Transformer in Medical Analysis

Mai Gia Chung [1], Nguyễn Tuấn Kiệt [2], Phan Đặng Anh Khôi [3†]

[1]University of Science, Vietnam National University, 227 Nguyen Van Cu, Ho Chi Minh city, 700000, District 5, Vietnam.

Contributing authors: 20127415@student.hcmus.edu.vn; 21127088@student.hcmus.edu.vn; 21127325@student.hcmus.edu.vn;
[†]These reporters contributed equally to this work.

Tóm tắt nội dung

Bài báo cáo này trình bày tổng quan về sự trỗi dậy và ứng dụng rộng rãi của kiến trúc Transformer trong lĩnh vực phân tích ảnh y tế. Ban đầu, kiến trúc này đã thống trị lĩnh vực xử lý ngôn ngữ tự nhiên (NLP), Transformer đã nhanh chóng chứng minh được hiệu quả vượt trội trong nhiều tác vụ thị giác máy tính, bao gồm phân loại, phân đoạn và phát hiện đối tượng trong ảnh y tế. Bài báo cáo này đi sâu vào các phương pháp tiếp cận khác nhau, các ứng dụng cụ thể trên nhiều phương thức hình ảnh và bệnh lý, cũng như những ưu điểm và thách thức của việc sử dụng Transformer trong bối cảnh y tế. Bài báo cáo này lấy cảm hứng từ bài nghiên cứu tổng quát Transformer trong ứng dụng phân tích dữ liệu y khoa - "Transformers in Medical Image Analysis: A Review"[1].

Keywords: Transformers, Phân tích ảnh y tế, Học sâu, Chẩn đoán, Phân đoạn, Tổng hợp ảnh, Học đa nhiệm, Học đa phương thức, Học bán giám sát

1 Giới thiệu

Trong những năm gần đây, lĩnh vực thị giác máy tính đã chứng kiến những tiến bộ vượt bậc nhờ sự phát triển của các mạng nơ-ron tích chập (CNN). CNN đã trở thành kiến trúc chủ đạo cho nhiều bài toán phân tích hình ảnh y tế, đóng góp quan trọng vào việc cải thiện độ chính xác và hiệu quả của chẩn đoán và điều trị bệnh. Tuy nhiên, CNN thường gặp khó khăn trong việc mô hình hóa các mối quan hệ phụ thuộc dài hạn trong hình ảnh do tính chất cục bộ của các phép tích chập.

Kiến trúc Transformer, được đề xuất lần đầu tiên cho bài toán dịch máy, đã mang đến một cách tiếp cận mới dựa trên cơ chế tự chú ý (self-attention), cho phép mô

hình học được các tương tác giữa các phần khác nhau của dữ liệu đầu vào, bất kể khoảng cách không gian giữa chúng. Nhờ khả năng này, Transformer đã nhanh chóng trở thành kiến trúc ưu việt trong xử lý ngôn ngữ tự nhiên (NLP) và gần đây đã được khám phá và ứng dụng rộng rãi trong thị giác máy tính.

Ở bối cảnh phân tích hình ảnh y tế, Transformer hứa hẹn sẽ khắc phục những hạn chế của CNN và mang lại những cải tiến đáng kể trong nhiều nhiệm vụ quan trọng như phân loại bệnh, phân đoạn các cấu trúc giải phẫu và bệnh lý, cũng như phát hiện các dấu hiệu bất thường. Báo cáo này nhằm mục đích cung cấp một cái nhìn tổng quan toàn diện về việc ứng dụng kiến trúc Transformer trong phân tích hình ảnh y tế, tập trung vào cơ sở lý thuyết, các ứng dụng chính và những triển vọng trong tương lai.

Trong bài báo cáo này, nhóm sẽ đề xuất ra một phương pháp đại diện để nêu bật khả năng ứng dụng của mô hình Transformer trong lĩnh vực phân tích dữ liệu y khoa, đó là nghiên cứu "COVID-19 automatic diagnosis with CT images using the novel Transformer architecture do Gabriel Sousa Silva Costa và các cộng sự phát triển - sẽ được nhóm trình bày ở phần 4.

1.1 Động lực nghiên cứu

Về mặt khoa học, Transformer với cơ chế self-attention giúp mô hình nắm bắt được mối liên hệ toàn cục giữa các đặc trưng trong ảnh, giải quyết vấn đề của CNN. Các biến thể như Vision Transformer (ViT) [2], Swin Transformer [3] hay TransUNet [4] đã chứng minh được tính ưu việt trong nhiều tác vụ y học như phân đoạn khối u, nhận diện bệnh lý trong ảnh X-quang, MRI, CT...

Về mặt thực tiễn, Transformer đã đạt được nhiều tiến bộ trong xử lý ngôn ngữ tự nhiên (NLP) và thị giác máy tính. Điều này mở ra tiềm năng ứng dụng rộng rãi của Transformer trong phân tích ảnh y khoa, giúp cải thiện độ chính xác và khả năng tự động hóa trong lĩnh vực này.

1.2 Phát biểu bài toán

Input: Dữ liệu y khoa dưới dạng văn bản (hồ sơ bệnh án), hình ảnh (X-quang, MRI) hoặc tín hiệu sinh học (ECG, EEG).
Output: Mô hình Transformer có thể đưa ra các dự đoán hoặc phân loại dựa trên dữ liệu đầu vào, chẳng hạn như:

- Phân loại bệnh lý từ hình ảnh y khoa;
- Phát hiện triệu chứng từ văn bản bệnh án;
- Dự đoán nguy cơ mắc bệnh dựa trên hồ sơ sức khỏe.

2 Cơ sở lý thuyết

2.1 Cơ chế Chú ý (Attention Mechanism)

Cơ chế chú ý [5] là một thành phần cốt lõi của kiến trúc Transformer, được giới thiệu lần đầu trong bài toán dịch máy thần kinh. Mục tiêu chính của cơ chế này là cho phép mô hình tự động tìm kiếm và tập trung vào những phần quan trọng nhất của dữ liệu đầu vào khi đưa ra dự đoán cho dữ liệu đầu ra. Thay vì cố gắng mã hóa toàn

bộ thông tin đầu vào vào một vector có độ dài cố định, cơ chế chú ý cho phép mô hình lựa chọn một cách linh hoạt các phần thông tin liên quan nhất để xử lý .

Trong kiến trúc Transformer, cơ chế chú ý thường được thực hiện thông qua việc tính toán sự tương tác giữa ba thành phần: Query (Q), Key (K) và Value (V). Đối với mỗi vị trí trong chuỗi đầu vào, một vector query, key và value được tạo ra. Attention weight (α_{ij}) giữa phần tử thứ i của đầu ra và phần tử thứ j của đầu vào được tính toán dựa trên độ tương đồng giữa query thứ i (q_i) và key thứ j (k_j^T):

$$\alpha'_{ij} = q_i \cdot k_j^T \tag{1}$$

$$\alpha_{ij} = Softmax(\frac{\alpha'_{ij}}{\sqrt{d_k}}) = \frac{\exp(\alpha'_{ij}/\sqrt{d_k})}{\sum_j \exp(\alpha'_{ij}/\sqrt{d_k})} \tag{2}$$

Trong đó, d_k là một hệ số tỷ lệ (thường là căn bậc hai của chiều của vector key) giúp ổn định quá trình huấn luyện. Cuối cùng, đầu ra (z_i) là tổng trọng số của các vector value (v_j), với trọng số là attention weight:

$$z_i = \sum_j \alpha_{ij} \cdot v_j \tag{3}$$

Cơ chế self-attention còn có thể được viết dưới dạng ma trận. Gọi $X \in R^{s \times c}$ là ma trận đầu vào, Q, K, V lần lượt là ma trận truy vấn, ma trận khóa và ma trận giá trị, trong đó s là số lượng mẫu, và mỗi ma trận được cấu thành từ các phần tử như sau: $X = [x_1; x_2; ...; x_s]^T$. Như vậy, ma trận Attention A và ma trận đầu ra Z được tính toán như sau:

$$A = Softmax(\frac{Q \times K^T}{\sqrt{d_k}}) \in R^{s \times s} \tag{4}$$

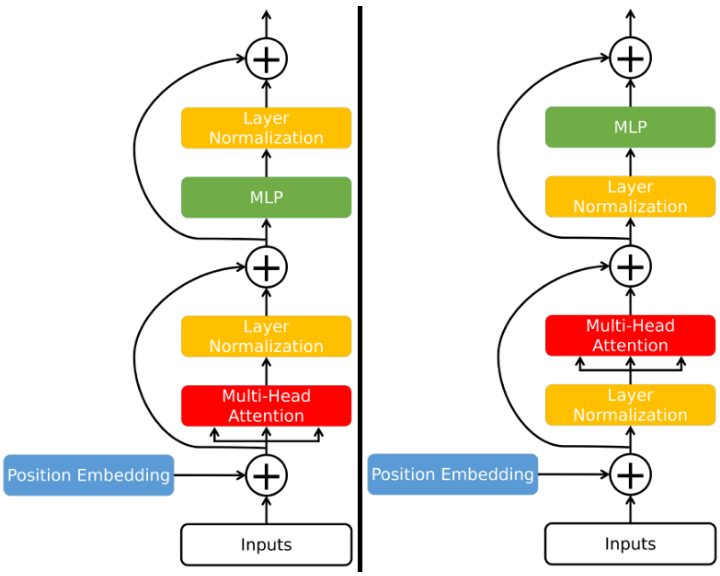
$$Z = A \times V \in R^{s \times d_v} \tag{5}$$

Khi áp dụng nhiều cơ chế self-attention lên cùng một đầu vào có thể giúp mô hình nắm bắt tốt hơn các đặc trưng phân cấp (hierachiacal features). Với h cơ chế self-attention (heads), mô-đun sẽ cho ra kết quả cuối cùng bằng cách dùng phép nối (concatenate) các self-attention đã tính toán lại với nhau.

$$Z_i = Attention(Q \times W_i^Q, K \times W_i^K, V \times W_i^V) \tag{6}$$

$$MultiHead(Q, K, V) = Concat(Z_1, Z_2, ..., Z_h)W^O \tag{7}$$

trong đó W_i^Q, W_i^K, W_i^V là các ma trận chiếu tuyến tính (linear projection matrices) dùng để ánh xạ các ma trận Q, K, V vào các không gian con khác nhau tương ứng. Ma trận W^O là ma trận chiếu đầu ra dùng để kết hợp (concatenate) các kết quả của tất cả các đầu attention lại với nhau.



Hình 1: Bên trái là Transformer gốc; bên phải là Vision Transformer

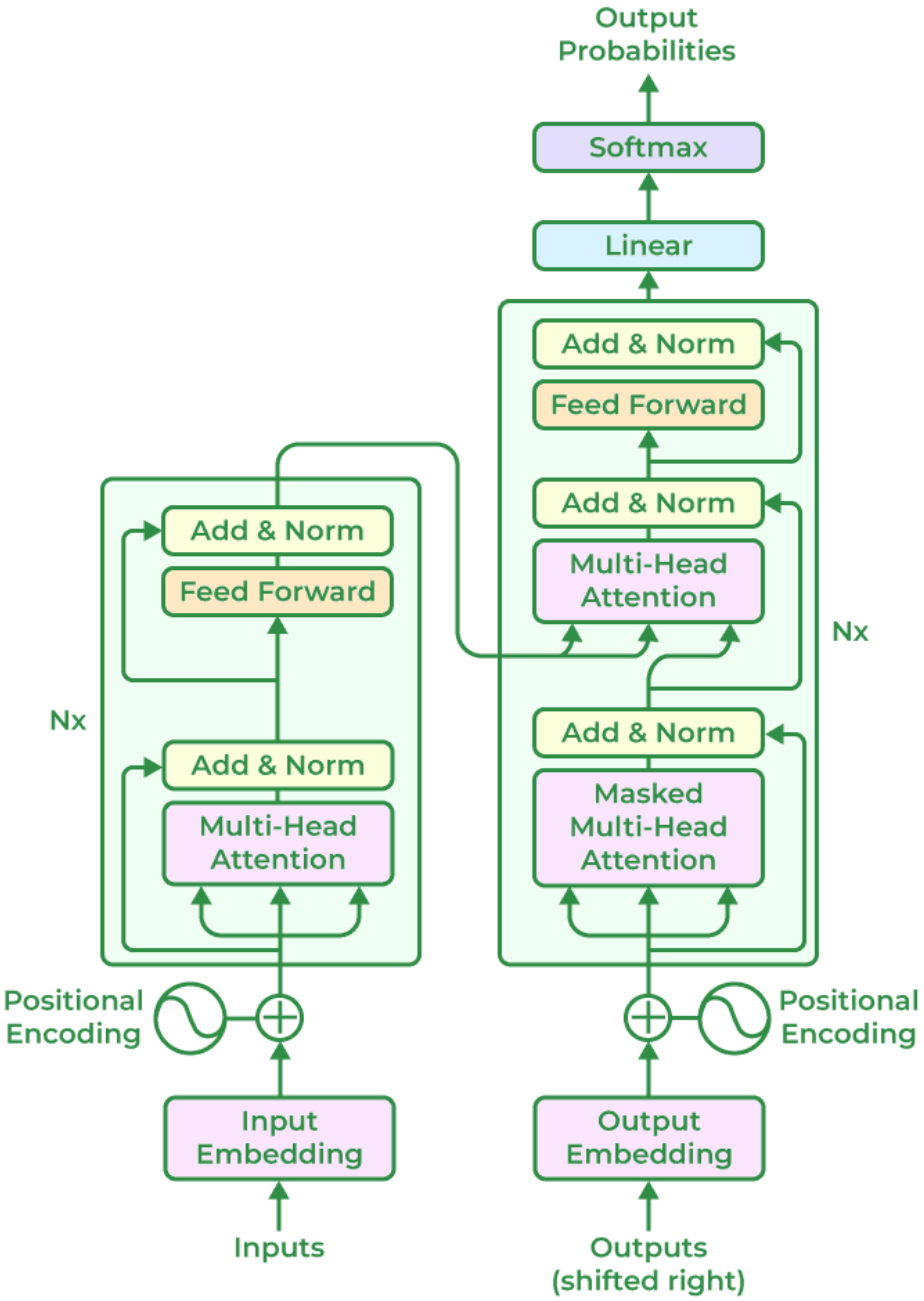
2.2 Kiến trúc Transformer

Kiến trúc Transformer ban đầu được thiết kế cho các bài toán xử lý ngôn ngữ tự nhiên và sau đó đã được chứng minh là rất hiệu quả trong thị giác máy tính. Một Transformer điển hình bao gồm một encoder và một decoder. Encoder xử lý chuỗi đầu vào và tạo ra các biểu diễn trung gian, trong khi decoder sử dụng các biểu diễn này để tạo ra chuỗi đầu ra.

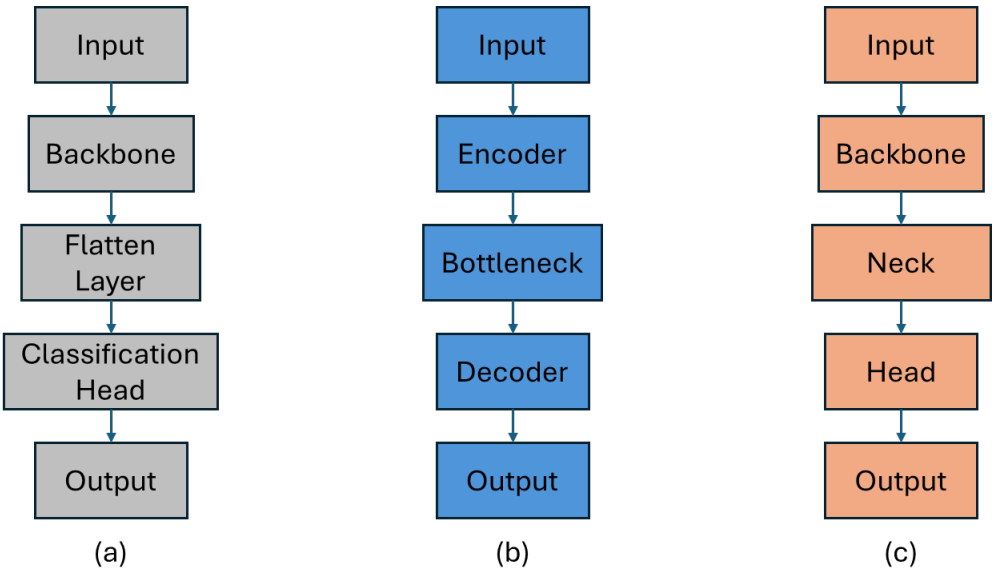
Mỗi lớp trong encoder và decoder thường bao gồm hai khối chính: một lớp multi-head self-attention (MSA) và một lớp feed-forward network (FFN). Các lớp này thường được kết nối bằng các kết nối dư (residual connections) và chuẩn hóa lớp (layer normalization).

Đối với các bài toán thị giác máy tính, chẳng hạn như Vision Transformer (ViT), hình ảnh đầu vào thường được chia thành một chuỗi các patch không chồng lấn và mỗi patch được coi như một "token". Một lớp embedding tuyến tính sau đó được áp dụng cho mỗi patch để tạo ra các vector đầu vào cho Transformer encoder. Thông tin về vị trí của các patch trong hình ảnh thường được thêm vào thông qua vị trí mã hóa (positional encoding), cho phép mô hình nhận biết được cấu trúc không gian của hình ảnh. Các hàm sin và cos thường được sử dụng để mã hóa vị trí.

Ở hình 1, có sự thay đổi của vị trí Layer Norm. Cụ thể từ Transformer gốc là Post-Normalization (nghĩa là input đi vào MSA hay MLP trước rồi mới tới Layer Norm) đến Vision Transformer là Pre-Normalization (ngược lại với Post-Normalization). Trong bài báo [2], người ta nhận thấy rằng việc thay đổi vị trí của Layer Norm giúp quá trình huấn luyện ổn định hơn, đặc biệt là các mạng Transformer sâu.



Hình 2: Mô hình kiến trúc của Transformer



Hình 3: (a) Framework chung phân loại trong ảnh y khoa; (b) Framework chung phân đoạn trong ảnh y khoa; (c) Framework chung phát hiện đối tượng trong ảnh y khoa

2.2.1 Framework chung phân loại trong ảnh y tế (hình 3a)

Các mô hình phân loại điển hình chủ yếu gồm 3 phần:

- **Backbone**: là thành phần cốt lõi, học cách trích xuất các đặc trưng hình ảnh như các mẫu (pattern), kết cấu, hình dạng và các thông tin khác liên quan đến các lớp cần phân loại.
- **Lớp flatten**: có nhiệm vụ giảm kích thước không gian của bản đồ đặc trưng cuối cùng từ Backbone thành một vector đặc trưng có kích thước cố định, đại diện cho toàn bộ ảnh.
- **Classification Head**: lấy vector đặc trưng từ lớp flatten và đưa ra dự đoán về lớp cuối cùng. Thường là một hay nhiều lớp Fully Connected. Đầu ra là nhãn lớp được dự đoán hoặc phân phối xác suất trên các lớp.

Việc áp dụng Transformer vào phân loại ảnh y tế gồm: Áp dụng trực tiếp Transformer vào ảnh y khoa như ViT (được làm rõ ở phần 3.1.1). Hoặc kết hợp với các tích chập để học thêm các biểu diễn đặc trưng cục bộ.

2.2.2 Framework chung phân đoạn trong ảnh y tế (hình 3b)

Hầu hết các mô hình phân đoạn trong ảnh y tế đều có cấu tạo giống như hình chữ U (UNet) và gồm 3 phần chính: Encoder, Bottleneck và Decoder.

- **Encoder**: trích xuất các đặc trưng ở nhiều cấp độ khác nhau. Khi đi sâu hơn vào Encoder, độ phân giải không gian (resolution) của bản đồ đặc trưng giảm xuống,

nhưng số lượng kênh đặc trưng (chiều sâu) tăng lên, giúp mô hình học được các đặc trưng phức tạp hơn.
- **Bottleneck**: dùng nối giữa encoder và decoder. Đây là nơi bản đồ đặc trưng có độ phân giải không gian thấp nhất nhưng biểu diễn các đặc trưng ở mức độ trừu tượng cao nhất.
- **Decoder**: tăng dần độ phân giải không gian của bản đồ đặc trưng và định vị chính xác các đối tượng cần phân đoạn.

Các ý tưởng liên quan đến Transformer là thay thế hoặc bổ sung các phần trên. Như TransUNet đã giữ nguyên Decoder của UNet và bổ sung thêm Encoder Transformer sau Encoder CNN của UNet (phần 3.2.1) hay Swin Transformer (phần 3.2.2) đã thay thế hoàn toàn 3 phần đó bằng cách tạo ra một khối Transformer mới (gọi là Swin Transformer Block).

2.2.3 Framework chung phát hiện đối tượng trong ảnh y tế (hình 3c)

Kiến trúc của các mô hình phát hiện đối tượng cũng bao gồm 3 thành phần chính: Backbone, Neck và Head.

- **Backbone**: giống như trong mô hình phân loại, Backbone có nhiệm vụ trích xuất đặc trưng từ ảnh đầu vào. Nó học các biểu diễn hình ảnh ở nhiều cấp độ trừu tượng khác nhau, từ các cạnh, góc đơn giản đến các cấu trúc phức tạp hơn.
- **Neck**: kết hợp và tinh chỉnh các đặc trưng được trích xuất từ các tầng khác nhau của Backbone - kết hợp ngữ cảnh (từ lớp sâu) và chi tiết không gian (từ lớp nông). Điều này rất quan trọng để phát hiện các đối tượng ở nhiều kích thước khác nhau.
- **Head**: dự đoán vị trí bounding box và phân loại lớp cho các đối tượng được phát hiện. Tùy theo các họ mô hình mà kiến trúc phần head khác nhau.
  - **Two-Stage Detectors (ví dụ: R-CNN)**: gồm 2 giai đoạn. Giai đoạn 1 là RPN (Region Proposal Network (RPN): mạng này quét qua bản đồ đặc trưng và đề xuất các vùng có khả năng chứa đối tượng (Region of Interest - RoI). Giai đoạn 2 là RoI Head: trích xuất đặc trưng từ các vùng RoI đã được đề xuất ở giai đoạn 1, sau đó đưa các đặc trưng RoI cố định này vào các lớp fully connected để phân loại đối tượng.
  - **One-Stage Detectors (ví dụ: YOLO)**: không có giai đoạn 1 như mô hình trên. Mô hình dự đoán trực tiếp tọa độ bounding box và xác suất lớp.
  - **Transformer-based Detectors (ví dụ: DETR)**: sử dụng một tập hợp các "object queries"(vector truy vấn đối tượng) có thể học được đưa vào Transformer Encoder-Decoder cùng với các đặc trưng được trích xuất từ Backbone + FFN (Prediction Head), mỗi object query đầu ra từ decoder được đưa qua một mạng này để dự đoán trực tiếp lớp (bao gồm cả lớp "no objects") và tọa độ bounding box. Mô hình COTR được xây dựng dựa trên DETR sẽ được nói rõ hơn ở phần 3.3.

3 Ứng dụng của Transformer trong Phân tích Hình ảnh Y tế:

3.1 Phân loại hình ảnh y tế

3.1.1 Vision Transformer (ViT) trong phân loại hình ảnh y tế

Vision Transformer (ViT) [2] là một trong những ứng dụng đầu tiên và nổi bật của Transformer trong phân loại hình ảnh y tế. ViT xử lý hình ảnh bằng cách chia nhỏ thành các *patch* (khối ảnh nhỏ), sau đó mã hóa chúng thành chuỗi *embedding* để đưa vào Transformer encoder. Các nghiên cứu đã chỉ ra rằng ViT, khi được huấn luyện trên lượng lớn dữ liệu, có thể đạt được hiệu suất ngang bằng hoặc thậm chí vượt trội so với các mạng CNN tiên tiến trong nhiều bài toán phân loại hình ảnh y tế.

- **Chẩn đoán COVID-19**: ViT đã được áp dụng trong việc phân loại ảnh chụp CT phổi/Xray để phát hiện COVID-19, với nhiều nghiên cứu kết hợp ViT với các cơ chế chú ý hiệu quả hơn như Performer để giảm chi phí tính toán.
- **Phân loại các bệnh khác**: Transformer đã được sử dụng thành công trong các bài toán phân loại như ung thư vú từ ảnh siêu âm, bạch cầu cấp tính từ ảnh tế bào máu, và melanoma từ ảnh da.
- **Phân loại ảnh đáy mắt**: Một biến thể của ViT là MIL-VT (Multiple Instance Learning enhanced Vision Transformer) đã được đề xuất để phân loại ảnh đáy mắt.

Nguyên lý hoạt động của ViT:

1. **Phân chia ảnh thành các bản vá (Patch Splitting)**: Hình ảnh đầu vào được chia thành một lưới các bản vá không chồng chéo, mỗi bản vá có kích thước cố định (ví dụ: 16×16 hoặc 32×32 pixel). Mỗi bản vá này sẽ đóng vai trò như một phần tử trong chuỗi đầu vào của mô hình.
2. **Nhúng bản vá vào không gian đặc trưng (Patch Embedding)**: Mỗi bản vá được làm phẳng thành một vector và sau đó ánh xạ vào một không gian có số chiều cao hơn thông qua một lớp fully connected layer. Việc này giúp chuyển đổi dữ liệu hình ảnh thành dạng mà Transformer có thể xử lý được.
3. **Mã hóa vị trí (Positional Encoding)**: Do Transformer không có cơ chế nhận biết vị trí tương đối của các phần tử đầu vào như CNN, ViT sử dụng mã hóa vị trí (positional encoding) để cung cấp thông tin không gian cho mô hình. Mã hóa vị trí này sẽ được cộng trực tiếp vào vector nhúng của mỗi bản vá.
4. **Truyền dữ liệu vào Transformer Encoder**: Chuỗi các bản vá đã được nhúng sẽ đi qua Transformer Encoder, bao gồm các thành phần chính:
   - **Lớp Self-Attention**: Học mối quan hệ giữa các bản vá trên toàn bộ ảnh, giúp mô hình hiểu được các đặc trưng toàn cục.
   - **Lớp MLP (Multi-Layer Perceptron)**: Kết hợp thông tin và tăng cường biểu diễn đặc trưng sau khi qua Self-Attention.
   - **Normalization và Skip Connection**: Giúp ổn định quá trình huấn luyện và cải thiện khả năng hội tụ của mô hình.

5. **Cơ chế Self-Attention (Tự chú ý đa đầu - MHSA)**: ViT áp dụng cơ chế Multi-Head Self-Attention (MHSA), trong đó các trọng số chú ý được tính theo công thức:

$$\text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d}}\right)V$$

với Q, K, V lần lượt là ma trận truy vấn (Query), khóa (Key), và giá trị (Value), còn d là số chiều của vector truy vấn/khóa.

6. **Dự đoán phân loại (Classification)**: Để thực hiện phân loại, ViT sử dụng một *class token* đặc biệt, được thêm vào chuỗi bản vá đầu vào. Sau khi đi qua các lớp Transformer, vector biểu diễn của class token sẽ được đưa vào một MLP head để dự đoán lớp của hình ảnh.

3.1.2 Học tập hiệu quả dữ liệu với DEiT (Data-efficient Image Transformers)

Mặc dù ViT có khả năng đạt hiệu suất cao, nó yêu cầu lượng dữ liệu huấn luyện lớn, điều này không phải lúc nào cũng khả thi trong lĩnh vực y tế. DEiT (Data-efficient Image Transformers) là một cải tiến quan trọng, được phát triển để huấn luyện Transformer hiệu quả hơn trên các tập dữ liệu nhỏ mà vẫn duy trì hiệu suất cao.

Cơ chế hoạt động của DEiT

DEiT sử dụng chiến lược giáo viên-học sinh (*teacher-student*) để truyền đạt kiến thức từ một mô hình CNN đã được huấn luyện trước sang một mô hình Transformer. Cách tiếp cận này giúp Transformer học nhanh hơn và chính xác hơn, ngay cả khi tập dữ liệu huấn luyện bị giới hạn.

- **Teacher Model**: Một mạng CNN đóng vai trò như một mô hình giáo viên, đã được huấn luyện trước trên dữ liệu y tế.
- **Student Model**: Một Transformer (ViT hoặc biến thể) sẽ học từ dữ liệu gốc và cả những thông tin do mô hình giáo viên cung cấp.

Kiến trúc của DEiT

DEiT kế thừa kiến trúc ViT nhưng được tối ưu hóa để học hiệu quả hơn với dữ liệu hạn chế. Kiến trúc của DEiT bao gồm các thành phần chính sau:

1. **Backbone – Trích xuất đặc trưng**: Sử dụng Convolution Layer (CNN) để học đặc trưng cục bộ và toàn cục trước khi đưa vào Transformer. Các đặc trưng chính bao gồm:
   - **Đặc trưng cấu trúc (Structural Features)**: Skeletonization, cạnh biên, nút phân nhánh.
   - **Đặc trưng toàn cục (Global Features)**: Diện tích, chu vi, độ lệch tâm.
2. **Encoding – Biểu diễn dữ liệu**: Mỗi bản vá ảnh được biến đổi thành vector embedding và được thêm mã hóa vị trí như trong ViT. DEiT áp dụng Multi-Head Self-Attention (MHSA) để xác định mối quan hệ giữa các bản vá ảnh, giúp mô hình tập trung vào các vùng quan trọng.

3. **Decoding – Xác định đối tượng**: Khác với CNN xử lý ảnh theo từng phần tuần tự, DEiT chạy song song trên toàn bộ ảnh, giúp tăng tốc quá trình nhận diện. Dự đoán bounding box dựa trên attention scores để xác định vị trí chính xác của đối tượng.
4. **Prediction Heads – Dự đoán đối tượng**: DEiT sử dụng một mạng Feed-Forward (FFN) để thực hiện phân loại ảnh và dự đoán vị trí đối tượng. Các hàm mất mát chính được sử dụng để tối ưu mô hình:
   - **IoU Loss (Intersection over Union)**: Đánh giá mức độ trùng khớp giữa bounding box dự đoán và ground-truth.

$$IoU = \frac{TP}{TP + FP + FN}$$

   - **L1 Loss**: Đo sai số vị trí giữa bounding box dự đoán và ground-truth, giúp tối ưu độ chính xác.

Lợi ích của DEiT trong phân loại hình ảnh y tế

- **Giảm chi phí tính toán**: Không yêu cầu lượng lớn dữ liệu để đạt hiệu suất cao.
- **Tăng độ chính xác trên tập dữ liệu nhỏ**: Phù hợp với bài toán y tế, nơi dữ liệu thường bị giới hạn.
- **Tận dụng được đặc trưng từ CNN**: Kết hợp sức mạnh của CNN và Transformer để đạt hiệu suất tối ưu.

3.1.3 Hạn chế của ViT và cải tiến với DEiT

Mặc dù Vision Transformer (ViT) đã chứng minh được hiệu quả vượt trội trong phân tích hình ảnh, nhưng nó vẫn tồn tại một số hạn chế quan trọng:

- **Yêu cầu lượng dữ liệu huấn luyện lớn**: ViT hoạt động tối ưu khi được huấn luyện trên các tập dữ liệu lớn, chứa hàng triệu hình ảnh. Điều này khiến nó khó áp dụng trong các lĩnh vực có dữ liệu hạn chế như y tế.
- **Không tận dụng được tri thức từ CNN**: ViT không khai thác các đặc trưng cục bộ vốn rất quan trọng trong phân tích ảnh y tế, nơi các mô hình CNN đã chứng minh được hiệu quả cao trong việc phát hiện và trích xuất đặc trưng từ hình ảnh.

Giải pháp cải tiến của DEiT

Data-efficient Image Transformer (DEiT) được đề xuất để khắc phục những hạn chế trên bằng các kỹ thuật tối ưu hóa sau:

- **Học chuyển giao từ CNN (Teacher-Student Learning)**: DEiT sử dụng cơ chế huấn luyện teacher-student, trong đó một mô hình CNN đóng vai trò là giáo viên (teacher) để hướng dẫn ViT, giúp nó học tốt hơn ngay cả với tập dữ liệu nhỏ.
- **Tăng cường dữ liệu thông minh**: DEiT kết hợp nhiều kỹ thuật tăng cường dữ liệu tiên tiến như RandAugment, Mixup và CutMix để cải thiện khả năng tổng quát hóa của mô hình, giúp ViT học tốt hơn trên tập dữ liệu giới hạn.

- **Giảm chi phí tính toán mà vẫn đảm bảo hiệu quả**: Nhờ vào các cải tiến trong cơ chế huấn luyện và tối ưu hóa, DEiT giúp ViT đạt hiệu suất cao hơn với thời gian huấn luyện ngắn hơn và yêu cầu ít tài nguyên tính toán hơn so với ViT gốc.

3.2 Phân đoạn hình ảnh y tế

Xác định và khoanh vùng các vùng hoặc cấu trúc cụ thể trong ảnh y tế, chẳng hạn như khối u, cơ quan hoặc các vùng bệnh lý. Các phương pháp truyền thống thường dựa trên mạng nơ-ron tích chập (CNNs), đặc biệt là kiến trúc Unet, đã đạt được thành công lớn trong nhiều tác vụ phân đoạn hình ảnh y tế. Tuy nhiên, Unet và các CNNs khác bị giới hạn trong việc mô hình hóa các phụ thuộc dài hạn do bản chất cục bộ của các phép toán tích chập.

Để vượt qua những hạn chế này, các nhà nghiên cứu đã nỗ lực thiết kế các kiến trúc Transformer kết hợp mạnh mẽ với kiến trúc Unet (mô hình hybrid). Ngoài ra, một số phương pháp cũng áp dụng Transformer thuần túy cho các nhiệm vụ phân đoạn.

3.2.1 Mô hình Hybrid: TransUNet

TransUNet [4] là một kiến trúc hybrid nổi bật kết hợp Transformer vào trong kiến trúc U-Net truyền thống cho nhiệm vụ phân đoạn hình ảnh y tế. Phương pháp này tận dụng khả năng học các đặc trưng không gian có độ phân giải cao của CNN (trong phần encoder của U-Net) và khả năng mô hình hóa ngữ cảnh toàn cục của Transformer.

Kiến trúc TransUNet bao gồm một encoder dựa trên CNN và một decoder dựa trên U-Net. Điểm khác biệt chính là một lớp Transformer được chèn vào giữa encoder và decoder.

- **Encoder CNN**: Phần encoder của U-Net được sử dụng để trích xuất các bản đồ đặc trưng ở các độ phân giải khác nhau từ ảnh đầu vào.
- **Transformer Encoder**: Các bản đồ đặc trưng có độ phân giải thấp nhất từ encoder CNN sau đó được làm phẳng thành một chuỗi các token và đưa vào Transformer Encoder. Transformer này có khả năng nắm bắt các phụ thuộc dài hạn và mô hình hóa ngữ cảnh toàn cục trong dữ liệu. Các lớp self-attention trong Transformer cho phép mỗi token tương tác với tất cả các token khác, từ đó học được các mối quan hệ trên toàn bộ hình ảnh.
- **Decoder U-Net**: Các đặc trưng được mã hóa bởi Transformer sau đó được upsample và kết hợp với các đặc trưng đa tỷ lệ được trích xuất từ đường dẫn encoder thông qua các kết nối tắt (skip-connections). Các kết nối tắt này giúp khôi phục thông tin không gian chi tiết và cải thiện độ chính xác của bản đồ phân đoạn.

Ưu điểm

- **Kết hợp ưu điểm của CNN và Transformer**: TransUNet tận dụng khả năng trích xuất đặc trưng cục bộ của CNN và khả năng mô hình hóa phụ thuộc toàn cục của Transformer.
- **Bảo toàn thông tin chi tiết**: Các Residual Connection từ encoder giúp decoder có được thông tin không gian chi tiết cần thiết cho việc định vị chính xác các vùng phân đoạn.

Các biến thể và công trình liên quan Ý tưởng chèn Transformer vào kiến trúc U-shaped đã được nhiều nghiên cứu tiếp tục phát triển như:

- Yao et al. kết hợp Transformer với Claw U-Net.
- Xu et al. đề xuất LeViT-UNet.
- Sha et al. thiết kế Transformer-Unet.
- Li et al. thêm thành phần Attention Upsample vào decoder.
- Gao et al. giới thiệu UTNet áp dụng các module tự chú ý ở cả encoder và decoder.

3.2.2 Mô hình Transformer thuần túy: SwinUNet

Swin UNet [3] là một mô hình lấy cảm hứng từ UNet, áp dụng Transformer thuần túy vào các khối Swin Transformer. Kiến trúc Swin UNet bao gồm một Encoder-Bottleneck-Decoder, giữ nguyên hình dạng của UNet trong khi tận dụng khả năng học biểu diễn mạnh mẽ của Transformer.

Kiến trúc

- **Encoder**: Ảnh đầu vào được chia thành các ô không chồng lấp có kích thước 4×4. Mỗi ô có chiều đặc trưng $4\times4\times3 = 48$. Một lớp nhúng tuyến tính (Linear Embedding Layer) được sử dụng để chiếu chiều đặc trưng này thành một kích thước tùy ý (biểu diễn là C).
  - **Patch Merging**: Giảm kích thước không gian xuống 2 lần và tăng số chiều đặc trưng.
  - **Swin Transformer Block x2 (hai khối liên tiếp)**: Học các biểu diễn đặc trưng bằng cách sử dụng Window-based Multi-Head Self-Attention (W-MSA) (khối đầu) và Shifted Window Multi-Head Self-Attention (SW-MSA) (khối sau).
- **Bottleneck**: Gồm hai khối Swin Transformer liên tiếp, giữ nguyên số chiều đặc trưng và độ phân giải.
- **Decoder**: Ngược với Encoder, sử dụng Patch Expanding (đối của Patch Merging) để tăng kích thước không gian từng bước, phục hồi độ phân giải ban đầu của ảnh. Lớp cuối cùng là Linear Projection để tạo ra bản đồ phân đoạn.
- **Skip Connection**: Kết hợp đặc trưng từ Encoder với Decoder theo từng tỉ lệ (1/4, 1/8, 1/16) để giữ lại thông tin chi tiết.

Ưu điểm

- **Học biểu diễn mạnh mẽ hơn CNN** : Swin UNet tận dụng Swin Transformer, giúp mô hình học ngữ cảnh dài hạn và các mối quan hệ không gian hiệu quả hơn so với mạng CNN truyền thống.
- **Phân cấp đặc trưng**: Kiến trúc Patch Merging và Patch Expanding giúp tạo ra các đặc trưng phân cấp, tương tự như trong CNN.

3.3 Phát hiện đối tượng trong hình ảnh y tế

Phát hiện đối tượng trong hình ảnh y tế là việc xác định vị trí và phân loại các đối tượng quan tâm (ví dụ: polyp, khối u, tổn thương) trong hình ảnh. Tương tự như

phân đoạn, các mô hình Transformer xử lý tác vụ phát hiện đối tượng thường được kết hợp với các khối CNN, trong đó CNN được sử dụng để trích xuất đặc trưng từ hình ảnh y tế, trong khi kiến trúc Transformer được dùng để tăng cường các đặc trưng đã trích xuất nhằm phục vụ cho các tác vụ phát hiện sau đó.

Nổi bật trong lĩnh vực này, ta có một mô hình dựa trên kiến trúc DETR (Detection Transformer), có tên là COTR (Convolutional Transformer for object detection) [6] bao gồm Backbone, N x Encoder Transformer có tích hợp tích chập (convolution in transformer), N x Decoder Transformer và một Predictor (N = 6 trong bài báo gốc) trong đó:

- **Backbone**: Là một feature extractor sử dụng ResNet18. Cho ra f.
- **Encoder**: Mỗi Encoder gồm một Transformer Encoder làm phẳng cấu trúc không gian đặc trưng f thành s (s < f), theo sau là một lớp tích chập để tái dựng lại cấu trúc không gian s thành f.
  - **Transformer Encoder**: kiến trúc tiêu chuẩn gồm MSA (Multi-head Self-Attention), Feed Forward Network (FFN) và các nhúng vị trí (positional embeddings) được thêm vào mỗi tầng attention.
  - **Lớp tích chập**: gồm tích chập 3×3, batch normalization và hàm kích hoạt ReLU.
- **Decoder**: Mỗi Decoder sử dụng kiến trúc tiêu chuẩn theo Decoder của Transformer, ngoại trừ việc các đối tượng truy vấn được giải mã song song. Các Decoder nhận đầu vào là N đối tượng truy vấn cùng với các nhúng vị trí, kết hợp với đầu ra từ encoder, và biến chúng thành các vector nhúng để đưa vào FFN ở Predictor.
- **Predictor**: Hay còn gọi là bộ dự đoán, là một FFN gồm 2 lớp Fully Connected:
  - Một lớp hồi quy hộp (box-regression) để dự đoán vị trí đối tượng (x,y,w,h).
  - Một lớp phân loại hộp (box-classification) để dự đoán nhãn.

  Vì thế, từ N đối tượng truy vấn tạo thành N dự đoán cuối cùng, bao gồm cả các hộp chứa và các hộp không có đối tượng.

4 Phương pháp đại diện chính

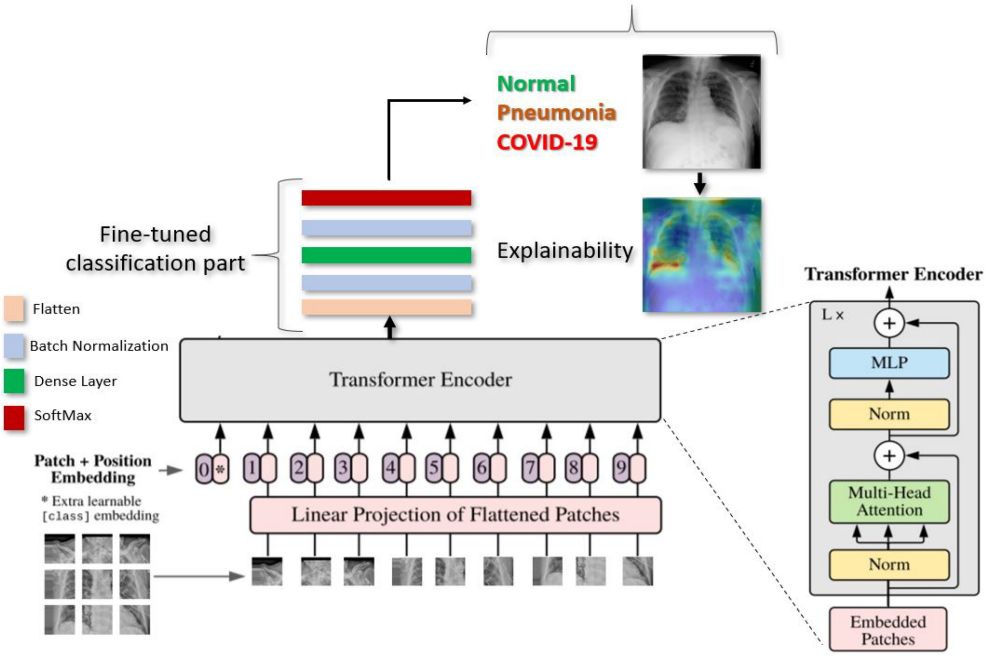
Đại dịch COVID-19 đã gây ra những ảnh hưởng nghiêm trọng đến sức khỏe toàn cầu, với số lượng ca tử vong lớn và tình trạng tái thiết lập phong tỏa ở nhiều quốc gia.

Với nghiên cứu "Explainable Vision Transformers and Radiomics for COVID-19 Detection in Chest X-rays do Mohamed Chetoui và Moulay A. Akhloufi [7], các tác giả đã điều tra việc sử dụng Vision Transformer (ViT) để phát hiện COVID-19 trong hình ảnh X-quang ngực (CXR).

Các phương pháp Mạng nơ-ron tích chập sâu (Deep Convolutional Neural Network - CNN) hiện có thường không thể nắm bắt được ngữ cảnh toàn cục do thiên hướng quy nạp cố hữu của chúng đối với hình ảnh. Để giải quyết vấn đề này, Chetoui và Akhloufi đã tinh chỉnh (fine-tune) nhiều mô hình ViT cho bài toán phân loại đa lớp gồm: COVID-19, Viêm phổi (Pneumonia), và các trường hợp Bình thường (Normal).

Nghiên cứu này đã sử dụng một tập dữ liệu lớn, bao gồm 7598 hình ảnh CXR dương tính với COVID-19, 8552 hình ảnh CXR của bệnh nhân khỏe mạnh, và 5674 hình ảnh CXR bị Viêm phổi, tổng cộng là 21.824 hình ảnh CXR.

Ở phần huấn luyện, các mô hình ViT được đề xuất, cụ thể là ViT-B16, ViT-B32 và ViT-L32, đã được huấn luyện theo cách có giám sát. Kết quả là mô hình ViT-B32 đã vượt trội hơn hẳn các mô hình CNN so sánh và các kiến trúc dựa trên Transformer khác trong việc phát hiện COVID-19 trên hình ảnh CXR. Điều này cho thấy khả năng tổng quát hóa tốt hơn của mô hình trên dữ liệu chưa từng thấy. Một đóng góp quan trọng khác của nghiên cứu là khả năng giải thích (explainability) của mô hình: bản đồ chú ý (attention map) cho thấy mô hình có khả năng xác định hiệu quả các dấu hiệu của COVID-19 và các dấu hiệu viêm phổi khác trên hình ảnh CXR (Chest XRay).



Hình 4: Mô hình mô tả phương pháp đại diện chính

4.1 Thu thập và tiền xử lý dữ liệu

4.1.1 Bộ Dữ Liệu `COVID19_Pneumonia_Normal_Chest_Xray_PA_Dataset`

- **Gồm: 6939 ảnh XRay ngực**
- **Bộ dữ liệu được chia thành ba tập dữ liệu**:

1. **Tập covid (viêm phổi do covid-19)**: 2313 ảnh.
2. **Tập normal (bình thường)**: 2313 ảnh.
3. **Tập pneumonia (viêm phổi khác)**: 2313 ảnh.

4.1.2 Quá Trình Thu Thập và Tiền Xử Lý Dữ Liệu

Quá trình tiền xử lý dữ liệu được thực hiện theo các bước sau:

- **Tải dữ liệu**: Từng tập dữ liệu (covid, normal, pneumonia) được tải vào.
- **Chia dữ liệu**: Lặp qua từng tập dữ liệu để chia ra thành 3 tập huấn luyện, tập xác thực và tập kiểm tra với tỉ lệ lần lượt là 0.75, 0.15, 0.15.
- **Tăng Cường Dữ Liệu (Data Augmentation):**
  - **Tập huấn luyện** Thay đổi kích thước và áp dụng các phép tăng cường để tăng tính đa dạng và khả năng khái quát hóa của mô hình.
    - Thay đổi kích thước ảnh: Kích thước ảnh được điều chỉnh thành kích thước cố định **224x224**.
    - Phép tăng cường 1: Lật ngẫu nhiên theo chiều ngang.
    - Phép tăng cường 2: Xoay ảnh ngẫu nhiên trong khoảng từ -30 đến 30 độ.
    - Chuyển sang dạng Tensor của PyTorch.
    - Chuẩn hóa dữ liệu: chuẩn hóa các giá trị pixel về khoảng giá trị có trung bình [0.485, 0.456, 0.406] và độ lệch chuẩn [0.229, 0.224, 0.225] tương ứng với các kênh RGB, theo chuẩn ImageNet.
  - **Tập xác thực và kiểm tra**: tương tự với tập huấn luyện nhưng không áp dụng hai phép tăng cường dữ liệu.

4.2 Vision Transformer

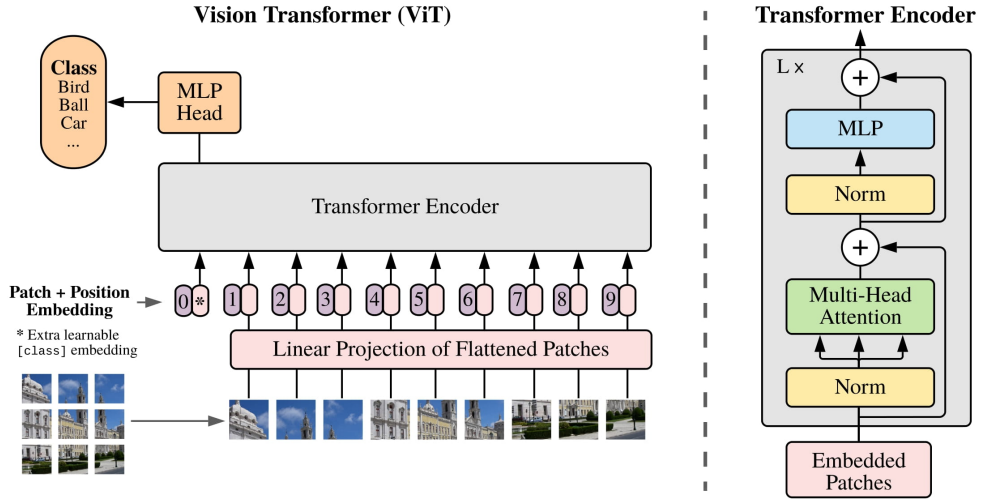
Trong nghiên cứu này, Vision Transformer (ViT) [2] được sử dụng như một công cụ trích xuất đặc trưng để phân loại ảnh Xray thành ba nhóm: bệnh nhân khỏe mạnh, viêm phổi do virus Corona chủng mới và viêm phổi do các nguyên nhân khác.

ViT là một kiến trúc mạng nơ-ron sâu được đề xuất bởi nhóm nghiên cứu của Google nhằm khắc phục những hạn chế của Transformer cổ điển khi áp dụng cho lĩnh vực thị giác máy tính, đặc biệt là vấn đề chi phí bộ nhớ lớn. ViT đã chứng minh hiệu quả vượt trội trên tập dữ liệu ImageNet, vượt qua nhiều mô hình phân loại ảnh tiên tiến đương thời. Hình ảnh tổng quát của phương pháp này được trình bày trong Hình 4.

Cốt lõi của ViT là mở rộng kiến trúc Transformer từ xử lý chuỗi trong ngôn ngữ tự nhiên sang xử lý ảnh bằng cách coi ảnh đầu vào như một chuỗi các đơn vị nhỏ (gọi là *patch*). Cụ thể, ảnh được chia thành N bản vá (patch) có kích thước cố định. Mỗi patch được ánh xạ qua một lớp tuyến tính để tạo thành một vector nhúng đặc trưng. Các vector này được kết hợp với một token lớp (*class token*) có thể huấn luyện và sau đó đưa vào một lớp tuyến tính để thực hiện phân loại.

Do Transformer không có khả năng nhận biết cấu trúc không gian vốn có trong hình ảnh, nên ViT bổ sung thông tin vị trí thông qua các vector nhúng vị trí (*positional embeddings*). Những vector này được cộng vào các vector đặc trưng đầu vào trước khi đưa vào các khối Transformer. Hình 5 mô tả chi tiết sơ đồ của Vision Transformer.

Khái niệm quan trọng nhất của ViT là Cơ chế tự chú ý (self-attention), cụ thể hơn là Cơ chế Tự chú ý Đa đầu (Multi-Head Self-Attention - MSA). Cơ chế chú ý này dựa trên một bộ nhớ liên kết có thể huấn luyện giữa các cặp vector truy vấn (query)



Hình 5: Mô hình chung về Vision Transformer. Ta chia một hình ảnh thành các bản vá (patch) có kích thước cố định, nhúng tuyến tính từng bản vá, thêm nhúng vị trí và đưa chuỗi vectơ kết quả vào bộ mã hóa Transformer chuẩn. Để thực hiện phân loại, ta sử dụng phương pháp tiếp cận chuẩn là thêm một "mã thông báo phân loại" có thể học được vào chuỗi. Minh họa về bộ mã hóa Transformer được lấy cảm hứng từ Vaswani và cộng sự (2017).

và vector khóa (key). Một vector truy vấn $q \in R^d$ được so sánh với một tập hợp các vector khóa $K \in R^{k \times d}$ bằng cách tính tích trong (inner product), sau đó được chuẩn hóa, đưa qua hàm Softmax để tạo ra trọng số.

Với một chuỗi đầu vào X, cơ chế tự chú ý tính toán ba ma trận: truy vấn (Q), khóa (K) và giá trị (V) bằng cách nhân X với các ma trận trọng số đã được học W_Q, W_K và W_V:

$$Q = XW_Q, \quad K = XW_K, \quad V = XW_V$$

Kết quả đầu ra là tổng có trọng số của một tập hợp các vector giá trị (value vectors) $V \in R^{k \times d}$ cho một chuỗi gồm N vector truy vấn $Q \in R^{N \times d}$, tạo thành một ma trận đầu ra có kích thước $N \times d$:

$$\text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d}}\right)V$$

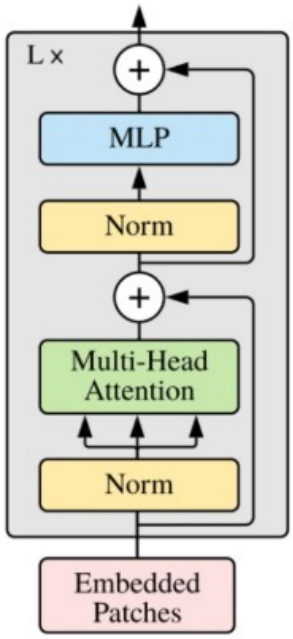
Cuối cùng, MSA được định nghĩa bằng cách xem xét h đầu chú ý (attention heads), với h hàm tự chú ý (self-attention) được áp dụng cho đầu vào. Mỗi đầu cung cấp một chuỗi có kích thước $N \times d$ và các chuỗi h này được sắp xếp lại thành một chuỗi có kích thước $N \times dh$, sau đó được chiếu lại thành $N \times D$ thông qua một lớp tuyến tính. Đối với chú ý đa đầu, chúng ta áp dụng cơ chế tự chú ý nhiều lần song song:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)W_O$$

Trong đó, mỗi *head* được tính như sau:

$$\text{head}_i = \text{Attention}(QW_Q^i, KW_K^i, VW_V^i)$$

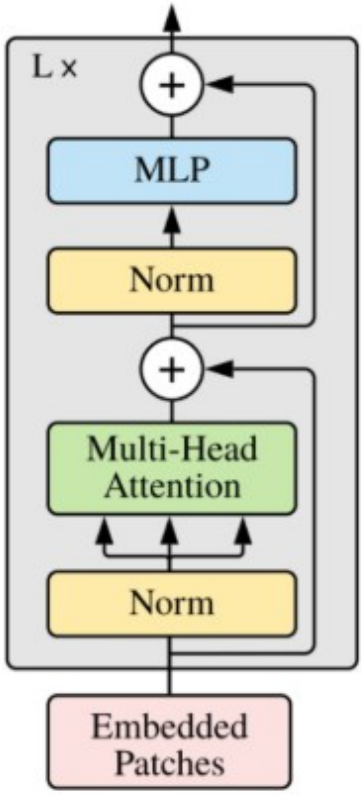
Kiến trúc của ViT bao gồm nhiều khối mã hóa (*Encoder block*) xếp chồng có thể huấn luyện. Khối đầu tiên nhận đầu vào là các bản vá ảnh đã nhúng và xuất ra đầu ra cho khối Bộ mã hóa tiếp theo, quá trình này tiếp tục cho đến khi đầu ra của khối cuối cùng được đưa vào một lớp MLP để phân loại ảnh. Mỗi khối Bộ mã hóa bao gồm hai lớp Chuẩn hóa theo lô (Batch Normalization) và hai lớp dư (residual layers), làm việc cùng với MSA và MLP như mô tả trong Hình 6. Tín hiệu đầu ra từ khối mã hóa cuối cùng sẽ được đưa qua lớp phân loại để xác định nhãn ảnh. Ngoài ra, kiến trúc này cho phép thay thế các khối mã hóa bằng các biến thể hiệu quả hơn nhằm giảm chi phí tính toán và bộ nhớ.



Hình 6: Mô hình mô tả khối mã hóa

4.3 Thực nghiệm và Kết quả

- **Link Dataset:** https://www.kaggle.com/datasets/amanullahasraf/covid19-pneumonia-normal-chest-xray-pa-dataset
- **Link Notebook trên Kaggle:** https://www.kaggle.com/code/senn1l/classification-with-vision-transformer
  - COVID: 2313 ảnh.
  - Normal: 2313 ảnh.
  - Pneumonia: 2313 ảnh.
- **Nhận xét dataset:** Phân bố ảnh giữa các lớp đồng đều.
- **Trình bày Model:**
  - Sử dụng mô hình `vit_base_patch16_224` – một phiên bản của Vision Transformer với patch size 16×16 và ảnh đầu vào 224×224.
  - Mô hình được khởi tạo với trọng số pretrained (huấn luyện trước) trên ImageNet-1k* để tăng khả năng khái quát và rút ngắn thời gian huấn luyện.
  - Về dữ liệu ImageNet*: đây là tập dữ liệu huấn luyện phổ biến với hàng triệu ảnh quen thuộc trong đời sống hằng ngày như động vật (chó, mèo, bò,...) hay đồ vật (bút, viết, ghế,...) và các vật thể nhân tạo khác. Thường được sử dụng để huấn luyện trước các mô hình thị giác máy tính. Có 2 phiên bản: ImageNet-1k: xấp xỉ 1.2 triệu ảnh - 1000 lớp và ImageNet-21k: xấp xỉ 14 triệu ảnh - 21.000 lớp.



Hình 7: Kiến trúc encoder của Transformer cũng như mô hình dùng để huấn luyện

Trong đó:

- Kích thước ảnh đầu vào: 224x224 pixels.
- N = 12 (số lớp, là L trên hình 7).
- Kích thước patch: 16x16.
- Các thông số còn lại được mô tả rõ trong [2], page 5, table 1.

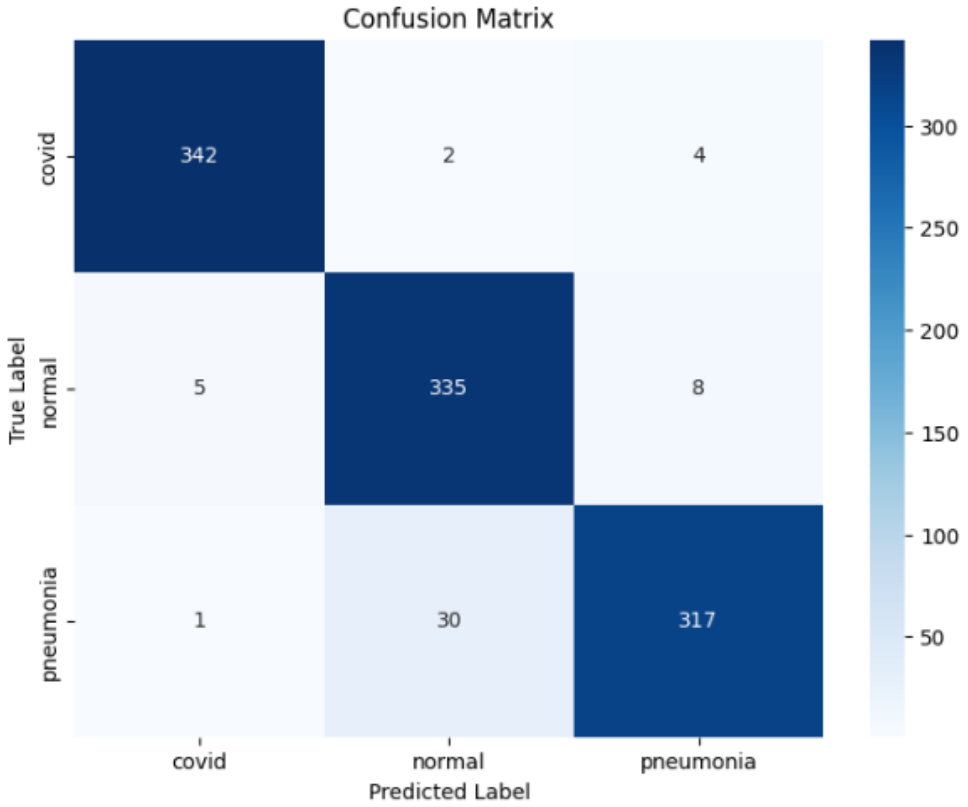
– Mô hình được huấn luyện trên kaggle (có sử dụng GPU P100) với các siêu tham số như sau:

- Số epoch = 30.
- Learning rate = 0.0001.
- Hàm mất mát là Cross Entropy.
- Trình tối ưu (optimizer) là Adamw với `weight_decay` = 1e-5.
- Sử dụng scheduler để thực hiện giảm learning rate qua từng epoch.
- Thư viện chính: Pytorch.

Sau khi huấn luyện mô hình, chúng tôi tiến hành đánh giá hiệu suất trên tập kiểm tra gồm 1.044 ảnh (đã được split trong phần tiền xử lý), được chia đều cho ba lớp: COVID, Normal, và Pneumonia.

Hình 8: Biểu đồ quá trình huấn luyện

- **Nhận xét biểu đồ quá trình huấn luyện:** Biểu đồ quá trình huấn luyện cho thấy sự thay đổi của độ mất mát (loss) và độ chính xác (accuracy) qua 16 epoch, được ghi nhận cho cả tập huấn luyện và tập xác thực.
  - Loss trên tập huấn luyện giảm đều từ khoảng 0.62 xuống còn xấp xỉ 0.05, cho thấy mô hình học được tốt từ dữ liệu.
  - Loss trên tập xác thực cũng giảm nhanh trong các epoch đầu và ổn định dần từ epoch thứ 6 trở đi. Tuy vậy, có một vài dao động (đặc biệt ở epoch 5).
  - Accuracy trên tập huấn luyện tăng liên tục, đạt xấp xỉ 98% ở epoch cuối cùng.
  - Accuracy trên tập xác thực cũng tăng nhanh, đạt đỉnh gần 95% và duy trì ổn định sau epoch thứ 6. Mặc dù có sự sụt giảm tạm thời tại epoch thứ 5, nhưng sau đó mô hình đã phục hồi và giữ vững hiệu suất cao trên tập xác thực.



Hình 9: Confusion Matrix

- **Nhận xét Confusion Matrix:** Confusion matrix cho thấy mô hình có khả năng phân loại tốt trên cả ba lớp. Cụ thể:
  - **Lớp COVID:** 342/348 ảnh được phân loại đúng, chỉ có 6 ảnh bị nhầm sang lớp khác.
  - **Lớp Normal:** 335 ảnh đúng, với một số ảnh bị nhầm sang Pneumonia (8 ảnh) và COVID (5 ảnh).
  - **Lớp Pneumonia:** Có 317 ảnh đúng, tuy nhiên bị nhầm khá nhiều sang Normal (30 ảnh), cho thấy mô hình gặp khó khăn hơn khi phân biệt Pneumonia với ảnh ngực bình thường.

	precision	recall	f1-score	support
covid	0.98	0.98	0.98	348
normal	0.91	0.96	0.94	348
pneumonia	0.96	0.91	0.94	348
accuracy			0.95	1044
macro avg	0.95	0.95	0.95	1044
weighted avg	0.95	0.95	0.95	1044

Hình 10: Precision, Recall và F1-score

- **Nhận xét Precision, Recall và F1-score:**
  - Lớp COVID đạt hiệu suất gần như tuyệt đối, cho thấy mô hình rất nhạy và chính xác trong việc phát hiện ảnh X-ray có dấu hiệu COVID.
  - Lớp Normal có recall cao (0.96) nhưng precision thấp hơn (0.91), nghĩa là mô hình có xu hướng gán nhãn "Normal"cho một số ảnh không phải là bình thường (false positives).
  - Lớp Pneumonia có precision cao (0.96), nhưng recall thấp hơn (0.91), cho thấy mô hình chưa phát hiện hết các ca viêm phổi, dễ nhầm lẫn với Normal.
  - Độ chính xác tổng thể (accuracy) đạt 95%, cho thấy mô hình hoạt động hiệu quả trong phân loại ba lớp ảnh X-ray.
- **Kết luận:**
  - Từ quá trình huấn luyện và đánh giá, có thể nhận thấy rằng mô hình đạt được hiệu suất tốt trong bài toán phân loại ảnh X-ray ngực thành ba lớp: COVID, Normal, và Pneumonia. Mô hình hội tụ nhanh, đạt độ chính xác cao trên cả tập huấn luyện và tập xác thực, đồng thời không có dấu hiệu overfitting cụ thể.
  - Kết quả confusion matrix cho thấy mô hình phân loại rất chính xác các ca COVID, trong khi vẫn duy trì mức độ chính xác tốt đối với hai lớp còn lại. Chỉ số precision, recall và F1-score ở cả ba lớp đều đạt từ 0.91 trở lên, với độ chính xác tổng thể lên tới 95%, cho thấy mô hình có khả năng phân loại tổng quát cao và ổn định.
  - Mặc dù mô hình có xu hướng nhầm lẫn nhẹ giữa lớp Pneumonia và Normal, đây là điều dễ hiểu do đặc điểm hình ảnh có thể tương đồng ở một số trường hợp. Tuy nhiên, với kết quả hiện tại, mô hình đã cho thấy hiệu quả và tiềm năng ứng dụng trong hỗ trợ chẩn đoán hình ảnh X-ray ngực, đặc biệt là trong bối cảnh cần phát hiện nhanh các ca nhiễm COVID-19.

Tài liệu

[1] K. He, C. Gan, Z. Li **andothers**, "Transformers in Medical Image Analysis: A Review," *IEEE Transactions on Medical Imaging*, **journvol** 40, **number** 9, **pages** 2572–2583, 2021. DOI: 10.1109/TMI.2021.3091329.

[2] A. Dosovitskiy, L. Beyer, A. Kolesnikov **andothers**, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," **in***Proc. International Conference on Learning Representations (ICLR)* 2021.

[3] H. Cao, Y. Wang, J. Chen **andothers**, "Swin-Unet: Unet-like Pure Transformer for Medical Image Segmentation," *arXiv preprint arXiv:2105.05537*, 2021. **url**: https://arxiv.org/abs/2105.05537.

[4] J. Chen, Y. Lu, Q. Yu **andothers**, "TransUNet: Transformers Make Strong Encoders for Medical Image Segmentation," *arXiv preprint arXiv:2102.04306*, 2021. **url**: https://arxiv.org/abs/2102.04306.

[5] D. Bahdanau, K. Cho **and** Y. Bengio, *Neural Machine Translation by Jointly Learning to Align and Translate*, 2016. arXiv: 1409.0473 [cs.CL]. **url**: https://arxiv.org/abs/1409.0473.

[6] Z. Shen, C. Lin **and** S. Zheng, "COTR: Convolution in Transformer Network for End-to-End Polyp Detection," **in***Proc. International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)* 2021. **url**: https://arxiv.org/abs/2103.14117.

[7] M. Chetoui **and** M. A. Akhloufi, "Explainable Vision Transformers and Radiomics for COVID-19 Detection in Chest X-rays," *Journal of Clinical Medicine*, **journvol** 11, **number** 11, 2022, ISSN: 2077-0383. DOI: 10.3390/jcm11113013. **url**: https://www.mdpi.com/2077-0383/11/11/3013.