

ĐẠI HỌC QUỐC GIA TP HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA KHOA HỌC MÁY TÍNH



BÁO CÁO ĐỒ ÁN CUỐI KÌ
CÁC KỸ THUẬT HỌC SÂU VÀ ỨNG DỤNG - CS431

VIETNAMESE IMAGE CAPTIONING

Giảng viên: TS. Nguyễn Vinh Tiệp

Nhóm sinh viên:

Tên	MSSV
Hồ Yến Nhi	21520380
Nguyễn Hà Anh Vũ	21520531
Nguyễn Đinh Minh Chí	21520648
Hứa Bảo Duy	21521993

12/2024, TP Hồ Chí Minh

Mục lục

1 CHƯƠNG 1. BÀI TOÁN	3
1.1 Định nghĩa Image Captioning	3
1.2 Mô tả bài toán	3
1.2.1 Dữ liệu đầu vào (Input)	3
1.2.2 Dữ liệu đầu ra (Output)	4
1.2.3 Ràng buộc (Constraints)	4
1.2.4 Yêu cầu (Requirements)	4
1.3 Thách thức của bài toán Image Captioning	5
1.3.1 Hiểu ý nghĩa hình ảnh	5
1.3.2 Sự đa dạng ngôn ngữ	6
1.4 Ứng dụng của bài toán Image Captioning	6
2 CHƯƠNG 2. PHƯƠNG PHÁP GIẢI QUYẾT BÀI TOÁN	8
2.1 Pipeline	8
2.2 Mô hình	8
2.2.1 SWIN và BART	8
2.2.2 GRIT: Grid- and Region-based Image captioning Transformer	11
2.2.3 EfficientNet_v2 và Transformer	15
3 CHƯƠNG 3. THỰC NGHIỆM	25
3.1 Dataset	25
3.2 Độ đo	25
3.2.1 BLEU	25
3.2.2 ROUGE	26
3.2.3 CIDEr	27
3.3 Training	28
3.4 Kết quả	28
4 CHƯƠNG 4. KẾT LUẬN, HƯỚNG PHÁT TRIỂN	30
4.1 Kết luận	30
4.2 Hướng phát triển	30
5 CHƯƠNG 5. TRẢ LỜI CÂU HỎI	32
5.1 Grid feature là gì?	32
5.2 Tại sao phương pháp EfficientNetV2 chỉ train có 30p?	32
5.3 Dataset cấu trúc như thế nào? Một ảnh có nhiều caption hay chỉ 1? 4000 ảnh dùng để train có quá ít không?	32
5.4 Cơ chế attention của swin transformer cho nó lợi thế như thế nào so với transformer truyền thống trong bài toán liên quan đến ảnh?	32

5.5	Mô hình có thể hiểu được các lễ hội cụ thể của Việt Nam trong hình không?	33
5.6	GRIT sử dụng phương pháp gì để kết hợp đặc trưng lại với nhau?	33
5.7	Model Transformer nếu dùng global attention thì có cho kết quả tốt hơn không?	33
5.8	Giải thích ý nghĩa các metric đánh giá.	33
5.9	Hệ thống này tạo caption bằng cách dựa vào các mẫu đã có hay hoàn toàn mang tính sáng tạo ?	33
5.10	Thời gian inference trung bình là bao lâu?	33
5.11	GRIT viết tắt chữ gì?	34
5.12	Phương pháp cho độ chính xác cao nhất, theo nhóm em điều gì tạo nên sự khác biệt so với các mô hình còn lại?	34
5.13	Mô hình có thể đọc được chữ tiếng Việt? Ví dụ: trong ảnh có chữ tiếng Việt nó đọc được ko?	34

CHƯƠNG 1. BÀI TOÁN

1.1 Định nghĩa Image Captioning

Image Captioning là quá trình tạo ra câu mô tả (caption) cho một hình ảnh dựa trên ngôn ngữ tự nhiên. Quá trình này kết hợp giữa hệ thống hiểu biết thị giác (visual understanding system) và mô hình ngôn ngữ (language model) để sinh ra các câu chính xác về mặt ngữ pháp, ngữ nghĩa và mạch lạc về nội dung.



Hình 1: Ví dụ.

1.2 Mô tả bài toán

1.2.1 Dữ liệu đầu vào (Input)

$$D = \{I_k, C_k\}_{k=1}^N \text{ và } I_{\text{new}} \in F$$

với $\begin{cases} I_k \in F = \mathbb{R}^d, \\ C_k = \{c_{k1}, c_{k2}, \dots, c_{km}\} \end{cases}$

Trong đó:

- + D : Bộ dữ liệu gồm các ảnh và tập các caption tương ứng.
- + I_k : Ảnh thứ k trong tập dữ liệu.
- + C_k : Tập các caption mô tả cho ảnh I_k .
- + N : Số lượng ảnh trong bộ dữ liệu.
- + I_{new} : Một ảnh mới cần được mô tả.
- + F : Không gian chứa các ảnh (\mathbb{R}^d).

- **Tập dữ liệu:**

- Một tập hợp các hình ảnh, mỗi hình ảnh được gắn kèm một hoặc nhiều câu mô tả chi tiết về nội dung của nó.
- Mỗi hình ảnh có thể chứa một hoặc nhiều đối tượng, hành động, ngữ cảnh hoặc các yếu tố khác mà mô hình cần nhận diện và hiểu.

- **Hình ảnh mới:**

- Một hình ảnh đầu vào không có câu mô tả kèm theo.

1.2.2 Dữ liệu đầu ra (Output)

$$\hat{c} = f(I_{new}) \text{ với } f \text{ là model Image Captioning}$$

- Một câu mô tả bằng ngôn ngữ tự nhiên thể hiện nội dung trực quan của hình ảnh.
- Câu mô tả cần bao gồm thông tin về:
 - Các đối tượng quan trọng trong ảnh.
 - Hành động đang diễn ra.
 - Ngữ cảnh hoặc các chi tiết liên quan khác.
- Đảm bảo câu văn rõ ràng, mạch lạc, phù hợp về mặt ngữ pháp và truyền tải đúng ý nghĩa.

1.2.3 Ràng buộc (Constraints)

- **Về hình ảnh:**

- Ảnh đầu vào có thể chứa nhiều đối tượng chồng chéo, ngữ cảnh phức tạp, hoặc ánh sáng không đồng đều.

- **Về câu mô tả:**

- Phải chính xác về mặt ngữ pháp và ngữ nghĩa.
- Độ dài câu mô tả không quá ngắn để thiếu ý, cũng không quá dài gây khó hiểu.
- Không chứa thông tin không liên quan hoặc chưa xuất hiện trong ảnh.

- **Về thời gian xử lý:**

- Hệ thống cần tạo ra câu mô tả trong thời gian hợp lý để đảm bảo hiệu quả sử dụng thực tế.

1.2.4 Yêu cầu (Requirements)

Hiểu biết thị giác:

- Nhận diện chính xác các đối tượng xuất hiện trong ảnh.
- Hiểu được mối quan hệ giữa các đối tượng (ví dụ: vị trí tương đối, hành động tương tác).

Khả năng ngôn ngữ:

- Sinh ra câu mô tả tự nhiên, giống với cách con người giao tiếp.
- Đảm bảo câu văn đúng ngữ pháp, có cấu trúc rõ ràng và nội dung đầy đủ.

Khả năng tổng quát hóa:

- Hệ thống phải hoạt động tốt trên các hình ảnh có các chủ đề khác nhau.
- Không phụ thuộc vào một loại nội dung hình ảnh cụ thể (ví dụ: chỉ mô tả động vật hoặc chỉ mô tả cảnh vật).

Tích hợp hiệu quả:

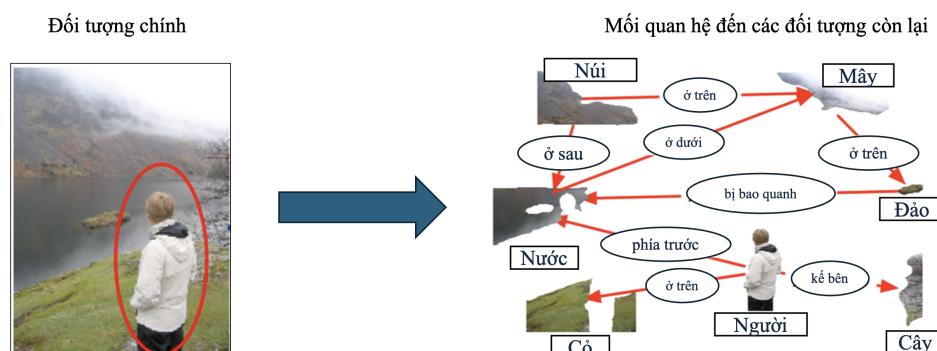
- Dễ dàng tích hợp vào các ứng dụng thực tế như công cụ hỗ trợ người khiếm thị, tìm kiếm hình ảnh, hoặc sản xuất nội dung tự động.

1.3 Thách thức của bài toán Image Captioning

1.3.1 Hiểu ý nghĩa hình ảnh

Hiểu ý nghĩa hình ảnh là nền tảng quan trọng trong bài toán image captioning, giúp hệ thống tạo ra các mô tả phù hợp và có ý nghĩa.

- **Hiểu các đối tượng trong ảnh:** Trước tiên, mô hình cần xác định các đối tượng chính trong hình ảnh. Ví dụ: nhận biết rằng có một "chiếc ô tô" hoặc "một chú chó" xuất hiện. Đây là bước cơ bản nhưng không đủ để tạo ra chú thích hoàn chỉnh.
- **Hiểu mối quan hệ giữa các đối tượng:** Bài toán captioning không chỉ đơn thuần liệt kê các đối tượng mà cần mô tả cách chúng tương tác. Ví dụ: "Một chú chó đang ngồi trong ô tô" khác hoàn toàn với "Một chiếc ô tô đang đậu gần chú chó." Để làm được điều này, mô hình cần khai thác các đặc trưng ngữ nghĩa sâu hơn từ ảnh và mối quan hệ giữa các vùng ảnh.

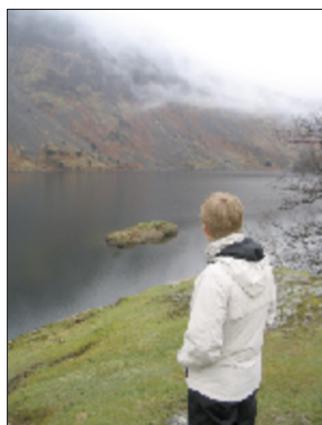


Hình 2: Semantic Understanding

1.3.2 Sự đa dạng ngôn ngữ

Da dạng hóa mô tả là một thách thức lớn trong image captioning:

- **Nhiều cách diễn đạt cho cùng một bức ảnh:** Với một bức ảnh chứa "một chú mèo trên ghế sofa", mô hình cần tạo ra các mô tả khác nhau như:
 - "Một chú mèo đang nằm trên ghế."
 - "Có một con mèo thư giãn trên ghế sofa."
 - "Chú mèo đang ngủ ngon lành trên ghế."
- **Tạo ngữ cảnh phù hợp:** Một bức ảnh có thể được mô tả khác nhau tùy thuộc vào ngữ cảnh hoặc mục đích sử dụng. Ví dụ, trong bối cảnh thương mại, mô tả có thể tập trung vào chi tiết sản phẩm như "Một chiếc ghế sofa sang trọng với một chú mèo dễ thương đang nằm."



Caption 1: Người đàn ông mặc áo khoác đang nhìn xuống hồ

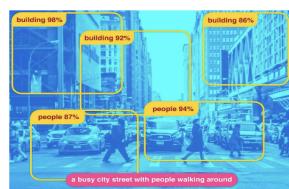
Caption 2: Một người đang mặc áo trắng, tóc vàng đứng trước mặt hồ

Hình 3: Alternative captions.

1.4 Ứng dụng của bài toán Image Captioning



Hỗ trợ người khiếm thị



Hệ thống tạo mô tả cho hình ảnh



Tích hợp vào các công cụ tìm kiếm và truy vấn hình ảnh dựa trên mô tả

Hình 4: Ứng dụng của bài toán Image Captioning.

• **1. Hỗ trợ người khuyết tật:** Image captioning có thể đóng vai trò quan trọng trong việc hỗ trợ người khiếm thị tiếp cận thông tin hình ảnh:

- **Chuyển đổi nội dung hình ảnh sang văn bản hoặc âm thanh:** Các mô tả tự động về hình ảnh có thể được đọc bằng giọng nói thông qua screen readers, giúp người khiếm thị hiểu nội dung mà họ khó có thể nhìn thấy, giúp họ nhận biết và tưởng tượng được tình huống.
- **Tăng cường trải nghiệm giao tiếp:** Ứng dụng trong các nền tảng mạng xã hội như Facebook, Twitter, cho phép người khiếm thị tiếp cận nội dung hình ảnh được chia sẻ qua các mô tả tự động.
- **Hỗ trợ học tập:** Trong giáo dục, các tài liệu hình ảnh hoặc biểu đồ có thể được chuyển đổi thành văn bản mô tả, giúp học sinh khiếm thị nắm bắt được nội dung bài học.

• **2. Hệ thống hỗ trợ tạo mô tả cho hình ảnh:** Hệ thống Image captioning có thể giúp đơn giản hóa các công việc chuyên môn:

- **Tự động hóa công việc chỉnh sửa và chú thích hình ảnh:**
 - * Trong các ngành sáng tạo như báo chí, quảng cáo và thiết kế, image captioning giúp tạo ra các mô tả nhanh chóng, tiết kiệm thời gian chỉnh sửa. Ví dụ: khi tải lên một bộ ảnh, hệ thống có thể tự động tạo chú thích cho từng ảnh để xuất bản hoặc chia sẻ.
 - * Trong thương mại điện tử, tạo mô tả tự động cho hình ảnh có thể được dùng để mô tả cho các sản phẩm mới.
- **Hỗ trợ tạo nội dung chuyên biệt:**
 - * Trong y tế, image captioning có thể mô tả các hình ảnh X-quang hoặc MRI, giúp bác sĩ chẩn đoán dễ dàng hơn khi có các chú thích tự động.
 - * Trong pháp lý, các hệ thống này có thể giúp mô tả hình ảnh hoặc video liên quan đến bằng chứng.

• **3. Tích hợp vào các công cụ tìm kiếm và truy vấn hình ảnh dựa trên mô tả**
Image captioning nâng cao khả năng truy vấn hình ảnh và video thông qua mô tả tự nhiên:

- **Cải thiện tìm kiếm bằng văn bản:** Các công cụ tìm kiếm hình ảnh như Google Images có thể sử dụng captioning để gắn thẻ hình ảnh một cách chính xác và tự động. Điều này giúp cải thiện độ chính xác khi người dùng nhập các truy vấn phức tạp như “một chiếc xe thể thao màu đỏ đỗ gần biển”.
- **Truy vấn ngược bằng hình ảnh:** Khi người dùng tải lên một hình ảnh, hệ thống có thể tạo ra chú thích tự động để tìm kiếm các nội dung tương tự trên internet.
- **Ứng dụng trong quản lý dữ liệu:** Các thư viện hình ảnh hoặc video lớn có thể tự động tổ chức và gắn thẻ nội dung thông qua mô tả, giúp việc tìm kiếm và phân loại hiệu quả hơn.

CHƯƠNG 2. PHƯƠNG PHÁP GIẢI QUYẾT BÀI TOÁN

2.1 Pipeline

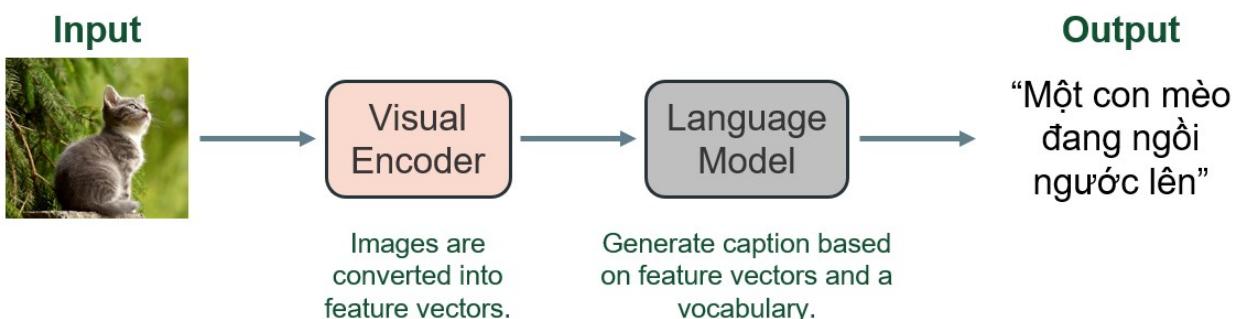
Quá trình tạo caption cho ảnh (Image Captioning) được thực hiện qua một pipeline gồm nhiều bước liên tiếp, từ việc trích xuất đặc trưng hình ảnh đến việc sinh ra câu mô tả chi tiết cụ thể như sau:

Quá trình tạo caption cho ảnh bắt đầu với ảnh đầu vào mà mô hình cần mô tả. Ảnh này sẽ được đưa qua một Visual Encoder để trích xuất các đặc trưng của ảnh. Visual Encoder, thường là các mô hình deep learning, có nhiệm vụ chuyển hình ảnh thành các vector đặc trưng (feature vectors) phản ánh nội dung của hình ảnh, bao gồm các đối tượng, cảnh vật và mối quan hệ giữa chúng. Các đặc trưng này có thể là một vector duy nhất hoặc một tập hợp các vector đại diện cho các phần khác nhau trong hình ảnh.

Tiếp theo, các đặc trưng từ Visual Encoder sẽ được đưa vào một Language Model để tạo ra chuỗi từ mô tả. Mô hình này sử dụng các đặc trưng ảnh cùng với các từ đã được sinh ra trước đó (hoặc một token bắt đầu như <start>) để dự đoán từ tiếp theo trong câu caption. Quá trình này diễn ra tuần tự, với mỗi từ mới được tạo ra dựa trên ngữ cảnh (từ trước đó) và đặc trưng hình ảnh, cho đến khi mô hình sinh ra từ kết thúc (thường là <end>).

Cuối cùng, mô hình Language Model sẽ trả về một output caption dưới dạng một câu văn hoàn chỉnh. Caption này mô tả nội dung của hình ảnh đầu vào, bao gồm các đối tượng, hành động và mối quan hệ giữa chúng.

Toàn bộ quá trình trên giúp mô hình tự động tạo ra mô tả ngữ nghĩa chính xác và mạch lạc cho bất kỳ hình ảnh nào, dựa trên các đặc trưng được học từ dữ liệu huấn luyện.

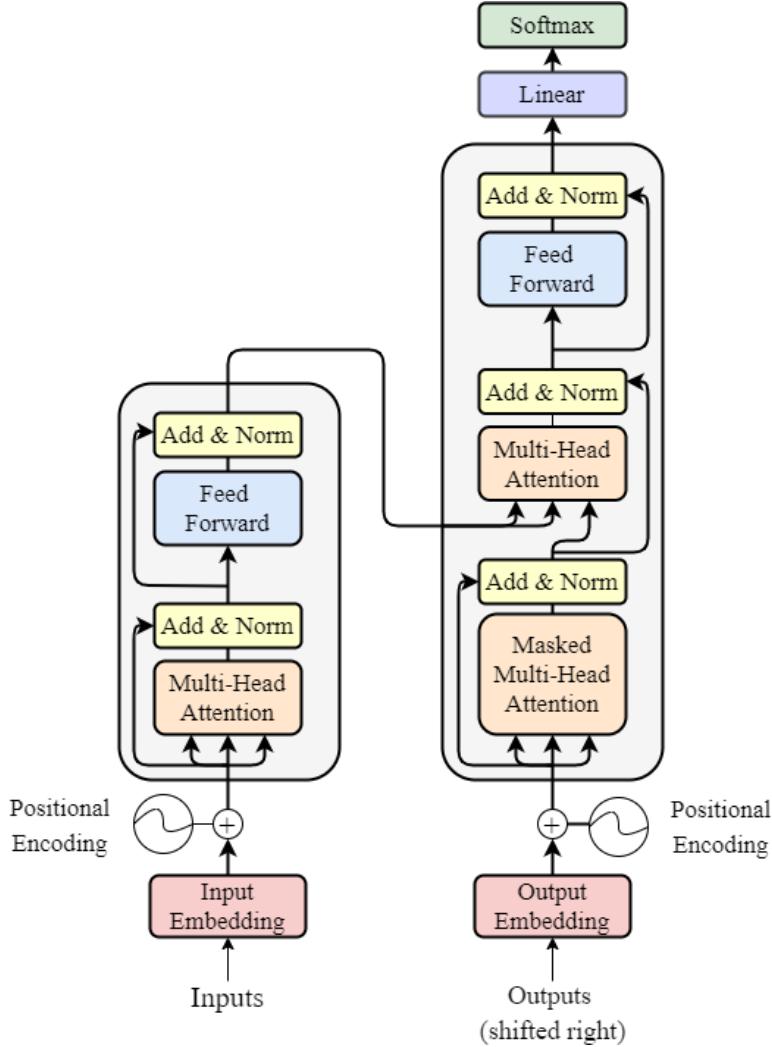


Hình 5: Pipeline.

2.2 Mô hình

2.2.1 SWIN và BART

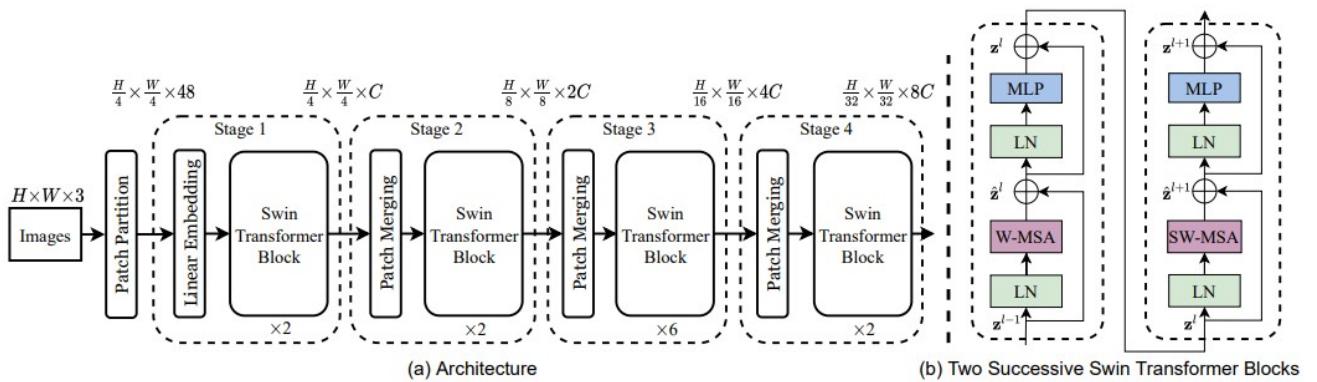
Mô hình được thiết kế dựa theo Transformer-based với backbone cho encoder là mô hình Swin dùng để trích xuất đặc trưng của ảnh và decoder của mô hình được lấy từ phần decoder của mô hình Bart.



Hình 6: Kiến trúc tổng thể của mô hình SWIN Transformer và BART.

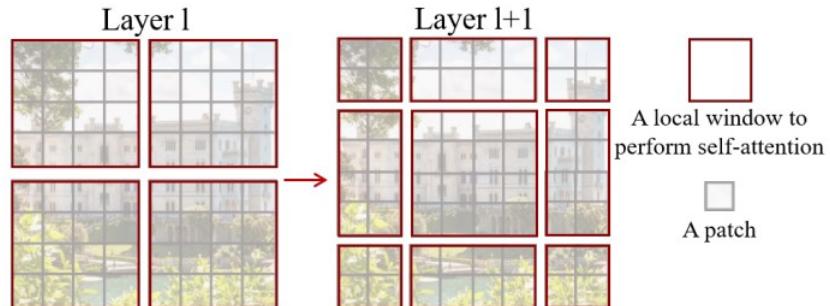
Swin Transformer

- Swin Transformers có kiến trúc gồm 4 layer, layer đầu gồm 1 Linear Embedding để tạo embedding vector cùng với Swin Transformer Block, các layer sau mỗi patch sẽ được gom lại qua Patch Merging, làm giảm kích thước không gian và tăng chiều của vector, tiếp theo là Swin Transformer Block tương tự như layer đầu tiên.



Hình 7: Mô hình Swin Transformer.

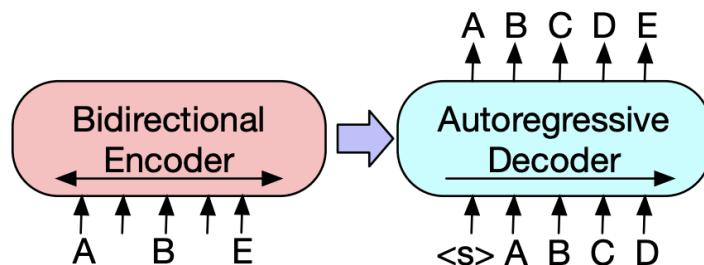
- Khác với việc tính global attention của Transformer thì Swin Transformer dùng window-based attention để tính attention trong phạm vi window của các patch được chia từ input, giúp giảm độ phức tạp. Sau đó, cửa sổ này sẽ được dịch chuyển (shifted) để các patch ngoài cửa sổ ban đầu có thể giao tiếp với nhau, giúp mô hình học được mối quan hệ giữa các patch không liền kề.



Hình 8: Cơ chế local attention của Swin Transformer.

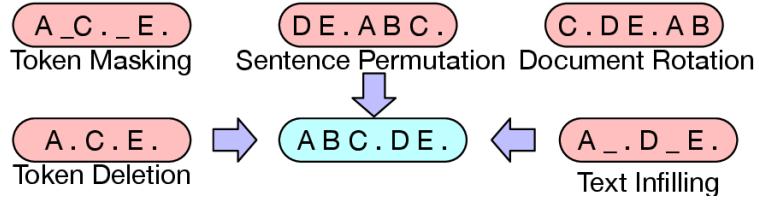
BART

- BART là một mô hình denoising autoencoder được thiết kế để tái tạo lại văn bản gốc từ một phiên bản đã bị làm nhiễu, bằng cách học cách khôi phục các thông tin bị thiếu hoặc sắp xếp sai.
- BART được xây dựng như một mô hình sequence-to-sequence, với bộ mã hóa hai chiều (bidirectional encoder) và bộ giải mã tự hồi quy từ trái sang phải (left-to-right decoder). Ở phiên bản base, mô hình bao gồm 6 lớp trong cả encoder và decoder, trong khi phiên bản large mở rộng lên 12 lớp cho mỗi thành phần. Mỗi lớp trong decoder đều thực hiện cross-attention trực tiếp với đầu ra từ lớp cuối cùng của encoder, tạo ra sự kết nối chặt chẽ giữa hai thành phần và tăng cường khả năng học biểu diễn ngữ cảnh.



Hình 9: Mô hình Bart.

- BART được tiền huấn luyện(pretrain) bằng cách làm hỏng các văn bản gốc sau đó tối ưu reconstruction loss(phép đo cross-entropy giữa đầu ra của bộ giải mã và văn bản gốc). Các tác vụ được thực hiện trong quá trình tiền huấn luyện bao gồm: che token(token masking), xóa token(token deletion), điền văn bản(text infilling), hoán vị câu(sentence permutation) và xoay vòng văn bản(document rotation).



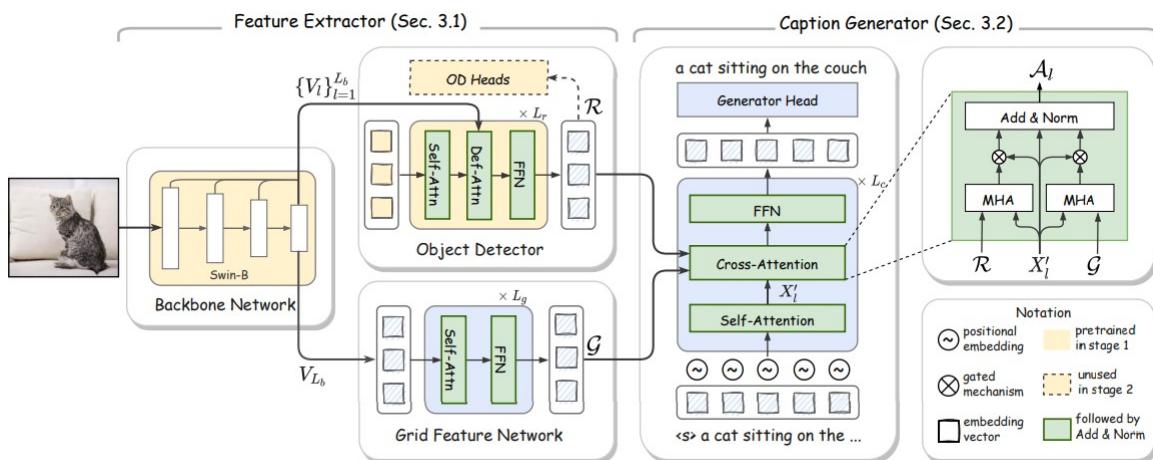
Hình 10: Các tác vụ tiền huấn luyện Bart.

2.2.2 GRIT: Grid- and Region-based Image captioning Transformer

GRIT (Grid- and Region-based Image captioning Transformer) là một phương pháp tiên tiến được phát triển cho bài toán image captioning với mục tiêu chính là khắc phục những hạn chế của các phương pháp truyền thống, đồng thời nâng cao cả độ chính xác trong việc tạo caption lẫn hiệu quả tính toán trong quá trình xử lý.

GRIT nổi bật nhờ khả năng kết hợp đặc trưng thị giác dựa trên vùng (region-based), cung cấp thông tin chi tiết về đối tượng trong hình ảnh, và đặc trưng dựa trên lưới (grid-based), bổ sung ngữ cảnh tổng thể. Cách tiếp cận này không chỉ tăng cường khả năng hiểu nội dung hình ảnh mà còn giảm thiểu các lỗi liên quan đến phát hiện đối tượng không chính xác. Hơn nữa, GRIT sử dụng DETR (DEtection TRansformer) thay vì các bộ phát hiện đối tượng truyền thống như Faster R-CNN, giúp cải thiện tốc độ và giảm chi phí tính toán.

Mô hình gồm 2 phần, đầu tiên là trích xuất 2 đặc trưng (region feature và grid feature) từ ảnh input sau đó là tạo caption dựa trên các đặc trưng đã trích xuất.

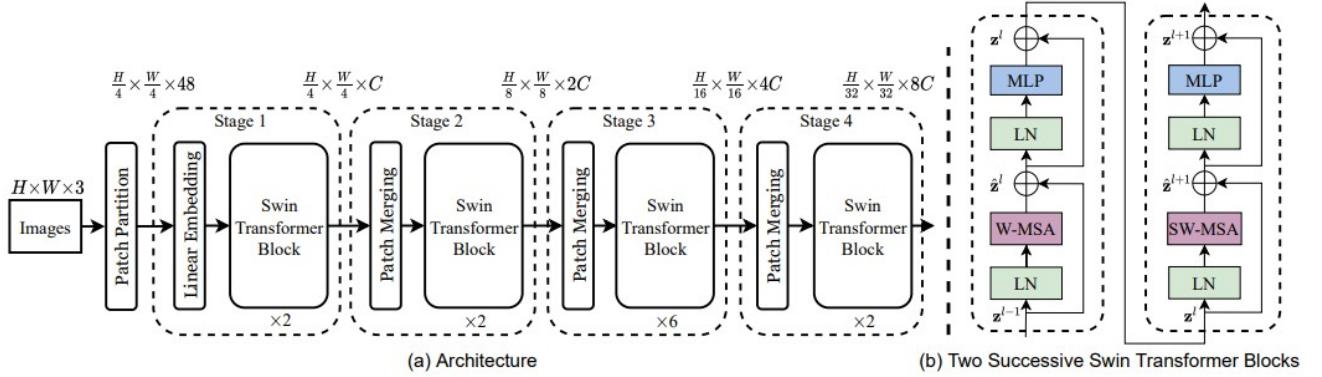


Hình 11: Tổng quan Kiến trúc mô hình GRIT.

Đầu tiên, dùng 1 backbone network để trích xuất initial feature, ở đây dùng **Swin Transformer**, đây là 1 bản cải tiến của vision transformer. Với ViT chia ảnh thành từng patch sau đó trích xuất đặc trưng toàn cục, điều này thì không phù hợp với các task phức tạp vì độ chính xác khá thấp cũng như tốn thời gian tính toán. Vì thế dùng Swin Transformer đã khắc phục được các vấn đề đó bằng cách giảm kích thước patch và shifted window cùng với cơ chế local self-attention.

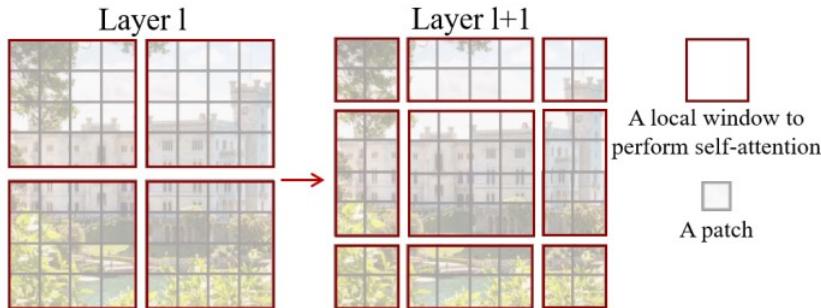
Swin Transformer:

- Swin Transformers có kiến trúc gồm 4 layer, layer đầu gồm 1 Linear Embedding để tạo embedding vector cùng với Swin Transformer Block, các layer sau mỗi patch sẽ được gom lại qua Patch Merging, làm giảm kích thước không gian và tăng chiều của vector, tiếp theo là Swin Transformer Block tương tự như layer đầu tiên.



Hình 12: Mô hình Swin Transformer.

- Khác với việc tính global attention của Transformer thì Swin Transformer dùng window-based attention để tính attention trong phạm vi window của các patch được chia từ input, giúp giảm độ phức tạp. Sau đó, cửa sổ này sẽ được dịch chuyển (shifted) để các patch ngoài cửa sổ ban đầu có thể giao tiếp với nhau, giúp mô hình học được mối quan hệ giữa các patch không liền kề.

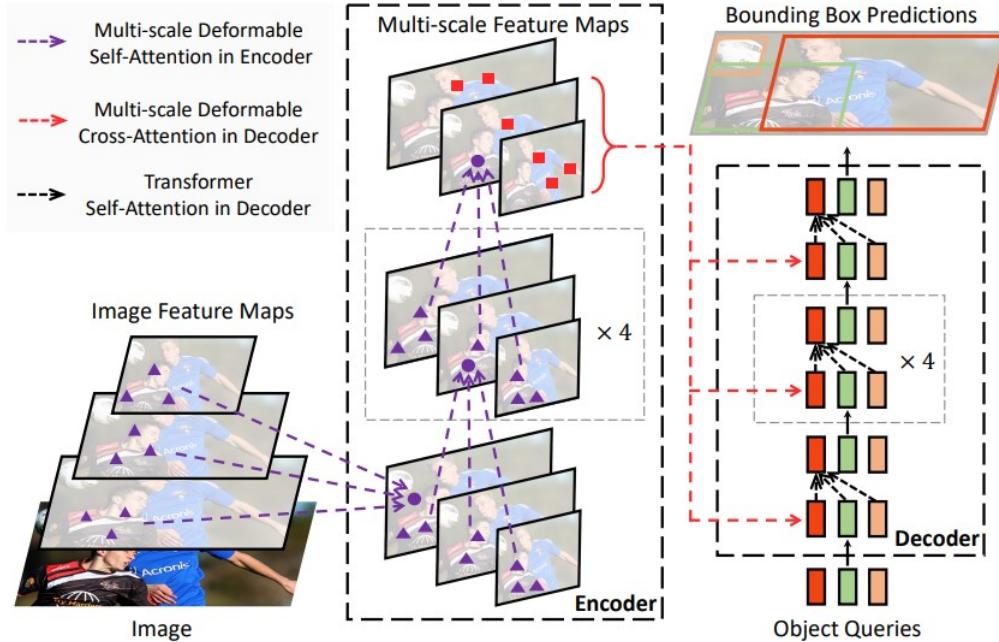


Hình 13: Cơ chế local attention của Swin Transformer.

Tiếp theo, quá trình tạo region features từ initial feature được trích xuất bởi Backbone Network được thực hiện thông qua decoder của mô hình **Deformable DETR** (DEtection TRansformer). Cụ thể như sau:

- Multi-scale feature maps từ Backbone được đưa vào cùng với object queries được khởi tạo ngẫu nhiên.
- Sau đó, qua lớp Self-Attention tính mối quan hệ giữa các object queries và feature maps, học cách kết nối các vùng trong ảnh để tạo thông tin tổng quát.
- Tiếp theo, lớp Deformable Attention tập trung vào các vùng quan trọng hơn trong ảnh, giảm độ phức tạp tính toán trong khi vẫn giữ lại các đặc trưng cần thiết.

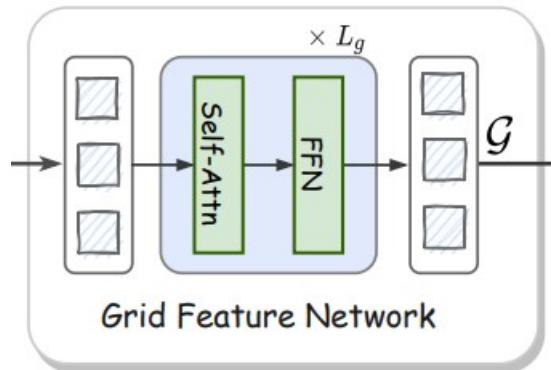
- Cuối cùng, lớp Feed-Forward Network (FFN) xử lý và tăng cường biểu diễn để tạo ra region features, đại diện cho các đối tượng trong ảnh.



Hình 14: Kiến trúc mô hình DERT (GRIT chỉ dùng phần Decoder).

Ngoài region feature thì phương pháp GRIT còn dùng 1 loại feature nữa là grid feature để học thông tin ngữ cảnh của ảnh.

Grid feature được tạo như sau: Đầu tiên, feature map cuối cùng trong các multi-scale feature maps từ Swin Transformer backbone được lấy làm đầu vào sau đó biến đổi tuyến tính bằng ma trận W_g để chuyển vào không gian d . Rồi tiếp tục đi qua 1 model Transformer với cơ chế self-attention gồm L_g lớp học mối quan hệ và ngữ cảnh giữa các vùng trong ảnh từ đó tạo ra grid feature.



Hình 15: Mô hình tạo Grid Feature.

Sau khi trích xuất hai loại đặc trưng hình ảnh là region features và grid features, quá trình tạo caption bắt đầu.

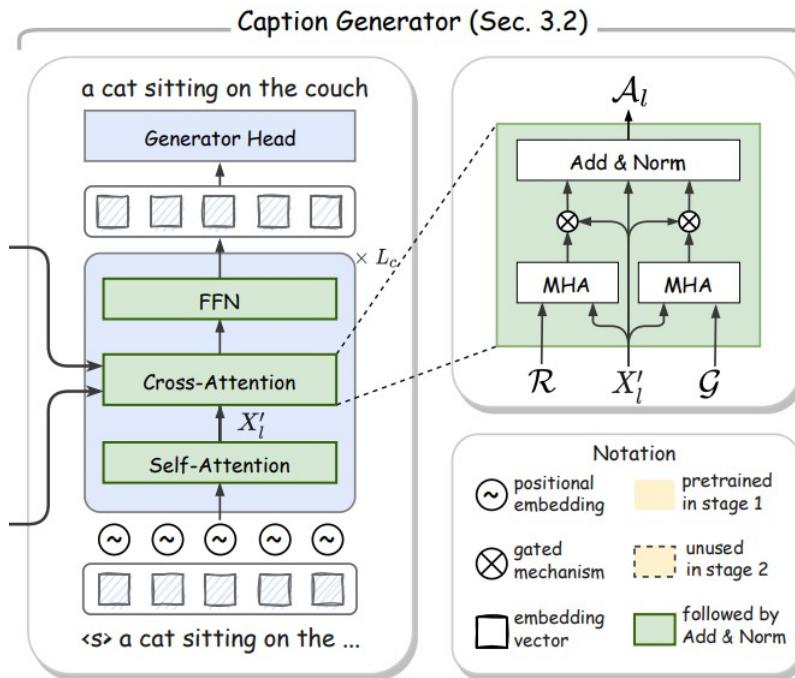
Caption Generator được thiết kế dựa trên kiến trúc Transformer, hoạt động theo cơ chế

tự hồi quy (autoregression) tức là tại thời điểm t , mô hình nhận embedding của các từ đã được dự đoán ở thời điểm $t-1$, kết hợp với thông tin vị trí để dự đoán từ tiếp theo.

Caption Generator bao gồm một ngăn xếp L_c lớp Transformer giống hệt nhau, với cấu trúc như sau:

- Đầu tiên là lớp Masked Self-Attention: Lớp này nhận chuỗi từ đã dự đoán trước đó, sử dụng attention mask để đảm bảo mô hình không truy cập thông tin từ các từ ở tương lai trong quá trình huấn luyện giúp việc dự đoán từ tiếp theo chỉ dựa trên ngữ cảnh đã biết.
- Sau đó qua lớp Cross-Attention: Đầu ra từ self-attention được kết hợp với cả region features và grid features thông qua Multi-Head Attention (MHA). Trong lớp này, các đặc trưng từ self-attention (X'_l) sẽ là queries, trong khi region features và grid features làm keys và values. MHA sẽ thực hiện tính attention song song với hai loại đặc trưng, giúp mô hình học cách liên kết ngữ cảnh giữa các từ trong câu và thông tin trực quan từ ảnh. Kết quả từ MHA sẽ được nối với X'_l , chiếu lại thành vector có kích thước d , chuẩn hóa và cộng với X'_l , sau đó đưa qua lớp chuẩn hóa để có đầu ra cuối cùng A_l .
- Cuối cùng qua Feedforward Network (FFN): để tăng cường khả năng biểu diễn trước khi truyền đến lớp tiếp theo.

Sau khi qua L_c lớp Transformer, output cuối cùng được đưa vào một linear layer để chuyển thành một vector có kích thước bằng từ vựng, dùng để dự đoán từ tiếp theo của caption. Theo



Hình 16: Cơ chế tạo Caption cho ảnh dựa trên Region feature và Grid feature.

kiến trúc như trên thì mô hình GRIT: Grid- and Region-based Image Captioning Transformer có các ưu điểm và nhược điểm như sau:

Ưu điểm:

- Hiệu suất cao: Tạo chú thích chính xác và chi tiết nhờ việc kết hợp grid-based và region-based representation.
- Khả năng hiểu ngữ cảnh tốt: Sử dụng Transformer để xử lý các mối quan hệ ngữ nghĩa phức tạp.
- Tính linh hoạt: Áp dụng tốt cho các hình ảnh với độ phân giải và kích thước khác nhau.
- Xử lý các vùng quan trọng: Tự động xác định và tập trung vào những khu vực quan trọng của hình ảnh.

Nhược điểm:

- Yêu cầu tài nguyên tính toán cao: Cần nhiều bộ nhớ và tài nguyên tính toán.
- Phức tạp trong triển khai: Cấu trúc mô hình phức tạp, khó tối ưu hóa và triển khai trên hệ thống hạn chế tài nguyên.
- Độ phụ thuộc vào dữ liệu huấn luyện: Cần dữ liệu đa dạng để tạo ra chú thích chính xác.

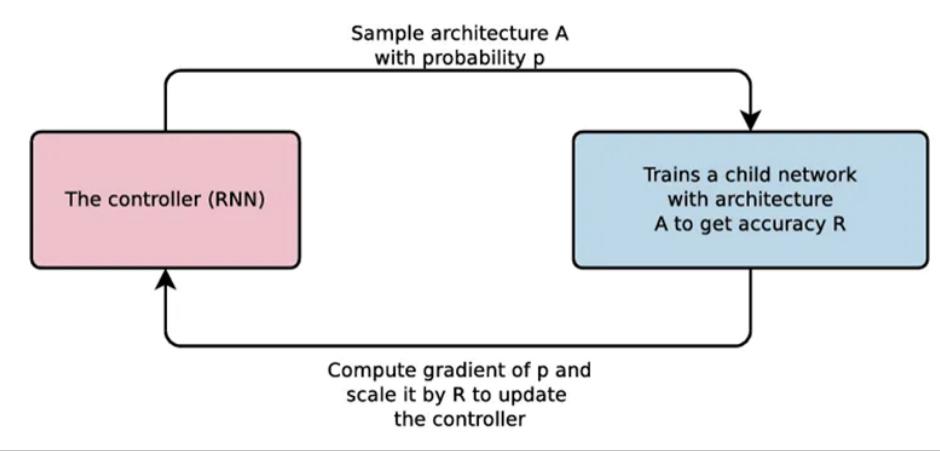
Tóm lại, nhờ vào các ưu điểm nổi bật, GRIT không chỉ nâng cao chất lượng chú thích hình ảnh mà còn có tiềm năng ứng dụng rộng rãi. Tuy nhiên, yêu cầu tài nguyên tính toán cao và sự phức tạp trong triển khai là những yếu tố cần xem xét khi áp dụng mô hình vào thực tế.

2.2.3 EfficientNet_v2 và Transformer

Encoder

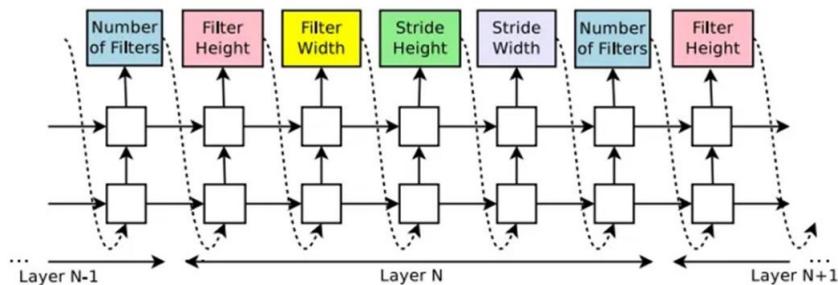
- EfficientNetv1:

Ý tưởng của EfficientNet là sử dụng bộ điều khiển (mạng chẵng hạn như RNN) và lấy mẫu kiến trúc mạng từ không gian tìm kiếm có xác suất ' p '. Kiến trúc này sau đó được đánh giá bằng cách huấn luyện mạng đầu tiên, sau đó xác thực nó trên một bộ thử nghiệm để có được độ chính xác ' R '. Độ dốc của ' p ' được tính toán và chia tỷ lệ theo độ chính xác ' R '. Kết quả (phần thưởng) được đưa đến bộ điều khiển RNN. Bộ điều khiển đóng vai trò là tác nhân, quá trình đào tạo, kiểm tra mạng đóng vai trò là môi trường và kết quả đóng vai trò là phần thưởng. Đây là vòng lặp của Học tăng cường (Reinforcement learning) phổ biến. Vòng lặp này chạy nhiều lần cho đến khi bộ điều khiển tìm thấy kiến trúc mạng mang lại phần thưởng cao (độ chính xác kiểm tra cao).



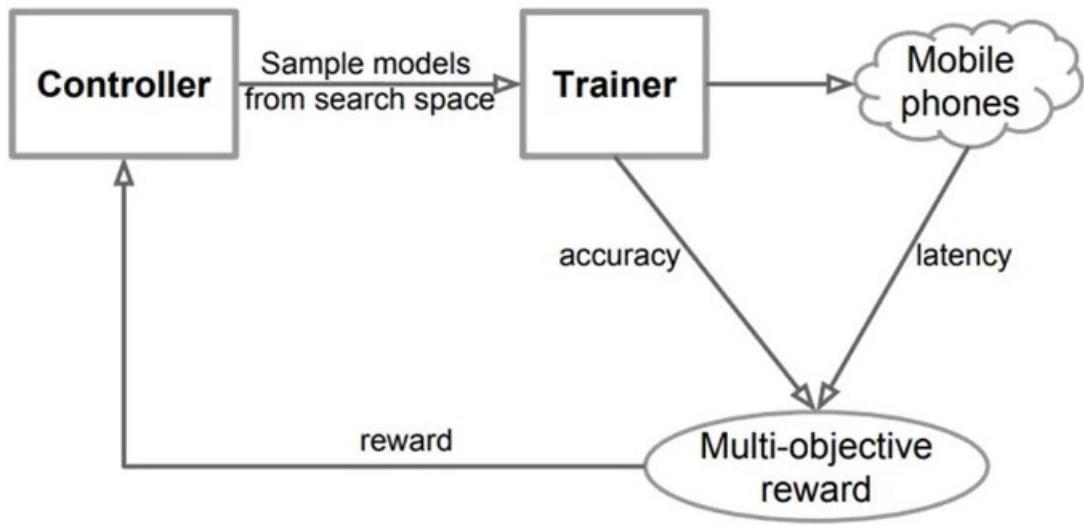
Hình 17: Cách hoạt động của bộ điều khiển (controller).

Bộ điều khiển RNN lấy mẫu các tham số kiến trúc mạng khác nhau — chẳng hạn như số lượng filters, độ cao filter, độ rộng filter, độ cao stride, và độ rộng stride cho mỗi lớp. Các tham số này có thể khác nhau đối với từng lớp của mạng. Cuối cùng, mạng có kết quả (phần thưởng) cao nhất được chọn làm kiến trúc mạng cuối cùng.



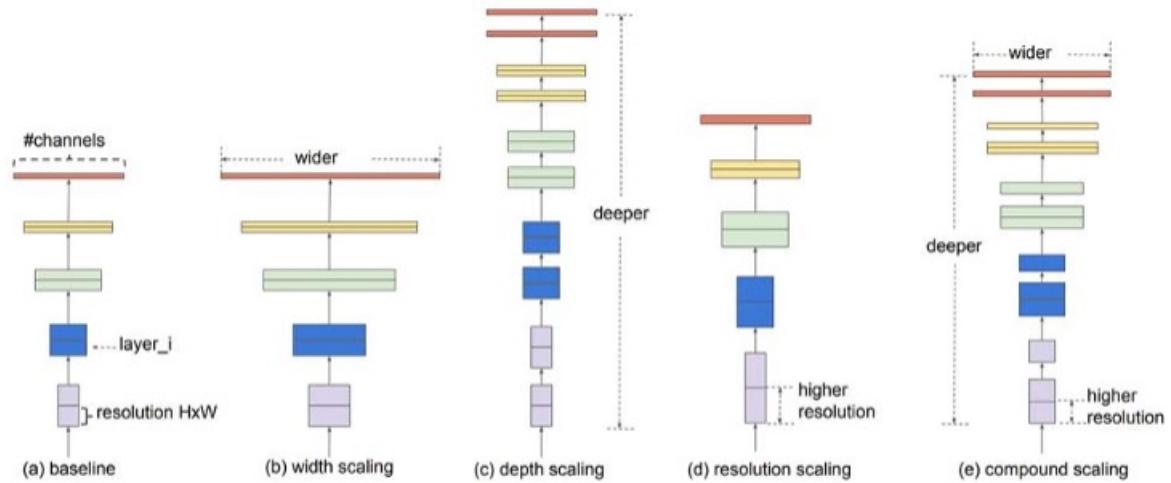
Hình 18: Cách hoạt động cụ thể của bộ điều khiển

Mặc dù phương pháp này hoạt động tốt, nhưng một trong những vấn đề với phương pháp này là nó đòi hỏi một lượng lớn sức mạnh tính toán cũng như thời gian. Các kiến trúc này không có các tham số khác nhau trong mỗi lớp, mà có một khối với nhiều lớp tích chập (còn gọi là ConvNet / CNN) và lớp tổng hợp, và trong toàn bộ kiến trúc mạng, các khối này được sử dụng nhiều lần. Các tác giả đã sử dụng ý tưởng này để tìm các khối như vậy bằng bộ điều khiển Reinforcement learning và chỉ cần lặp lại các khối này N lần để tạo kiến trúc NASNet có thể mở rộng. Trong mạng này, các tác giả đã chọn 7 khối và một lớp của khối được lấy mẫu và lặp lại cho mỗi khối. Ngoài các tham số này, một tham số rất quan trọng khác đã được xem xét khi quyết định phần thưởng, tham số này được đưa vào bộ điều khiển và đó là “độ trễ”. Vì vậy, đối với MnasNet, các tác giả đã xem xét cả độ chính xác và độ trễ để tìm ra kiến trúc mô hình tốt nhất. Điều này được thể hiện trong hình dưới. Điều này làm cho kiến trúc trở nên nhỏ gọn và nó có thể chạy trên thiết bị di động hoặc thiết bị biên.



Hình 19: Kiến trúc

Quy trình tìm kiếm kiến trúc EfficientNet rất giống với MnasNet, nhưng thay vì coi 'độ trễ' là tham số phần thưởng, 'FLOP (floating point operations per second)' đã được xem xét. Tìm kiếm theo tiêu chí này đã cho ra một mô hình cơ sở được gọi là EfficientNetB0. Tiếp theo, tăng tỷ lệ độ sâu, chiều rộng và độ phân giải hình ảnh của mô hình cơ sở (sử dụng tìm kiếm dạng lưới hay vét cạn) để tạo thêm 6 mô hình, từ EfficientNetB1 đến EfficientNetB7. Tỷ lệ này được hiển thị trong hình dưới.

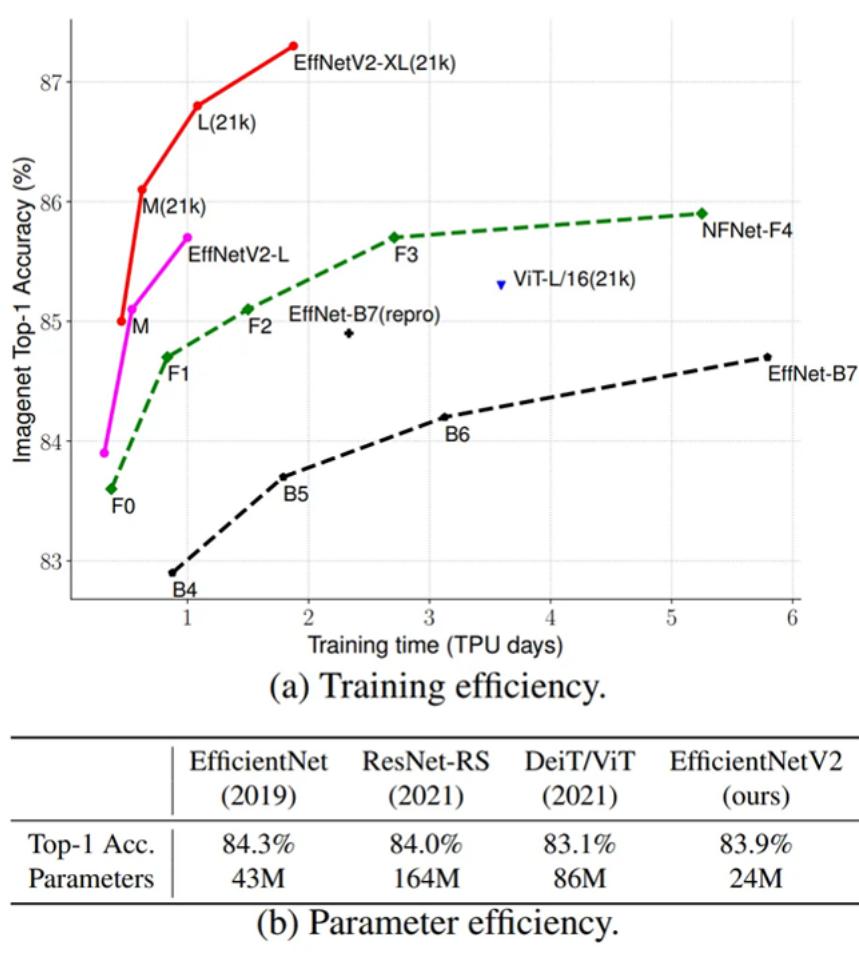


Hình 20: Tỷ lệ độ sâu, rộng và độ phân giải ảnh

- Vấn đề của EfficientNetv1:

EfficientNetV2 tiến xa hơn một bước so với EfficientNet để tăng tốc độ đào tạo và hiệu quả tham số. Mạng này được tạo bằng cách sử dụng kết hợp chia tỷ lệ (chiều rộng, độ sâu, độ phân giải) và tìm kiếm kiến trúc neural. Mục tiêu chính là tối ưu hóa tốc độ đào tạo và hiệu quả tham số. Ngoài ra, lần này không gian tìm kiếm cũng bao gồm các khối tích chập

mới như Fused-MBConv. Cuối cùng, các tác giả đã thu được kiến trúc EfficientNetV2 nhanh hơn nhiều so với các mô hình hiện đại trước đây và mới hơn, đồng thời nhỏ hơn nhiều (lên tới 6,8 lần). Điều này được thể hiện trong hình dưới.



Hình 21: Parameter efficiency

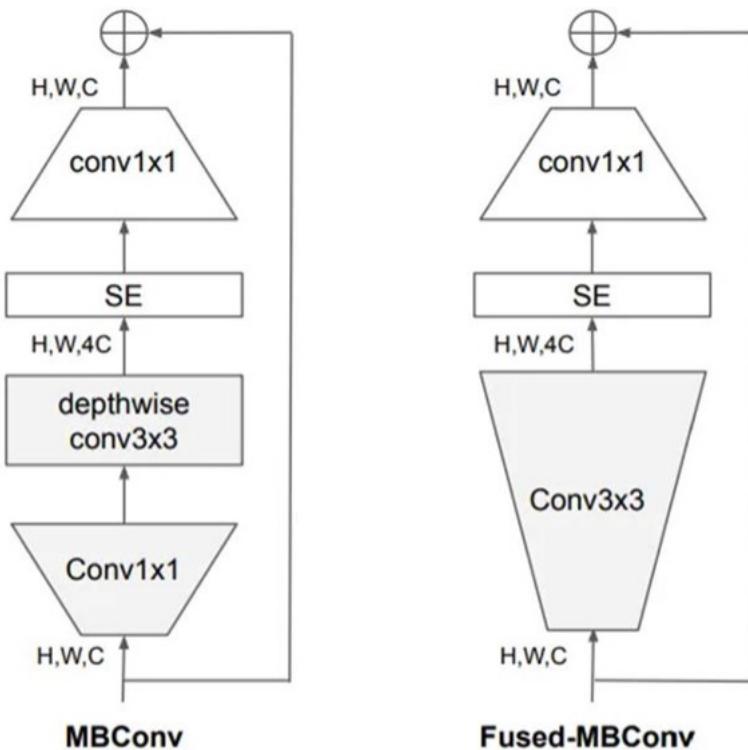
Hình Parameter efficiency cho thấy rõ ràng rằng EfficientNetV2 có 24 triệu tham số, trong khi Vision Transformer (ViT) có 86 triệu tham số. Phiên bản V2 cũng có gần một nửa thông số của EfficientNet ban đầu. Mặc dù nó làm giảm đáng kể kích thước tham số, nhưng nó vẫn duy trì độ chính xác tương tự hoặc cao hơn so với các mô hình khác trên bộ dữ liệu ImageNet. Ngoài ra cũng thực hiện progressive learning, đây là một phương pháp để tăng dần kích thước hình ảnh cùng với các quy định như bỏ học và tăng cường dữ liệu. Phương pháp này tiếp tục tăng tốc đào tạo. EfficientNets thường đào tạo nhanh hơn các mô hình CNN lớn khác. Tuy nhiên, khi độ phân giải hình ảnh lớn được sử dụng để huấn luyện các mô hình (mô hình B6 hoặc B7), quá trình huấn luyện diễn ra chậm. Điều này là do các mô hình EfficientNet lớn hơn yêu cầu kích thước hình ảnh lớn hơn để có được kết quả tối ưu và khi sử dụng hình ảnh lớn hơn, kích thước lô cần phải được hạ xuống để phù hợp với những hình ảnh này trong bộ nhớ GPU/TPU, khiến quá trình tổng thể chậm lại.

Trong các lớp đầu tiên của kiến trúc mạng, các lớp tích chập theo chiều sâu (MBConv) hoạt động chậm. Các lớp tích chập theo chiều sâu thường có ít tham số hơn các lớp

tích chập thông thường, nhưng vấn đề là chúng không thể tận dụng triệt để các modern accelerator. Để khắc phục vấn đề này, EfficientNetV2 sử dụng kết hợp MBConv và Fused MBConv để đào tạo nhanh hơn mà không cần tăng các tham số. Tỷ lệ bằng nhau được áp dụng cho chiều cao, chiều rộng và độ phân giải hình ảnh để tạo các mô hình EfficientNet khác nhau từ B0 đến B7. Tỷ lệ bằng nhau của tất cả các lớp này không phải là tối ưu. Ví dụ: nếu độ sâu được chia tỷ lệ 2, tất cả các khối trong mạng sẽ được tăng tỷ lệ 2 lần, làm cho mạng trở nên rất lớn/sâu. Có thể tối ưu hơn nếu chia tỷ lệ một khối hai lần và khối kia 1,5 lần (tỷ lệ không đồng đều), để giảm kích thước mô hình trong khi vẫn duy trì độ chính xác tốt.

- EfficientNetV2:

Như đã đề cập ở trên, khái MBConv thường không thể tận dụng triệt để các modern accelerator. Các lớp Fused-MBConv có thể sử dụng tốt hơn các trình tăng tốc máy chủ/thiết bị di động.



Hình 22: Kiến trúc 2 khối MBCov và Fused MBCov

Lớp MBConv lần đầu tiên được giới thiệu trong MobileNets. Như đã thấy trong hình trên, sự khác biệt duy nhất giữa cấu trúc của MBConv và Fused-MBConv là hai khái cuối cùng. Trong khi MBConv sử dụng tích chập theo chiều sâu (3x3) sau là lớp tích chập 1x1, thì Fused-MBConv thay thế/kết hợp hai lớp này bằng lớp tích chập 3x3 đơn giản. Các lớp MBConv được hợp nhất có thể giúp đào tạo nhanh hơn chỉ với một lượng nhỏ tham số tăng lên, nhưng nếu nhiều khái trong số này được sử dụng, nó có thể làm chậm quá trình đào tạo với nhiều tham số được thêm vào. Để khắc phục vấn đề này, các tác giả đã chuyển

cả MBCConv và Fused-MBCConv trong tìm kiếm kiến trúc neural, tự động quyết định sự kết hợp tốt nhất của các khối này để có hiệu suất và tốc độ đào tạo tốt nhất.

Việc tìm kiếm kiến trúc neural được thực hiện để cùng nhau tối ưu hóa độ chính xác, hiệu quả tham số và hiệu quả đào tạo. Mô hình EfficientNet được sử dụng làm xương sống và quá trình tìm kiếm được tiến hành với các lựa chọn thiết kế khác nhau, chẳng hạn như — khối tích chập, số lớp, kích thước bộ lọc, tỷ lệ mở rộng, v.v. Gần 1000 mô hình là mẫu và được đào tạo trong 10 epochs và kết quả của chúng được so sánh. Mô hình được tối ưu hóa tốt nhất về độ chính xác, thời gian bước đào tạo và kích thước tham số được chọn làm mô hình cơ sở cuối cùng cho EfficientNetV2. Hình dưới cho thấy kiến trúc mô hình cơ sở của mô hình EfficientNetV2 (EfficientNetV2-S). Mô hình chứa các lớp Fused-MBCConv lúc đầu nhưng sau đó chuyển sang các lớp MBCCConv. Để so sánh, chúng tôi cũng đã chỉ ra kiến trúc mô hình cơ sở cho bài báo EfficientNet trước đó trong Hình 9. Phiên bản trước chỉ có các lớp MBCCConv và không có các lớp Fused-MBCConv. EfficientNetV2-S cũng có tỷ lệ mở rộng nhỏ hơn so với EfficientNet-B0. EfficientNetV2 không sử dụng bộ lọc 5x5 và chỉ sử dụng bộ lọc 3x3.

Sau khi có được mô hình EfficientNetV2-S, nó sẽ được mở rộng quy mô để có được các mô hình EfficientNetV2-M và EfficientNetV2-L. Một phương pháp chia tỷ lệ hỗn hợp đã được sử dụng, tương tự như EfficientNet, nhưng một số thay đổi khác đã được thực hiện để làm cho các mô hình nhỏ hơn và nhanh hơn

Đầu tiên, kích thước hình ảnh tối đa được giới hạn ở 480x480 pixel để giảm mức sử dụng bộ nhớ GPU/TPU, do đó tăng tốc độ đào tạo.

Thứ hai, nhiều lớp hơn đã được thêm vào các giai đoạn sau để tăng dung lượng mạng mà không làm tăng nhiều chi phí thời gian chạy.

Kích thước hình ảnh lớn hơn thường có xu hướng cho kết quả đào tạo tốt hơn nhưng tăng thời gian đào tạo. Một số bài báo trước đây đã đề xuất kích thước hình ảnh thay đổi linh hoạt, nhưng nó thường dẫn đến mất độ chính xác trong đào tạo.

Các tác giả của EfficientNetV2 cho thấy rằng khi kích thước hình ảnh được thay đổi linh hoạt trong khi đào tạo mạng, do đó, việc chuẩn hóa cũng nên được thay đổi tương ứng. Thay đổi kích thước hình ảnh, nhưng vẫn giữ nguyên chuẩn hóa dẫn đến mất độ chính xác. Hơn nữa, các mô hình lớn hơn đòi hỏi sự chính quy hóa nhiều hơn các mô hình nhỏ hơn.

Các tác giả kiểm tra giả thuyết của họ bằng cách sử dụng các kích thước hình ảnh khác nhau và các phần mở rộng khác nhau. Như đã thấy trong dưới, khi kích thước hình ảnh nhỏ, phần mở rộng yếu hơn sẽ cho kết quả tốt hơn, nhưng khi kích thước hình ảnh lớn, phần tăng cường mạnh hơn sẽ cho kết quả tốt hơn.

Cân nhắc giả thuyết này, các tác giả của EfficientNetV2 đã sử dụng phương pháp Progressive Learning with Adaptive Regularization. Ý tưởng rất đơn giản. Trong các bước đầu tiên, mạng đã được đào tạo về hình ảnh nhỏ và regularization yếu. Điều này cho phép mạng học các tính năng nhanh chóng. Sau đó, kích thước hình ảnh được tăng dần và các

regularizations cũng vậy. Điều này làm cho mạng khó học. Nhìn chung, phương pháp này cho độ chính xác cao hơn, tốc độ đào tạo nhanh hơn và ít trang bị thừa hơn.

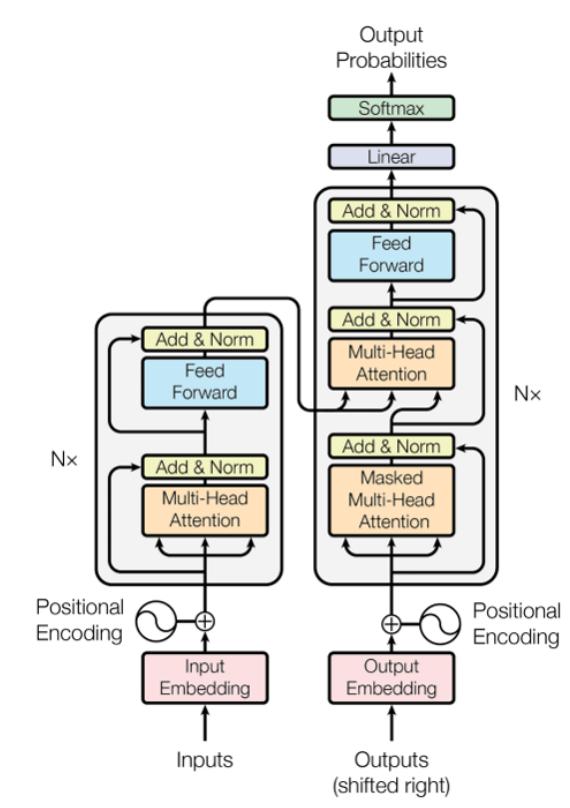
Kích thước hình ảnh ban đầu và tham số chuẩn hóa do người dùng xác định. Linear interpolation sau đó được áp dụng để tăng kích thước hình ảnh và chuẩn hóa sau một giai đoạn cụ thể (M), như thể hiện trong hình dưới. Điều này được giải thích trực quan hơn trong cuối cùng. Khi số lượng epochs tăng kích thước hình ảnh và các phần mở rộng cũng tăng dần. EfficientNetV2 sử dụng ba loại regularization khác nhau — Dropout, RandAugment và Mixup.

Decoder Transformer

- Transformer:

Transformer là một mô hình học sâu được giới thiệu năm 2017, được dùng chủ yếu ở lĩnh vực xử lý ngôn ngữ tự nhiên (NLP). Đây được coi là mô hình mạng học sâu hiện đại và mang lại hiệu quả cao hiện nay (state-of-the-art).

Sau khi thu được những đặc trưng ảnh dưới dạng các ma trận image embedding thông qua encoder, chúng em sẽ sử dụng kiến trúc decoder của mô hình mạng học sâu Transformer để tiến hành tạo câu caption cho ảnh.



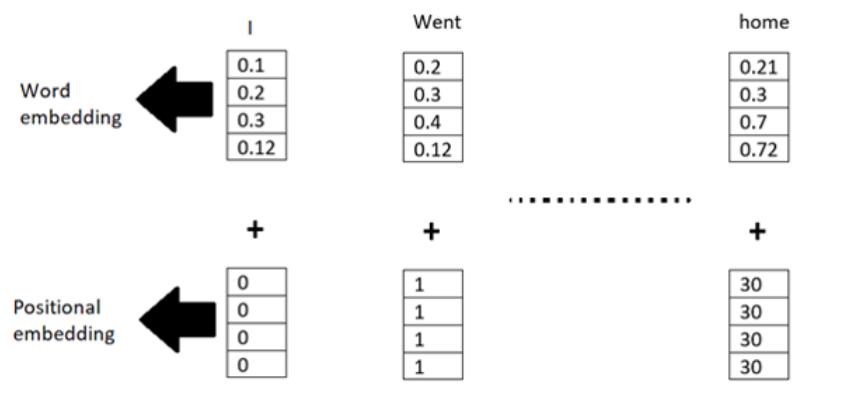
Hình 23: Kiến trúc Transformer

- Positional Encoding:

Positional Encoding là một kỹ thuật được sử dụng để đưa thông tin vị trí vào input embedding của mô hình mạng nơ-ron. Mục đích là cung cấp cho mô hình thông tin về thứ

tự và vị trí của các từ trong câu. Điều này rất quan trọng vì các neural networks thường hoạt động trên các vectơ có kích thước cố định và chúng không có ý nghĩa vốn có về thứ tự của các phần tử trong chuỗi đầu vào. Do đó, nếu không có positional encoding, mô hình sẽ gặp khó khăn trong việc phân biệt giữa các chuỗi khác nhau chứa các từ giống nhau theo các thứ tự khác nhau.

Bằng việc cộng thêm positional encoding vào input embedding sẽ giúp mô hình hiểu rõ hơn về cấu trúc và ý nghĩa của chuỗi văn bản đầu vào.



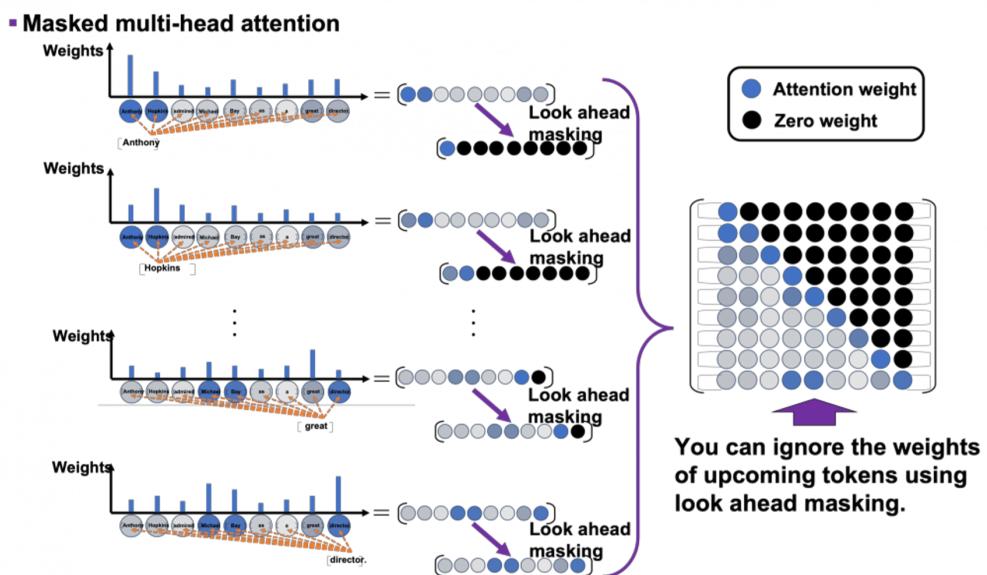
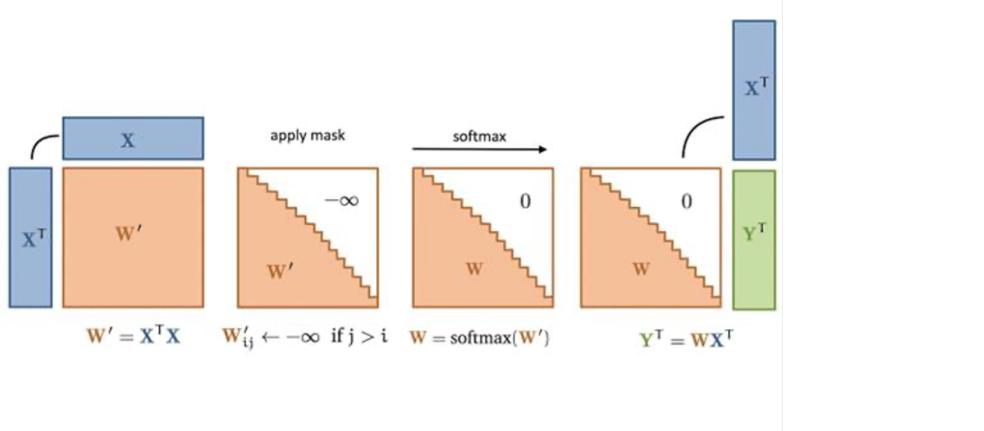
Hình 24: Positional embedding và word embedding

- Masked Multi-Head Attention:

Masked Multi-Head Attention là một cơ chế attention được sử dụng trong kiến trúc mạng neural sequence-to-sequence. Mục đích là cho phép decoder chỉ tập trung vào các mã thông báo đã tạo trước đó thay vì toàn bộ chuỗi đầu vào. Cơ chế Masked Multi-Head Attention hoạt động như sau:

1. Đầu vào của cơ chế là một chuỗi các embeddings, biểu thị các mã thông báo đầu ra đã tạo ra trước đó.
2. Các embeddings được chuyển đổi bằng ba phép chiếu tuyến tính riêng biệt, tạo ra các vectơ truy vấn, vectơ chìa khóa và vectơ giá trị cho mỗi mã thông báo. Những vectơ này sau đó được chia thành nhiều head khác nhau, cho phép mô hình tập trung vào các phần khác nhau của chuỗi.
3. Các điểm attention giữa các vectơ query và key được tính toán cho mỗi head, sử dụng cơ chế scaled dot-product attention. Các điểm attention này sau đó được sử dụng để trọng số hóa các vectơ giá trị, và các giá trị được trọng số này kết hợp để tạo ra một tập hợp các đầu ra attention.
4. Các đầu ra attention được nối lại và thông qua một phép chiếu tuyến tính khác, tạo ra đầu ra cuối cùng của cơ chế.

Trong quá trình huấn luyện, chuỗi đầu vào được che để ngăn decoder “nhìn thấy” các tokens kết quả chưa được tạo ra. Điều này được thực hiện bằng cách đặt các điểm attention cho các mã thông báo trong tương lai thành một giá trị âm vô cực.



Hình 25: Quá trình huấn luyện Masked Multi-Head Attention

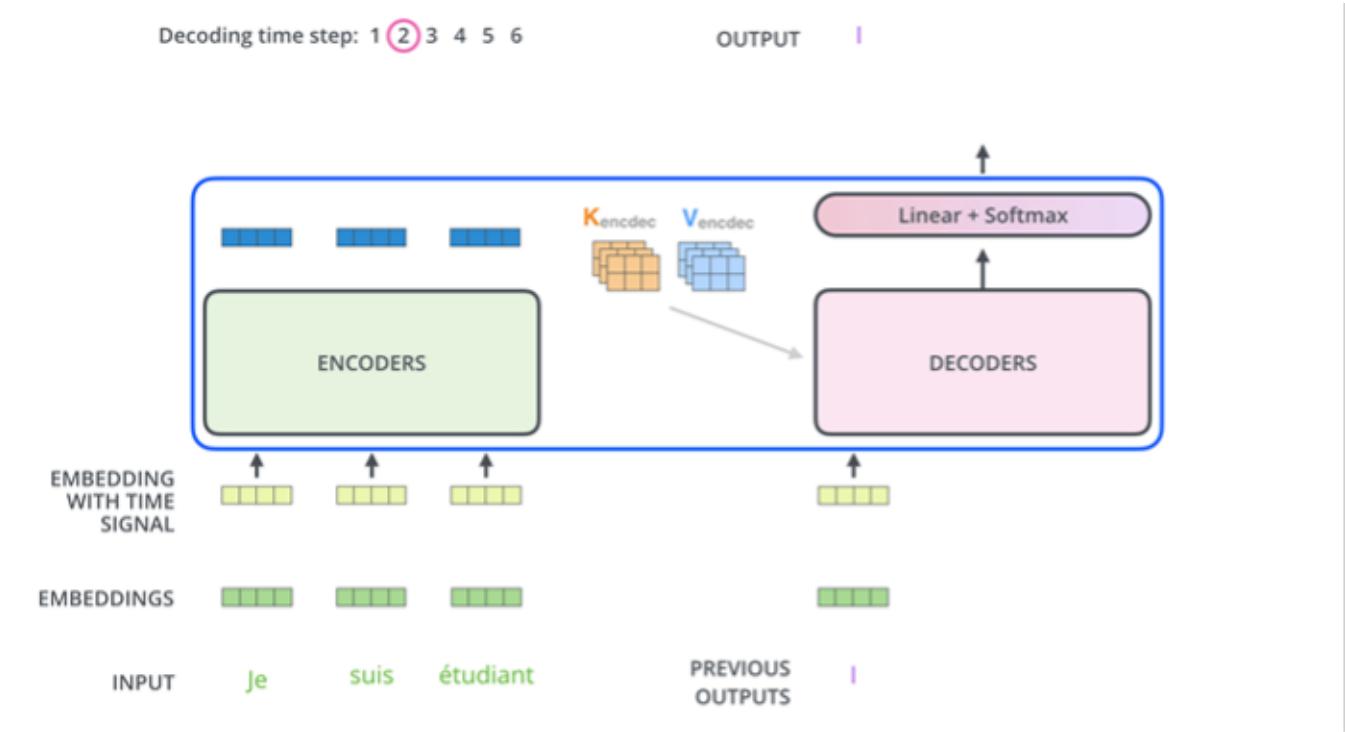
- Multi-Head Attention:

Multi-Head Attention trong decoder Transformer được sử dụng để tạo ra các từ trong câu mô tả ảnh. Cụ thể, Multi-Head Attention được sử dụng để tính toán độ quan trọng của các từ trong câu mô tả ảnh đối với các hình ảnh đã được encoder. Việc này giúp mô hình tập trung vào các phần khác nhau của hình ảnh để tạo ra các từ mô tả chính xác hơn trong câu caption.

Để thực hiện Multi-Head Attention, đầu vào của cơ chế attention sẽ bao gồm các vector truy vấn (query) và các vector giá trị (value). Các vector truy vấn được tạo ra bằng cách lấy đầu ra của lớp xử lý tuyến tính trước đó (Masked Multi-Head Attention) và truyền qua một lớp xử lý tuyến tính khác để tạo ra các vector truy vấn (Add & Norm). Các vector giá trị được tạo ra bằng cách lấy đầu ra của lớp encoder hình ảnh và truyền qua một lớp xử lý tuyến tính khác để tạo ra các vector giá trị (image embedding).

Sau đó, các vector query và value được chia thành nhiều head và truyền vào hàm attention. Các điểm attention giữa các vector query và value được tính toán cho mỗi head, và các giá trị được trọng số này kết hợp để tạo ra một tập hợp các đầu ra attention. Các đầu ra attention này sau đó được nối lại và thông qua một lớp xử lý tuyến tính khác để tạo ra

các đại diện cho các từ trong câu mô tả ảnh.



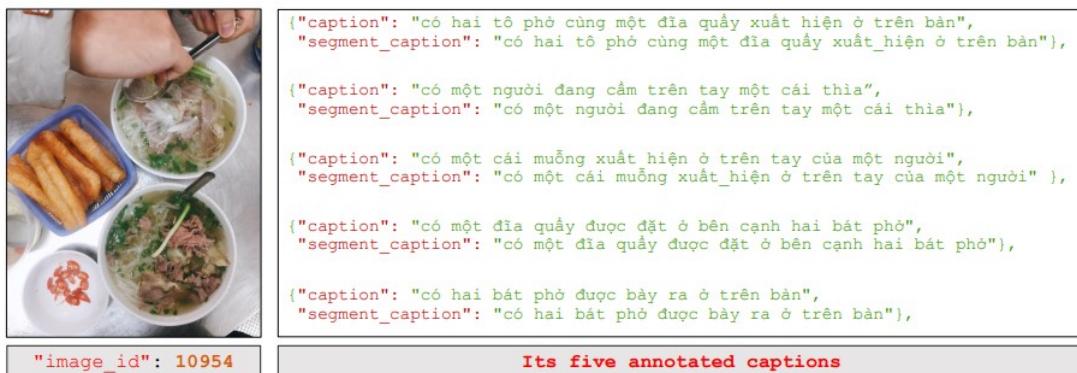
Hình 26: Quy trình hoạt động Multi-Head Attention

CHƯƠNG 3. THỰC NGHIỆM

3.1 Dataset

Trong bài báo cáo này, nhóm em sử dụng bộ dữ liệu KTVIC (Knowledge Technology Lab's Vietnamese Image Captioning) để huấn luyện mô hình image captioning cho ngữ cảnh tiếng Việt.

KTVIC là bộ dữ liệu chú thích hình ảnh được phát triển với trọng tâm là lĩnh vực đời sống và các hoạt động hàng ngày. Các hình ảnh trong bộ dữ liệu này được lấy từ bộ UIT-EVJVQA và mỗi ảnh được chú thích với năm câu mô tả khác nhau, tạo ra tổng cộng 21.635 câu chú thích chất lượng cao. Điều này giúp mô hình có thể học được sự đa dạng và phong phú của ngữ cảnh hình ảnh trong đời sống thực tế. So với các bộ dữ liệu hiện có như VieCap4H, KTVIC nổi bật ở sự đa dạng về đối tượng và cảnh vật trong hình ảnh, cũng như số lượng câu chú thích cho mỗi hình ảnh, giúp nâng cao chất lượng và độ chính xác của mô hình chú thích hình ảnh. Việc sử dụng KTVIC trong bài toán image captioning sẽ đóng góp quan trọng vào việc cải thiện mô hình chú thích hình ảnh tiếng Việt.



Hình 27: Ví dụ về 1 sample trong bộ dữ liệu KTVIC, trong đó mỗi hình ảnh kèm 5 câu caption.

3.2 Độ đo

Các độ đo phổ biến dùng để đánh giá mô hình image captioning có thể được chia thành hai nhóm chính: NLP-based metrics và image-specific metrics. Trong nhóm NLP-based metrics, các độ đo như BLEU và ROUGE thường được sử dụng, trong khi CIDEr là độ đo phổ biến trong nhóm image-specific metrics. Các độ đo này giúp đánh giá chất lượng mô hình image captioning, mỗi loại đóng góp một góc nhìn khác nhau, từ đó cung cấp cái nhìn toàn diện về hiệu quả của mô hình.

3.2.1 BLEU

BLEU (Bilingual Evaluation Understudy) là một trong những độ đo phổ biến được sử dụng để đánh giá chất lượng của các mô hình tạo văn bản, trong đó gồm hệ thống tạo caption cho ảnh (image captioning). BLEU đo lường mức độ tương đồng giữa các câu caption do mô

hình tạo ra (candidate captions) và các câu tham chiếu được viết bởi con người (reference captions).

Mặc dù BLEU không phải là một thước đo hoàn hảo, nhưng nó có nhiều ưu điểm như tính toán nhanh chóng, chi phí thấp, không phụ thuộc vào ngôn ngữ và quan trọng nhất là có mối tương quan cao với đánh giá của con người. Điều này làm cho BLEU trở thành một công cụ quan trọng trong việc đánh giá chất lượng của các hệ thống tạo văn bản tự động.

BLEU hoạt động dựa trên việc so sánh các n-gram (chuỗi liên tiếp của n từ) giữa câu caption dự đoán và các câu tham chiếu. Điểm số BLEU càng cao thì câu dự đoán càng gần với câu tham chiếu.

BLEU tính toán trên các thành phần chính:

- Precision của n-gram: Xác định tỉ lệ n-gram trong câu dự đoán xuất hiện trong câu tham chiếu.
- Brevity Penalty (BP): Một hệ số phạt được áp dụng nếu câu dự đoán quá ngắn so với câu tham chiếu, để tránh việc mô hình tạo ra những câu cực ngắn nhưng có độ chính xác cao.

Công thức tính BLEU Score như sau:

$$\text{BLEU} = BP \cdot \exp \left(\sum_{n=1}^N w_n \log p_n \right)$$

Trong đó, Brevity Penalty (BP) được tính như sau:

$$BP = \begin{cases} 1 & \text{nếu } c > r, \\ \exp \left(1 - \frac{r}{c} \right) & \text{nếu } c \leq r, \end{cases}$$

Trong đó:

p_n : Precision của n -gram,

w_n : Trọng số của mỗi n -gram (thường $w_n = \frac{1}{N}$),

c : Độ dài của câu dự đoán,

r : Độ dài trung bình của các câu tham chiếu.

3.2.2 ROUGE

ROUGE (Recall-Oriented Understudy for Gisting Evaluation) là một độ đo thường được sử dụng để đánh giá hiệu quả của các mô hình tạo văn bản, bao gồm việc tạo caption cho ảnh (image captioning). ROUGE kiểm tra mức độ trùng khớp giữa các cụm từ trong câu do mô hình tạo ra (candidate captions) và các câu tham chiếu được viết bởi con người (reference captions), từ đó đánh giá khả năng bao phủ nội dung.

ROUGE không hoàn hảo, nhưng nó nổi bật nhờ tính linh hoạt, hỗ trợ nhiều biến thể để phân tích đa chiều, và quy trình tính toán đơn giản. Nhờ đó, ROUGE trở thành một công cụ hữu ích để đánh giá chất lượng của các hệ thống tạo văn bản tự động, đặc biệt trong các trường hợp cần so sánh ý nghĩa và nội dung tổng thể giữa các văn bản.

Công thức tính ROUGE:

$$\text{ROUGE-N} = \frac{\text{Số lượng n-gram trùng khớp}}{\text{Tổng số n-gram trong câu tham chiếu}}$$

3.2.3 CIDEr

CIDEr (Consensus-based Image Description Evaluation) là một độ đo được thiết kế đặc biệt để đánh giá chất lượng các mô hình tạo caption cho ảnh (image captioning). CIDEr được phát triển nhằm cải thiện quá trình đánh giá các câu chú thích do mô hình tạo ra, dựa trên mức độ tương đồng không chỉ với một mà với nhiều câu tham chiếu. Đồng thời, CIDEr cũng xem xét các yếu tố ngữ nghĩa và sự phong phú trong việc mô tả hình ảnh.

CIDEr là công cụ hữu ích trong việc đánh giá các mô hình tạo chú thích, đặc biệt khi các câu chú thích cần phải dài hoặc sáng tạo, nơi các độ đo như BLEU hoặc ROUGE có thể không đủ khả năng đánh giá toàn diện tính phong phú của ngữ nghĩa. Để có cái nhìn đầy đủ về chất lượng mô hình, CIDEr thường được sử dụng kết hợp với các độ đo khác.

CIDEr đánh giá chất lượng chú thích hình ảnh dựa trên sự đồng thuận giữa câu dự đoán và các câu tham chiếu. Điểm số CIDEr được tính dựa trên các yếu tố chính:

- Tính chất n-gram: Đánh giá sự trùng khớp n-gram giữa câu dự đoán và câu tham chiếu.
- Sự phù hợp với nhiều tham chiếu: Đánh giá sự trùng khớp với nhiều câu tham chiếu để giảm thiên lệch.
- Khả năng bao phủ: Sử dụng trọng số cho các n-gram xuất hiện thường xuyên trong câu tham chiếu.
- Phù hợp với ngữ nghĩa: Đánh giá mức độ phong phú và chính xác trong mô tả ngữ nghĩa của câu chú thích.

$$\text{CIDEr} = \frac{1}{N} \sum_{i=1}^N \text{IDF}_i \cdot \text{TF-IDF}_i$$

Trong đó:

N : Số lượng n-gram trong câu.

TF-IDF_i : Tần suất của n-gram thứ i trong câu dự đoán với trọng số TF-IDF.

IDF_i : Tỉ lệ đảo ngược tần suất tài liệu (Inverse Document Frequency), cho biết tầm quan trọng

3.3 Training

Model	Môi trường Training	Tham số Training	Thời gian
SWIN + BART	Kaggle (GPU T4x2)	Epoch: 8, Batchsize: 32	8 hours
GRIT	Kaggle (GPU T4x2)	Epoch: 20, Batchsize: 8	8 hours
EfficientNet_v2 + Transformer	GoogleColab (L4 GPU)	Epoch: 20, Batchsize: 64	30 mins

Bảng 1: Môi trường và thời gian Training của từng Model.

Bảng trên mô tả thông tin về ba mô hình nhóm đã sử dụng cùng với thông tin về môi trường huấn luyện và các tham số liên quan.

- Mô hình đầu tiên, SWIN + BART, được huấn luyện trên Kaggle với GPU T4x2, sử dụng batch size 32 và huấn luyện trong 8 epoch. Thời gian huấn luyện cho mô hình này là 8 giờ.
- Mô hình thứ hai, GRIT, cũng được huấn luyện trên Kaggle (GPU T4x2) với batch size 8 và 20 epoch, kéo dài 8 giờ.
- Cuối cùng, EfficientNet_v2 + Transformer được huấn luyện trên Google Colab sử dụng GPU L4, với batch size 64 và 20 epoch, nhưng chỉ mất 30 phút để hoàn thành.

Bảng này chỉ ra sự khác biệt về thời gian huấn luyện và các tham số giữa các mô hình và môi trường, đồng thời phản ánh hiệu quả tính toán và độ phức tạp của các mô hình.

3.4 Kết quả

Model	BLEU-4	ROUGE	CIDEr
SWIN + BART	0.382	0.562	1.229
GRIT	0.417	0.609	1.281
EfficientNet_v2 + Transformer	0.257	0.48	0.606

Bảng 2: Bảng kết quả đánh giá các Model theo 3 độ đo BLEU-4, ROUGE và CIDEr.

Bảng trên cho thấy kết quả của ba mô hình khác nhau (SWIN + BART, GRIT, và EfficientNet_v2 + Transformer) trong bài toán Image Captioning dựa trên ba độ đo: BLEU-4, ROUGE và CIDEr.

- SWIN + BART có BLEU-4 là 0.382, ROUGE là 0.562, và CIDEr là 1.229, cho thấy mô hình này có hiệu suất khá tốt trong việc tạo chú thích, với sự tương đồng về ngữ nghĩa và cấu trúc tương đối cao so với các câu tham chiếu.
- GRIT đạt BLEU-4 cao hơn một chút là 0.417 và ROUGE score là 0.609 cùng CIDEr là 1.281 đều cao hơn so với SWIN + BART. Điều này cho thấy GRIT có khả năng tạo ra các câu chú thích có sự tương đồng tốt hơn với các câu tham chiếu và đánh giá ngữ nghĩa hiệu quả hơn. Mô hình GRIT được đánh giá cao nhờ vào việc sử dụng hai loại feature: grid feature và region feature trong khi 2 mô hình còn lại chỉ dựa vào một loại đặc trưng,

điều này giúp mô hình học được nhiều đặc điểm chi tiết hơn về hình ảnh, từ đó cải thiện khả năng mô tả ngữ nghĩa.

- EfficientNet_v2 + Transformer có BLEU-4 khá thấp 0.257, ROUGE là 0.48 và CIDEr là 0.606, cho thấy mô hình này có hiệu suất thấp hơn đáng kể so với hai mô hình còn lại. Điều này có thể chỉ ra rằng mặc dù mô hình này có cấu trúc hiệu quả, nhưng khả năng tạo ra các chú thích tương đồng với tham chiếu và đánh giá ngữ nghĩa không mạnh mẽ như GRIT hay SWIN + BART.

Tóm lại, GRIT là mô hình tốt nhất trong ba mô hình, nhờ vào việc sử dụng hai loại đặc trưng giúp mô hình học sâu hơn và tạo ra các câu caption chính xác hơn.

CHƯƠNG 4. KẾT LUẬN, HƯỚNG PHÁT TRIỂN

4.1 Kết luận

Trong bài báo cáo này, nhóm em đã nghiên cứu, thực nghiệm và đánh giá hiệu quả của ba mô hình trong bài toán Image Captioning, bao gồm: SWIN + BART, GRIT, và EfficientNet_v2 + Transformer. Mỗi mô hình đều có những ưu điểm riêng, và kết quả thực nghiệm cho thấy GRIT là mô hình nổi bật nhất, với khả năng tạo ra các chú thích có mức độ tương đồng cao với các câu tham chiếu, thể hiện qua các độ đo BLEU-4, ROUGE và CIDEr. Thành công của GRIT đạt được nhờ việc kết hợp hai loại đặc trưng: grid feature và region feature, giúp mô hình học được nhiều chi tiết hơn về hình ảnh, từ đó nâng cao chất lượng chú thích.

SWIN + BART cũng đạt được các kết quả khá quan, với hiệu suất ổn định, nhưng mô hình này chủ yếu dựa vào kiến trúc transformer để xử lý thông tin hình ảnh. Mặc dù cho kết quả khá tốt, nhưng SWIN + BART không thể đạt được mức độ chi tiết và chính xác tương tự như GRIT. Trong khi đó, EfficientNet_v2 + Transformer, dù có cấu trúc đơn giản và hiệu quả về mặt tính toán, nhưng lại gặp khó khăn trong việc mô tả đầy đủ các đặc điểm của hình ảnh, thể hiện qua các chỉ số đánh giá thấp hơn so với GRIT và SWIN + BART.

Các kết quả thực nghiệm này nhấn mạnh rằng việc sử dụng các đặc trưng phong phú và kết hợp nhiều loại dữ liệu từ hình ảnh là yếu tố quan trọng để nâng cao chất lượng mô hình tạo chú thích. Tuy nhiên, hiệu suất của mô hình không chỉ phụ thuộc vào việc sử dụng các đặc trưng mạnh mẽ mà còn vào việc tối ưu hóa mô hình sao cho phù hợp với tài nguyên tính toán và thời gian huấn luyện.

Tóm lại, dù GRIT có hiệu suất vượt trội nhờ vào việc sử dụng các đặc trưng phong phú, nhưng việc triển khai mô hình này yêu cầu tài nguyên tính toán mạnh mẽ và thời gian huấn luyện đáng kể. Các mô hình như SWIN + BART và EfficientNet_v2 + Transformer, mặc dù có một số hạn chế nhất định, nhưng vẫn có thể được áp dụng hiệu quả cho các bài toán chú thích hình ảnh khi tài nguyên tính toán có hạn.

4.2 Hướng phát triển

Hướng phát triển tiếp theo có thể tập trung vào việc tối ưu hóa các mô hình hiện tại để giảm yêu cầu tài nguyên tính toán mà vẫn duy trì được hiệu suất cao. Các kỹ thuật như pruning, quantization, và transfer learning có thể được áp dụng để giảm bớt khối lượng tính toán mà không làm giảm chất lượng mô hình, giúp mô hình hoạt động hiệu quả hơn trên các hệ thống với tài nguyên hạn chế. Việc cải tiến các thuật toán học sâu để tăng cường khả năng tận dụng tài nguyên tính toán hiện có sẽ là một yếu tố quan trọng trong việc mở rộng khả năng áp dụng các mô hình chú thích hình ảnh vào các ứng dụng thực tế.

Bên cạnh đó, việc kết hợp thêm các đặc trưng như semantic features (đặc trưng ngữ nghĩa) và spatial features (đặc trưng không gian) có thể nâng cao khả năng hiểu ngữ nghĩa của hình ảnh, từ đó giúp mô hình tạo ra các chú thích chính xác hơn và có độ chi tiết cao hơn. Những cải tiến này sẽ giúp mô hình không chỉ nhận diện được các đối tượng trong hình ảnh

mà còn hiểu được mối quan hệ giữa các đối tượng đó, từ đó tạo ra những chú thích phản ánh đầy đủ hơn về ngữ cảnh của hình ảnh.

Cùng với đó, việc áp dụng các phương pháp sáng tạo như Generative Adversarial Networks (GANs) để tạo ra các chú thích đa dạng và phong phú cũng sẽ giúp mô hình tiếp cận các bài toán chú thích hình ảnh một cách toàn diện hơn. Sử dụng GANs có thể thúc đẩy khả năng sáng tạo của mô hình trong việc tạo ra các chú thích tự nhiên, linh hoạt và khác biệt hơn, đồng thời duy trì sự chính xác trong việc mô tả hình ảnh. Thực hiện các nghiên cứu thêm về việc kết hợp GANs với các mô hình chú thích hình ảnh sẽ mở ra những hướng đi mới, giúp nâng cao cả chất lượng và tính sáng tạo của các mô hình chú thích trong tương lai.

CHƯƠNG 5. TRẢ LỜI CÂU HỎI

5.1 Grid feature là gì?

Grid feature là các đặc trưng được trích xuất từ toàn bộ hình ảnh thay vì chỉ tập trung vào các đối tượng cụ thể. Các đặc trưng này giúp mô hình hiểu được ngữ cảnh tổng thể của hình ảnh, thay vì chỉ tập trung vào từng đối tượng riêng lẻ, và chúng chủ yếu mô tả các mối quan hệ hoặc cấu trúc chung trong hình ảnh.

5.2 Tại sao phương pháp EfficientNetV2 chỉ train có 30p?

EfficientNetV2 sử dụng kết hợp giữa MBConv và FusedMBConv giúp giảm đáng kể số lượng tham số cần phải huấn luyện mà vẫn giữ được độ chính xác cao nên thời gian train chỉ tốn khoảng 30-45 phút.

5.3 Dataset cấu trúc như thế nào? Một ảnh có nhiều caption hay chỉ 1? 4000 ảnh dùng để train có quá ít không?

Dataset là bộ KTVIC gồm 4327 ảnh, ứng với mỗi ảnh là 5 caption khác nhau. Tỉ lệ train, valid, test nhóm chia 8:1:1.

Vì dataset của bài toán Image Captioning với ngôn ngữ tiếng việt cũng khá hạn chế, tài nguyên dùng để training cũng có hạn nên nhóm mình chỉ lấy 1 bộ để retrain lại các model để hiểu hơn về code cũng như kiến trúc model, cách train chứ model này áp dụng vào thực tế thì không có độ chính xác cao lắm.

5.4 Cơ chế attention của swin transformer cho nó lợi thế như thế nào so với transformer truyền thống trong bài toán liên quan đến ảnh?

Cơ chế attention của Swin Transformer mang lại lợi thế so với transformer truyền thống trong bài toán xử lý ảnh nhờ vào các đặc điểm sau:

- Window-based Attention: Swin chia ảnh thành các cửa sổ nhỏ, tính attention trong từng cửa sổ, giúp giảm độ phức tạp tính toán so với transformer truyền thống.
- Shifted Window: Cơ chế "dịch chuyển cửa sổ" giúp kết hợp thông tin từ các vùng biên, cải thiện khả năng học các mối quan hệ dài hạn giữa các vùng trong ảnh.
- Hiệu quả với ảnh độ phân giải cao: Swin có thể xử lý ảnh lớn mà không tốn quá nhiều tài nguyên tính toán, nhờ vào cơ chế cửa sổ linh hoạt.

Tóm lại, Swin Transformer xử lý ảnh hiệu quả hơn transformer truyền thống nhờ vào việc giảm độ phức tạp tính toán và học được các mối quan hệ giữa các vùng ảnh.

5.5 Mô hình có thể hiểu được các lẽ hội cụ thể của Việt Nam trong hình không?

Không vì trong dataset các caption không đề cập cụ thể đến tên các loại lẽ hội.

5.6 GRIT sử dụng phương pháp gì để kết hợp đặc trưng lại với nhau?

GRIT kết hợp đặc trưng grid và region thông qua một lớp Cross-Attention. Sau khi áp dụng self-attention, đầu ra sẽ được kết hợp với cả region features và grid features thông qua Multi-Head Attention (MHA). Trong lớp này, kết quả từ self-attention sẽ làm queries, trong khi region features và grid features sẽ là keys và values. MHA giúp mô hình học được sự liên kết giữa các từ trong câu và thông tin từ hình ảnh bằng cách tính attention song song với hai loại đặc trưng này. Sau đó, kết quả từ MHA sẽ được kết hợp với đầu ra của self-attention, chiếu lại thành vector có kích thước phù hợp, chuẩn hóa và cộng thêm với đầu ra ban đầu để qua lớp chuẩn hóa, tạo ra kết quả cuối cùng.

5.7 Model Transformer nếu dùng global attention thì có cho kết quả tốt hơn không?

Chỉ có phần Swin Transformer mới dùng local attention để giảm chi phí tính toán nhưng vẫn đảm bảo được độ chính xác, còn Model Transformer truyền thống thì vẫn dùng global attention.

5.8 Giải thích ý nghĩa các metric đánh giá.

BLEU: Tập trung vào precision, đo tỷ lệ các n-gram trong câu sinh ra trùng với câu tham chiếu. Mô hình có điểm BLEU cao nếu tạo ra nhiều n-gram trùng khớp với câu tham chiếu.

ROUGE: Tập trung vào recall, đo tỷ lệ các n-gram trong câu tham chiếu mà mô hình đã sinh ra. Điểm ROUGE cao cho thấy mô hình đã tạo ra nhiều n-gram quan trọng từ câu tham chiếu.

CIDEr: Dành cho bài toán image captioning, đánh giá mức độ tương đồng giữa chú thích và câu tham chiếu, chú trọng vào tầm quan trọng của từng từ trong ngữ cảnh hình ảnh.

5.9 Hệ thống này tạo caption bằng cách dựa vào các mẫu đã có hay hoàn toàn mang tính sáng tạo ?

Hệ thống tạo caption dựa trên các caption đã được training trong tập dataset.

5.10 Thời gian inference trung bình là bao lâu?

Thời gian inference trung bình tầm 0.3-0.5s.

5.11 GRIT viết tắt chữ gì?

GRIT viết tắt của chữ Grid- and Region-based Image captioning Transformer.

5.12 Phương pháp cho độ chính xác cao nhất, theo nhóm em điều gì tạo nên sự khác biệt so với các mô hình còn lại?

GRIT là mô hình có độ chính xác cao nhất bởi vì mô hình sử dụng hai loại feature: grid feature và region feature trong khi 2 mô hình còn lại chỉ dựa vào một loại đặc trưng, điều này giúp mô hình học được nhiều đặc điểm chi tiết hơn về hình ảnh, từ đó cải thiện khả năng mô tả ảnh.

5.13 Mô hình có thể đọc được chữ tiếng Việt? Ví dụ: trong ảnh có chữ tiếng Việt nó đọc được ko?

Mô hình này không phải mô hình tích hợp OCR nên không đọc được chữ tiếng Việt có trong ảnh.

Tài liệu

- [1] Van-Quang Nguyen, Masanori Suganuma, and Takayuki Okatani, *GRIT: Faster and Better Image captioning Transformer Using Dual Visual Features*, arXiv:2207.09666, 2022. <https://arxiv.org/pdf/2207.09666.pdf>
- [2] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, Bain-ing Guo, *Swin Transformer: Hierarchical Vision Transformer using Shifted Windows*, arXiv:2103.14030, 2021. <https://arxiv.org/pdf/2103.14030.pdf>
- [3] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, Jifeng Dai, *DEFORMABLE DETR: DEFORMABLE TRANSFORMERS FOR END-TO-END OBJECT DETECTION*, arXiv:2010.04159, 2020. <https://arxiv.org/pdf/2010.04159.pdf>
- [4] Matteo Stefanini, Marcella Cornia, Lorenzo Baraldi, Silvia Cascianelli, Giuseppe Fiameni, and Rita Cucchiara, *From Show to Tell: A Survey on Deep Learning-based Image Captioning*, arXiv:2107.06912, 2021. <https://arxiv.org/pdf/2107.06912.pdf>
- [5] Anh-Cuong Pham, Van-Quang Nguyen, Thi-Hong Vuong, Quang-Thuy Ha, *KTVIC: A VIETNAMESE IMAGE CAPTIONING DATASET ON THE LIFE DOMAIN*, arXiv:2401.08100, 2024. <https://arxiv.org/pdf/2401.08100.pdf>