Vision Transformer

CS331.P21 - Thị giác máy tính nâng cao

Trương Huỳnh Thúy An Hoàng Đức Chung Nguyễn Hải Đăng

Khoa Khoa học Máy tính
Trường Đại học Công nghệ Thông tin, ĐHQG-HCM

Thứ Ba, ngày 27 tháng 5, 2025

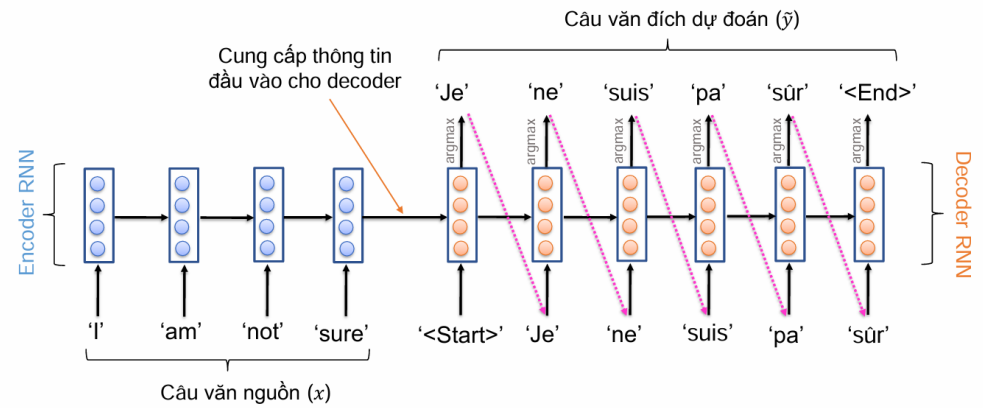
Overview

1. Attention
2. Transformer
3. Vision Transformer

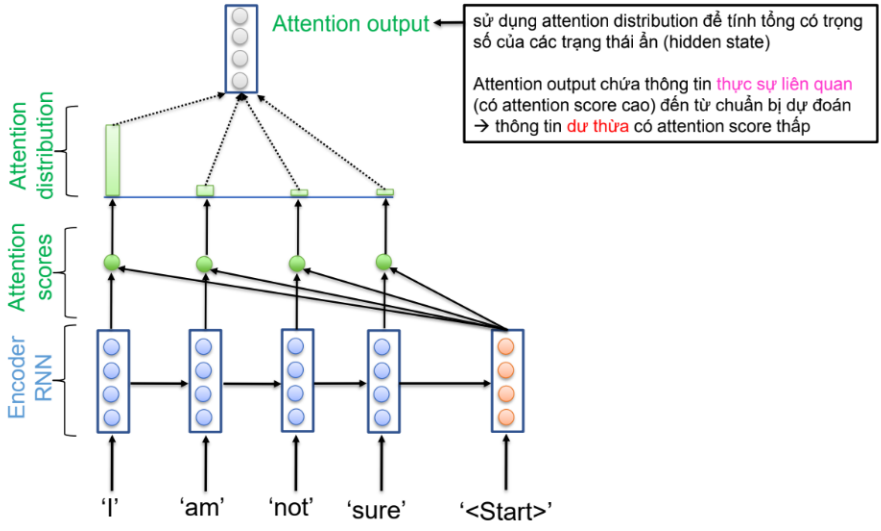
Overview

1. **Attention**
2. Transformer
3. Vision Transformer

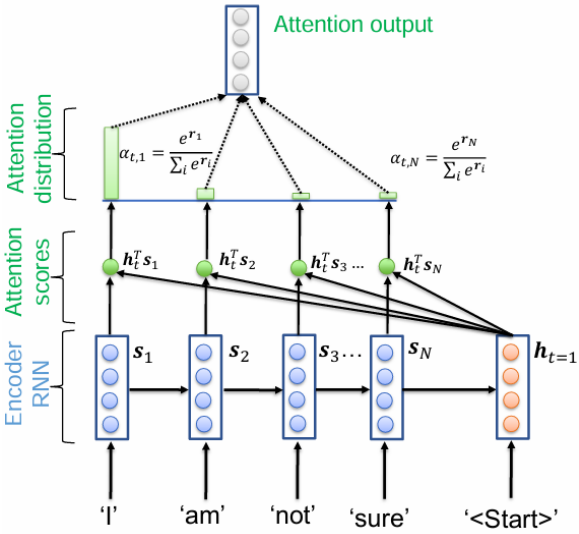
Mô hình Seq2Seq



Cơ chế Attention



Cơ chế Attention



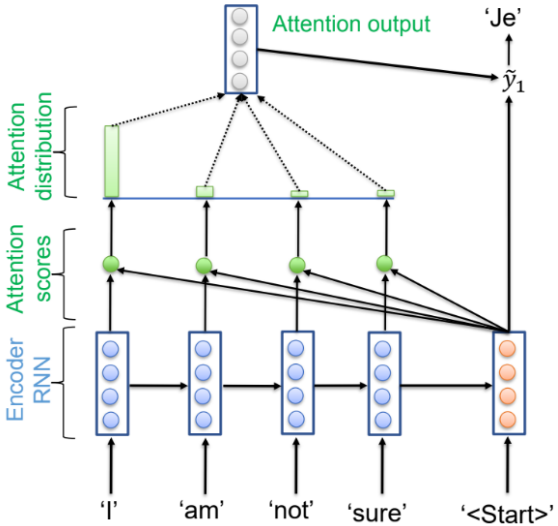
$$c_t = \sum_{i=1}^{N} \alpha_{t,i} s_i$$

$$\alpha_t = \text{softmax}(r_t)$$

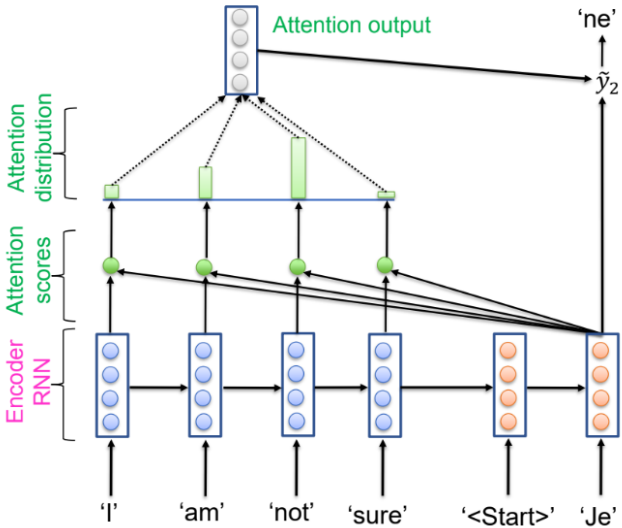
$$r_t = [h_t^T s_1, h_t^T s_2, \dots, h_t^T s_N]$$

Decoder RNN

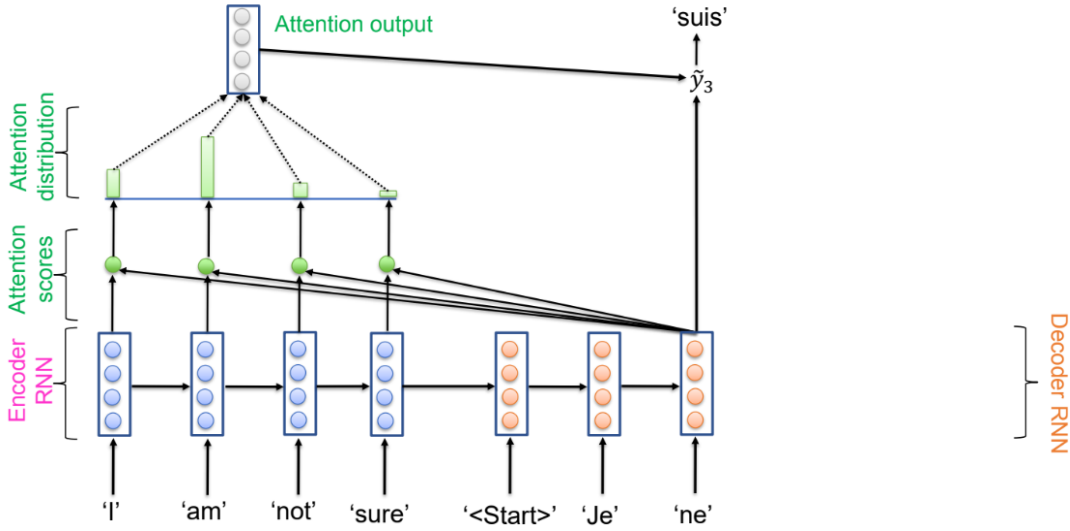
Cơ chế Attention



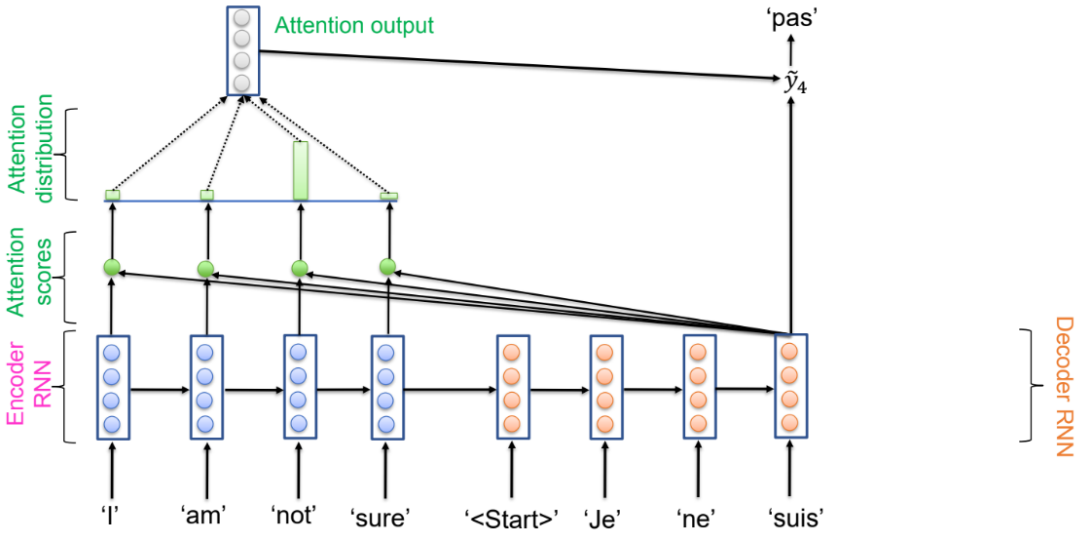
Cơ chế Attention



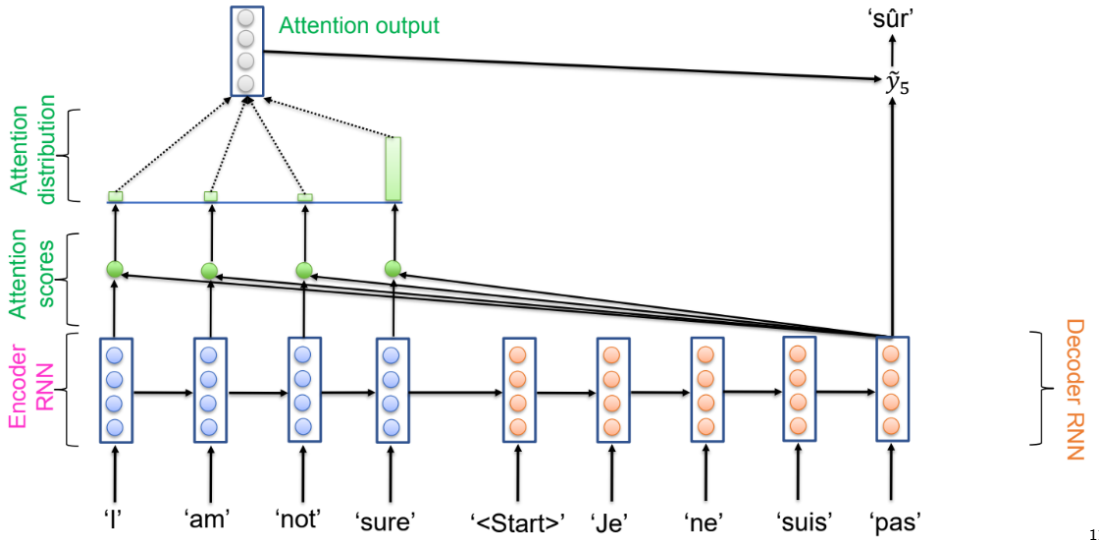
Cơ chế Attention



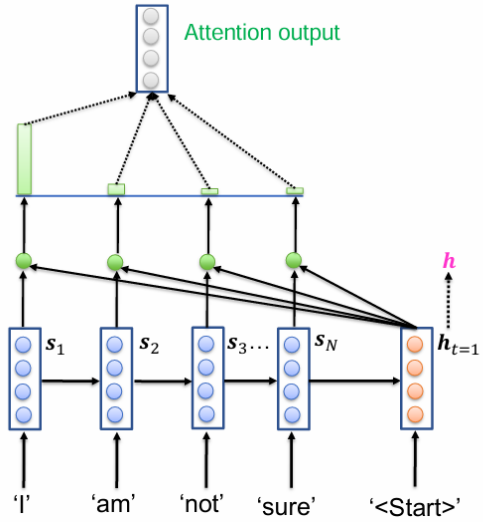
Cơ chế Attention



Cơ chế Attention



Cơ chế Attention



- Trạng thái ẩn ở encoder (values):

$$s_1, s_2, \ldots, s_N \in \mathbb{R}^{d_1}$$

- Vector truy vấn (query) $h \in \mathbb{R}^{d_2}$
- Attention thực hiện các bước sau:

(1) Tính attention scores: r

(2) Tính attention distribution α:

$$\alpha = \text{softmax}(r)$$

(3) Tính attention output c để tổng hợp thông tin:

$$c = \sum_{i=1}^{N} \alpha_i s_i$$

Cơ chế Attention

- Attention cho hiệu suất dịch cao hơn hẳn so với các phương pháp trước
  - Cho phép decoder nhìn lại toàn bộ câu văn nguồn
  - Cho phép decoder tập trung một số phần nhất định của câu văn nguồn
- Attention giải quyết vấn đề "điểm nghẽn" của trạng thái ẩn từ cuối
- Attention giải quyết vấn đề vanishing gradient
  - Tạo các "đường tắt" khi huấn luyện với back propagation
- Attention cho khả năng diễn đạt, giải thích kết quả
  - Với attention distribution, ta có thể quan sát decoder đang tập trung vào đâu

Attention và Seft-attention

- **Attention**: một từ truy vấn (query) ở decoder, truy vấn và tổng hợp thông tin từ các tập giá trị (values) của encoder
- **Selt-attention**: là cơ chế attention trên encoder-encoder (hoặc decoder-decoder), trong đó mỗi từ "chú ý" đến nhau trong cùng input (hoặc output)

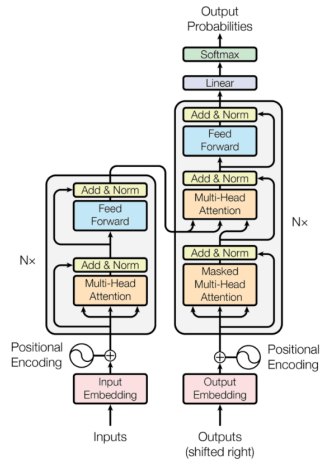
Overview

1. Attention
2. **Transformer**
3. Vision Transformer

Transformer sử dụng Self-Attention

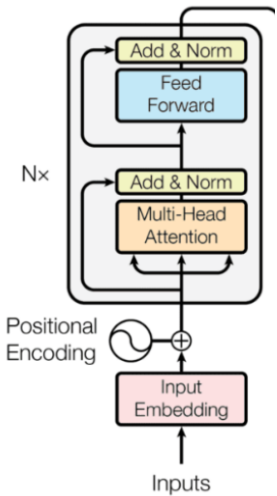
- Mô hình sequence-to-sequence, gồm 2 thành phần chính: **Encoder** và **Decoder**
- Loại bỏ tính tuần tự (recurrence)
- Self-Attention là tầng quan trọng trong mỗi lớp của Transformer. Với mỗi token, tính toán truy vấn (Q), khóa (K), giá trị (V).

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



Encoder

- Encoder gồm $N = 6$ lớp giống nhau (thường là 6, 12 hoặc 24).
- Mỗi lớp gồm:
  - Multi-head Self-Attention
  - Feed-Forward Network
- Mỗi tầng con được bao quanh bởi:
  - Residual Connection
  - Layer Normalization

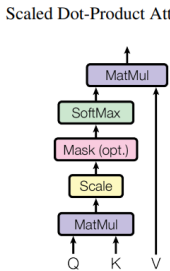


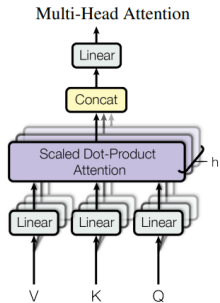
Multi-Head Self-Attention

- Cho phép mô hình học mối quan hệ giữa các từ trong câu.
- Công thức:

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

- Multi-head: nhiều attention song song để học các mối quan hệ khác nhau.





Feed Forward và Residual

- Feed Forward Network:

$$\text{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2$$

- Áp dụng độc lập tại từng vị trí.
- Residual Connection: $x + \text{Sublayer}(x)$
- Layer Normalization: ổn định huấn luyện.

Tổng kết Encoder

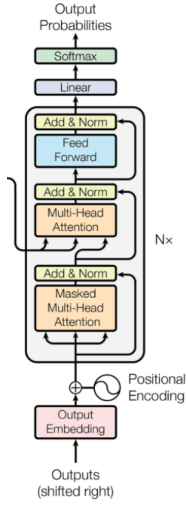
Thành phần	Vấn đề giải quyết	Cơ chế
Self-Attention	Phụ thuộc dài hạn	Attention đa hướng
Positional Encoding	Thiếu vị trí từ	Sin/Cos tuần hoàn
Feed Forward	Phi tuyến tính	Mạng 2 lớp
Residual + Norm	Gradient	Lan truyền ổn định

Decoder

Mỗi lớp decoder gồm **3** tầng con:

- Masked Multi-head Self-Attention layer (bỏ qua các token tương lai)
- Encoder-Decoder Attention layer (chú ý đến đầu ra của encoder)
- Feed Forward layer

Mỗi tầng con cũng theo sau bởi bước "Add & Normalize"



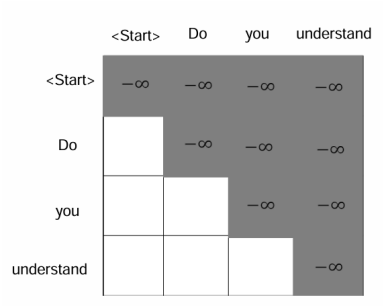
Masked Multi-head Self-Attention layer

Vấn đề: Decoder là quá trình giải mã tuần tự dạng mô hình ngôn ngữ, ta không thể "nhìn đáp án phía sau".

$\Rightarrow$ Tại mỗi bước tính của Decoder, ta mở rộng dần tập key và value.

Vấn đề mới: Không tính toán song song được.

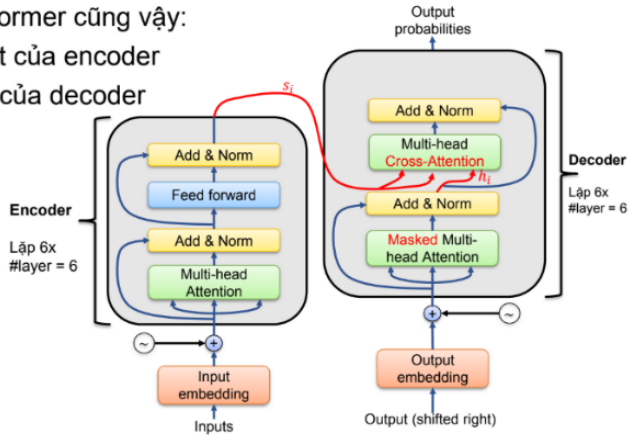
Giải pháp: Sử dụng Masked Multi-head Self-Attention, che các attention của các từ phía sau bằng cách gán attention score bằng $-\infty$.



Encoder-Decoder Attention (Cross-Attention)

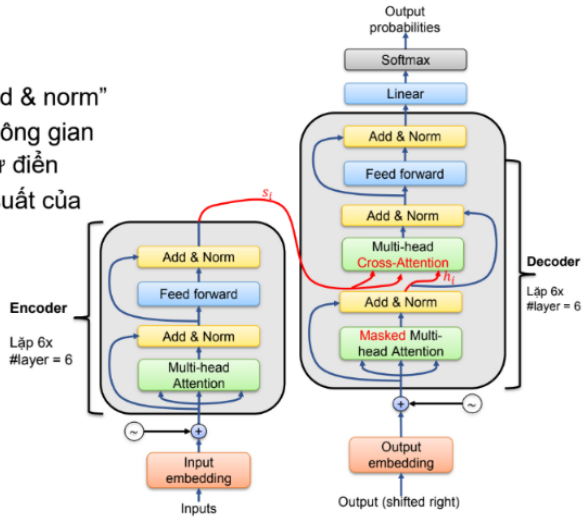
- Trong bài Cơ chế Attention, key đến từ decoder, value đến từ encoder, Transformer cũng vậy:
- $s_1, s_2, \ldots, s_T \in \mathbb{R}^d$ là các output của encoder
- $h_1, h_2, \ldots, h_T \in \mathbb{R}^d$ là các input của decoder
- Khi đó key, value và query:

$$k_i = Ks_i$$
$$v_i = Vs_i$$
$$q_i = Qh_i$$



Decoder - Những bước cuối cùng

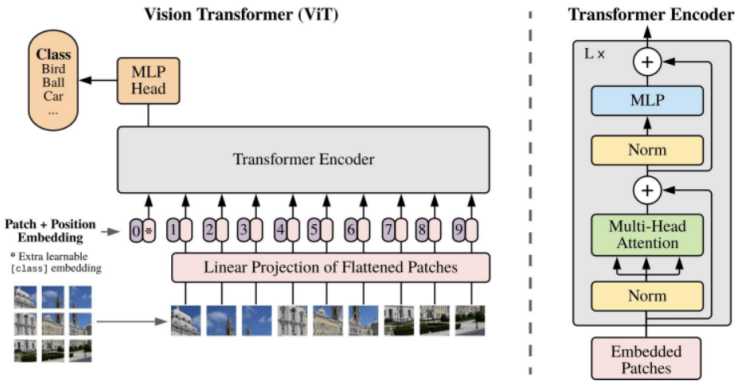
- Thêm “Feed forward” với “Add & norm”
- Thêm “Linear” để chiếu từ không gian đặc trưng sang không gian từ điển
- Thêm “softmax” để tính xác suất của từ tiếp theo



Overview

Vision Transformer

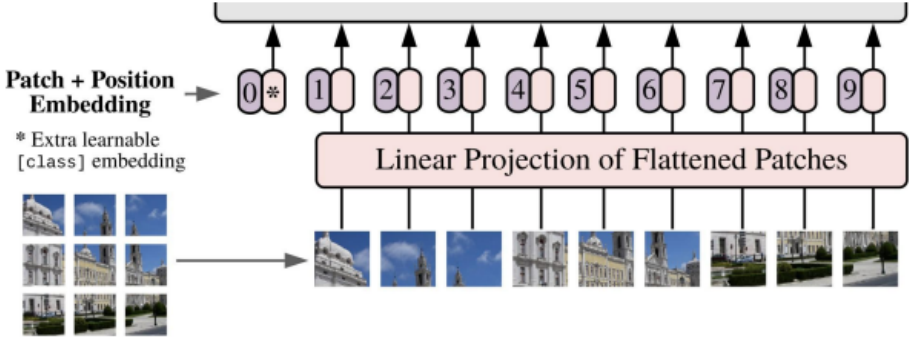
Bài toán: Làm thế nào để đưa dữ liệu hình ảnh vào mô hình Transformer?



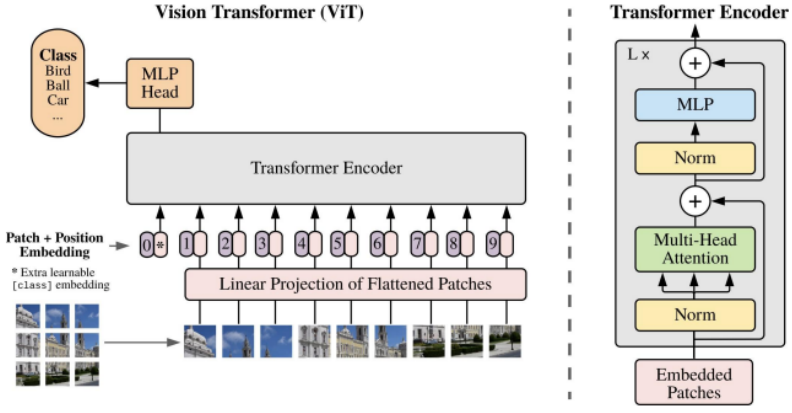
Ảnh: Tổng quan mô hình Vision Transformer

Vision Transformer - Lớp Linear Projection

Giải pháp: Chia ảnh đầu vào $x \in \mathbb{R}^{H \times W \times C}$ thành các mảnh (patches), trải phẳng (flatten), và mã hóa (encode) bằng một lớp tuyến tính (linear projection).



Vision Transformer - Cho bài toán phân loại ảnh



$$\boldsymbol{x}_p \in \mathbb{R}^{P^2 C} \qquad N = HW/P^2$$