

ĐẠI HỌC QUỐC GIA VIỆT NAM THÀNH PHỐ HỒ CHÍ MINH
ĐẠI HỌC BÁCH KHOA THÀNH PHỐ HỒ CHÍ MINH
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH



**BÁO CÁO
TIỂU LUẬN**

NGHIÊN CỨU PHƯƠNG PHÁP PHÂN LOẠI DỰA TRÊN MÔ TẢ VỚI TẬP DỮ LIỆU POKEMON

Ngành: Khoa học máy tính
Môn học: Xử lý ảnh số và thị giác máy tính

GIẢNG VIÊN HƯỚNG DẪN: TS. Nguyễn Đức Dũng

SINH VIÊN THỰC HIỆN: Đoàn Trần Cao Trí - 2010733

Thành phố Hồ Chí Minh, 12/2023

Danh sách hình ảnh

1.1	Hình minh họa: động lực của đề tài	3
2.1	Kiến trúc của mô hình Transformer [3]	6
2.2	Kiến trúc của mô hình Multi-Head Attention [3]	8
2.3	Kiến trúc của mô hình ViT [1]	9
3.1	Mô hình GIT - Generative Image-to-text Transformer	12
4.1	Tập dữ liệu Pokemon tạo bởi Gemini	13
4.2	Quá trình huấn luyện Yolov5	14
4.3	Trường hợp dự đoán đúng tên Pokemon	15
4.4	Trường hợp dự đoán sai tên Pokemon	15

Danh sách chỉ mục

1	Giới thiệu	3
1.1	Phát biểu vấn đề	3
1.2	Mục tiêu đề tài	3
1.3	Giới hạn đề tài	4
1.4	Cấu trúc đề tài	4
1.5	Github	4
2	Kiến thức nền tảng	5
2.1	Tập dữ liệu bài toán phân loại	5
2.1.1	Phân loại hình ảnh (Image Classification)	5
2.1.2	Phân loại câu chữ (Sequence Classification)	5
2.2	Kiến trúc mô hình Transformer	5
2.2.1	Mã hoá vị trí	7
2.2.2	Bộ mã hoá và bộ giải mã	7
2.2.2.1	Bộ mã hoá	7
2.2.2.2	Bộ giải mã	7
2.2.3	Cơ chế Attention	7
2.2.3.1	Cơ chế Self-Attention	7
2.2.3.2	Kiến trúc Multi-Head Attention	8
2.3	Phân loại hình ảnh với mô hình Mạng neuron tích chập (CNN)	8
2.4	Chú thích hình ảnh với mô hình Transformer	9
2.5	Phân loại câu chữ với mô hình Transformer	10
3	Phương pháp đề xuất	11
3.1	Chuẩn bị tập dữ liệu	11
3.1.1	Tập dữ liệu Pokemon	11
3.2	Chú thích hình ảnh với mô hình GIT	11
3.2.1	Ứng dụng phổ biến của GIT	11
3.3	Phân loại câu chữ với mô hình RoBERTa	12
4	Kết quả nghiên cứu	13
4.1	Tập dữ liệu Pokemon	13
4.2	Image Classification với YOLOVv5	13
4.3	Image Classification với Resnet50	13
4.4	Image Captioning với GIT model	14
4.5	Sequence Classification với Roberta model	14
4.6	Tổng hợp kết quả	14
5	Tổng kết đề tài	16

Chương 1

Giới thiệu

1.1 Phát biểu vấn đề

Thời đại thông tin bùng nổ, nhu cầu kiểm soát thông tin đáng bởi người dùng trong cộng đồng trở nên ngày càng quan trọng. Trong ngữ cảnh này, một trong những thách thức đặt ra là kiểm duyệt hình ảnh, nơi mà việc đặt vấn đề bằng hình ảnh trở thành khía cạnh quan trọng.

Ví dụ: một hình ảnh hiển thị người lính cầm súng trên chiến trường có thể được cấp một câu mô tả về súng, người lính, hành động bắn và sau đó được gán nhãn "chiến tranh" để phát đi cảnh báo. Ngược lại, một hình ảnh cổ vũ hòa bình cũng sẽ được mô tả những con chim bồ câu, hành động thả vì hòa bình và gán nhãn "hòa bình" tương ứng để đảm bảo sự đa dạng trong việc kiểm soát thông tin. Hình minh họa 1.1



(a) Biểu tượng hòa bình



(b) Biểu tượng chiến tranh

Hình 1.1: Hình minh họa: động lực của đề tài

Tuy nhiên, vấn đề đặt ra là các mô hình phân loại hình ảnh thường không thể bao quát được tất cả loại nhãn mà hình ảnh cần gán. Đồng thời, theo tư duy trực quan của con người, chúng ta có khả năng gán nhãn hình ảnh dựa vào mô tả và sự hiểu biết về nội dung của hình ảnh.

Do đó, nghiên cứu này tập trung vào phát triển một mô hình gán nhãn hình ảnh dựa trên mô tả từ hình ảnh để cải thiện khả năng kiểm soát thông tin.

1.2 Mục tiêu đề tài

Mục tiêu chính của đề tài này là thực hiện hai nhiệm vụ chính:

1. Mô hình mô tả hình ảnh

2. Mô hình phân loại câu văn mô tả
3. So sánh với mô hình CNN phân loại

1.3 Giới hạn đề tài

Để giới hạn phạm vi của đề tài, chúng tôi quyết định sử dụng tập dữ liệu chứa các thông tin cơ bản như hình ảnh, mô tả và nhãn. Ngoài ra, để thực hiện các thử nghiệm và đánh giá, chúng tôi tập trung vào việc xây dựng một tập dữ liệu về Pokemon để đảm bảo tính đa dạng và phức tạp trong quá trình nghiên cứu.

1.4 Cấu trúc đề tài

Cấu trúc của đề tài bao gồm các phần chính sau:

- Giới thiệu
- Kiến thức nền tảng
- Phương pháp đề xuất
- Kết quả nghiên cứu
- Tổng kết đề tài

1.5 Github

Toàn bộ code train và process data được đăng tại

<https://github.com/tri218138/Research-Image-Captioning-Classification>

Chương 2

Kiến thức nền tảng

2.1 Tập dữ liệu bài toán phân loại

Trong lĩnh vực máy học và trí tuệ nhân tạo, bài toán phân loại đóng vai trò quan trọng trong việc gán nhãn cho dữ liệu. Có hai dạng phổ biến của bài toán phân loại là phân loại hình ảnh (*image classification*) và phân loại câu chữ (*sequence classification*).

2.1.1 Phân loại hình ảnh (Image Classification)

Phân loại hình ảnh là bài toán nhận diện và gán nhãn cho một hình ảnh dựa trên nội dung của nó. Các ứng dụng thường thấy của phân loại hình ảnh bao gồm hệ thống nhận diện khuôn mặt, nhận diện đối tượng trong hình ảnh, và nhận diện chữ thể trong y học. Ví dụ, trong y học, phân loại hình ảnh có thể được sử dụng để xác định các bệnh lý từ hình ảnh chụp cắt lớp của cơ thể.

Hướng tiếp cận thường được áp dụng cho bài toán phân loại hình ảnh là sử dụng các mô hình học máy sâu như Convolutional Neural Networks (CNN). Các mô hình này được thiết kế để hiệu quả trong việc học đặc trưng từ dữ liệu hình ảnh và đưa ra dự đoán chính xác.

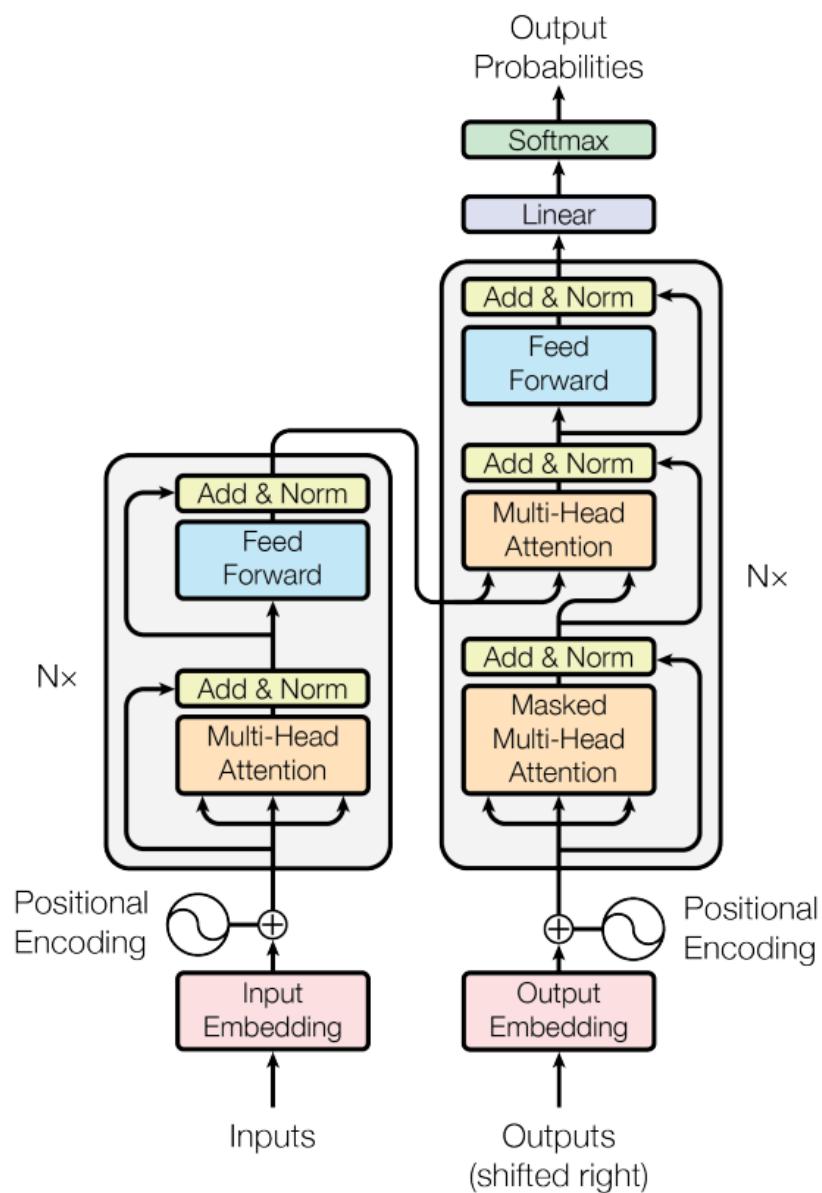
2.1.2 Phân loại câu chữ (Sequence Classification)

Phân loại câu chữ là bài toán nhận diện và gán nhãn cho một chuỗi dữ liệu, thường được áp dụng trong các tình huống như phân loại văn bản hoặc dự đoán chuỗi thời gian. Ứng dụng thường thấy của phân loại câu chữ là trong lĩnh vực xử lý ngôn ngữ tự nhiên, như phân loại cảm xúc trong các bình luận trực tuyến hoặc dự đoán ngôn ngữ người dùng.

Hướng tiếp cận cho bài toán phân loại câu chữ thường sử dụng các kiến trúc mô hình như Long Short-Term Memory (LSTM) hoặc Gated Recurrent Unit (GRU) để hiệu quả trong việc xử lý thông tin chuỗi và giữ lại thông tin quan trọng.

2.2 Kiến trúc mô hình Transformer

Transformer là một mô hình ngôn ngữ được Google phát triển và công bố vào năm 2017 trong bài báo "Attention is All You Need" [3]. Kiến trúc của mô hình này bao gồm hai thành phần chính là Bộ mã hóa (Encoder) và Bộ giải mã (Decoder). Ở đây, bộ mã hóa sẽ ánh xạ chuỗi ký tự đầu vào $x = (x_1, \dots, x_n)$ thành một chuỗi liên tục tương ứng $z = (z_1, \dots, z_n)$ và bộ giải mã sẽ sử dụng chuỗi này để tạo ra một chuỗi đầu ra $y = (y_1, \dots, y_m)$. Ở mỗi bước, mô hình tự động hồi quy [?]. sử dụng các ký hiệu được tạo trước đó làm đầu vào bổ sung khi tạo bước tiếp theo. Hình 2.1 mô tả kiến trúc của mô hình Transformer.



Hình 2.1: Kiến trúc của mô hình Transformer [3]

2.2.1 Mã hoá vị trí

Như chúng ta đã biết, máy tính không thể hiểu các câu văn được viết trực tiếp vào nó, thay vào đó, nó cần một hình thức biểu diễn thông tin để có thể biểu diễn các ký tự, câu văn thành dạng số để có thể hiểu được. Hình thức biểu diễn thông tin này được gọi là nhúng từ (word embedding) để có thể biểu diễn chuỗi ký tự ban đầu thành một véc-tơ nhúng (embedding). Các véc-tơ này sau đó được nối với nhau trở thành một ma trận hai chiều và được xử lý bởi các tiền trình tiếp theo của mô hình. Quá trình này rất phổ biến và thường được áp dụng ở các mô hình tuần tự (sequence to sequence model). Nhưng trong Transformer, có một vấn đề khác nảy sinh là word embedding không đủ thông tin để có thể biểu diễn vị trí của từ do cơ chế xử lý các từ song song của mô hình. Để giải quyết vấn đề này, các tác giả của mô hình Transformer đã giới thiệu cơ chế mã hóa mới có tên gọi là mã hóa vị trí (position encoding), với mục tiêu là mã hóa vị trí của các từ bằng một véc-tơ có kích thước bằng với kích thước của véc-tơ nhúng từ sẽ được cộng trực tiếp vào vectơ véc-tơ nhúng từ tương ứng. Dưới đây là công thức mã hóa vị trí được nêu trong bài báo.

$$PE_{(pos,2i)} = \sin(pos/10000^{2i/d_{model}})$$
$$PE_{(pos,2i+1)} = \cos(pos/10000^{2i/d_{model}})$$

Với pos là vị trí và i là kích thước.

2.2.2 Bộ mã hoá và bộ giải mã

2.2.2.1 Bộ mã hoá

Bộ mã hoá đóng vai trò mã hóa đầu vào thành một biểu diễn mới giàu thông tin hơn gọi là *véc-tơ ngữ cảnh* (context véc-tơ). Kiến trúc của bộ mã hóa được tạo nên bằng cách xếp chồng các lớp mã hóa với nhau (thường là 6 lớp), tạo thành một mạng truyền thẳng (Feedforward Neural Networks). Kiến trúc này cho phép các lớp xử lý đồng thời các từ, thay vì xử lý tuần tự giống các mô hình khác như LSTM. Trong mỗi bộ mã hóa lại có hai thành phần chính là tầng tập trung đa đầu (multi-head attention) và mạng truyền thẳng (feedforward network), ngoài ra còn có bỏ kết nối (skip connection) và lớp chuẩn hóa (normalization).

2.2.2.2 Bộ giải mã

Giống như bộ mã hóa, bộ giải mã cũng có kiến trúc xếp chồng với 6 lớp tạo thành mạng truyền thẳng. Mỗi lớp sẽ nhận thông tin đầu vào từ bộ mã hóa để thực hiện tác vụ giải mã vectơ của câu nguồn thành đầu ra tương ứng. Kiến trúc của những lớp này cơ bản giống với những lớp trong bộ mã hóa, ngoại trừ có thêm một lớp chú ý đa đầu (multi-head attention) nằm ở giữa đóng vai trò học mối quan hệ tương quan giữa các từ đang được dịch với các từ trong văn bản gốc. (Query), Ket (value) chứa định danh của

2.2.3 Cơ chế Attention

2.2.3.1 Cơ chế Self-Attention

Tự chú ý (Self-Attention) là một cơ chế cho phép mô hình xem xét những thông tin có liên quan tới một từ trong ngữ cảnh văn bản (thường là những từ khác trong chính văn bản đó) để tìm ra manh mối có thể giúp dẫn đến mã hóa tốt hơn cho từ này.

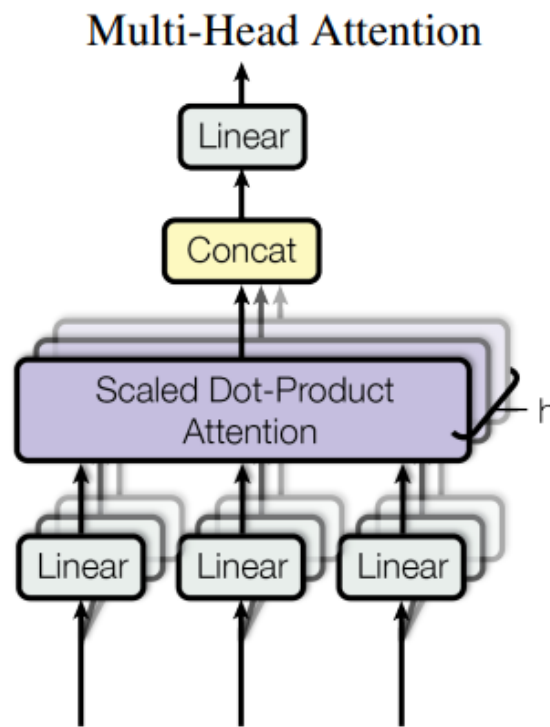
Để tính toán giá trị attention cho toàn bộ các từ trong một câu, mô hình Transformer đã sử dụng ba ma trận là Query, Keys và Values (viết tắt là Q, K, V tương ứng). Trong đó, Query dùng

để chứa thông tin của từ được tìm kiếm, mỗi dòng của Q sẽ là một vectơ đại diện cho các từ đầu vào, Key dùng để biểu diễn thông tin của các từ được so sánh với từ cần tìm kiếm đó (mỗi dòng cũng là một vectơ của từ đầu vào) và Values dùng để biểu diễn nội dung, ý nghĩa của các từ. Hai ma trận Q, K sẽ được sử dụng để tính giá trị attention của các từ trong câu đối với một từ xác định. Tiếp theo, các giá trị này sẽ được sử dụng để tính ra các vectơ attention dựa trên việc trung bình hóa có trọng số với ma trận V. Phương trình tính toán Attention được mô tả như sau:

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

2.2.3.2 Kiến trúc Multi-Head Attention

Áp dụng kỹ thuật tự chú ý (Self Attention), Transformer có thể học được những mối quan hệ giữa các từ với nhau trong một văn bản. Tuy nhiên, trong thực tế những mối quan hệ này cũng rất đa dạng và phong phú, không thể chỉ gói gọn trong một hình thức thể hiện nhất định. Do đó, để mở rộng cũng như cải thiện hiệu suất của mô hình, các tác giả đã xuất đề sử dụng nhiều lớp tự chú mục đích bắt giữ nhiều nhất có thể những mối quan hệ trong văn bản. Và để phân biệt các lớp tự chú ý với nhau, các ma trận trọng số Query, Key, Value sẽ có thêm một chiều "depth" chứa định danh của mình. Hình 2.2 mô tả kiến trúc của Multi-Head Attention.



Hình 2.2: Kiến trúc của mô hình Multi-Head Attention [3]

2.3 Phân loại hình ảnh với mô hình Mạng neuron tích chập (CNN)

Phương pháp phân loại hình ảnh sử dụng mô hình Mạng neuron tích chập (CNN) đã trở thành một trong những phương tiện hiệu quả và phổ biến nhất trong lĩnh vực thị giác máy tính. CNN

được thiết kế đặc biệt để nhận diện các đặc trưng trong dữ liệu hình ảnh, giúp cải thiện khả năng hiểu biểu đồ của mô hình.

Có nhiều mô hình CNN nổi tiếng và phổ biến được sử dụng trong các ứng dụng phân loại hình ảnh. Trong số đó, mô hình ResNet50 và YOLOv5 đã đạt được sự chú ý đặc biệt.

Mô hình ResNet50: ResNet50 là một kiến trúc mạng neuron sâu có khả năng học cực kỳ sâu và chịu được vấn đề biến mất đặc trưng trong quá trình học. Nó sử dụng khối cơ bản được gọi là "Residual Block," giúp tránh vấn đề vanishing gradient và cho phép xây dựng các mô hình sâu hơn mà vẫn duy trì hiệu suất tốt.

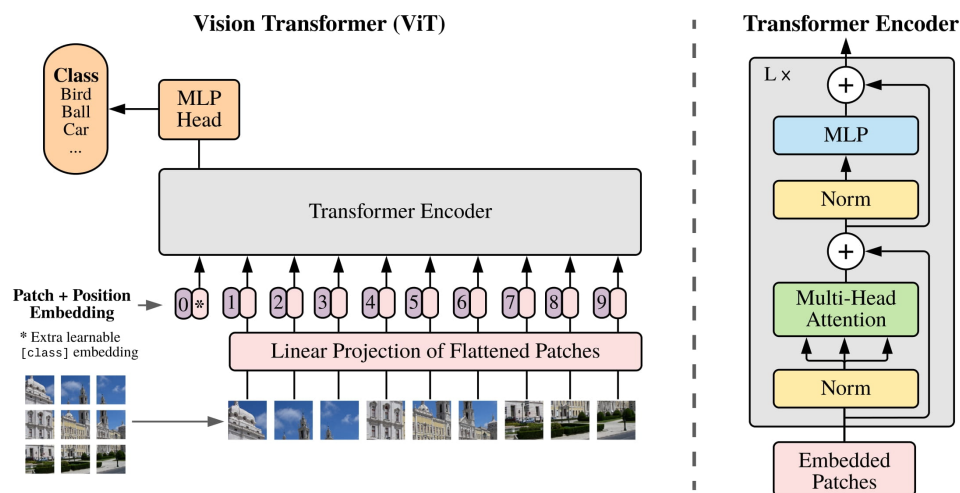
Mô hình YOLOv5: YOLOv5, hay "You Only Look Once," là một trong những mô hình phân loại hình ảnh và đồng thời xác định vị đối tượng trong thời gian thực. Nó được xây dựng dựa trên kiến trúc YOLO, với cải tiến đáng kể về tốc độ và độ chính xác. YOLOv5 thường được sử dụng trong các ứng dụng yêu cầu khả năng xác định vị đối tượng nhanh và chính xác.

Sự kết hợp giữa phương pháp phân loại hình ảnh và các mô hình CNN mạnh mẽ như ResNet50 và YOLOv5 không chỉ mang lại hiệu suất cao mà còn mở ra nhiều ứng dụng trong thực tế, như nhận diện đối tượng, nhận diện khuôn mặt, và nhiều lĩnh vực khác.

2.4 Chú thích hình ảnh với mô hình Transformer

Chú thích hình ảnh là một phần quan trọng của xử lý hình ảnh và thị giác máy tính. Mô hình Transformer, ban đầu được thiết kế cho xử lý dữ liệu chuỗi như ngôn ngữ tự nhiên, đã được mở rộng để áp dụng vào các tác vụ xử lý hình ảnh, bao gồm cả chú thích hình ảnh.

Mô hình Transformer trong ngữ cảnh chú thích hình ảnh thường sử dụng kiến trúc như mô hình ViT (Vision Transformer) 2.3. ViT chia hình ảnh thành các "điểm chú ý" và sau đó chuyển chúng thành dạng chuỗi để xử lý bởi các lớp Transformer. Điều này cho phép mô hình học được các mối quan hệ và đặc trưng quan trọng trong không gian hình ảnh, giúp nó thực hiện chú thích một cách hiệu quả.



Hình 2.3: Kiến trúc của mô hình ViT [1]

Đo độ lỗi WER (Word Error Rate) là một phương pháp đánh giá hiệu suất của các hệ thống nhận dạng tiếng nói hoặc xử lý ngôn ngữ tự nhiên. WER được tính bằng cách đo sự khác biệt giữa văn bản dự đoán và văn bản thực tế, dựa trên số từ được chèn (Insertions - I), số từ bị xóa (Deletions - D) và số từ bị thay đổi (Substitutions - S).

Công thức tính WER là:

$$WER = \frac{I + D + S}{N} \times 100$$

Trong đó:

I là số từ được chèn,
 D là số từ bị xóa,
 S là số từ bị thay đổi,
 N là tổng số từ trong văn bản thực tế.

Với công thức này, giá trị WER càng thấp thì mô hình càng chính xác.

Ví dụ, nếu có một câu có 10 từ, trong đó 2 từ được chèn, 1 từ bị xóa và 1 từ bị thay đổi, WER sẽ là:

$$WER = \frac{2 + 1 + 1}{10} \times 100 = 40\%$$

2.5 Phân loại câu chữ với mô hình Transformer

Mô hình Transformer không chỉ có ứng dụng trong xử lý hình ảnh, mà còn rất hiệu quả trong các tác vụ phân loại câu chữ. Các ứng dụng phổ biến của phân loại câu chữ bao gồm phân loại cảm xúc trong văn bản, phân loại văn bản thư rác, và dự đoán chủ đề từ một đoạn văn bản.

Trong mô hình Transformer được sử dụng cho phân loại câu chữ, các chuỗi đầu vào được chuyển qua lớp đầu tiên của Transformer để tạo ra các biểu diễn đầu vào. Sau đó, các biểu diễn này được đưa qua các lớp Transformer để học các mối quan hệ và đặc trưng quan trọng trong chuỗi. Điều này giúp mô hình hiệu quả trong việc hiểu nghĩa và phân loại các đoạn văn bản một cách chính xác.

Độ đo chính xác dự đoán Accuracy là tổng số dự đoán chính xác trên tổng số phép thử

$$Accuracy = \frac{TotalTruthPrediction}{TotalSamplesPrediction}$$

Chương 3

Phương pháp đề xuất

3.1 Chuẩn bị tập dữ liệu

Tập dữ liệu cần cho nghiên cứu này phải là tập dữ liệu gồm

- Hình ảnh - image
- Chú thích về nội dung hình ảnh đó - caption/text
- Lớp phân loại - label

Trong quá trình tìm kiếm tập dữ liệu như trên, em không đạt được kết quả mong muốn. Chính vì lý do đó, em tiến hành xây dựng tập dữ liệu nhờ vào Google Gemini API.

3.1.1 Tập dữ liệu Pokemon

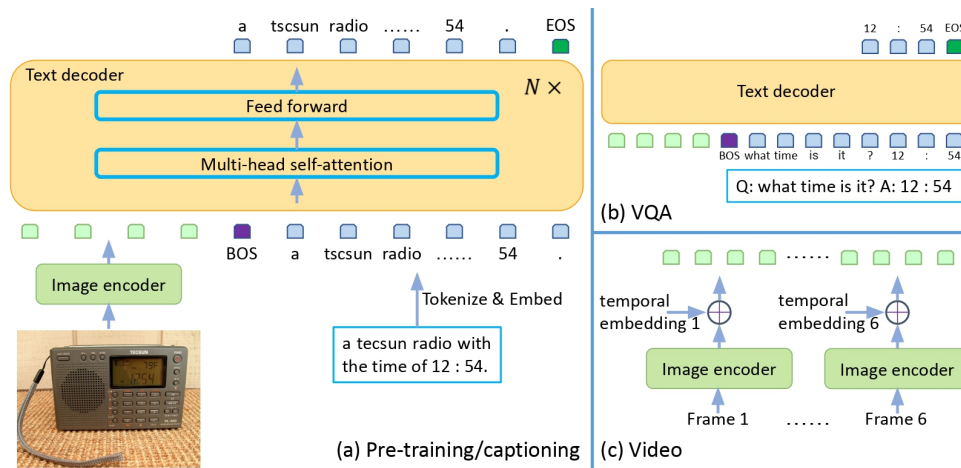
- Tập dữ liệu Pokemon <https://huggingface.co/datasets/keremberke/pokemon-classification> gồm:
 - 4.87k images train, 1.39k images validation, 732 images test,
 - Mỗi tập dữ liệu gồm hình ảnh pokemon và tên pokemon (label)
- Google Gemini Pro Version API: em tham khảo CLI API từ notebook https://colab.research.google.com/github/google/generative-ai-docs/blob/main/site/en/tutorials/python_quickstart.ipynb#scrollTo=MCzr5ZpNhxLm

3.2 Chú thích hình ảnh với mô hình GIT

Mô hình GIT, hay Generative Image-to-text Transformer, là một mô hình hiệu quả trong việc kết hợp thông tin từ hình ảnh và văn bản. Trích từ bài báo "GIT: A Generative Image-to-text Transformer for Vision and Language" [4] mô hình này đạt được nhiều thành công trong việc tạo ra mô tả văn bản cho hình ảnh, cung cấp khả năng gắn kết giữa thị giác và ngôn ngữ.

3.2.1 Ứng dụng phổ biến của GIT

Mô hình GIT có nhiều ứng dụng phổ biến trong thực tế. Một trong những ứng dụng quan trọng là tạo mô tả tự động cho hình ảnh, giúp người máy hiểu và tạo ra mô tả chính xác dựa trên nội dung hình ảnh. Điều này có thể hỗ trợ trong việc tổ chức và tìm kiếm hình ảnh trực quan, đặc biệt trong các ứng dụng như quản lý thư viện ảnh và tìm kiếm thông tin dựa trên nội dung hình ảnh.



Hình 3.1: Mô hình GIT - Generative Image-to-text Transformer

3.3 Phân loại câu chữ với mô hình RoBERTa

Mô hình RoBERTa [2] (Robustly optimized BERT approach) là một trong những mô hình ngôn ngữ hiện đại và mạnh mẽ trong lĩnh vực xử lý ngôn ngữ tự nhiên. Được xây dựng trên cơ sở của kiến trúc BERT (Bidirectional Encoder Representations from Transformers), RoBERTa đã trở thành một công cụ hiệu quả trong nhiều tác vụ xử lý ngôn ngữ, đặc biệt là trong phân loại câu chữ.

Mô hình RoBERTa đã chứng minh khả năng xuất sắc trong nhiều tác vụ phân loại câu chữ, bao gồm phân loại cảm xúc, phân loại chủ đề, và nhiều ứng dụng khác. Với khả năng hiểu biểu đồ ngôn ngữ tự nhiên đặc sắc, RoBERTa có thể học được các mối quan hệ phức tạp trong các đoạn văn bản và đưa ra dự đoán chính xác về loại câu chữ.

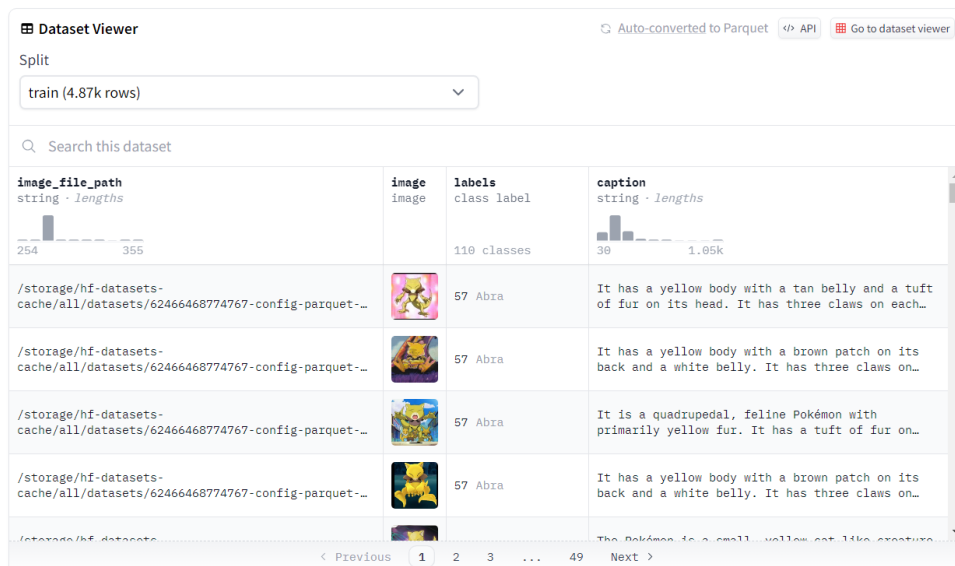
Trong phân loại câu chữ, mỗi câu được coi như một chuỗi, và mô hình cần hiểu cấu trúc chuỗi này để đưa ra dự đoán. RoBERTa, với khả năng biểu diễn bằng mã hóa chuỗi đầu vào một cách hiệu quả, là lựa chọn lý tưởng cho nhiều tác vụ sequence classification. Mô hình này sử dụng các lớp tự học để xử lý thông tin chuỗi và tạo ra biểu diễn đầu ra phản ánh nghĩa của câu.






Chương 4

Kết quả nghiên cứu

4.1 Tập dữ liệu Pokemon

Sau khi áp dụng Google Gemini API và một số xử lý gen caption từ hình ảnh pokemon, em thu được tập dữ liệu tại <https://huggingface.co/datasets/TeeA/Pokemon-Captioning-Classification> với kích thước tương tự tập dữ liệu gốc 3.1.1



image_file_path	image	labels	caption
string · lengths	image	class label	string · lengths
254 355		110 classes	39 1.05k
/storage/hf-datasets-cache/all/datasets/62466468774767-config-parquet-...		57 Abra	It has a yellow body with a tan belly and a tuft of fur on its head. It has three claws on each...
/storage/hf-datasets-cache/all/datasets/62466468774767-config-parquet-...		57 Abra	It has a yellow body with a brown patch on its back and a white belly. It has three claws on...
/storage/hf-datasets-cache/all/datasets/62466468774767-config-parquet-...		57 Abra	It is a quadrupedal, feline Pokémon with primarily yellow fur. It has a tuft of fur on...
/storage/hf-datasets-cache/all/datasets/62466468774767-config-parquet-...		57 Abra	It has a yellow body with a brown patch on its back and a white belly. It has three claws on...
/storage/hf-datasets-cache/all/datasets/62466468774767-config-parquet-...		57 Abra	The Pokémon is a small, yellow, cat-like creature...

Hình 4.1: Tập dữ liệu Pokemon tạo bởi Gemini

4.2 Image Classification với YOLOVv5

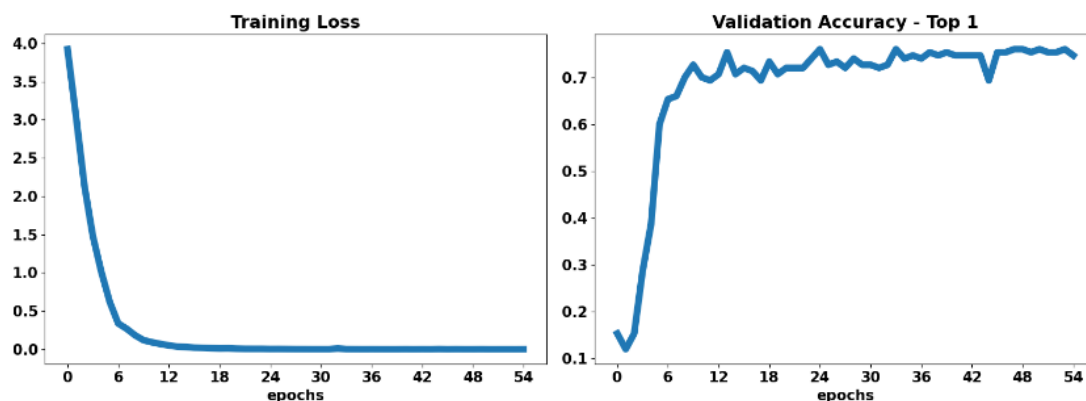
Sử dụng framework train sẵn từ [Roboflow](https://universe.roboflow.com/teea/pokemon-epaek/model/1), em đặt lệnh và nhận được một số kết quả sau:

Tham khảo trực tiếp từ: <https://universe.roboflow.com/teea/pokemon-epaek/model/1>

Mô hình YOLOv5 đạt độ chính xác **76.0%**, điều này cho thấy tập dữ liệu Pokemon là tập dữ liệu khó và phức tạp. Có lẽ bởi vì một số pokemon có bản tiến hóa cấp cao hơn không mang nhiều đặc trưng khác biệt so với cấp tiến hóa trước đó.

4.3 Image Classification với Resnet50

Sử dụng base model từ Microsoft <https://huggingface.co/microsoft/resnet-50>



Hình 4.2: Quá trình huấn luyện YOLOv5

Mô hình Resnet50 đạt độ chính xác **8.49%**

4.4 Image Captioning với GIT model

Sử dụng base model GIT từ Microsoft <https://huggingface.co/microsoft/git-base>

Mô hình GIT khi đo đặc, chỉ **4.65%** phần trăm lỗi từ

4.5 Sequence Classification với Roberta model

Sử dụng base model Roberta từ Huggingface Team <https://huggingface.co/roberta-base>

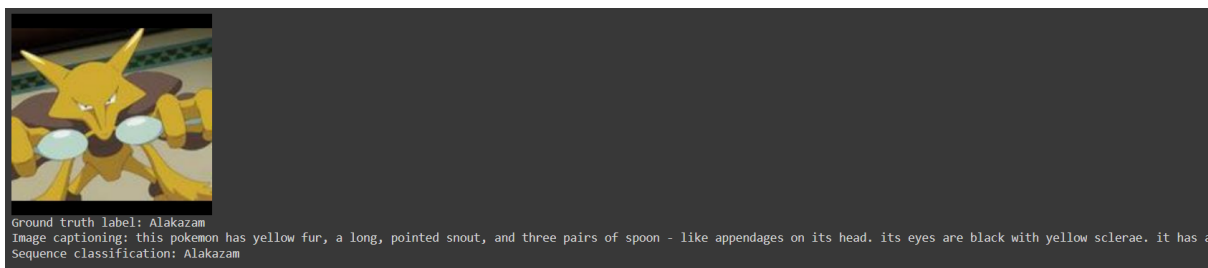
Mô hình Resnet50 đạt độ chính xác **6.19%**

4.6 Tổng hợp kết quả

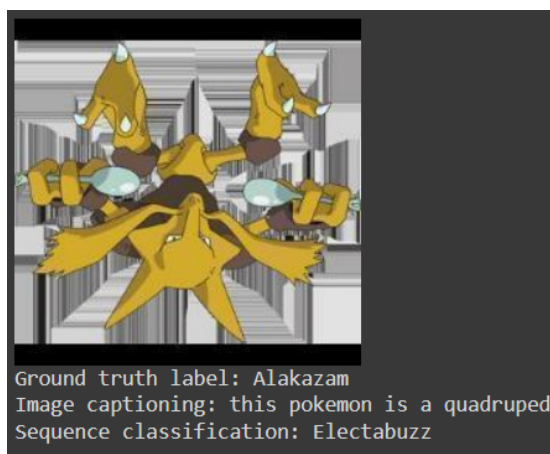
Models	Task(s)	
	Image Classification	
	Image Captioning	Sequence Classification
YOLOv5		76.0
Resnet50		8.49
GIT*	4.65	-
Roberta	-	6.19
Ours		1.09

Bảng 4.1: Bảng đánh giá độ chính xác (%) của mô hình

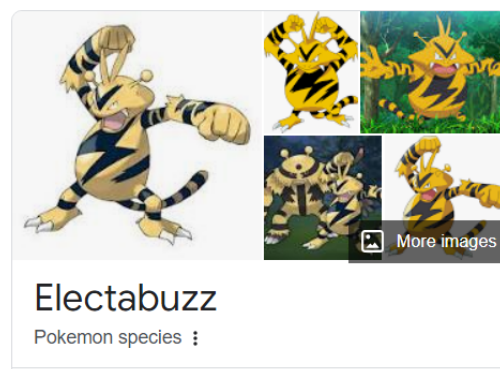
- Với mô hình GIT, độ đo mặc định là phần trăm lỗi - kết quả đo đặc là **4.65%**. Tuy nhiên, để đồng bộ về độ đo chính xác chung của bảng, em điều chỉnh thành độ chính xác là **100.0% - 4.65% = 95.35%**
- Mặc dù kết quả dự đoán là quá thấp và không thể áp dụng. Tuy nhiên, kết quả dự đoán là một Pokemon khác có đặc điểm tương đồng về một số đặc điểm với Pokemon đầu vào. Ví dụ Pokemon ở cấp 1 và cấp 2



Hình 4.3: Trường hợp dự đoán đúng tên Pokemon



(a) Dự đoán của mô hình



(b) Pokemon Ground Truth Electabuzz (ảnh từ Google)

Hình 4.4: Trường hợp dự đoán sai tên Pokemon

- Training YOLOv5: <https://universe.roboflow.com/teea/pokemon-epaek/model/1>
- Training Resnet50: <https://huggingface.co/TeeA/resnet-50-finetuned-pokemon>
- Training Roberta Classifier: <https://huggingface.co/TeeA/roberta-classifier-pokemon>
- Training GIT Image Captioning: <https://huggingface.co/TeeA/git-base-pokemon>

Chương 5

Tổng kết đề tài

Phương pháp thử nghiệm Image-Captioning-Classification không đạt được kết quả cao trên tập dữ liệu Pokemon phải chăng câu chữ không đủ diễn tả nội dung hình ảnh mà ở đây là hình ảnh đặc điểm trên cơ thể của các Pokemon, qua đó làm mất đi thông tin từ hình ảnh và từ đó năng lực dự đoán của mô hình gặp sai sót. Mặc dù đạt kết quả không được tốt trên tập Pokemon, em vẫn hy vọng đây là hướng nghiên cứu tiềm năng với các tập dữ liệu thực tế hơn như đã đề cập ở phần giới thiệu đề tài, qua đó giúp sàng lọc dữ liệu hiệu quả hơn.

Một số hướng nghiên cứu tiếp theo

- Thu thập tập dữ liệu thực tế trên các mạng xã hội: gồm hình ảnh, mô tả hình ảnh (có thể là caption của bài viết đó) và định nghĩa tập nhãn lớn
- Thử nghiệm mô hình với tập dữ liệu trên
- Đánh giá và chọn lọc mô hình tốt nhất đối với các mô hình của cùng 1 tác vụ

Mặc dù kết quả đạt được không như mong muốn nhưng hy vọng đây sẽ là bước đệm kết nối 3 thành tố: hình ảnh, chú thích và nhãn cho một số nhiệm vụ khác tốt đẹp hơn.

Sau cùng, em xin gửi lời cảm ơn đến thầy Nguyễn Đức Dũng đã hỗ trợ và hướng dẫn em hoàn thành đề tài này.

Tài liệu tham khảo

- [1] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021.
- [2] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach, 2019.
- [3] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.
- [4] Jianfeng Wang, Zhengyuan Yang, Xiaowei Hu, Linjie Li, Kevin Lin, Zhe Gan, Zicheng Liu, Ce Liu, and Lijuan Wang. Git: A generative image-to-text transformer for vision and language, 2022.