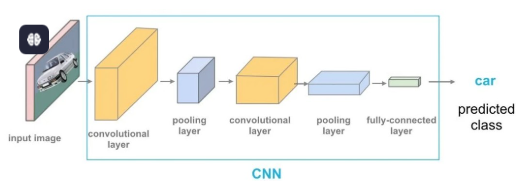
1. Giới thiệu về Vision Transformers ( ViT )
   1. **Vision Transformer (ViT) trong nhận diện hình ảnh**

Trong khi mô hình Transformer đã trở thành tiêu chuẩn cho các tác vụ liên quan đến Xử lý ngôn ngữ tự nhiên (NLP), thì các trường hợp sử dụng nó trong thị giác máy tính (Computer Vision) vẫn chỉ là một con số ít ỏi. Trong thị giác máy tính, cơ chế Attention sẽ hoặc được sử dụng kết hợp với các mạng nơ-ron tích chập(CNN), hoặc sử dụng để thay thế các khía cạnh của CNN.



Tuy nhiên, sự phụ thuộc này vào CNN là không bắt buộc, và một mô hình Transformer thuần túy được áp dụng trực tiếp vào chuỗi các mảng hình ảnh( image patch – là một hình chữ nhật chứa nhiều điểm ảnh gần nhau, đủ lớn để có thể chứa được các bộ phận có thể mô tả được vật thể trong ảnh) sẽ có thể vận hành đặc biệt tốt trong các nhiệm vụ phân loại ảnh.

* 1. **Vission Transformer (ViT) là gì ?**

Vision Transformers (ViT) là một kiến trúc mạng học sâu tiên tiến dành cho các bài toán thị giác máy tính (Computer Vision). ViT sử dụng cơ chế **transformer**, vốn rất phổ biến trong xử lý ngôn ngữ tự nhiên (NLP), để thay thế hoàn toàn các lớp tích chập (CNN). Ý tưởng cốt lõi của ViT là chuyển hình ảnh thành các patch (mảnh nhỏ) và xử lý chúng giống như các token trong bài toán NLP.

ViT được giới thiệu bởi Google Research trong bài báo nổi tiếng **“An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale”** (2020). ViT đã đạt được độ chính xác cao trên nhiều bộ dữ liệu như ImageNet, đồng thời thể hiện hiệu suất vượt trội trên các mô hình CNN truyền thống như ResNet khi được huấn luyện trên dữ liệu quy mô lớn.

1. Kiến trúc của Vision Transformers
2. **Chia hình ảnh thành các patch**

Hình ảnh đầu vào được chia thành các mảnh có kích thước cố định (ví dụ: 16×16 pixel). Mỗi mảnh sau đó được làm phẳng thành một vector và đưa vào không gian có chiều cao hơn.

Hình ảnh đầu vào có kích thước H × W × C, trong đó:

* + H: Chiều cao của hình ảnh.
  + W: Chiều rộng của hình ảnh.
  + C: Số kênh màu (thường là 3 đối với RGB).

ViT chia hình ảnh thành các mảnh (patches) không chồng lấn, mỗi patch có kích thước P ×P

Số lượng patch N được tính bởi:

Ví dụ: Với hình ảnh kích thước 224 × 224 và P = 16, ta có N = 196 patches.

1. **Tạo Embedding cho các patch**

Mỗi patch được **flatten** (trải phẳng) thành một vector độ dài P2 \* C.

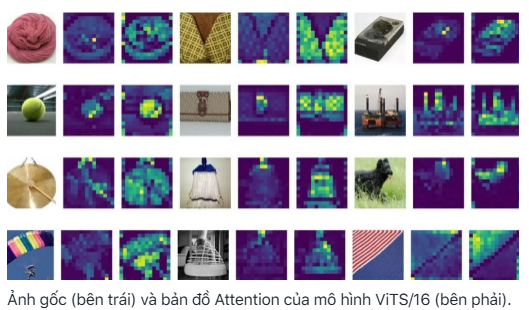
Các vector này được ánh xạ vào không gian embedding nhờ một lớp **linear projection**, tạo ra các vector đặc trưng z1,z2,…zn .

1. **Transformer Encoder**

**Lớp Multi-Head Self Attention (MSP) :** Lớp này nối tất cả các kết quả đầu ra của cơ chế Attention một cách tuyết tính theo đúng kích thước. Nhiều Attention head sẽ giúp huấn luyện những yếu tố phụ thuộc cục bộ và toàn bộ trong hình ảnh.

**Lớp Multi-Layer Perceptrons(MLP) :** Lớp này chứa một hàm Gaussian Error Linear Unit hai lớp.

**Lớp thường :** Lớp này được thêm vào trước mỗi khối, vì nó không bao gồm bất ký yếu tố phụ thuộc mới nào giữa các hình ảnh huấn luyện. Điều này giúp cải thiện thời gian đào tạo và hiệu suất tổng thể.



1. **Classification Head**

ViT thêm một token đặc biệt (classification token [CLS] vào các patch đầu vào.

Sau khi qua các lớp transformer encoder, token [CLS] chứa thông tin tổng hợp của toàn bộ hình ảnh và được đưa qua một **fully connected layer** để dự đoán nhãn.

1. Cơ chế hoạt động của Vision Transformers
2. **Multi-Head Self-Attention (MHSA)**

**Cơ chế MHSA** là trung tâm của ViT, cho phép mô hình học được mối quan hệ giữa các patch (các phần của hình ảnh) thông qua việc tính toán **attention**. MHSA giúp mô hình chú ý tới các vùng quan trọng trong ảnh, cả ở gần và xa, để hiểu rõ hơn cấu trúc tổng thể.Cơ chế attention được tính như sau:

Trong đó:

* + Q (Query): Ma trận truy vấn, biểu diễn vector đặc trưng của các patch cần "chú ý".
  + K (Key): Ma trận khóa, biểu diễn vector đặc trưng của các patch được dùng để tính mức độ liên quan với truy vấn.
  + V (Value): Ma trận giá trị, chứa thông tin thực tế để trích xuất từ các patch dựa trên mức độ chú ý.
  + dk​: Số chiều của vector key.

1. **Multi-Head Attention**

Thay vì chỉ tính một lớp attention, **Multi-Head Self-Attention** (MHSA) tính nhiều attention song song (gọi là "heads"). Mỗi "head" học các mối quan hệ khác nhau giữa các patch, giúp mô hình nắm bắt được thông tin đa chiều.

**Các bước của MHSA**:

* Chia các vector Q, K, V thành h phần (với hhh là số "head").
* Tính attention cho từng "head" độc lập.
* Gộp kết quả từ các "head" bằng phép nối (concatenate).
* Nhân với ma trận trọng số WO để tổng hợp thông tin.

1. **Thêm thông tin vị trí (Positional Encoding)**

Vì transformer không có khái niệm về không gian (spatial locality), ViT thêm thông tin vị trí vào các vector embedding thông qua **Positional Encoding**.

Công thức tính vector đầu vào:

1. Sự khác nhau giữa mạng nơ-ron tích chập( Convolutional Neural Network – CNN) và Vision Transformer (ViT)
2. Ưu điểm của Vision Transformers
3. **Hiệu suất vượt trội trên dữ liệu lớn:**

Khi được huấn luyện trên tập dữ liệu lớn (như JFT-300M), ViT đạt hiệu suất vượt trội so với các mô hình CNN tiên tiến như ResNet hoặc EfficientNet.

Cơ chế tự chú ý (self-attention) của ViT cho phép học tốt hơn các mối quan hệ dài hạn trong dữ liệu.

1. **Khả năng tổng quát tốt:**

Không giống CNN, ViT không phụ thuộc vào giả định về "tính cục bộ" (locality). Điều này làm cho ViT dễ dàng tổng quát hóa với dữ liệu khác biệt.

ViT có thể học được các mối quan hệ phức tạp giữa các phần của hình ảnh, ngay cả khi chúng không liên quan gần nhau trong không gian.

1. **Cơ chế attention mạnh mẽ:**

MHSA giúp ViT tập trung vào các vùng quan trọng trong ảnh, bất kể khoảng cách giữa các vùng này.

Điều này đặc biệt hữu ích trong các bài toán yêu cầu hiểu các mối liên hệ dài hạn (long-range dependencies).

1. Hạn chế của Vision Transformers
2. **Yêu cầu dữ liệu lớn:**

ViT cần bộ dữ liệu huấn luyện khổng lồ để đạt hiệu suất tốt, trong khi CNN hoạt động tốt ngay cả trên bộ dữ liệu nhỏ.

1. **Tính toán phức tạp:**

MHSA có độ phức tạp O(N2)O(N^2)O(N2), đòi hỏi tài nguyên tính toán lớn khi số lượng patch NNN tăng.

1. **Khả năng xử lý thông tin cục bộ:**

ViT kém hiệu quả hơn CNN trong việc học các đặc trưng cục bộ như cạnh, góc trong hình ảnh nhỏ.

1. Ứng dụng của Vision Transformers
2. Y tế

Chuẩn đoán hình ảnh y tế : ViT có thể hổ trợ phân tích các hình ảnh y tế như X-Quang, MRI và CT, giúp phát hiện các bất thường và chuẩn đoán có độ chính xác cao.

Giải phẩu : ViT có thể được sử dụng phân tích và hỗ trợ phát hiện bệnh ung thư và các bệnh lý khác.

1. Bán lẻ và thương mại điện tử

Nhận diện sản phẩm : ViT giúp nhận diện sản phẩm trong hình ảnh giúp quản lý tồn kho

và thanh toán của cửa hàng.

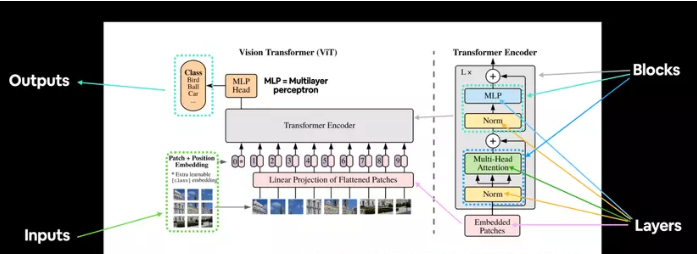
Đề xuất cá nhân hóa : Bằng cách phân tích nội dung hình ảnh, ViT có thể cung cấp đề xuất sản phẩm tương tự phù hợp với nhu cầu của mình

1. An ninh và giám sát

Nhận diện khuôn mặt : ViT cải thiện độ chính xác của hệ thống an ninh và giám sát trong việc nhận diện khuôn mặt

Phát hiện bất thường : ViT có thể phát hiện các hoạt động hoặc đối tượng bất thường trong camera giám sát, giúp tăng cường an ninh.

1. Kiến trúc của ViT



Patch + Position Embedding(đầu vào) : Chuyển hình ảnh đầu vào thành một chuỗi các patches và thêm số vị trí để chỉ định thứ tự cho các patches hợp lí.

Embedded patches : Các patches được chuyển thành Embedding, lợi ích của việc sử dụng Embedding thay vì chỉ các giá trị hình ảnh. Embedding là một biểu diễn có thể học được( thường ở dạng vector) của hình ảnh.

Norm – đây là viết tắt của “Layer Nỏmalization” hoặc “LayerNorm”, một kỹ thuật để chuẩn hóa( giảm overfitting) một mạng thần kinh, chúng ta có thể sử dụng LayerNorm thông qua lớp PyTorch.

Multi-Head Attention : Đây là một lớp Multi-Head Self-Attention hoặc viết tắt là “MSA”. Bạn có thể tạo một lớp MSA thông qua lớp PyTorch.

MLP (hoặc Multilayer Perceptron) - MLP thường đề cập đến bất kỳ tập hợp các lớp feedforward nào (hoặc trong trường hợp của PyTorch, một tập hợp các lớp với một phương thức). Trong bài báo ViT, các tác giả gọi MLP là "khối MLP" và nó chứa hai lớp torch.nn.Linear() với kích hoạt phi tuyến tính torch.nn.GELU () ở giữa chúng và lớp torch.nn.Dropout().

Transformer Encoder - Bộ mã hóa Transformer, là một tập hợp các lớp được liệt kê ở trên. Có hai kết nối bỏ qua bên trong bộ mã hóa Transformer (ký hiệu "+") có nghĩa là đầu vào của lớp được đưa trực tiếp đến các lớp ngay lập tức cũng như các lớp tiếp theo. Kiến trúc ViT tổng thể bao gồm một số bộ mã hóa Transformer xếp chồng lên nhau.

MLP Head - Đây là lớp đầu ra của kiến trúc, nó chuyển đổi các features đã học của đầu vào thành đầu ra lớp. Vì chúng ta đang nghiên cứu phân loại hình ảnh, bạn cũng có thể gọi đây là "đầu phân loại". Cấu trúc của MLP Head tương tự như khối MLP bên trên.