



UNIVERSITY OF
SCIENCE
VIETNAM NATIONAL UNIVERSITY HO CHI MINH CITY

Seminar Data Science

IMAGE CAPTION GENERATOR

Sinh viên thực hiện:

Nguyễn Lê Công Duy

Giảng viên hướng dẫn

TS. Trần Anh Tuấn

INDEX PAGE

Abstract	2
CHAPTER 1: INTRODUCTION	4
CHAPTER 2: LITERATURE REVIEW	5
1. Related Works or Conventional Approaches	5
2. Proposed Approach	5
CHAPTER 3: METHODOLOGY	6
3.1 Convolutional neural network	6
3.2 Recurrent neural network (LSTM)	6
3.3 Architecture	7
3.4 Implement	7
CHAPTER 4: RESULTS AND CONCLUSION	11

ABSTRACT

Image Captioning, còn được biết đến là chú thích ảnh, là một ứng dụng thú vị và phát triển nhanh chóng của công nghệ học sâu trong những năm gần đây. Đây là sự kết hợp tuyệt vời giữa Computer Vision (thị giác máy tính) và Xử lý Ngôn ngữ Tự nhiên (NLP), đặt ra thách thức làm thế nào chúng ta có thể mô tả nội dung của một bức ảnh một cách tự động và hiệu quả.

Image Captioning bắt đầu từ một hình ảnh và tạo ra một đoạn văn bản ngắn, chính xác mô tả bức ảnh đó. Quá trình này đòi hỏi mô hình không chỉ hiểu nội dung của hình ảnh mà còn có khả năng biểu diễn ý nghĩa bằng ngôn ngữ tự nhiên.

Trong bài báo cáo này, em sẽ đi sâu vào phương pháp để giải quyết bài toán này. Em sẽ bàn về các kiến trúc mạng neural được thiết kế để học biểu diễn của cả hình ảnh và ngôn ngữ, cũng như cách chúng được tích hợp để tạo ra mô hình Image Captioning. Độ phức tạp của nhiệm vụ yêu cầu sự tương tác mạnh mẽ giữa các thành phần khác nhau của mạng neural, và em sẽ mô tả cách mối quan hệ này được xây dựng.

Ngoài ra, em sẽ giới thiệu các phương pháp đánh giá cho mô hình Image Captioning. Điều này bao gồm cách đo lường độ chính xác của mô hình trong việc mô tả hình ảnh và cách mô hình được đánh giá dựa trên sự "đúng" của câu mô tả so với câu mô tả thực tế. Những phương pháp này không chỉ giúp đánh giá chất lượng của mô hình mà còn mang lại cái nhìn sâu sắc về cách mô hình hiểu và tái tạo cả ngôn ngữ và hình ảnh.

Keywords - image captioning ; convolution neural network (CNN) ; recurrent neural network (RNN) ; Long short-term memory (LSTM) ; Network Image Caption (NIC)

CHAPTER 1: INTRODUCTION

1.1 Motivation .

Các hệ thống mô tả hình ảnh (Neural image captioning – NIC) có nhiều ứng dụng trong công nghiệp, như hỗ trợ người sử dụng có vấn đề thị giác hiểu nội dung trang web, tạo phụ đề tự động, hoặc xác định các vai trò ngữ nghĩa của đối tượng trong hình ảnh. Mô tả hình ảnh cũng có giá trị lý thuyết, vì giải quyết nhiệm vụ này sẽ đại diện cho một bước tiến trong việc hiểu một cảnh hoàn chỉnh - nơi máy tính có thể "nhìn thấy" và giải thích thông tin giác quan giống như con người, một ước mơ lâu dài của các chuyên gia thị giác máy tính và trí tuệ nhân tạo.

Ứng dụng nhiều trong các lĩnh vực khác nhau như y học, sinh học, thương mại, tìm kiếm web và quân sự, v.v. Trong lĩnh vực y học, NIC có thể hỗ trợ việc chẩn đoán và theo dõi các bệnh lý dựa trên hình ảnh y tế. Trong lĩnh vực thương mại, NIC có thể giúp tăng cường trải nghiệm mua sắm trực tuyến và thu hút sự chú ý của khách hàng. Trong lĩnh vực tìm kiếm web, NIC có thể tăng cường khả năng tìm kiếm và phân loại hình ảnh trên các công cụ tìm kiếm. Trong lĩnh vực quân sự, NIC có thể hỗ trợ việc phân tích hình ảnh quân sự và trích xuất thông tin quan trọng từ hình ảnh.

Các mạng xã hội như Instagram, Facebook, ... cũng sử dụng NIC tự động để cung cấp trải nghiệm người dùng tốt hơn. Việc tự động tạo mô tả từ hình ảnh giúp người dùng dễ dàng chia sẻ và tương tác với những hình ảnh mà họ đăng. Đồng thời, cũng giúp tăng cường khả năng tìm kiếm và khám phá nội dung trên các mạng xã hội.

Trong tương lai, việc phát triển và cải tiến các phương pháp NIC sẽ tiếp tục là một lĩnh vực nghiên cứu quan trọng. Việc tạo ra các mô hình NIC chính xác và tự nhiên sẽ đóng vai trò quan trọng trong việc cải thiện trải nghiệm người dùng và ứng dụng của công nghệ trí tuệ nhân tạo.

1.2 Problem Description

Việc tạo chú thích cho hình ảnh là một thách thức, yêu cầu mô hình không chỉ hiểu sâu về ngữ cảnh hình ảnh mà còn có khả năng biểu đạt điều đó bằng ngôn ngữ tự nhiên một cách sáng tạo. Quá trình này không chỉ bao gồm việc nhận diện đối tượng và hành động trong hình ảnh, mà còn đặt ra yêu cầu về độ chính xác và tính đa dạng của mô tả. Sự độc đáo và sáng tạo trong việc tạo ra các mô tả không chỉ ngăn chặn sự trùng lặp mà còn tạo ra trải nghiệm hấp dẫn cho người xem. Đồng thời, công việc này cần sử dụng một bộ dữ liệu thực tế và phức tạp hơn. Điều này tạo ra thách thức lớn khi huấn luyện mạng chú thích hình ảnh trên dữ liệu không có hướng dẫn rõ ràng. Công trình này thảo luận về khó khăn và hành vi không mong muốn trong quá trình huấn luyện và thử nghiệm nhiều kiến trúc để tìm ra kiến trúc phù hợp nhất. Tích hợp giữa thị giác máy tính và xử lý ngôn ngữ tự nhiên, nhiệm vụ này là một bước tiến đáng chú ý trong lĩnh vực trí tuệ nhân tạo, đặt ra những thách thức đòi hỏi sự đồng bộ và sáng tạo từ các nhà nghiên cứu.

Nội dung các phần còn lại được trình bày theo thứ tự :

- Ở chương 2, em sẽ trình bày ý tưởng chung và phương pháp đề xuất cho bài toán
- Ở chương 3, em sẽ trình bày chi tiết về mô hình cũng như các bước thực hiện
- Ở chương 4, em sẽ trình bày kết quả và kết luận báo cáo

CHAPTER 2: LITERATURE REVIEW

1. Related Works or Conventional Approaches

Trong các nhiệm vụ dự đoán chuỗi - như dự báo chuỗi thời gian và mô hình ngôn ngữ - liên quan đến việc sử dụng một chuỗi đầu vào có độ dài biến đổi, và dự đoán một đầu ra duy nhất. Mạng neural được trang bị tốt để giải quyết loại vấn đề "many-to-one" này vì nó có cấu trúc đơn giản hơn và ít phức tạp hơn trong việc xử lý dữ liệu. Trong many-to-one, bạn có một loạt dữ liệu đầu vào và chỉ một đầu ra, giúp giảm độ phức tạp của mô hình.

Tuy nhiên, một loại nhiệm vụ dự đoán chuỗi khác là những nhiệm vụ có chuỗi đầu vào và đầu ra có độ dài biến đổi (many-to-many) này khó khăn hơn, vì mạng phải học cách tạo ra các dự đoán có độ dài biến đổi. Một ví dụ quan trọng cho many to many là Static machine translation (SMT). Các mô hình SMT cố gắng nhận vào một chuỗi từ trong một ngôn ngữ và đầu ra là một chuỗi từ trong một ngôn ngữ khác, trong khi bảo toàn ý nghĩa của chuỗi đầu vào và tính liên kết của chuỗi đầu ra. Một phương pháp đã được chứng minh hiệu quả cho máy dịch là kiến trúc "Encoder-Decoder". Kiến trúc này bao gồm hai phần: một "bộ mã hóa," nhận vào một chuỗi đầu vào có độ dài biến đổi và mã hóa nó thành một biểu diễn vector có độ dài cố định. Một mô hình "bộ giải mã" giải mã biểu diễn vector này thành một dự đoán chuỗi có độ dài biến đổi.

Gần đây, cộng đồng học sâu đã thành công trong các nhiệm vụ quan trọng của thị giác máy tính. Cùng với sự tiến bộ trong dịch máy neuron và mô hình ngôn ngữ, nhà nghiên cứu đã phát triển các mô hình mạng nơ-ron end-to-end cho hệ thống chú thích hình ảnh. Người ta sử dụng mạng nơ-ron tích chập (CNN) và mạng nơ-ron hồi quy (RNN) để tạo ra mô hình chú thích hình ảnh. Họ lấy ý tưởng từ cấu trúc encoder-decoder trong SMT, sử dụng CNN để mã hóa hình ảnh và RNN để tạo ra chú thích.

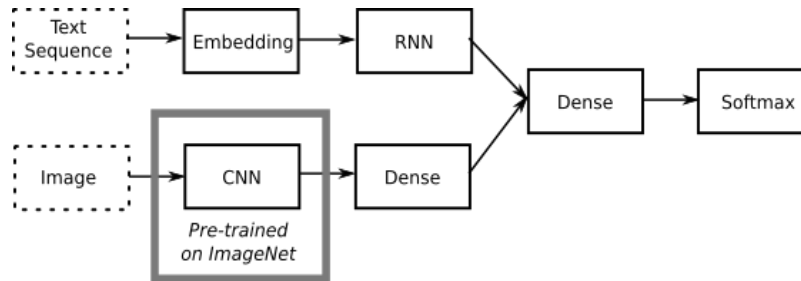
Bài báo "Where to put the image in an image caption generator" [1] đã thử nghiệm các kiến trúc khác nhau cho mô hình chú thích hình ảnh, tìm ra kiến trúc phù hợp nhất trên các bộ dữ liệu chú thích hình ảnh cơ bản và đưa ra giả thuyết về nhiệm vụ phù hợp nhất với RNNs.

Bài báo "Show, attend and tell: Neural image caption generation with visual attention"[2], đạt hiệu suất xuất sắc bằng cách tích hợp cơ chế chú ý, giúp mạng tập trung vào đặc trưng quan trọng của hình ảnh. Cơ chế chú ý cũng mang lại tính giải thích cho mô hình chú thích hình ảnh.

2. Proposed Approach

- Trích xuất đặc trưng hình ảnh (Image Feature Extraction):
 - Sử dụng mô hình mạng nơ-ron học sâu (CNN) để trích xuất đặc trưng từ hình ảnh.
 - Biến đổi hình ảnh thành biểu diễn vector đặc trưng.
- Tạo biểu diễn văn bản (Text Representation):
 - Sử dụng mô hình mạng nơ-ron tái lập (RNN) hoặc các biến thể như LSTM để tạo biểu diễn ngôn ngữ từ mô tả.

- Merge (Kết hợp) thông tin:
 - Kết hợp hai biểu diễn từ hình ảnh (CNN) và ngôn ngữ (RNN).
- Tạo câu:
 - Sử dụng biểu diễn tổng hợp từ bước merge để tạo ra mô tả cuối cùng.
 - Mô tả thường được tạo ra bằng cách sử dụng mô hình mạng nơ-ron tái lập để dự đoán từng từ mô tả một cách tuần tự.

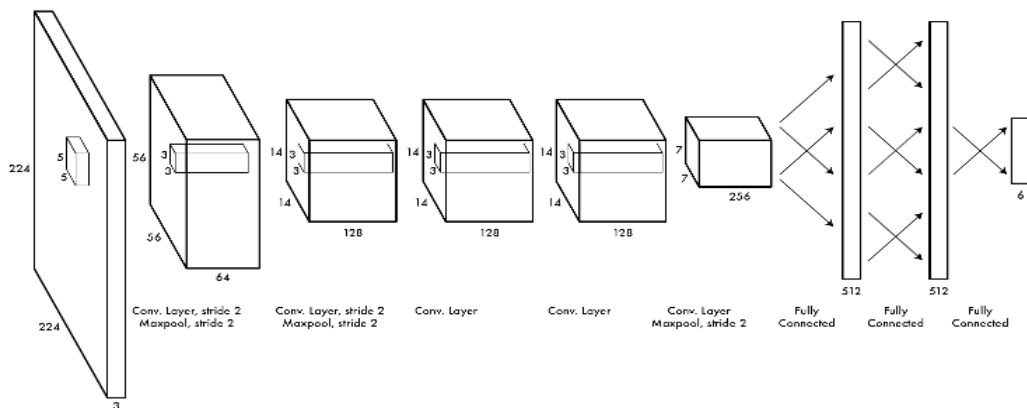


Hình 1. Tổng quan mô hình đề xuất

CHAPTER 3: METHODOLOGY

3.1 Convolutional neural network

Mạng nơ-ron tích chập (CNN) là một loại mạng neural đặc biệt, được thiết kế để xử lý dữ liệu có hình dạng tương tự như một ma trận 2D. Hình ảnh, trong ngữ cảnh này, có thể được xem như ma trận 2D, tạo nên một lưới các giá trị pixel. Vai trò quan trọng của CNN nằm trong khả năng xử lý và rút trích đặc trưng từ hình ảnh. Khi nhận một hình ảnh làm đầu vào, CNN gán trọng số (weight) và độ chệch (bias) cho các khía cạnh và đối tượng khác nhau trong hình ảnh, từ đó hiểu và phân biệt chúng với nhau. Việc này giống như cách não người nhận biết và phân loại các đối tượng trong thời gian và không gian. Đặc biệt, CNN sử dụng các bộ lọc, hay các Kernel, để thực hiện quá trình học các đặc trưng. Các bộ lọc này giúp mô hình nhận biết các khái niệm trừu tượng trong hình ảnh, như làm mờ, phát hiện biên, làm sắc nét, và nhiều khái niệm khác. Qua việc này, CNN trở thành một công cụ mạnh mẽ để hiểu và biểu diễn thông tin quan trọng trong hình ảnh.

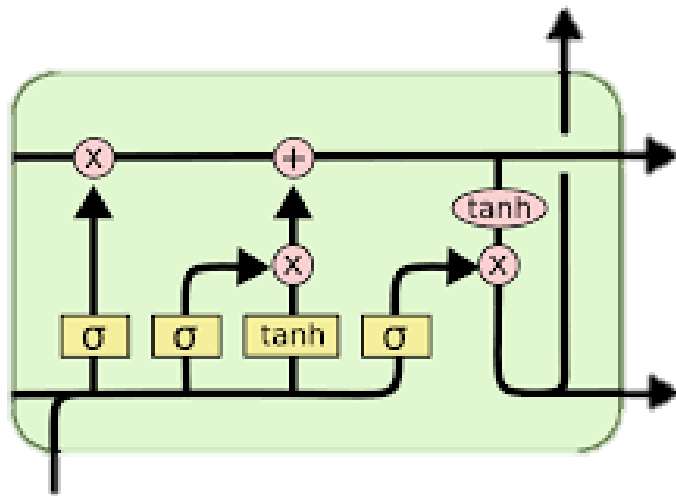


Hình 2. Ví dụ của Convolutional Neural Network

3.2 Recurrent neural network (LSTM)

Trong báo cáo này, mô hình LSTM (Long Short-Term Memory) được sử dụng như một bộ mã hóa cho các từ vựng. LSTM thường được sử dụng rộng rãi trong Xử lý Ngôn ngữ Tự nhiên (NLP). Mô hình LSTM có một lớp bộ nhớ, giúp nó duy trì thông tin qua các chuỗi dài. Điều này giúp nó xử lý hiệu quả các phụ thuộc dài hạn và ghi nhớ thông tin trong khoảng thời gian dài. Với cơ chế cổng như cổng đầu vào, cổng đầu ra và cổng quên, LSTM có khả năng kiểm soát thông tin để lựa chọn thông tin nào quan trọng và thông tin nào có thể bị loại bỏ. Quá trình này cung cấp khả năng học hiệu quả và tận dụng thông tin từ quá khứ để dự đoán tốt hơn.

$$\begin{aligned} i_t &= \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i) \\ f_t &= \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f) \\ o_t &= \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o) \\ g_t &= \phi(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \\ c_t &= f_t \odot c_{t-1} + i_t \odot g_t \\ h_t &= o_t \odot \phi(c_t) \end{aligned}$$



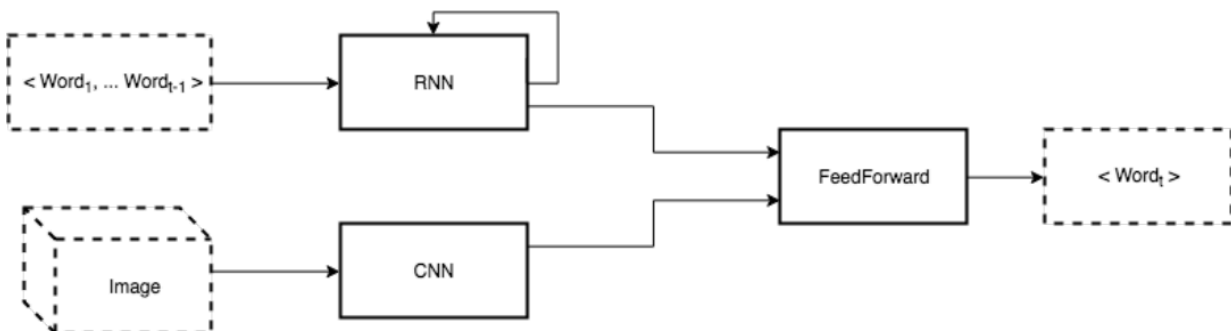
Hình 3. Ảnh minh họa mạng LSTM

3.3 Architecture

Lấy ý tưởng từ bài toán Machine translation (dịch máy), những mô hình cho bài toán Image captioning (NIC) cũng thường được chia làm hai phần chính : decoder và encoder. Các input đầu vào (hình ảnh và mô tả) sẽ được mã hóa với CNNs cho hình ảnh và RNN cho văn bản, sau đó các vector “đặc trưng” của hình ảnh được kết hợp với các token đã dự đoán trước đó để dự đoán token tiếp theo trong một chuỗi, kiến trúc này còn được gọi là kiến trúc “inject”.

Tuy nhiên, một giải pháp thay thế được gọi là kiến trúc “merge” đã được tìm thấy để tạo ra kết quả tốt hơn (được chứng minh ở bài báo “Where to put image in an image caption generator” [1]). Thay vì kết nối bộ mã hóa hình ảnh làm đầu vào của bộ giải mã trình tự, hai thành phần này hoạt động độc lập với nhau. Nói cách khác, ta không kết hợp hai dạng thức: hình ảnh với văn bản. Mạng CNNs chỉ xử lý hình ảnh và mạng LSTM chỉ hoạt động trên chuỗi token được tạo ra.

Tiếp theo, kết quả đầu ra của cả hai mạng được kết hợp với nhau. Chức năng của lớp này không chỉ giúp hợp nhất thông tin từ cả hai nguồn kết quả, mà còn thực hiện công việc diễn giải cho cả hai đầu ra. Sau đó, trình tạo câu sẽ tiếp tục dự đoán và tạo ra chú thích cuối cùng cho hình ảnh. Một lợi ích quan trọng của phương pháp này là khả năng sử dụng transfer learning không chỉ cho Bộ mã hóa hình ảnh mà còn mở rộng sang Bộ giải mã trình tự. Chúng ta có thể áp dụng mô hình ngôn ngữ đã được đào tạo trước đó cho việc huấn luyện Bộ giải mã trình tự, tận dụng kiến thức từ ngôn ngữ đã biết để cải thiện khả năng dự đoán chú thích cho ảnh



Hình 4. Kiến trúc “merge” cho bài toán Image caption generator

3.4 Implement

3.4.1 Dataset

Em đã sử dụng bộ dữ liệu Flickr 8K - một bộ dữ liệu điển hình để giải quyết bài toán tạo mô tả cho ảnh. Bộ dữ liệu này bao gồm khoảng 8,000 hình ảnh, mỗi hình ảnh được đi kèm với tới 5 chú thích khác nhau. Sự hiện diện của 5 chú thích cho một hình ảnh cụ thể giúp ta nắm bắt được sự đa dạng câu và các tình huống có thể xuất hiện trong ảnh đó.

Các hình ảnh được lựa chọn từ sáu nhóm đa dạng trên nền tảng Flickr, và đặc biệt, chúng không chứa bất kỳ cá nhân hay địa điểm nổi tiếng nào. Tuy nhiên, mỗi hình ảnh trong bộ dữ liệu được lựa chọn thủ công để mang lại sự đa dạng và độ phong phú trong cảnh quan và tình huống thể hiện

Link dataset : <https://www.kaggle.com/datasets/adityajn105/flickr8k/data>

3.4.2 Preprocessing

- Việc tiền xử lý dữ liệu được thực hiện trong hai phần, hình ảnh và chú thích tương ứng được làm sạch và tiền xử lý một cách riêng biệt.
- **Mã hóa hình ảnh:** Mô hình NIC phải tích hợp các đặc trưng được trích xuất từ cả một hình ảnh đầu vào và một đoạn chú thích một phần để dự đoán từ tiếp theo của chú thích. Điều này đòi hỏi việc nén một hình ảnh đầu vào thành một vector cố định, mã hóa các “đặc trưng” hình ảnh - do đó tạo thành một loại vector hình ảnh. Điều

này được thực hiện bằng cách đưa dữ liệu đầu vào vào mô hình Xception của API Keras chạy trên TensorFlow. Các lớp cuối cùng của kiến trúc Xception đều là các lớp fully connected, có kích thước lần lượt là 2048, 2048 và 1000. Để tạo ra một vector nhúng hình ảnh cho mỗi hình ảnh trong tập dữ liệu, hai lớp fully-connected cuối cùng từ Xception được bỏ đi và sử dụng các dự đoán của mô hình được rút gọn này như là các vector đặc trưng của hình ảnh. Hai lớp cuối cùng được bỏ đi vì trong khi các lớp đầu của mạng chịu trách nhiệm cho việc trích xuất đặc trưng, các lớp cuối cùng được chuyên biệt cho nhiệm vụ mà chúng được đào tạo. Do đó, việc loại bỏ hai lớp cuối cùng cho phép em mã hóa mỗi hình ảnh thành một vector 2048 chiều, có thông tin và có khả năng tổng quát hóa cho nhiệm vụ mô tả hình ảnh.

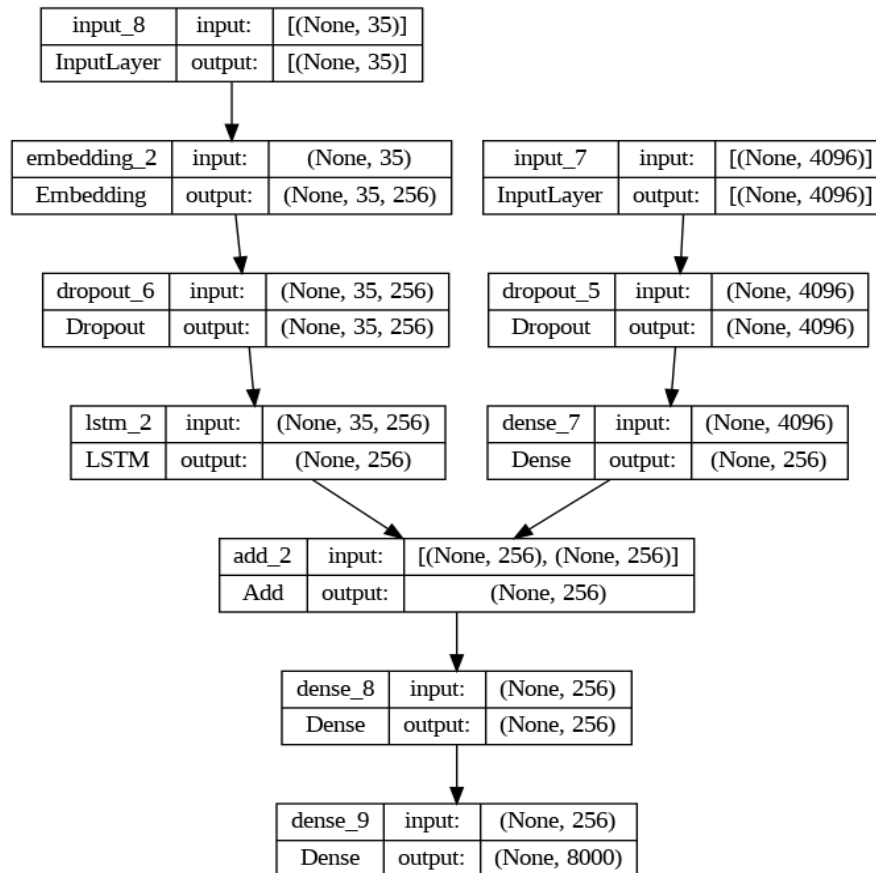
- **Tiền xử lý văn bản :** Một thách thức phổ biến trong các nhiệm vụ xử lý ngôn ngữ tự nhiên (NLP) là từ vựng của tập văn bản thường lặp lại (một số từ xuất hiện nhiều lần). Điều này dẫn đến việc mô hình có nhiều tham số và làm cho việc tổng hợp các mẫu học được trong quá trình đào tạo trở nên khó khăn. Do đó, em đã thực hiện một quá trình làm sạch sơ bộ cho các chú thích được cung cấp trong tập dữ liệu. Các thao tác được dùng :
 - Chuyển tất cả các ký tự thành chữ thường.
 - Loại bỏ các ký tự xuống dòng.
 - Loại bỏ các chữ số, ký tự đặc biệt và bất kỳ ký tự nào không phải là chữ cái từ chú thích. Bước này làm sạch văn bản bằng cách giữ lại chỉ các ký tự chữ cái.
 - Loại bỏ các khoảng trắng dư thừa
 - Thêm các từ khởi đầu và kết thúc vào chú thích.
- **Định dạng dữ liệu để huấn luyện:** ta có thể hiểu nhiệm vụ sinh chú thích hình ảnh theo khung xác suất. Trong phương trình (1), nó chỉ ra rằng, với giả định nhẹ, xác suất của một chú thích đúng, khi biết một hình ảnh, có thể được phân tách thành xác suất của mỗi từ dựa trên các từ trước đó và hình ảnh.

$$\log P(S_1, \dots, S_k | I; \theta) = \sum_{t=1}^k \log P(S_t | S_1, \dots, S_{t-1}, I; \theta)$$

Từ phương trình trên, em điều chỉnh lại các chú thích để phù hợp với nhiệm vụ mô hình hóa ngôn ngữ. Đầu tiên, em trích xuất một biểu diễn hình ảnh cho mỗi hình ảnh trong tập dữ liệu, và tiền xử lý văn bản của chú thích tương ứng. Sau đó, đối với mỗi cặp biểu diễn/hình ảnh, em tạo ra một huấn luyện mới cho mỗi từ trong chú thích.

	Xi		Yi	
i	Image feature vector	Partial Caption	Target word	
1	Image_1	startseq	the	data points corresponding to image 1 and its caption
2	Image_1	startseq the	black	
3	Image_1	startseq the black	cat	
4	Image_1	startseq the black cat	sat	
5	Image_1	startseq the black cat sat	on	
6	Image_1	startseq the black cat sat on	grass	
7	Image_1	startseq the black cat sat on grass	endseq	
8	Image_2	startseq	the	data points corresponding to image 2 and its caption
9	Image_2	startseq the	white	
10	Image_2	startseq the white	cat	
11	Image_2	startseq the white cat	is	
12	Image_2	startseq the white cat is	walking	
13	Image_2	startseq the white cat is walking	on	
14	Image_2	startseq the white cat is walking on	road	
15	Image_2	startseq the white cat is walking on road	endseq	

Bộ giải mã trong quá trình hoạt động này sẽ thực hiện quá trình hợp nhất giữa đầu ra của hai tầng trước đó để tạo ra dự đoán cuối cùng. Cụ thể, em sẽ kết hợp thông tin từ trình trích xuất đặc trưng và trình xử lý chuỗi, mỗi cái đều tạo ra một vector có chiều dài cố định. Sau đó, thông tin hợp nhất này sẽ được đưa vào một tầng Dense để thực hiện các biến đổi cuối cùng và đưa ra dự đoán. Tầng Dense này có vai trò quan trọng trong việc kết hợp thông tin từ cả hai nguồn và tạo ra một đầu ra cuối cùng đồng nhất với kích thước của từ vựng của em.



Tạo caption : Khi thực hiện quá trình sinh chú thích (captioning) trong mô hình máy học, có hai phương pháp chính được sử dụng để tạo ra chuỗi từ hình ảnh: Greedy Search và Beam Search.

Greedy Search là một phương pháp đơn giản, bắt đầu với một chuỗi trống, sau đó, ở mỗi bước, tìm kiếm toàn bộ không gian của bước đó, chọn ra từ có xác suất lớn nhất để đi tiếp. Tuy nhiên, thay vì xem xét tất cả các lựa chọn, Greedy Search chỉ lựa chọn duy nhất một kết quả có điểm số cao nhất. Khi chọn xong, quá trình mở rộng chỉ dựa trên kết quả duy nhất đã chọn ở bước trước đó. Điều này có thể dẫn đến việc bỏ qua các lựa chọn tốt nhất tại mỗi bước, vì mô hình chỉ tập trung vào việc chọn ra kết quả tốt nhất ngay tại thời điểm đó.

Một sự lựa chọn khác là Beam Search giúp cải thiện quá trình sinh chú thích bằng cách mở rộng không gian tìm kiếm từ nhiều ứng viên hàng đầu hơn. Thay vì chỉ chọn một kết quả duy nhất, Beam Search lựa chọn k kết quả có điểm số cao nhất. Sau đó, tất cả các kết quả này được tiếp tục mở rộng ở bước tiếp theo. Điều này giúp mô hình xem xét nhiều lựa chọn hơn và duy trì một "dải" các ứng viên hàng đầu, có khả năng sinh ra các chuỗi chú thích chất lượng cao hơn, linh hoạt hơn và phong phú hơn. Beam Search thường tạo ra các kết quả dự đoán có sự đa dạng và phù hợp hơn với nhiều khía cạnh của hình ảnh đầu vào.

Algorithm 1 Inference: Greedy Selection

```

1: procedure GREEDYSELECT(IMG-FEATURES, MODEL)
2:   caption  $\leftarrow$  [ < startseq > ] ▷ Initialize caption as start token.
3:   while Length(caption) < 15 do ▷ Repeat until generated caption is of maximum length.
4:     predictions  $\leftarrow$  model.predict(img-features, caption) ▷ Vector of predicted probabilities
5:     next-word  $\leftarrow$  argmax(predictions) ▷ Predicted next word is the one with the highest predicted probability.
6:     if next-word == <endseq> then
7:       break ▷ If end token is predicted, return caption as-is.
8:     else
9:       caption  $\leftarrow$  caption.append(next-word)
10:    return caption
```

Thuật toán Greedy search

Algorithm 2 Inference: Beam Search

```

1: procedure BEAMSEARCH(IMG-FEATURES, MODEL,  $\beta, \kappa, \alpha$ )
2:   population  $\leftarrow$  [ < startseq > ] ▷ Initialize population as a single 'starter' caption.
3:   i  $\leftarrow$  0 ▷ Iteration Number
4:   for i < 15 do
5:     for each candidate caption S  $\in$  population do
6:       if Last token in candidate == <endseq> then
7:         break
8:       predictions  $\leftarrow$  model.predict(img-features, caption) ▷ Vector of predicted probabilities
9:       Add top  $\kappa$  predicted words to caption to create  $\kappa$  new candidates
10:    Truncate population top  $\beta$  candidates, according to quality metric.
11:    i  $\leftarrow$  i + 1
12:  return population
```

Thuật toán Beam search




Đánh giá:

Trong quá trình đánh giá mô hình Image Caption Generator chúng ta không thể bỏ qua quan trọng của việc áp dụng các metrics NLP để đo lường chất lượng của mô tả tạo ra. Một trong những metrics phổ biến và được sử dụng rộng rãi là BLEU [4]. Đây là một công cụ hữu ích để đánh giá độ chính xác và sự giống nhau giữa câu mô tả sinh ra bởi mô hình và câu mô tả mục tiêu từ con người. BLEU hoạt động dựa trên nguyên tắc đơn giản và dễ hiểu: một mô hình được coi là tốt nếu câu mô tả mà nó tạo ra càng giống với câu mô tả mục tiêu từ con người

Giá trị Bleu Score nằm trong khoảng từ 0 đến 1. Mức độ 0.6 hoặc 0.7 thường được coi là tốt, nhưng cũng cần lưu ý rằng việc đạt được điểm gần với 1 có thể là dấu hiệu của việc mô hình đã bị overfitting

	BLEU-1	BLEU-2	BLEU-3
Greedy	0.549983	0.195940	0.080740
Beam (k=3)	0.541465	0.196471	0.083715
Beam (k=5)	0.529275	0.192826	0.080727

CHAPTER 4: RESULTS AND CONCLUSION

	Xception + LSTM	Actual
	<p>Greedy : <i>startseq</i> man in yellow life jacket sailing in the water <i>endseq</i></p> <p>Beam k = 3: <i>startseq</i> man in yellow life vest sailing in the water <i>endseq</i></p> <p>Beam k = 5: <i>startseq</i> the man is wearing life jacket and blue life jacket <i>endseq</i></p> <p>Beam k = 7: <i>startseq</i> the man is wearing life jacket and blue life jacket on the water <i>endseq</i></p>	<ul style="list-style-type: none"> • <i>startseq</i> man in black swimming gear parasails <i>endseq</i> • <i>startseq</i> man windsurfing <i>endseq</i> • <i>startseq</i> man windsurfs with group of other windsurfers <i>endseq</i> • <i>startseq</i> the man in the black suit is in the water on white and red board with sail <i>endseq</i> • <i>startseq</i> the man is on ski boat in the water <i>endseq</i>
	<p>Greedy : <i>startseq</i> woman in pink dress and black hat is standing in front of brick wall <i>endseq</i></p> <p>Beam k = 3: <i>startseq</i> woman in black dress and pink hat is standing in front of brick wall <i>endseq</i></p> <p>Beam k = 5: <i>startseq</i> woman in black dress and black hat is standing in front of brick wall <i>endseq</i></p> <p>Beam k = 7: <i>startseq</i> woman in black dress and black hat is standing in front of brick wall <i>endseq</i></p>	<ul style="list-style-type: none"> • <i>startseq</i> bride being escorted under an umbrella to large black car <i>endseq</i> • <i>startseq</i> bride takes cover under an umbrella <i>endseq</i> • <i>startseq</i> newly wed wife is under an umbrella held by pastor <i>endseq</i> • <i>startseq</i> an older newly wed couple stand under an umbrella in the rain <i>endseq</i> • <i>startseq</i> newlyweds stand outside in the rain <i>endseq</i>
	<p>Greedy : <i>startseq</i> brown dog is running through the grass <i>endseq</i></p> <p>Beam k=3: <i>startseq</i> brown dog is running in the grass <i>endseq</i></p> <p>Beam k=5: <i>startseq</i> brown dog is running in the grass <i>endseq</i></p> <p>Beam k=7: <i>startseq</i> the brown dog is running through the grass <i>endseq</i></p>	

Tổng kết:

Trong các phương pháp ngày nay,, kiến trúc merge, sự hòa trộn giữa Convolutional Neural Networks (CNNs) và Recurrent Neural Networks (RNN), là một cách tiếp cận đối với bài toán Image Caption Generator, bên cạnh đó đây cũng là một cánh cửa mở ra sự hiểu biết sâu sắc về nội dung hình ảnh và sức mạnh của thông tin ngôn ngữ. Tuy nhiên, những tiến

bộ mới trong lĩnh vực này đang chứng minh rằng có nhiều phương pháp khác có thể đưa ra mô tả chi tiết hơn, chính xác hơn cùng câu văn có ngữ nghĩa tốt hơn. Việc đó bao gồm việc sử dụng các mô hình trích xuất đặc trưng hình ảnh tiên tiến hơn cũng như xử lý ngôn ngữ tự nhiên tốt hơn

Đề đối mặt với thách thức ngày càng phức tạp và đa dạng của việc mô tả hình ảnh, cần tiếp tục nghiên cứu và cải tiến. Càng hiểu rõ về cách kết hợp thông tin hình ảnh và ngôn ngữ một cách tinh tế, mô hình sẽ trở nên mạnh mẽ hơn trong việc tạo ra các mô tả không chỉ chính xác mà còn phản ánh đúng bản chất và ngữ cảnh của hình ảnh. Tính ổn định và độ tin cậy của mô hình là chìa khóa để đảm bảo ứng dụng thực tế. Cần duy trì sự nhạy bén và khả năng đổi mới với những tình huống đa dạng để mô hình có thể đáp ứng một cách linh hoạt và hiệu quả. Những nỗ lực này không chỉ mở ra những triển vọng mới trong lĩnh vực Image Captioning mà còn góp phần quan trọng vào sự phát triển của trí tuệ nhân tạo tổng thể.

Trích nguồn

- [1] Tanti, Marc, Albert Gatt, and Kenneth P. Camilleri. "Where to put the image in an image caption generator." *Natural Language Engineering* 24.3 (2018): 467-489.
- [2] Xu, Kelvin, et al. "Show, attend and tell: Neural image caption generation with visual attention." *International conference on machine learning*. PMLR, 2015
- [3] Vinyals, Oriol, et al. "Show and tell: A neural image caption generator." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015.
- [4] Papineni, Kishore, et al. "Bleu: a method for automatic evaluation of machine translation." *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*. 2002.

Github :

[https://github.com/Zui1506/Image_caption/blob/66fbbc6a6e69adf731acab34a29250ff5e8b074a/Image_caption%20\(Xception%20BLSTM\).ipynb](https://github.com/Zui1506/Image_caption/blob/66fbbc6a6e69adf731acab34a29250ff5e8b074a/Image_caption%20(Xception%20BLSTM).ipynb)