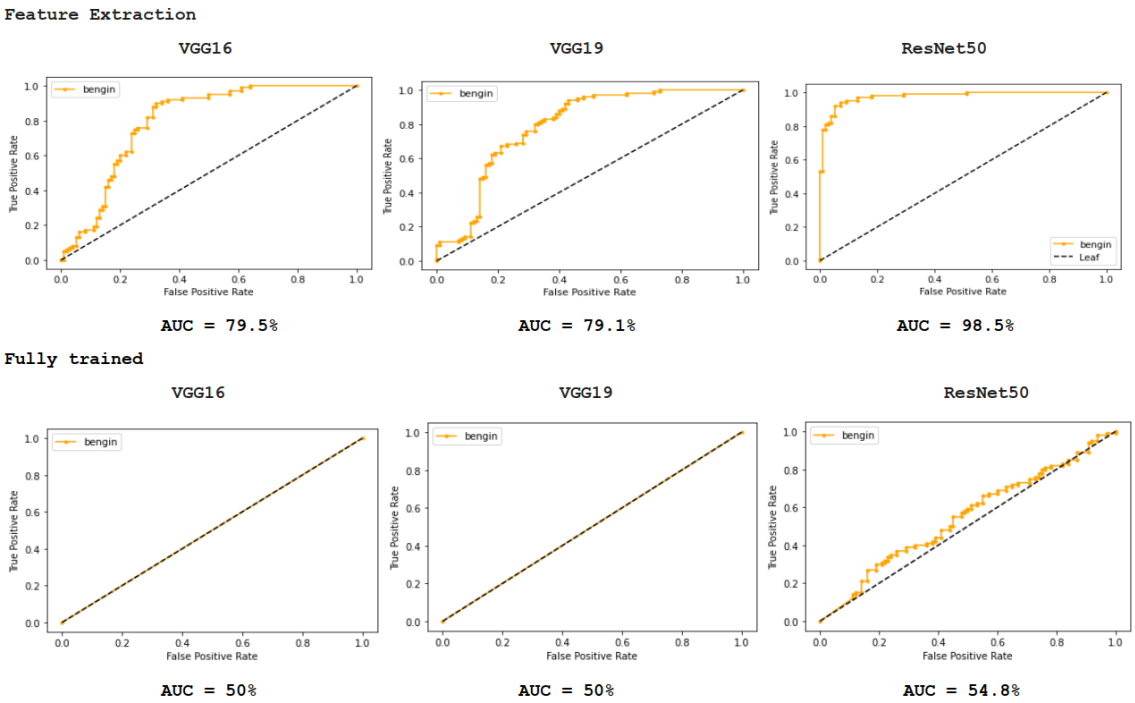
Có thể thấy cũng như ma trận nhầm lẫn, mô hình Fully trained với epoch bằng 1 vẫn chưa có đủ thời gian học, vì vậy cho nên chỉ số AUC rất thấp, chỉ nằm ở mức 50%, hoàn toàn không có khả năng áp dụng thực tế.

2.2.3 Transferlearning và Training from scratch với mẫu thử 500 ảnh

Nhóm chúng em có lấy ra một phần dữ liệu gồm 500 ảnh train theo tỷ lệ dữ liệu train-test 80 – 20%. Dữ liệu sẽ bao gồm 400 ảnh của tập train, 100 ảnh của tập test. Đối với fully trained sẽ chạy với số epoch là 50, batchsize là 30.

Có thể thấy, với 500 ảnh, ResNet50 vẫn cho kết quả ngoài mong đợi ngay cả khi trong tình huống hạn chế dữ liệu. VGG16 và VGG19 giảm đáng kể (gần 20%) so với khi

tập dữ liệu 20000 ảnh. Đối với fully trained thì kết quả vẫn không mấy khả quan khi không có mô hình nào vượt ngưỡng AUC 60%.



Hình 43 Biểu đồ ROC-AUC của mẫu thử 500 ảnh

KẾT LUẬN

Xét về tổng quan có thể thấy rằng Transfer Learning giải quyết được nhiều vấn đề bất cập mà Full-trained không làm được.

- Thứ nhất là về mặt hạn chế dữ liệu. Full-trained cần có một lượng dữ liệu cực lớn thì mới có thể học một cách hoàn chỉnh và cho độ chính xác cao.
- Thứ hai là thời gian học của mô hình Transfer Learning ngắn hơn nhiều so với Full-trained.
- Thứ ba là về độ chính xác. Theo như 2 bảng số liệu của tác giả thì độ chính xác của Transfer Learing vượt mặt Full-trained hoàn toàn.

Còn xét về 3 cấu trúc VGG16, VGG19 và ResNet50 thì chúng em thấy rằng ResNet50 vẫn luôn vượt trội hơn hẳn dù là Transfer Learning hay Full-trained.

Ở những nghiên cứu kế tiếp, chúng em hy vọng sẽ cải thiện kết quả nhằm mang lại những đóng góp trên cơ sở thực nghiệm, góp phần nào vào việc chẩn đoán ung thư có kết quả chính xác hơn.

ĐÁNH GIÁ CÔNG VIỆC

STT	Họ và tên	Vai trò	Nhiệm vụ	Mức độ hoàn thành	Đánh giá
1	Phan Mai Kiều Anh	Nhóm trưởng	• Báo cáo: *Viết, phân tích:* ✓ *Tiền xử lý dữ liệu, Mạng kiến trúc tích chập* ✓ *Nghiên cứu về hồi quy Logistic; Phương pháp xử lý dữ liệu sử dụng trong đồ án.* ✓ *Tìm hiểu mạng kiến trúc VGG16, VGG19, ResNet50.* • Chương trình demo: ✓ *Phân tách dữ liệu, tiền xử lý và đánh giá mô hình học.* ✓ *Chạy thực nghiệm Transfer learning với 3 kiến trúc mạng trên 3 bộ dữ liệu phân tách 90% -10%, 80%-20%, 70%-30%*	*Hoàn thành tốt*	*10/10*
2	Nguyễn Thị Hồng Yến	Thành viên	• Báo cáo: *Viết, phân tích:* ✓ *Trình bày tổng quan về đề tài. Tìm hiểu bài toán phân lớp.* ✓ *Những giải pháp khoa học nghiên cứu.* ✓ *Tìm hiểu và trình bày về bộ dữ liệu Break-his.* • Chương trình demo: ✓ *Chạy thực nghiệm Fully - trained với 3 kiến trúc*	*Hoàn thành tốt*	*10/10*

			mạng trên phân tách dữ liệu 80%-20%		
3	Hồ Thị Thanh Vân	Thành viên	• Báo cáo: *Viết, phân tích:* ✓ *Mạng kiến trúc tích chập.* ✓ *Trình bày, tìm hiểu độ đo đánh giá. Thực nghiệm, đánh giá so sánh giữa bài báo và thực nghiệm của nhóm.* ✓ *Kết luận* • Chương trình demo: ✓ *Chạy thực nghiệm Fully - trained với 3 kiến trúc mạng trên phân tách dữ liệu 70%-30%*	*Hoàn thành tốt*	*10/10*
4	Nguyễn Đức Quang	Thành viên	• Báo cáo: *Viết, phân tích:* ✓ *Trình bày, tìm hiểu mạng kiến trúc VGG16, VGG19, ResNet50.* ✓ *Tìm hiểu bài toán phân lớp.* • Chương trình demo: ✓ *Chạy thực nghiệm Fully trained với 3 kiến trúc mạng trên phân tách dữ liệu 90%-10%*	*Hoàn thành chưa tốt*	*6.5/10*

TÀI LIỆU THAM KHẢO

[Online]. Available: https://gco.iarc.fr/today/online-analysis-pie. [Accessed 4 1] 29 2022].

L. Anh, "Hơn 20.000 phụ nữ Việt Nam mắc ung thư vú trong năm 2020," 2] 2021. [Online]. Available: http://daidoanket.vn/hon-20000-phu-nu-viet-nam-mac-ung-thu-vu-trong-nam-2020-550727.html. [Accessed 26 5 2022].

"8 yếu tố nguy cơ gây ung thư vú và cách phòng ngừa," 2022. [Online]. 3] Available: 8 yếu tố nguy cơ gây ung thư vú và cách phòng ngừa. [Accessed 26 5 2022].

P. Thúy, "Ai dễ mắc ung thư vú và cách nhận biết?," 2021. [Online]. Available: 4] https://suckhoedoisong.vn/ai-de-mac-ung-thu-vu-va-cach-nhan-biet-16921100322494192 6.htm. [Accessed 26 5 2022].

J. Han, M. Kamber and J. Pei, Data Mining - Concepts and Techniques (3rd 5] Ed), 2012.

N. H. D. Trí, Writer, *Slide Chương 2. Các vấn đề về tiền xử lý dữ liệu.* 6] [Performance].

N. T. Hoàn, "Phương pháp trích chọn đặc trưng ảnh trong thuật toán học máy 7] tìm kiếm ảnh áp dụng vào bài toán tìm kiếm sản phẩm," 2010. [Online]. Available: http://vnlp.net/wp-content/uploads/2010/06/KLTN_NguyenThiHoan_final.pdf.

H. V. Hiệp, "Xử lý ảnh," 2011. [Online]. Available: http://www.zun.vn/tai-8] lieu/xu-ly-anh-chuong-5-trich-chon-cac-dac-trung-trong-anh-42253/.

M. Balck, "Research Gate," [Online]. Available: https://www.researchgate.net/figure/An-example-of-calculating-a-Grey-Level-Co-occurrence-Matrix-Adapted-from-Gonzalez-and_fig2_261490953. [9]

S. Saha, "A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way," 2018. [Online]. Available: https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53. [Accessed 26 5 2022]. [10]

H. T. Việt, "Tìm hiểu về mạng nơron tích chập (convolutional neural networks)," [Online]. Available: https://thanhvie.com/tim-hieu-ve-mang-no-ron-tich-chap-convolutional-neural-networks/#:~:text=M%E1%BA%A1ng%20n%C6%A1ron%20t%C3%ADch%20ch%E1%BA%ADp%20(c%C3%B2n,%C4%91%E1%BB%91i%20t%C6%B0%E1%BB%A3ng%20n%C3%A0y%20v%E1%BB%9Bi%20nhau. [11]

savyakhosla, "CNN | Introduction to Pooling Layer," 2021. [Online]. Available: https://www.geeksforgeeks.org/cnn-introduction-to-pooling-layer/. [12]

N. Đ. Trung, "Tìm hiểu về Convolutional Neural Network và làm một ví dụ nhỏ về phân loại ảnh," 2020. [Online]. Available: https://viblo.asia/p/tim-hieu-ve-convolutional-neural-network-va-lam-mot-vi-du-nho-ve-phan-loai-anh-aWj53WXo56m. [13]

G. Boesch, "VGG Very Deep Convolutional Networks (VGGNet) – What you need to know," [Online]. Available: https://viso.ai/deep-learning/vgg-very-deep-convolutional-networks/. [14]

P. Nepal, "VGGNet Architecture Explained," 2020. [Online]. Available: https://medium.com/analytics-vidhya/vggnet-architecture-explained-e5c7318aa5b6. [15]

Rismiyati, Ardytha Luthfiarta, "VGG16 Transfer Learning Architecture for
16] Salak Fruit Quality Classification," *Informasi Artikel,* vol. 18, 2021.

Yufeng Zhenga, Clifford Yangb, Alex Merkulovb, "Breast Cancer Screening
17] Using Convolutional Neural Network and Follow-up Digital Mammography," *Conference Paper,* 2018.

T. Đ. Thắng, "Giới thiệu mạng ResNet," 2020. [Online]. Available:
18] https://viblo.asia/p/gioi-thieu-mang-resnet-vyDZOa7R5wj.

"Tổng quan về bài toán phân lớp," [Online]. Available: https://tek4.vn/khoa-
19] hoc/machine-learning-co-ban/tong-quan-ve-bai-toan-phan-lop.

J. Brownlee, "4 Types of Classification Tasks in Machine Learning," 2020.
20] [Online]. Available: https://machinelearningmastery.com/types-of-classification-in-machine-learning/. [Accessed 26 5 2022].

"Multiclass Classification with NumPy and TMVA," [Online]. Available:
21] http://scikit-hep.org/root_numpy/auto_examples/tmva/plot_multiclass.html. [Accessed 26 5 2022].

"TOP 10 THUẬT TOÁN MACHINE LEARNING," [Online]. Available:
22] https://dataisg.org/tutorial/machine-learning/hoi-quy-logistic/. [Accessed 27 4 2022].

S. Stefanovic, "#005 PyTorch – Logistic Regression in PyTorch," 2020.
23] [Online]. Available: https://datahacker.rs/005-pytorch-logistic-regression-in-pytorch/. [Accessed 27 4 2022].

S. H. Ngọc, "Confusion Matrix / Ma trận nhầm lẫn / Ma trận lỗi," 2021.
24] [Online]. Available: https://viblo.asia/p/confusion-matrix-ma-tran-nham-lan-ma-tran-loi-V3m5WQB7ZO7.

N. H. Nam, "Tìm hiểu về Confusion matrix trong Machine Learning?," 2018.
25] [Online]. Available: https://www.noron.vn/post/tim-hieu-ve-confusion-matrix-trong-machine-learning-1fz9nhqo5ux.

A. Gad, "Evaluating Object Detection Models Using Mean Average
26] Precision," 2021. [Online]. Available: https://www.kdnuggets.com/2021/03/evaluating-object-detection-models-using-mean-average-precision.html. [Accessed 26 5 2022].

Jinchan Qu, Albert Steppi, Dongrui Zhong, Jie Hao, "Triage of documents
27] containing protein interactions affected by mutations using an NLP based machine learning approach," *BMC Genomics,* 2020.

N. H. Nam, "Tìm hiểu chi tiết về AUC - ROC trong Machine Learning?,"
28] 2018. [Online]. Available: https://www.noron.vn/post/tim-hieu-chi-tiet-ve-auc---roc-trong-machine-learning-1fz9nhqo5ut.

P. Đ. Khánh, "Bài 38 - Các kiến trúc CNN hiện đại," 2020. [Online].
29] Available: https://phamdinhkhanh.github.io/2020/05/31/CNNHistory.html. [Accessed 26 5 2022].

P. T. Đạt, "LDAP [Part 1] - Giới thiệu về LDAP," [Online]. Available:
30] https://blog.cloud365.vn/ldap/LDAP-part-1-gioi-thieu-ve-LDAP/. [Accessed 20 5 2022].