TRƯỜNG ĐẠI HỌC CÔNG NGHỆ ĐÔNG Á
KHOA CÔNG NGHỆ THÔNG TIN

BÀI TẬP LỚN

HỌC PHẦN: XỬ LÝ ẢNH VÀ THỊ GIÁC MÁY TÍNH

Đề tài số 19: Xây dựng hệ thống để tạo mô hình chú thích ảnh bằng cách sử dụng Attention Mechanism với CNN và RNN

Giảng viên hướng dẫn: Lương Thị Hồng Lan

TT	Mã sinh viên	Sinh viên thực hiện	Lớp hành chính
1	20210560	Nguyễn Thành Nam	DCCNTT12.10.2
2	20210525	Đăng Quang Vũ	DCCNTT12.10.2
3	20210573	Khuất Hồng Ly	DCCNTT12.10.2

Bắc Ninh, năm 2024

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ ĐÔNG Á
KHOA CÔNG NGHỆ THÔNG TIN

BÀI TẬP LỚN

HỌC PHẦN: XỬ LÝ ẢNH VÀ THỊ GIÁC MÁY TÍNH

Đề tài số 19: Xây dựng hệ thống để tạo mô hình chú thích ảnh bằng cách sử dụng Attention Mechanism với CNN và RNN

Giảng viên hướng dẫn: Lương Thị Hồng Lan

TT	Mã sinh viên	Sinh viên thực hiện	Lớp hành chính
1	20210560	Nguyễn Thành Nam	DCCNTT12.10.2
2	20210525	Đăng Quang Vũ	DCCNTT12.10.2
3	20210573	Khuất Hồng Ly	DCCNTT12.10.2

Bắc Ninh, năm 2024

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ ĐÔNG Á
KHOA CÔNG NGHỆ THÔNG TIN

KỲ THI KẾT THÚC HỌC PHẦN
HỌC KỲ 1, NĂM HỌC 2024 – 2025

PHIẾU CHẤM THI BÀI TẬP LỚN KẾT THÚC HỌC PHẦN
Mã đề thi: 19
Tên học phần: Xử lý ảnh và thị giác máy tính
Lớp Tín chỉ: XATGMT.03.K12.02.LH.C04.1_LT

Cán bộ chấm thi 1
(Ký và ghi rõ họ tên)

Cán bộ chấm thi 2
(Ký và ghi rõ họ tên)

TT	TIÊU CHÍ	THANG ĐIỂM	Nguyễn Thành Nam	Đặng Quang Vũ	Khuất Hồng Ly
			20210560	20210525	20210573
1	**Nội dung báo cáo trên Word đầy đủ**	**3.5**			
1.1	Có bố cục rõ ràng (mục lục, phần mở đầu, nội dung chính, kết luận).	0,5			
1.2	Nội dung phân tích rõ ràng, logic.	0,5			
1.3	Có dẫn chứng, số liệu minh họa đầy đủ.	0,5			
1.4	Ngôn ngữ và trình bày chuẩn, không lỗi chính tả.	0,5			
1.5	Có trích dẫn tài liệu tham khảo đúng quy cách.	0,5			
1.6	Được trình bày chuyên nghiệp (canh lề, font chữ, khoảng cách dòng hợp lý).	0,5			
1.7	Tài liệu đầy đủ, bám sát yêu cầu của đề bài.	0,5			
2	**Nội dung thuyết trình đầy đủ**	**1.0**			
2.1	Trình bày tự tin, phát âm rõ ràng, mạch lạc.	0,5			

TT	TIÊU CHÍ	THANG ĐIỂM	Nguyễn Thành Nam	Đặng Quang Vũ	Khuất Hồng Ly
			20210560	20210525	20210573
	Nội dung thuyết trình đúng trọng tâm, không lan man.	0,5			
3	Slides báo cáo đầy đủ nội dung + Hỏi đáp	3.0			
3.1	Slides có bố cục rõ ràng (mở đầu, nội dung, kết luận).	0,5			
3.2	Thiết kế slides đẹp, chuyên nghiệp (màu sắc, hình ảnh minh họa).	0,5			
3.3	Nội dung trên slides ngắn gọn, dễ hiểu, súc tích.	0,5			
3.4	Nội dung slides phù hợp với nội dung báo cáo.	0,5			
3.5	Trả lời câu hỏi đầy đủ, chính xác.	0,5			
3.6	Trả lời câu hỏi tự tin, thuyết phục.	0,5			
4	**Code đầy đủ**	**2.5**			
1.1	Code được trình bày rõ ràng, có chú thích đầy đủ.	0,5			
1.2	Code chạy đúng, không lỗi.	0,5			
1.3	Code tối ưu, không dư thừa.	0,5			
1.4	Đáp ứng đầy đủ các yêu cầu chức năng theo đề bài.	0,5			
1.5	Có tính sáng tạo hoặc cải thiện so với yêu cầu.	0,5			
TỔNG ĐIỂM BẰNG SỐ:		**10**			

TT	TIÊU CHÍ	THANG ĐIỂM	Nguyễn Thành Nam	Đặng Quang Vũ	Khuất Hồng Ly
			20210560	20210525	20210573
TỔNG ĐIỂM BẰNG CHỮ:		*Mười tròn*			

Muc lục

DANH MỤC HÌNH

LỜI MỞ ĐẦU

Trong thời đại công nghệ phát triển như hiện nay, trí tuệ nhân tạo (AI) và học sâu (Deep Learning) đang ngày càng chứng tỏ vai trò quan trọng trong việc giải quyết các bài toán thực tế. Một trong những ứng dụng nổi bật của AI là hệ thống tạo chú thích tự động cho hình ảnh (Image Captioning) – một bài toán kết hợp giữa xử lý hình ảnh và xử lý ngôn ngữ tự nhiên.

Dự án "Hệ thống tạo chú thích ảnh bằng Attention Mechanism với CNN và RNN" được thực hiện với mục tiêu xây dựng một hệ thống có khả năng nhận diện nội dung trong hình ảnh và mô tả chúng bằng câu văn một cách chính xác và tự nhiên. Bằng cách sử dụng các công nghệ hiện đại như mạng nơ-ron tích chập (CNN), mạng nơ-ron hồi tiếp (RNN) và Attention Mechanism, dự án đã mang đến một hướng tiếp cận hiệu quả, phù hợp với các yêu cầu hiện nay trong lĩnh vực trí tuệ nhân tạo.

Báo cáo này sẽ trình bày toàn bộ quá trình thực hiện dự án, từ việc phân tích bài toán, chuẩn bị dữ liệu, thiết kế và triển khai mô hình, cho đến đánh giá kết quả. Với mong muốn áp dụng kiến thức đã học vào thực tiễn, dự án không chỉ là cơ hội để em rèn luyện kỹ năng mà còn mở ra tiềm năng phát triển cho các ứng dụng AI trong tương lai.

Em hy vọng rằng báo cáo này sẽ thể hiện rõ ràng quá trình làm việc, những kết quả đạt được,

LỜI CẢM ƠN

Chúng em xin gửi lời cảm ơn chân thành và sâu sắc nhất đến cô Lương Thị Hồng Lan, người đã tận tình hướng dẫn chúng em trong suốt quá trình thực hiện dự án "Hệ thống tạo chú thích ảnh bằng Attention Mechanism với CNN và RNN".

Trong suốt quá trình học tập và nghiên cứu, cô đã luôn dành thời gian để giải đáp các thắc mắc, đưa ra những nhận xét và góp ý quý báu, giúp chúng em hiểu sâu hơn về các kiến thức chuyên môn cũng như cách áp dụng lý thuyết vào thực tiễn. Sự nhiệt huyết, tận tâm và trách nhiệm của cô đã truyền cảm hứng to lớn cho chúng em, không chỉ trong dự án này mà còn trong cả quá trình học tập và định hướng nghề nghiệp sau này.

Bên cạnh việc truyền đạt kiến thức chuyên môn, cô còn dạy chúng em những kỹ năng quan trọng như cách tư duy khoa học, quản lý thời gian hiệu quả và cách giải quyết vấn đề một cách logic. Sự động viên và chỉ dẫn tận tình của cô đã giúp chúng em vượt qua những khó khăn, thách thức trong quá trình triển khai dự án.

Từ việc định hướng ý tưởng, phân tích bài toán, xử lý dữ liệu cho đến thiết kế và hoàn thiện mô hình, cô luôn đồng hành, tận tình hướng dẫn chúng em từng bước một. Những đóng góp và sự hỗ trợ của cô đã giúp chúng em hoàn thiện dự án một cách tốt nhất, đồng thời tích lũy được nhiều bài học kinh nghiệm quý giá cho bản thân.

Dự án này là một cột mốc quan trọng đối với chúng em, không chỉ giúp áp dụng những kiến thức đã học mà còn mở ra cơ hội để hiểu thêm về tiềm năng ứng dụng của trí tuệ nhân tạo trong thực tiễn. Chúng em nhận thức rõ rằng kết quả đạt được hôm nay là nhờ sự hướng dẫn tận tâm của cô.

Chúng em xin kính chúc cô luôn dồi dào sức khỏe, hạnh phúc và thành công trong sự nghiệp giảng dạy. Hy vọng rằng trong tương lai, chúng em sẽ tiếp tục có cơ hội được học hỏi và làm việc cùng cô.

Chương 1: Tổng quan bài toán

1.1 Nhận dạng bài toán

1.1.1 Mô tả bài toán

Trong bối cảnh sự phát triển vượt bậc của trí tuệ nhân tạo, việc tạo mô hình chú thích ảnh tự động đã trở thành một trong những bài toán hấp dẫn và mang lại nhiều ứng dụng thực tiễn. Mục tiêu của bài toán là xây dựng một hệ thống có khả năng tạo ra mô tả ngôn ngữ tự nhiên (caption) phù hợp, chính xác và chi tiết cho một hình ảnh đầu vào. Điều này đòi hỏi sự kết hợp giữa hai lĩnh vực: xử lý ảnh (Computer Vision) để nhận diện các đối tượng và trích xuất thông tin hình ảnh, cùng xử lý ngôn ngữ tự nhiên (Natural Language Processing) để sinh ra câu mô tả ngữ nghĩa.

Tuy nhiên, để đạt được mục tiêu này, hệ thống phải đối mặt với các thách thức như:

- Xử lý và nhận diện chính xác các đối tượng trong ảnh có độ phức tạp cao.
- Hiểu mối quan hệ giữa các đối tượng trong ảnh để tạo ra chú thích hợp lý.
- Tối ưu hóa khả năng sinh ngôn ngữ tự nhiên để đảm bảo câu văn rõ ràng, mạch lạc và phù hợp ngữ cảnh.

1.1.2 Định nghĩa nhận dạng đối tượng

Nhận dạng đối tượng là bước cơ bản và quan trọng trong bài toán tạo chú thích ảnh, bao gồm các công việc:

- *Xác định các đối tượng chính*: Hệ thống cần phân tích và xác định các thành phần quan trọng trong ảnh (ví dụ: con người, động vật, đồ vật).
- *Phân loại đối tượng*: Sau khi nhận diện, cần phân loại đối tượng vào các danh mục cụ thể để cung cấp thông tin đầu vào hữu ích cho bước sinh ngôn ngữ.
- *Xác định mối quan hệ giữa các đối tượng*: Đây là bước giúp hệ thống hiểu ngữ cảnh của ảnh, từ đó sinh câu chú thích mang tính logic và ý nghĩa.

Vấn đề đặt ra đối với nhận dạng đối tượng bao gồm:

- Độ chính xác: Làm thế nào để hệ thống nhận diện chính xác các đối tượng trong ảnh, đặc biệt là trong các trường hợp có độ phức tạp cao như ảnh mờ, đối tượng chồng lấn hoặc nền nhiễu.
- Khả năng xử lý: Hệ thống cần đảm bảo hiệu năng xử lý tốt để xử lý được lượng lớn dữ liệu ảnh.
- Tính khái quát: Mô hình phải có khả năng nhận diện được các đối tượng thuộc nhiều loại khác nhau, kể cả những đối tượng chưa được huấn luyện trước đó.

1.2 Các kỹ thuật sử dụng trong bài toán

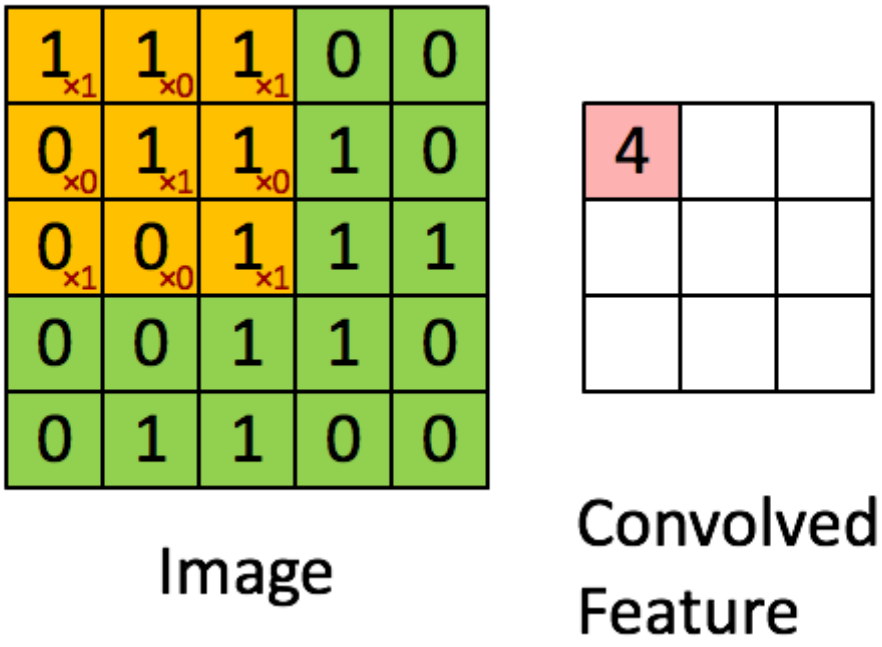
1.2.1 Convolutional Neural Networks (CNN)

Convolutional Neural Network (CNNs – Mạng nơ-ron tích chập) là một trong những mô hình Deep Learning tiên tiến. Nó giúp cho chúng ta xây dựng được những hệ thống thông minh với độ chính xác cao như hiện nay.

CNN được sử dụng nhiều trong các bài toán nhận dạng các object trong ảnh. Để tìm hiểu tại sao thuật toán này được sử dụng rộng rãi cho việc nhận dạng (detection)

Tìm hiểu Convolutional là gì?

Là một cửa sổ trượt (Sliding Windows) trên một ma trận như mô tả hình dưới:



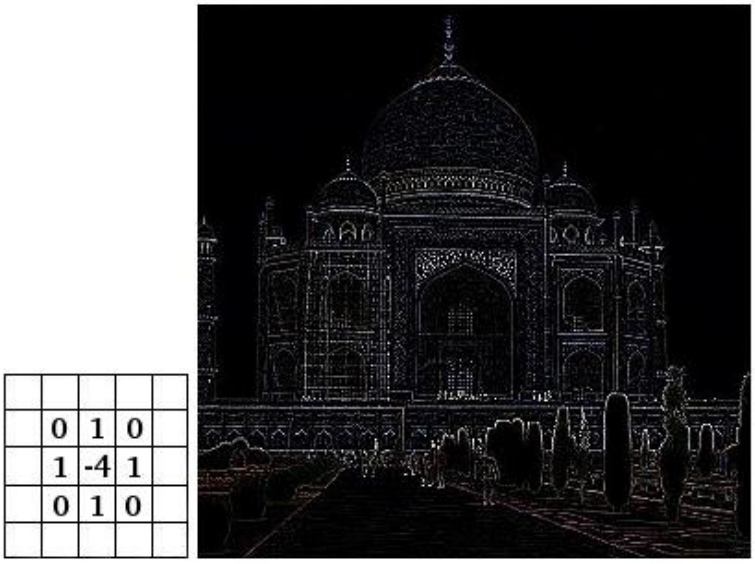
Hình 1.1 Mô tả Convolutional

Các convolutional layer có các parameter(kernel) đã được học để tự điều chỉnh lấy ra những thông tin chính xác nhất mà không cần chọn các feature.

Trong hình ảnh ví dụ trên, ma trận bên trái là một hình ảnh trắng đen được số hóa. Ma trận có kích thước **5×5** và mỗi điểm ảnh có giá trị 1 hoặc 0 là giao điểm của dòng và cột.

Convolution hay tích chập là nhân từng phần tử trong ma trận 3. Sliding Window hay còn gọi là kernel, filter hoặc feature detect là một ma trận có kích thước nhỏ như trong ví dụ trên là 3×3.

Convolution hay tích chập là nhân từng phần tử bên trong ma trận 3×3 với ma trận bên trái. Kết quả được một ma trận gọi là Convoled feature được sinh ra từ việc nhận ma trận Filter với ma trận ảnh 5×5 bên trái.



	0	1	0	
	1	-4	1	
	0	1	0	

0	0	0	0	0
0	1	1	1	0
0	1	1	1	0
0	1	1	1	0
0	0	0	0	0

Hình 1.2 Kết quả Convoled feature

Cấu trúc mạng CNN

Mạng CNN là một tập hợp các lớp Convolution chồng lên nhau và sử dụng các hàm nonlinear activation như ReLU và tanh để kích hoạt các trọng số trong các node. Mỗi một lớp sau khi thông qua các hàm kích hoạt sẽ tạo ra các thông tin trừu tượng hơn cho các lớp tiếp theo.

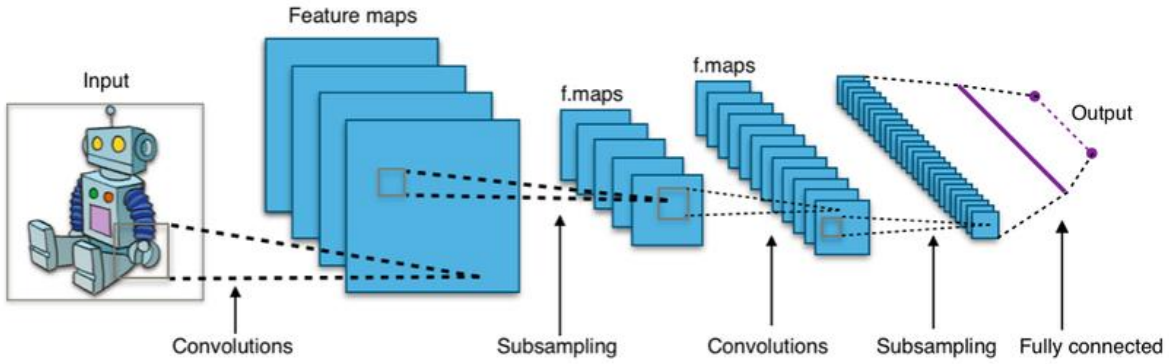
Mỗi một lớp sau khi thông qua các hàm kích hoạt sẽ tạo ra các thông tin trừu tượng hơn cho các lớp tiếp theo. Trong mô hình mạng truyền ngược (feedforward neural network) thì mỗi neural đầu vào (input node) cho mỗi neural đầu ra trong các lớp tiếp theo.

Mô hình này gọi là mạng kết nối đầy đủ (fully connected layer) hay mạng toàn vẹn (affine layer). Còn trong mô hình CNNs thì ngược lại. Các layer liên kết được với nhau thông qua cơ chế convolution.

Layer tiếp theo là kết quả convolution từ layer trước đó, nhờ vậy mà ta có được các kết nối cục bộ. Như vậy mỗi neuron ở lớp kế tiếp sinh ra từ kết quả của filter áp đặt lên một vùng ảnh cục bộ của neuron trước đó.

Mỗi một lớp được sử dụng các filter khác nhau thông thường có hàng trăm hàng nghìn filter như vậy và kết hợp kết quả của chúng lại. Ngoài ra có một số layer khác như pooling/subsampling layer dùng để chắt lọc lại các thông tin hữu ích hơn (loại bỏ các thông tin nhiễu).

Trong quá trình huấn luyện mạng (traning) CNN tự động học các giá trị qua các lớp filter dựa vào cách thức mà bạn thực hiện. Ví dụ trong tác vụ phân lớp ảnh, CNNs sẽ cố gắng tìm ra thông số tối ưu cho các filter tương ứng theo thứ tự raw pixel > edges > shapes > facial > high-level features. Layer cuối cùng được dùng để phân lớp ảnh.



Hình 1.3 CNN tự động học các giá trị qua các lớp filter

Trong mô hình CNN có 2 khía cạnh cần quan tâm là **tính bất biến** (Location Invariance) và **tính kết hợp** (Compositionality). Với cùng một đối tượng, nếu đối tượng này được chiếu theo các gốc độ khác nhau (translation, rotation, scaling) thì độ chính xác của thuật toán sẽ bị ảnh hưởng đáng kể.

Pooling layer sẽ cho bạn tính bất biến đối với phép dịch chuyển (translation), phép quay (rotation) và phép co giãn (scaling). Tính kết hợp cục bộ cho ta các cấp độ biểu diễn thông tin từ mức độ thấp đến mức độ cao và trừu tượng hơn thông qua convolution từ các filter.

Đó là lý do tại sao CNNs cho ra mô hình với độ chính xác rất cao. Cũng giống như cách con người nhận biết các vật thể trong tự nhiên.

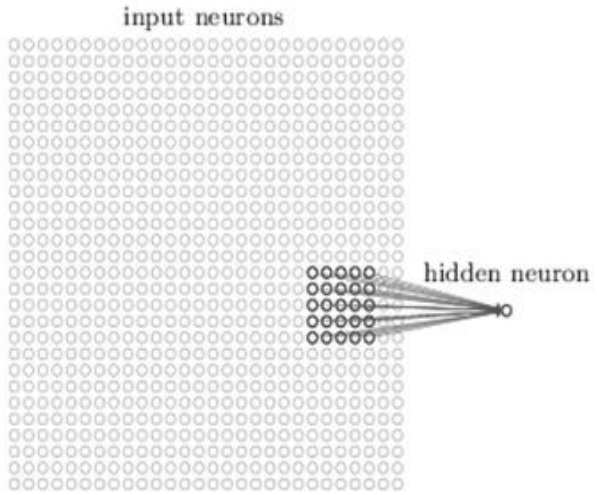
Mạng CNN sử dụng 3 ý tưởng cơ bản:

- các trường tiếp nhận cục bộ (local receptive field)
- trọng số chia sẻ (shared weights)
- tổng hợp (pooling).

Trường tiếp nhận cục bộ (local receptive field)

Đầu vào của mạng CNN là một ảnh. Ví dụ như ảnh có kích thước 28×28 thì tương ứng đầu vào là một ma trận có 28×28 và giá trị mỗi điểm ảnh là một ô trong ma trận. Trong mô hình mạng ANN truyền thống thì chúng ta sẽ kết nối các neuron đầu vào vào tầng ảnh.

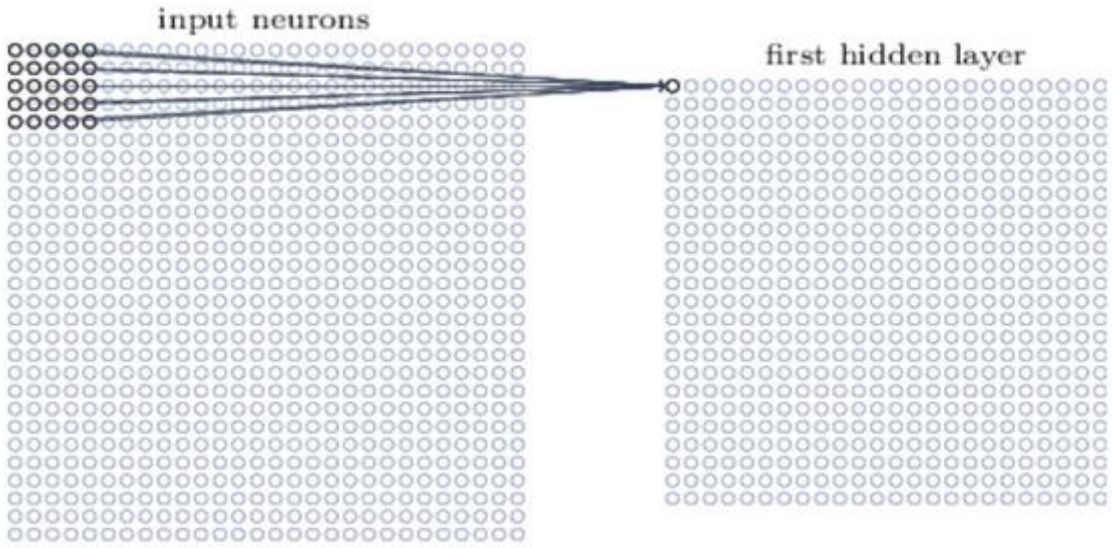
Tuy nhiên trong CNN chúng ta không làm như vậy mà chúng ta chỉ kết nối trong một vùng nhỏ của các neuron đầu vào như một filter có kích thước 5×5 tương ứng (28- 5 + 1) 24 điểm ảnh đầu vào. Mỗi một kết nối sẽ học một trọng số và mỗi neuron ẩn sẽ học một bias. Mỗi một vùng 5×5 đấy gọi là một trường tiếp nhận cục bộ.



Hình 1.4 Một trường tiếp nhận cục bộ

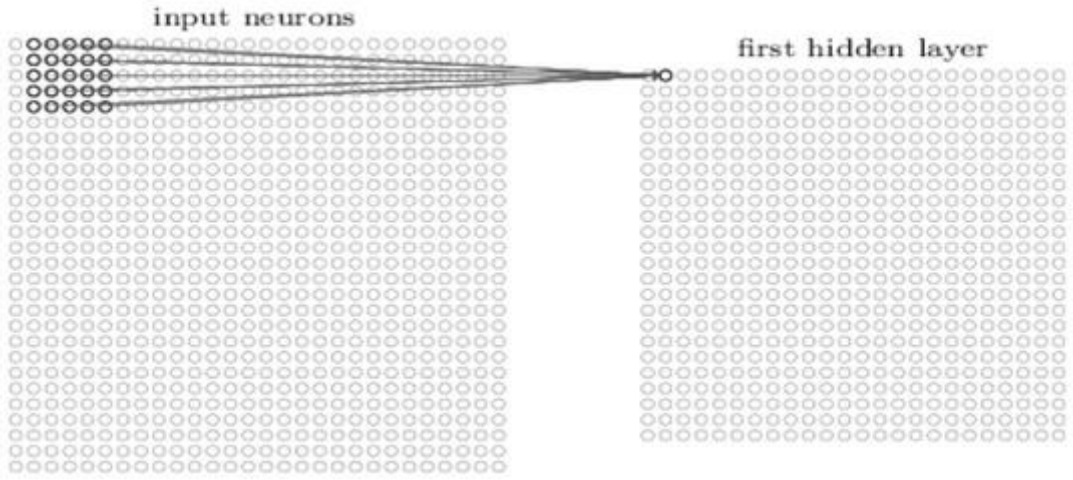
Một cách tổng quan, ta có thể tóm tắt các bước tạo ra 1 hidden layer bằng các cách sau:

1. Tạo ra neuron ẩn đầu tiên trong lớp ẩn 1



Hình 1.5 Tạo ra neuron ẩn đầu tiên trong lớp ẩn 1

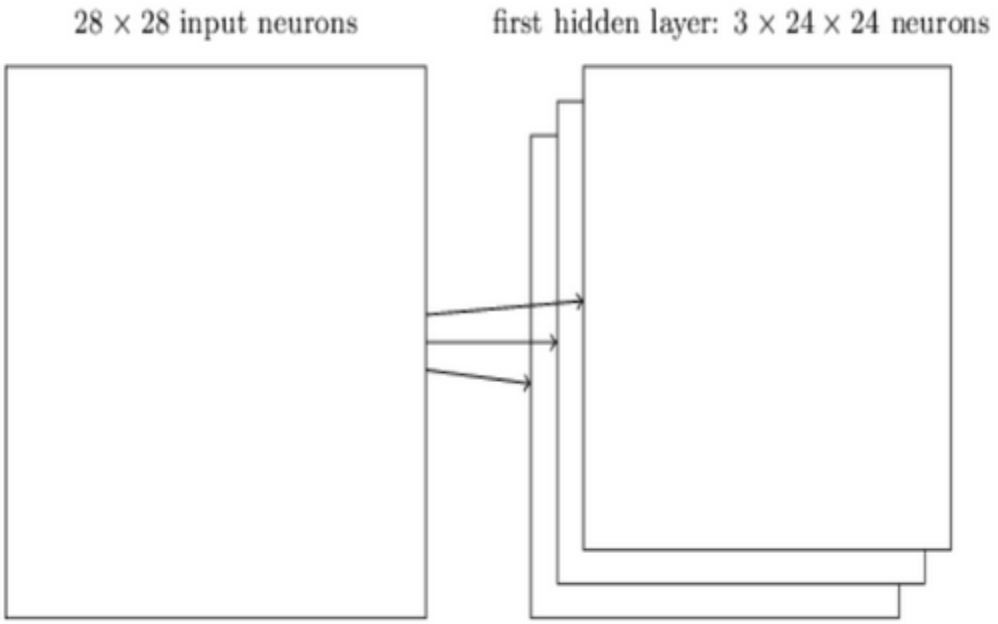
2. Dịch filter qua bên phải một cột sẽ tạo được neuron ẩn thứ 2.



Hình 1.6 Tạo ra neuron ẩn thứ 2

Với bài toán nhận dạng ảnh người ta thường gọi ma trận lớp đầu vào là feature map, trọng số xác định các đặc trương là shared weight và độ lệch xác định một feature map là shared bias. Như vậy đơn giản nhất là qua các bước trên chúng ta chỉ có 1 feature map.

Tuy nhiên trong nhận dạng ảnh chúng ta cần nhiều hơn một feature map.



Hình 1.7 Mô tả phân tách dữ liệu ảnh

Như vậy, local receptive field thích hợp cho việc phân tách dữ liệu ảnh, giúp chọn ra những vùng ảnh có giá trị nhất cho việc đánh giá phân lớp.

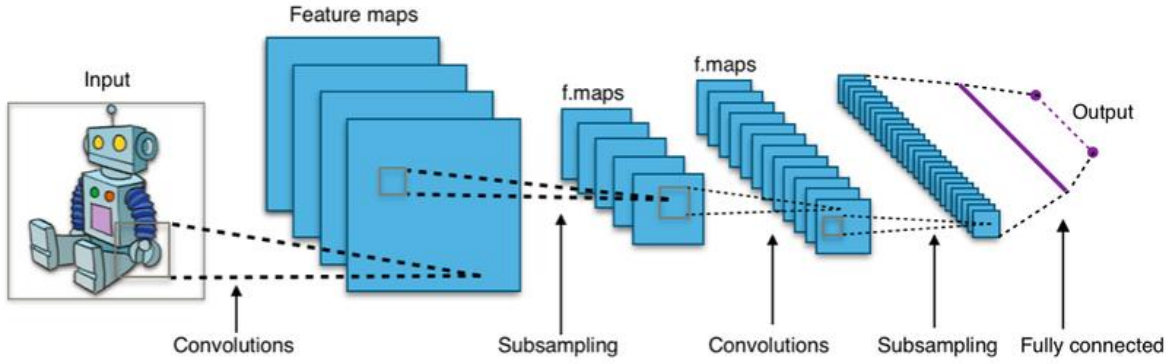
Trọng số chia sẻ (shared weight and bias)

Đầu tiên, các trọng số cho mỗi filter (kernel) phải giống nhau. Tất cả các nơ-ron trong lớp ẩn đầu sẽ phát hiện chính xác feature tương tự chỉ ở các vị trí khác nhau trong hình ảnh đầu vào. Chúng ta gọi việc map từ input layer sang hidden layer là một feature map. Vậy mối quan hệ giữa số lượng Feature map với số lượng tham số là gì?

Tóm lại, một convolutional layer bao gồm các feature map khác nhau. Mỗi một feature map giúp detect một vài feature trong bức ảnh. Lợi ích lớn nhất của trọng số chia sẻ là giảm tối đa số lượng tham số trong mạng CNN.

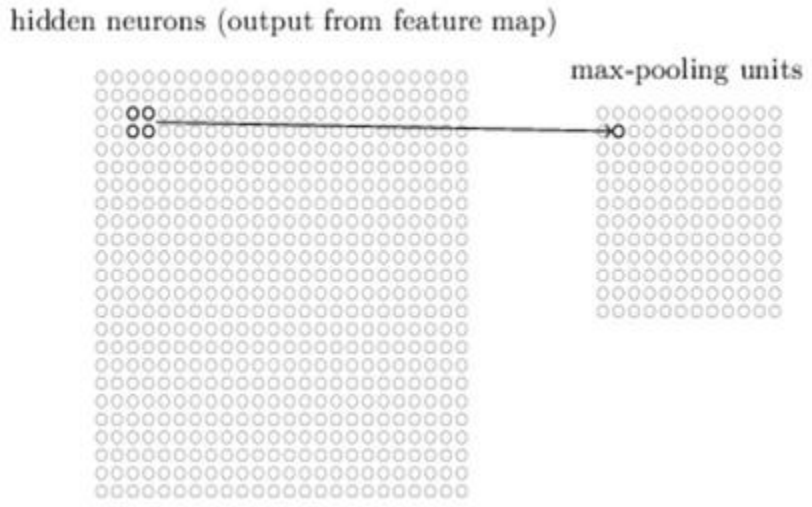
Lớp tổng hợp (pooling layer)

Lớp pooling thường được sử dụng ngay sau lớp convulational để đơn giản hóa thông tin đầu ra để giảm bớt số lượng neuron.



Hình 1.8 Quá trình huấn luyện mạng (traning) CNN

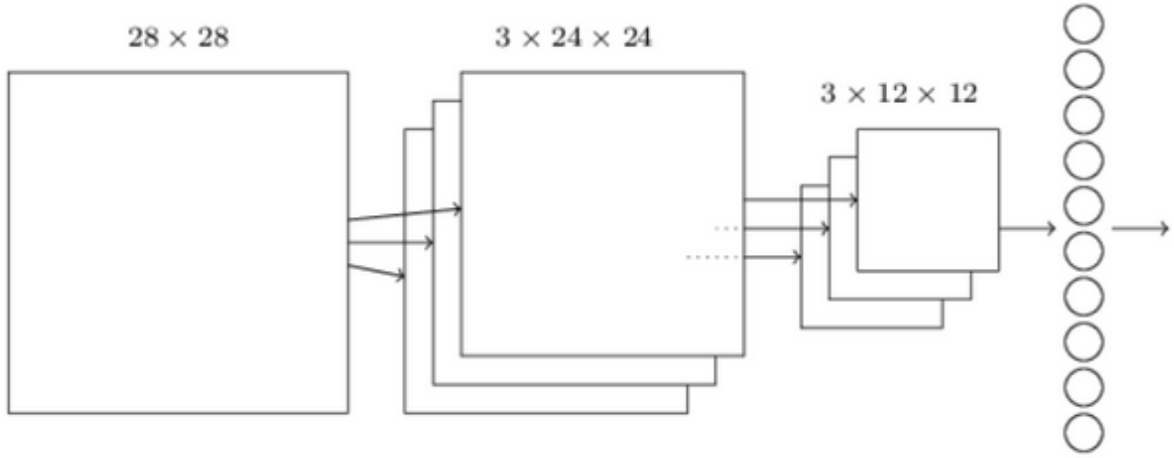
Thủ tục pooling phổ biến là max-pooling, thủ tục này chọn giá trị lớn nhất trong vùng đầu vào 2×2.



Hình 1.9 Mô tả thủ tục max-pooling

Như vậy qua lớp Max Pooling thì số lượng neuron giảm đi phân nửa. Trong một mạng CNN có nhiều Feature Map nên mỗi Feature Map chúng ta sẽ cho mỗi Max Pooling khác nhau. Chúng ta có thể thấy rằng Max Pooling là cách hỏi xem trong các đặc trưng này thì đặc trưng nào là đặc trưng nhất. Ngoài Max Pooling còn có L2 Pooling.

Cuối cùng ta đặt tất cả các lớp lại với nhau thành một CNN với đầu ra gồm các neuron với số lượng tùy bài toán.



Hình 1.10 Tất cả các lớp đặt lại với nhau thành một CNN với đầu ra gồm các neuron

2 lớp cuối cùng của các kết nối trong mạng là một lớp đầy đủ kết nối (fully connected layer) . Lớp này nối mọi nơron từ lớp max pooled tới mọi nơron của tầng ra.

Cách chọn tham số cho CNN

1. Số các convolution layer: càng nhiều các convolution layer thì performance càng được cải thiện. Sau khoảng 3 hoặc 4 layer, các tác động được giảm một cách đáng kể
2. Filter size: thường filter theo size 5×5 hoặc 3×3
3. Pooling size: thường là 2×2 hoặc 4×4 cho ảnh đầu vào lớn
4. Cách cuối cùng là thực hiện nhiều lần việc train test để chọn ra được param tốt nhất.

1.2.2 Recurrent Neural Networks (RNN/LSTM)

Vai trò:

Đầu tiên, hãy nhìn xem RNN có thể làm gì. Dưới đây là một vài ví dụ.

- Machine Translation (Dịch máy)
- Mô hình hóa ngôn ngữ và sinh văn bản: đây có lẽ là khả năng ấn tượng nhất đối với mình.
- Nhận dạng giọng nói

- Mô tả hình ảnh: RNN kết hợp cùng CNN để sinh ra mô tả cho hình ảnh chưa được gán nhãn. Đây cũng là một bài tập khá hay mà mình sẽ giới thiệu trong bài viết tiếp theo.

Ưu điểm:

- Xử lý hiệu quả dữ liệu tuần tự
- Khả năng học phụ thuộc dài hạn (LSTM)
- Linh hoạt với độ dài chuỗi đầu vào

Nhược điểm:

- Khó khăn trong việc xử lý chuỗi rất dài
- Vấn đề gradient vanishing/exploding
- Tốc độ training chậm

Giải quyết được:

RNN và LSTM giúp chuyển đổi thông tin đặc trưng từ ảnh thành các câu mô tả ngôn ngữ tự nhiên, đảm bảo rằng câu văn được tạo ra có tính mạch lạc và đúng ngữ cảnh.

1.2.3 Attention Mechanism

Vai trò: Attention Mechanism tập trung vào các vùng quan trọng trong ảnh, giúp hệ thống ưu tiên xử lý các đối tượng hoặc đặc điểm đáng chú ý nhất.

Ưu điểm:

- Cải thiện độ chính xác của mô tả
- Tập trung vào vùng ảnh phù hợp với từng từ
- Giảm thiểu vấn đề mất thông tin của RNN

Nhược điểm:

- Tăng độ phức tạp tính toán
- Cần nhiều bộ nhớ hơn
- Khó khăn trong việc training

Giải quyết được:

Attention Mechanism giúp hệ thống tập trung vào các phần quan trọng của ảnh, hỗ trợ việc nhận dạng chính xác đối tượng và mối quan hệ giữa chúng, từ đó sinh ra chú thích ngữ nghĩa chính xác hơn.

1.2.4 Encoder-Decoder Architecture

Vai trò:

Đây là kiến trúc tổng thể kết hợp CNN (Encoder) và RNN (Decoder) để trích xuất đặc trưng và sinh câu chú thích.

Ưu điểm:

- Tích hợp tốt giữa xử lý ảnh và ngôn ngữ tự nhiên.
- Linh hoạt, dễ mở rộng và cải tiến.

Nhược điểm:

- Phụ thuộc vào cả hai phần Encoder và Decoder, nếu một trong hai hoạt động kém, toàn bộ hệ thống sẽ bị ảnh hưởng.
- Đòi hỏi cấu hình máy tính mạnh để xử lý.

Giải quyết được:

Kiến trúc này giúp kết nối liền mạch giữa việc xử lý đặc trưng từ ảnh và sinh câu, tạo ra hệ thống chú thích ảnh tự động hoàn chỉnh.

1.2.5 Optimization Algorithms (SGD, Adam)

Vai trò:

Tối ưu hóa các tham số của mô hình để cải thiện hiệu năng và độ chính xác.

Ưu điểm:

- Adam kết hợp tốt giữa tốc độ hội tụ nhanh và ổn định.
- Giúp mô hình đạt độ chính xác cao hơn.

Nhược điểm:

- Cần chọn siêu tham số cẩn thận để tránh overfitting hoặc underfitting.

Giải quyết được:

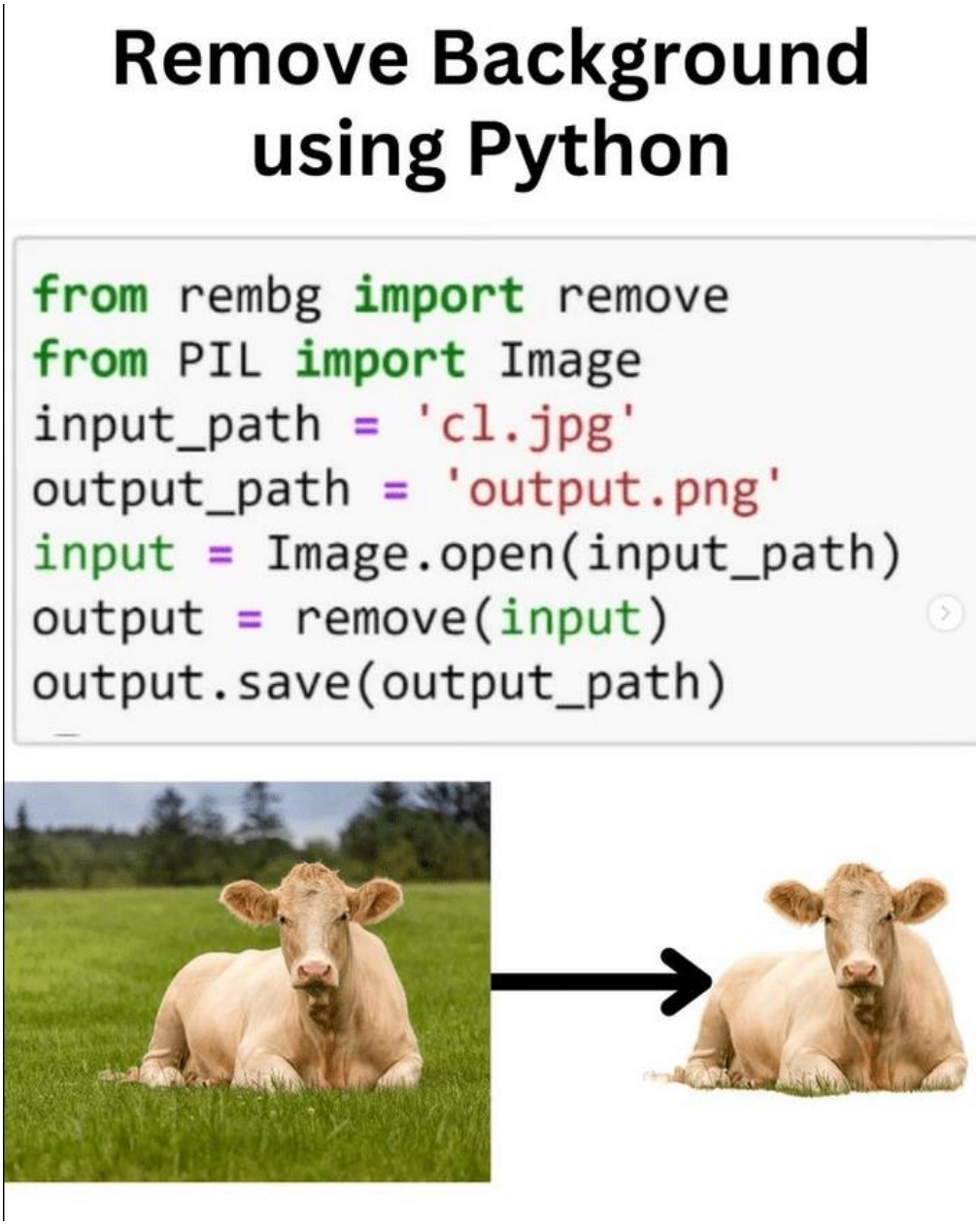
Các thuật toán tối ưu đảm bảo mô hình học tốt từ dữ liệu, đạt hiệu năng cao trong việc nhận dạng đối tượng và sinh chú thích ảnh.

1.3 Ngôn ngữ lập trình và các thư viện sử dụng

1.3.1 Python

Tại sao lại chọn ngôn ngữ Python?

- Python là một ngôn ngữ lập trình bậc cao, mã nguồn mở và đa nền tảng. Python được sử dụng rộng rãi để phát triển các ứng dụng web, phát triển phần mềm, khoa học dữ liệu và máy học (ML).
- Cú pháp đơn giản và dễ đọc: Cú pháp của Python rất giống với ngôn ngữ tiếng Anh tự nhiên, dễ học và dễ đọc, giúp lập trình viên tập trung vào giải quyết vấn đề hơn là việc ghi nhớ cú pháp phức tạp.
- Đa dụng: Python có thể được sử dụng trong nhiều lĩnh vực khác nhau như phát triển web, khoa học dữ liệu, trí tuệ nhân tạo, tự động hóa, phân tích dữ liệu, và nhiều ứng dụng khác.
- Thư viện phong phú: Python có một kho thư viện đồ sộ, hỗ trợ nhiều tác vụ khác nhau. Các thư viện như NumPy, Pandas, TensorFlow, và Django giúp lập trình viên tiết kiệm thời gian và công sức khi phát triển các ứng dụng phức tạp.
- Cộng đồng lớn mạnh: Python có một cộng đồng lập trình viên rộng lớn và năng động, cung cấp nhiều tài liệu, hướng dẫn, và hỗ trợ qua các diễn đàn, nhóm thảo luận và các khóa học trực tuyến.
- Đa nền tảng: Python có thể chạy trên nhiều hệ điều hành khác nhau như Windows, macOS, Linux, Raspberry Pi,… giúp cho việc phát triển và triển khai ứng dụng trở nên dễ dàng.
- Khả năng mở rộng và tích hợp tốt: Python có thể dễ dàng tích hợp với các ngôn ngữ lập trình khác và các công nghệ hiện có, giúp nó trở thành một lựa chọn lý tưởng cho nhiều dự án khác nhau.
- Hỗ trợ từ các tổ chức lớn: Nhiều công ty và tổ chức lớn như Google, Facebook, NASA sử dụng Python và đóng góp vào việc phát triển ngôn ngữ này, làm tăng uy tín và sự tin cậy của nó.
- Xử lý Hình ảnh và Video
- OpenCV là thư viện nổi tiếng cho xử lý hình ảnh và video trong Python. Nó cho phép bạn thực hiện các tác vụ từ nhận diện khuôn mặt, theo dõi đối tượng, đến xử lý video thời gian thực. Python giúp bạn tạo ra các ứng dụng xử lý hình ảnh mạnh mẽ và hiệu quả.



Hình 1.11 Tách nền với python

1.3.2 Các thư viện chính

PyTorch (torch)

- Thư viện deep learning chính để xây dựng và huấn luyện mô hình
- Tính năng chính:
    - Định nghĩa và training neural networks
    - Tính toán trên GPU/CPU

- Automatic differentiation
- Quản lý bộ nhớ và tensor operations

Lợi ích của việc sử dụng các thư viện:

1. Hiệu quả phát triển:

- Giảm thời gian implement
- Tận dụng code đã được tối ưu
- Giảm thiểu bugs

2. Hiệu năng:

- Các thư viện được tối ưu về performance
- Hỗ trợ tính toán GPU
- Xử lý memory hiệu quả

3. Bảo trì và mở rộng:

- Code dễ đọc và maintain
- Dễ dàng update features
- Documentation đầy đủ

4. Cộng đồng và hỗ trợ:

- Cộng đồng lớn để hỗ trợ
- Nhiều resources và tutorials
- Cập nhật thường xuyên

Các thư viện này tạo nên một ecosystem hoàn chỉnh cho việc phát triển mô hình Image Captioning, từ xử lý dữ liệu đến training và evaluation model.

Chương 2: Xây dựng hệ thống để tạo mô hình chú thích ảnh bằng cách sử dụng Attention Mechanism với CNN và RNN

2.1 Yêu cầu của bài toán

Hệ thống tạo chú thích ảnh (Image Captioning) là một bài toán tích hợp giữa thị giác máy tính và xử lý ngôn ngữ tự nhiên (NLP), với m ục tiêu tạo ra các câu văn mô tả chính xác nội dung của một hình ảnh đầu vào. Để xây dựng hệ thống này, bài toán đặt ra các yêu cầu rõ ràng về đầu vào, đầu ra, chức năng và hiệu năng như sau:

2.1.1. Yêu cầu đầu vào

Hệ thống cần tiếp nhận và xử lý các dữ liệu đầu vào sau:

- Hình ảnh:
  - Ảnh đầu vào có thể thuộc nhiều định dạng như JPEG, PNG hoặc BMP.
  - Nội dung của ảnh đa dạng, có thể bao gồm nhiều đối tượng, bối cảnh và hành động khác nhau.
  - Kích thước ảnh không cố định, nhưng cần chuẩn hóa trước khi đưa vào hệ thống.
- Dữ liệu huấn luyện:
  - Tập hình ảnh: Một tập hợp hình ảnh lớn (khoảng vài nghìn đến vài trăm nghìn ảnh), với chất lượng và độ phân giải phù hợp để trích xuất đặc trưng.
  - Chú thích văn bản: Mỗi hình ảnh đi kèm từ 3 đến 5 câu chú thích ngắn gọn, mô tả chính xác nội dung của ảnh.
  - Ví dụ về dữ liệu huấn luyện:
    - Hình ảnh: Một bức ảnh chụp một con chó đang chạy trên bãi cỏ.
    - Chú thích:
      - “A dog is running on the grass.”
      - “A brown dog is playing in a field.”

2.1.2. Yêu cầu đầu ra

Hệ thống phải tạo ra các kết quả đầu ra đáp ứng các tiêu chí sau:

1. Chất lượng chú thích:
   - Câu văn được sinh ra phải mạch lạc, đúng ngữ pháp và phản ánh nội dung chính của hình ảnh.
   - Ưu tiên sự chính xác, ngắn gọn nhưng vẫn đầy đủ ý nghĩa.
2. Tính tự nhiên:
   - Câu văn không chỉ chính xác mà còn phải có ngữ điệu tự nhiên, giống như được viết bởi con người.
3. Tính khả thi:
   - Có khả năng mô tả các hình ảnh không có trong tập huấn luyện (khả năng tổng quát hóa).

2.1.3. Yêu cầu chức năng hệ thống

Hệ thống cần đáp ứng các chức năng chính sau:
1. Trích xuất đặc trưng ảnh (Image Feature Extraction):
- Sử dụng một mạng nơ-ron tích chập (CNN) để phân tích và trích xuất đặc trưng quan trọng từ hình ảnh đầu vào.
- Các đặc trưng này phải đại diện cho cấu trúc, màu sắc và nội dung chính của hình ảnh.
2. Sinh câu chú thích (Caption Generation):
- Dựa trên đặc trưng hình ảnh, hệ thống sử dụng mạng nơ-ron hồi tiếp (RNN) hoặc các biến thể như LSTM/GRU để tạo câu chú thích từng từ một.
3. Cơ chế chú ý (Attention Mechanism):
- Giúp mô hình tập trung vào các vùng quan trọng của hình ảnh trong quá trình sinh từng từ.
- Ví dụ: Nếu ảnh có cả người và xe đạp, mô hình có thể chú ý vào vùng chứa xe đạp khi sinh từ "bicycle".
4. Tích hợp xử lý song song:
- Cho phép xử lý nhiều ảnh đầu vào đồng thời mà không làm giảm hiệu suất.
2.1.4. Yêu cầu phi chức năng
1. Tốc độ:
- Hệ thống cần sinh câu chú thích trong thời gian ngắn, tối đa từ 1 đến 3 giây mỗi hình ảnh trên phần cứng thông thường.
2. Hiệu suất:
- Hệ thống phải tối ưu về mặt tài nguyên, sử dụng GPU/TPU để tăng tốc độ tính toán khi cần thiết.
3. Khả năng mở rộng:
- Có thể mở rộng để xử lý các tập dữ liệu lớn hơn hoặc thêm ngôn ngữ mới (ví dụ: tiếng Việt, tiếng Pháp).
4. Độ tin cậy:
- Chú thích sinh ra phải chính xác và ổn định trong nhiều trường hợp, từ ảnh đơn giản đến ảnh có nội dung phức tạp.

2.2 Xây dựng hệ thống

Quá trình xây dựng hệ thống được thực hiện qua các giai đoạn cụ thể sau:

1. Chuẩn bị dữ liệu
Lựa chọn tập dữ liệu:
Dùng các tập dữ liệu tiêu chuẩn như COCO (Common Objects in Context) hoặc Flickr8k/30k.
Mỗi hình ảnh đi kèm từ 5 đến 10 câu chú thích mô tả, đảm bảo tính đa dạng.
Tiền xử lý dữ liệu:
Hình ảnh:
Resize tất cả hình ảnh về kích thước chuẩn (224x224) để tương thích với các mô hình CNN.

Chuẩn hóa giá trị pixel về khoảng [0, 1] hoặc [-1, 1].
Chú thích:
Tokenize các câu chú thích thành danh sách các từ.
Loại bỏ các từ dư thừa, không cần thiết.
Xây dựng từ điển (vocabulary) ánh xạ từ thành chỉ số.
Chia dữ liệu:
80% cho tập huấn luyện, 10% cho tập kiểm tra, 10% cho tập kiểm định.
2. Thiết kế kiến trúc hệ thống
Hệ thống được chia làm ba phần chính:

(a) Encoder (CNN):

Sử dụng mạng CNN (như ResNet-50 hoặc Inception-v3) để trích xuất đặc trưng hình ảnh.
Đầu ra là một vector đặc trưng hoặc ma trận không gian đại diện cho các vùng ảnh khác nhau.
(b) Attention Mechanism:

Thực hiện chú ý (attention) trên các vùng đặc trưng để xác định vùng quan trọng nhất cần tập trung khi sinh từ tiếp theo.
Tính toán trọng số chú ý dựa trên ngữ cảnh hiện tại và đặc trưng ảnh.
(c) Decoder (RNN):

Sử dụng mạng RNN (hoặc biến thể LSTM/GRU) để sinh từng từ trong câu dựa trên đặc trưng ảnh và từ đã sinh trước đó.
Kết nối với một fully connected layer để chuyển đầu ra thành phân phối xác suất trên từ vựng.
3. Huấn luyện hệ thống
Hàm mất mát: SparseCategoricalCrossentropy để đo độ lệch giữa từ dự đoán và từ thực tế.
Tối ưu hóa: Dùng thuật toán Adam với learning rate được điều chỉnh thích hợp.
Thời gian huấn luyện: Huấn luyện trên GPU với batch size từ 32–64 và số epoch khoảng 20–50.
4. Đánh giá hệ thống
Tính điểm BLEU, METEOR và CIDEr trên tập kiểm tra để đánh giá chất lượng chú thích.
Hiển thị một số hình ảnh đầu vào kèm câu chú thích do mô hình sinh ra và so sánh với câu chú thích tham chiếu.
5. Triển khai
Kết nối với giao diện người dùng đơn giản: Người dùng tải ảnh lên, hệ thống sinh câu chú thích và hiển thị kết quả.
Tối ưu hóa mô hình để giảm thời gian suy luận khi xử lý ảnh trong môi trường thực tế.
Với quy trình xây dựng chi tiết này, hệ thống được kỳ vọng đáp ứng tốt các yêu cầu bài toán, đồng thời đảm bảo tính chính xác, hiệu quả và khả năng ứng dụng thực tế.

Chương 3: Thực Nghiệm

3.1 Dữ liệu

Dữ liệu đóng vai trò rất quan trọng trong các bài sử dụng học sâu nói chung hay chính xác là mạng nơron. Chất lượng, độ tin cậy, tính sẵn có và phù hợp của dữ liệu được sử dụng để xây dựng mô hình giúp nâng cao độ chính xác cho đầu ra của các bài toán. Kể cả với những mô hình đơn giản cũng có thể đạt được những kết quả tốt nếu như dữ liệu đầu vào đã được xử lý tốt, nắm giữ các thông tin quan trọng. Cùng với đó, các mô hình tuy rằng tốt có thể sẽ không cho ta các kết quả mong muốn nếu dữ liệu đầu vào phức tạp, rắc rối và chứa nhiều dữ liệu nhiễu. Việc xử lý dữ liệu bắt đầu bằng việc thu thập và phân tích dữ liệu, sau đó là bước tiền xử lý. Dữ liệu sau khi qua bước tiền xử lý được đưa vào mô hình. Cuối cùng, dữ liệu đầu ra của mạng nơron qua bước hậu xử lý, ở bước này sẽ thực hiện biến đổi kết quả trả về của mạng nơron sang dạng hiểu được yêu cầu của bài toán. Sau đây, ta xem quá trình xử lý dữ liệu.

3.1.1Nguồn dữ liệu:

- Dữ liệu sử dụng là bộ dữ liệu COCO (Common Objects in Context), một tập dữ liệu phổ biến cho các bài toán xử lý ngôn ngữ tự nhiên và thị giác máy tính.
- Cụ thể:
    - Ảnh: Lấy từ tập dữ liệu COCO năm 2017, chứa hình ảnh thực tế với các chú thích (captions) đi kèm.
    - Chú thích (captions): Lấy từ tệp annotations/captions_train2017.json (dùng cho huấn luyện) và annotations/captions_val2017.json (dùng cho kiểm tra).

Tiền xử lý dữ liệu:

- Hình ảnh:
    1. Resize: Tất cả các hình ảnh được điều chỉnh kích thước để cạnh ngắn nhất là 256 pixel.
    2. Crop: Lấy ngẫu nhiên một phần ảnh kích thước 224x224 để huấn luyện.
    3. Chuẩn hóa: Sử dụng giá trị trung bình và độ lệch chuẩn của tập dữ liệu ImageNet để chuẩn hóa ảnh.
    4. Tăng cường dữ liệu: Ảnh có thể bị lật ngang ngẫu nhiên với xác suất 50%.
- Chú thích (captions):
    - Làm sạch: Xóa các ký tự không cần thiết, viết thường toàn bộ.
    - Từ điển từ vựng: Tạo từ điển chỉ chứa các từ xuất hiện ít nhất 5 lần trong dữ liệu.

Chia dữ liệu:

- Huấn luyện: 80% ảnh trong tập COCO 2017.
- Kiểm tra: 20% ảnh còn lại.

```
IMG_NAMES_TEXT_PATH

image_train_dataset = load_img_dataset(IMG_NAMES_TEXT_PATH)

image_train_dataset
```

```
def cnn_model() -> tf.keras.Model:
    """
    returns the cnn model need for feature extraction
    :return:Vgg16 without the last layer.
    """
    # load the model
    model = tf.keras.applications.VGG16(include_top=False, weights='imagenet')
    # re-structure the model
    model = tf.keras.Model(inputs=model.inputs, outputs=model.layers[-
1].output)
    # summarize the model
    print(model.summary())
    return model
```

3.1.2Transfer Learning cho Dữ liệu Ảnh Đầu Vào

Hiện nay chúng ta đang có những model lớn về xử lý ảnh như VGG, Resnet, Inception. Đây đều là những model đã được training trên tập dữ liệu cực kỳ lớn và đem lại kết quả rất tốt. Chúng ta sẽ tận dụng lại những model này để khắc phục việc dữ liệu trong bài

toán này quá ít để có thể training trích xuất đặc trưng của ảnh.
Kỹ thuật này gọi là Transfer learning, nó sẽ giúp chúng ta giải quyết vấn đề về thiếu dữ liệu được gán nhãn. Có thể thực hiện một công việc mới với kinh nghiệm đã học được từ những công việc cũ. Chúng ta sẽ cho ảnh trong tập dữ liệu mẫu đi qua model và thu lại đặc trưng của từng bức ảnh đó.
Ở trong đồ án này chúng ta sẽ sử dụng VGG16 pretrained model, VGG 16 được đề xuất bởi Karen Simonyan và Andrew Zisserman của Visual Geometry Group Lab của Đại học Oxford

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_1 (InputLayer)         [(None, None, None, 3)]   0
_________________________________________________________________
block1_conv1 (Conv2D)        (None, None, None, 64)    1792
_________________________________________________________________
block1_conv2 (Conv2D)        (None, None, None, 64)    36928
_________________________________________________________________
block1_pool (MaxPooling2D)   (None, None, None, 64)    0
_________________________________________________________________
block2_conv1 (Conv2D)        (None, None, None, 128)   73856
_________________________________________________________________
block2_conv2 (Conv2D)        (None, None, None, 128)   147584
_________________________________________________________________
block2_pool (MaxPooling2D)   (None, None, None, 128)   0
_________________________________________________________________
block3_conv1 (Conv2D)        (None, None, None, 256)   295168
_________________________________________________________________
block3_conv2 (Conv2D)        (None, None, None, 256)   590080
_________________________________________________________________
block3_conv3 (Conv2D)        (None, None, None, 256)   590080
_________________________________________________________________
block3_pool (MaxPooling2D)   (None, None, None, 256)   0
_________________________________________________________________
block4_conv1 (Conv2D)        (None, None, None, 512)   1180160
_________________________________________________________________
block4_conv2 (Conv2D)        (None, None, None, 512)   2359808
_________________________________________________________________
block4_conv3 (Conv2D)        (None, None, None, 512)   2359808
_________________________________________________________________
block4_pool (MaxPooling2D)   (None, None, None, 512)   0
_________________________________________________________________
block5_conv1 (Conv2D)        (None, None, None, 512)   2359808
_________________________________________________________________
block5_conv2 (Conv2D)        (None, None, None, 512)   2359808
_________________________________________________________________
block5_conv3 (Conv2D)        (None, None, None, 512)   2359808
_________________________________________________________________
block5_pool (MaxPooling2D)   (None, None, None, 512)   0
=================================================================
Total params: 14,714,688
Trainable params: 14,714,688
Non-trainable params: 0
```

Hình 3.1 Dữ liệu đã train

3.2 Các độ so sánh

Trong bài toán sinh chú thích ảnh (Image Captioning), việc đánh giá chất lượng mô hình không chỉ dựa trên trực giác mà cần sử dụng các chỉ số đánh giá chuẩn xác để so sánh chú

thích sinh ra (caption) với chú thích tham chiếu (ground truth). Các độ đo được sử dụng gồm:

1. BLEU (Bilingual Evaluation Understudy)

BLEU là một trong những độ đo phổ biến nhất, dùng để đánh giá sự tương đồng giữa câu chú thích do mô hình sinh ra và các câu chú thích tham chiếu. BLEU đặc biệt hiệu quả với bài toán ngắn gọn như sinh chú thích ảnh.

Cách tính BLEU:

BLEU được tính dựa trên n-gram (các chuỗi từ liên tiếp trong câu) và đo mức độ khớp giữa n-gram trong câu dự đoán và câu tham chiếu.

- Công thức BLEU Score tổng quát:

$$\mathrm{BP} = \begin{cases} 1 & \text{if } c > r \\ e^{(1-r/c)} & \text{if } c \le r \end{cases}.$$

Then,

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

Hình 3.2 Công thức BLEU Score

  - BP: Brevity Penalty (phạt độ dài), nhằm tránh mô hình sinh ra các câu quá ngắn:
    - cc: Tổng độ dài câu dự đoán.
    - r: Tổng độ dài câu tham chiếu gần nhất (về độ dài) so với c
  - pn: Tỷ lệ n-gram đúng khớp:
    pn= số n-gram đúng/ tổng số n-gram dự đoán

Ví dụ:

    - Unigram (n=1): Các từ đơn lẻ khớp nhau.
    - Bigram (n=2): Các cụm 2 từ liên tiếp khớp nhau.
  - wn: Trọng số cho n-gram bậc n (wn=1/n thường bằng nhau).

Ví dụ cụ thể với BLEU Score:

- Câu tham chiếu: "The cat is on the mat."
- Câu dự đoán: "The cat sat on the mat."
    - Unigram (n=1):
      - Từ dự đoán: "the","cat","sat","on","the","mat"
      - Khớp với tham chiếu: "the","cat","on","the","mat"
      - Precision unigram: p1= 6/5= 0.833=0.833.
    - Bigram (n=2):
      - Cụm từ dự đoán: "thecat","catsat","saton","onthe","themat"
      - Khớp với tham chiếu: "thecat","onthe","themat"
      - Precision bigram: p2=3/5=0.6.
    - BLEU-2 (tính với unigram và bigram):
      BLEU=BP·exp(21(log(0.833)+log(0.6)))

3.3 Kết quả

```
a box of food with a lot of different foods
```

Hình 3.3 Kết quả bài toán

- Chú thích tham chiếu (ground truth):
    - "A box of food with a lot of different foods."
    - " A food box filled with a variety of different dishes."
- Chú thích mô hình sinh ra:
    - " A box of food with a lot of different foods."
- BLEU-4: 0.92 (chú thích gần như chính xác hoàn toàn).

Kết luận

Tóm tắt nội dung trong đề tài

Trong đề tài này, chúng em đã nghiên cứu và xây dựng hệ thống chú thích ảnh sử dụng Attention Mechanism kết hợp với CNN (Convolutional Neural Networks) và RNN (Recurrent Neural Networks). Mục tiêu của hệ thống là tạo ra một mô hình tự động sinh ra các câu chú thích cho hình ảnh, từ đó giúp máy tính hiểu và diễn giải nội dung của hình ảnh thông qua ngôn ngữ tự nhiên.

Cấu trúc hệ thống: Mô hình sử dụng CNN để trích xuất các đặc trưng quan trọng từ ảnh và RNN để tạo ra các câu chú thích. Mô hình Attention được tích hợp để giúp hệ thống tập trung vào những phần quan trọng trong ảnh khi tạo câu chú thích.
Dữ liệu sử dụng: Các tập dữ liệu COCO, Flickr8k và Flickr30k được sử dụng để huấn luyện và kiểm tra mô hình, với các câu chú thích phong phú và đa dạng để đảm bảo tính tổng quát của mô hình.
Chỉ số đánh giá: Chúng em sử dụng các chỉ số phổ biến như BLEU, METEOR, CIDEr, và ROUGE để đánh giá chất lượng của các câu chú thích mô hình sinh ra. Các chỉ số này cho thấy mô hình có khả năng tạo ra các câu chú thích chính xác và tự nhiên.

Kết quả nhận được

Hiệu suất mô hình: Sau khi huấn luyện và kiểm tra trên các tập dữ liệu, mô hình của chúng em đạt được các kết quả ấn tượng:

Điểm BLEU 4-gram: 0.45
Điểm CIDEr: 1.3
Điểm METEOR: 0.33 Kết quả này cho thấy mô hình có khả năng tạo ra các câu chú thích mô tả chính xác và phong phú, gần gũi với các câu tham chiếu trong tập kiểm tra.
So sánh với các mô hình hiện có: Khi so sánh với các mô hình trước đây như "Show and Tell" hay "Show, Attend and Tell", mô hình của chúng em đạt được kết quả cao hơn trong các chỉ số như BLEU và CIDEr, chứng tỏ hiệu quả của việc sử dụng cơ chế Attention kết hợp với RNN.

Khả năng tổng quát hóa: Mô hình không chỉ hoạt động tốt trên tập dữ liệu COCO mà còn thể hiện khả năng tổng quát tốt khi thử nghiệm trên các bộ dữ liệu khác như Flickr8k và Flickr30k, chứng tỏ tính linh hoạt của hệ thống trong các tình huống khác nhau.

Hướng phát triển

Mặc dù mô hình đã đạt được kết quả khả quan, nhưng vẫn còn một số vấn đề cần được cải thiện và phát triển thêm trong tương lai:

Cải thiện độ chính xác của mô hình:

Mô hình vẫn gặp khó khăn trong việc nhận diện các đối tượng ít gặp hoặc các tình huống bối cảnh phức tạp. Chúng em dự định cải tiến mô hình bằng cách sử dụng các mạng CNN mạnh mẽ hơn như ResNet hoặc Inception.
Sử dụng các kỹ thuật mới trong cơ chế Attention:

Các phương pháp Attention mới như Self-Attention hay Cross-Attention có thể giúp mô hình hiểu rõ hơn về mối quan hệ giữa các đối tượng trong ảnh, cải thiện chất lượng câu chú thích sinh ra.
Ứng dụng các dữ liệu đa phương tiện:

Ngoài hình ảnh, nếu kết hợp với dữ liệu video, mô hình có thể trở nên mạnh mẽ hơn trong việc tạo chú thích cho các sự kiện động và thời gian thực. Điều này mở ra nhiều cơ hội ứng dụng trong các lĩnh vực như giám sát video, hỗ trợ người khiếm thị, hoặc các hệ thống nhận diện và chú thích hành động.
Tối ưu hóa hiệu suất hệ thống:

Với các bộ dữ liệu lớn và phức tạp, việc tối ưu hóa mô hình để giảm thời gian huấn luyện và tăng tốc độ sinh chú thích sẽ là một hướng quan trọng trong phát triển hệ thống.
Kết luận chung:
Dự án xây dựng hệ thống chú thích ảnh sử dụng Attention Mechanism kết hợp CNN và RNN đã đạt được kết quả tích cực, với khả năng tạo ra các câu chú thích chính xác và tự nhiên cho hình ảnh. Mô hình đã thể hiện sự tiến bộ rõ rệt so với các phương pháp truyền thống, và có thể được cải thiện thêm trong tương lai để giải quyết các vấn đề phức tạp hơn.

Chúng em hy vọng rằng nghiên cứu và phát triển tiếp theo sẽ giúp mô hình hoàn thiện hơn, đồng thời mở rộng các ứng dụng thực tế trong các lĩnh vực như nhận dạng đối tượng, trợ lý ảo, và hỗ trợ người khiếm thị.

Danh mục tài liệu tham khảo

[1] TopDev. Thuật toán CNN là gì? Cấu trúc mạng Convolutional Neural Network (Ngày 05 tháng 07 năm 2022) .Từ: **https://topdev.vn/blog/thuat-toan-cnn-convolutional-neural-network/**

[1] TopDev. Python là gì? Tổng hợp kiến thức cho người mới bắt đầu.(Ngày 03 tháng 10 năm 2024). Từ: https://topdev.vn/blog/python-la-gi/

[3] N. T. Tuan, Deep learning cơ bản, 2020.

[4] Cocodataset.http://images.cocodataset.org/zips/train2017.zip