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

ĐỒ ÁN MÔN NHẬN DẠNG

VIETNAMESE SCENE TEXT DETECTION AND RECOGNITION

Giảng viên hướng dẫn: ThS. Đỗ Văn Tiến

Nhóm sinh viên thực hiện:

1. Phan Tấn Thương – 20522001
2. Nguyễn Thanh Trọng – 20520330
3. Nguyễn Hồng Hậu – 20521300
4. Nguyễn Đặng Nhật Hào - 20520490

Lớp CS338.N21

Thành phố Hồ Chí Minh, tháng 07 năm 2023

THÔNG TIN CHUNG

Đề tài	Vietnamese Scene Text Detection and Recognition
Môn học	CS338 – Nhận Dạng
Lớp	CS419.N21
Giảng viên hướng dẫn	ThS Đỗ Văn Tiến
Sinh viên thực hiện	Phan Tấn Thương - 20522001 – Tham gia 100% Nguyễn Thanh Trọng - 20520330 – Tham gia 100% Nguyễn Hồng Hậu – 20521300 – Tham gia 100% Nguyễn Đặng Nhật Hào - 20520490 – Tham gia 100%
Nội dung đề tài	Đề tài nghiên cứu này nhằm tìm hiểu tổng quan, các hướng tiếp cận của bài toán Scene Text Detection and Recognition. Từ đó chạy đánh giá một số phương pháp trên tập dữ liệu Vintext và làm demo về tool label với các mô hình sử dụng.

LỜI CẢM ƠN

Nhóm xin chân thành gửi lời cảm ơn đến ThS. Đỗ Văn Tiến – Giảng viên khoa Khoa học máy tính, Trường Đại học Công nghệ thông tin, Đại học Quốc gia thành phố Hồ Chí Minh, đồng thời là giảng viên giảng dạy lớp CS338.N21 – Môn Nhận Dạng, trong thời gian đã tận tình hướng dẫn và định hướng cho nhóm trong suốt quá trình thực hiện và hoàn thành đồ án.

Trong quá trình thực hiện đồ án nhóm đã cố gắng để có thể hoàn thành đề tài đúng tiến độ, một cách hoàn thiện và tốt nhất, song cũng không tránh khỏi những sai sót ngoài ý muốn. Nhóm mong rằng sẽ nhận được lời nhận xét và lời góp ý chân thành nhất từ thầy để đề tài có thể ngày càng hoàn thiện hơn. Mọi thắc mắc cũng như góp ý xin được tiếp nhận qua một trong các địa chỉ email cá nhân của các thành viên trong nhóm: 20522001@gm.uit.edu.vn (Phan Tấn Thương), 20520330@gm.uit.edu.vn (Nguyễn Thanh Trọng), 20521300@gm.uit.edu.vn (Nguyễn Hồng Hậu), 20520490@gm.uit.edu.vn (Nguyễn Đặng Nhật Hào). Mỗi đóng góp ý kiến và nhận xét của thầy là động lực to lớn đối với nhóm để có thể cải tiến hệ thống hoàn thiện hơn và phát triển đồ án lên một mức cao hơn trong tương lai, nhóm cũng sẽ nhìn nhận ra được ưu và khuyến điểm còn tồn đọng ở bản thân để từ đó có thể hoàn thiện kỹ năng, cũng như kiến thức của nhóm hơn. Hy vọng đề tài "Vietnamese Scene Text Detection and Recognition" do nhóm thực hiện sẽ trở thành một đóng góp có ích cho những nghiên cứu về lĩnh vực Khoa học máy tính nói riêng, cũng như đóng góp cho sự phát triển của xã hội nói chung.

Thành phố Hồ Chí Minh, tháng 7 năm 2023

Nhóm sinh viên thực hiện

Phan Tấn Thương

Nguyễn Thanh Trọng

Nguyễn Hồng Hậu

Nguyễn Đặng Nhật Hào

Mục lục

CHƯƠNG 1. TỔNG QUAN ĐỀ TÀI

1.1 Giới thiệu bài toán

Bài toán Scene Text Detection và Recognition là một trong những bài toán quan trọng trong lĩnh vực thị giác máy tính. Bài toán liên quan đến hai bài toán con là phát hiện và nhận dạng Text (văn bản) trong Scene Text (Text xuất hiện trong ảnh được chụp từ môi trường thực tế, không phải được thêm vào sau khi chụp ảnh), cụ thể như hình ảnh biển hiệu, biểu báo, biển số xe hay hình ảnh chụp lại sách báo…

Trong đó bài toán Text Detection (Phát hiện văn bản) sẽ tập trung vào việc phát hiện vị trí của Text trong ảnh. Text Detection nhận đầu vào là ảnh, cho đầu ra là vị trí các vùng chứa Text trong ảnh, làm tiền đề cho bài toán Text Recognition.

Hình 1. Đầu vào và kết quả trả về của bài toán text detection.

Sau khi có được vị trí các vùng chứa Text trong ảnh, bài toán Text Recognition (Nhận diện văn bản) sẽ xác định nội dung của Text trong những vùng đó.

Hình 2 Kết quả trả về của text recognition

Một số ứng dụng của bài toán Scene Text Detection và Recognition như: Số hóa sách báo, ứng dụng đọc sách; dịch văn bản trong hình ảnh; xây dựng bản đồ và định vị để điều khiển robot; hỗ trợ người khiếm thị…

Phát hiện và nhận diện văn bản Tiếng Việt trong ảnh ngoại cảnh là một nhánh phát triển của bài toán trên.

1.2 Thách thức, phạm vi, mục tiêu

1.2.1 Thách thức

Một số thách thức gặp phải trong quá trình giải quyết bài toán như:

- So với bảng chữ cái Tiếng Anh thì bảng chữ cái Tiếng Việt có thêm 7 chữ cái biến thể thêm dấu cùng với năm dấu thanh, do đó tạo ra sự đa dạng trong từ ngữ cũng như gây khó khăn trong việc phân biệt những chữ có dấu và không có dấu, hay có dấu câu khác nhau. Ví dụ như phân biệt giữa các từ: hương, hường, hướng và huống.
- Scene Text có thể được chụp trong điều kiện thiếu sáng, bị nghiêng do góc chụp, biển báo biển hiệu không còn nguyên vẹn. Điều này làm chữ bị nét, bị quá nhỏ, khó xác định vị trí tương quan giữa các chữ cái với nhau. Gây ảnh hưởng đến quá trình phát hiện và nhận diện văn bản.
- Ngoài ra bài toán còn bị ảnh hưởng bởi sự đa dạng trong phong cách nghệ thuật: chữ được viết theo các kích thước, kiểu mẫu khác nhau hoặc viết chữ theo hình tròn.

1.2.2 Phạm vi

- Tìm hiểu hai mô hình Text Detection và Text Recognition trong bài toán nhận diện văn bản Tiếng Việt trong hình ảnh các biển hiệu, biển quảng cáo, bìa sách...
- Kết hợp giải quyết bài toán Scene Text Detection và Recognition.
- Đánh giá mô hình.
- Tìm hiểu các thức xây dựng API ứng dụng cho mình của bài toán.

1.2.3 Mục tiêu

- Tìm hiểu tổng quan về bài toán Scene Text Detection và Recognition, tập trung vào loại văn bản là văn bản Tiếng Việt.
- Tiếp cận và huấn luyện mô hình cho bài toán với tập dữ liệu Vintext.
- Xây dựng API phát hiện và nhận dạng văn bản Tiếng Việt.

CHƯƠNG 2. HƯỚNG TIẾP CẬN VÀ CÁC NGHIÊN CỨU LIÊN QUAN

2.1 Bài toán Text Detection

Bài toán Text Detection được phát biểu như sau: đầu vào của bài toán là ảnh số chứa các Text (đối tượng quan tâm) và đầu ra sẽ là khung bao quanh các Text thể hiện vị trí của nó trong bức ảnh. Kết quả bài toán Text Detection là yếu tố tiên quyết của bài táon Text Recognition. Mô hình nhận diện văn bản dù có tốt nhưng sẽ không có ý nghĩa nếu các văn bản trong ảnh không được phát hiện. Có thể tiếp cận bài toán Text Detection theo hướng bài toán Object Detection với một class duy nhất là "Text".

2.1.1 Một số hướng tiếp cận

a. Các phương pháp truyền thống

Stroke Width Transform(SWT)

Stroke Width Transform (SWT) là một phương pháp được sử dụng trong việc phát hiện vùng văn bản trong ảnh. Phương pháp này dựa trên ý tưởng rằng các nét chữ thường có độ dày gần như đồng nhất trong toàn bộ kí tự.

Cơ bản SWT thực hiện qua các bước sau để phát hiện văn bản:

- Phát hiện biên cạnh: sử dụng thuật thoán phát hiện biên cạnh như Canny.
- Tính toán và chuẩn hóa Stroke width.
- Phát hiện vùng chứa văn bản: Xác định vị trí vùng văn bản bằng cách so sánh Stroke width của từng pixel với độ rộng nét được chuẩn hóa. Nếu Stroke width của pixel nằm trong một phạm vi nhất định của Stroke width đã chuẩn hóa, thì nó được coi là một phần của vùng chứa văn bản.

Selective Search(SS)

Trong Text Detection Selective Search có thể được sử dụng để xác định vùng chứa văn bản trong ảnh. Đầu tiên nó sẽ chia hình ảnh thành các vùng (regions) khác nhau dựa trên dựa trên màu sắc, kết cấu hoặc các dấu hiệu nhận biết khác. Sau đó sử dụng thuật toán tham lam để tính toán độ tương đồng giữa các regions và rồi tiến hành hợp nhất các regions giống nhau nhất. Quá trình này lặp đi, lặp lại cho đến khi bức ảnh được biểu diễn bằng tập hợp các candidate regions. Trong candidate regions tiến hành lọc các regions không có khả năng chứa văn bản, cuối cùng thu được các regions chứa văn bản.

Nhìn chung những hướng tiếp cận truyền thống còn khá đơn giản, không phù hợp bộ dữ liệu đa dạng như Vintext được sử dụng cho bài toán của chúng em.

b. Ứng dụng Deep Learning

Học sâu đã được áp dụng rộng rãi trong việc phân đoạn ngữ nghĩa và phát hiện đối tượng chung. Các phương pháp tương tự cũng đang được áp dụng trong phát hiện văn bản. Trong phát hiện văn bản, việc dự đoán các khung giới hạn văn bản cần xem xét đặc điểm hướng của văn bản và không chỉ dựa trên thông tin khung giới hạn được điều chỉnh theo trục.

Một số phương pháp được sử dụng :

- Semantic Segmentation Based Methods.
- General Object Detection Based Methods.
- Hybrid Methods.

Semantic Segmentation Based Methods

Phương pháp phân đoạn ngữ nghĩa trong phát hiện văn bản sử dụng các mô hình mạng nơ-ron tích chập (CNN) để xác định các vùng văn bản dựa trên việc tìm hiểu các đặc trưng cấu trúc và ngữ nghĩa của văn bản.

Các phương pháp phát hiện văn bản dựa trên phân đoạn ngữ nghĩa sử dụng mô hình FCN (Fully Convolutional Network) hoặc các biến thể của nó để tạo ra bản đồ phân đoạn của văn bản trong hình ảnh. Mô hình này tạo ra một bản đồ toàn cục, đưa ra thông tin về vị trí, hình dạng và mối quan hệ giữa các vùng văn bản trong cảnh. Các phương pháp tiên tiến hơn kết hợp sự kết hợp của các mạng nơ-ron để thực hiện phân đoạn từ tương đối đến tốt hơn, cho phép xác định chính xác hơn các khối văn bản và các đặc trưng liên quan.

Phương pháp phân đoạn ngữ nghĩa trong phát hiện văn bản đem lại những kết quả ấn tượng, cho phép xác định vị trí và ranh giới của văn bản một cách chính xác và tự động. Điều này có ứng dụng rất rộng trong nhiều lĩnh vực như xử lý ảnh, truy vấn thông tin và nhận dạng văn bản, đóng góp vào việc tạo ra các ứng dụng thực tế như nhận dạng biển số xe, quét và chuyển đổi văn bản giấy thành văn bản số.

General Object Detection Based Methods

Hướng phát hiện văn bản dựa trên phương pháp phát hiện đối tượng chung (General Object Detection Based Methods) là một trong những hướng tiếp cận phổ biến.

Thay vì xác định các vùng văn bản bằng cách tìm kiếm các đặc trưng cụ thể của văn bản, các phương pháp dựa trên phát hiện đối tượng chung tập trung vào việc huấn luyện mô hình phát hiện đối tượng chung để xác định các vùng văn bản.

Các phương pháp phát hiện văn bản dựa trên phương pháp phát hiện đối tượng chung thường sử dụng các mô hình như R-CNN (Region-based Convolutional Neural Network), Faster R-CNN, hoặc SSD (Single Shot MultiBox Detector). Các mô hình này được huấn luyện trên tập dữ liệu chứa cả văn bản và các đối tượng khác, nhằm nhận dạng và phân loại các vùng văn bản trong cảnh.

Phương pháp phát hiện dựa trên phát hiện đối tượng chung có ưu điểm là khả năng phát hiện văn bản có hình dạng, kích thước và hướng khác nhau trong cảnh. Đồng thời, các mô hình phát hiện đối tượng chung đã được huấn luyện trên các tập dữ liệu lớn và đa dạng, giúp nâng cao hiệu suất và độ chính xác của phát hiện văn bản.

Tuy nhiên, hướng phát hiện dựa trên phát hiện đối tượng chung cũng đối mặt với một số thách thức đặc biệt khi áp dụng cho văn bản, bao gồm việc xử lý văn bản có hướng và việc xác định đúng ranh giới văn bản trong cảnh. Do đó, các nghiên cứu tiếp tục phát triển và cải tiến các phương pháp này để đảm bảo hiệu suất và độ chính xác tốt nhất trong việc phát hiện văn bản trong cảnh.

Hybrid Methods

Phương pháp kết hợp (Hybrid Methods) là một hướng tiếp cận quan trọng trong bài toán phát hiện văn bản trong cảnh. Gần đây, các nhà nghiên cứu đã tìm cách kết hợp hai loại phương pháp khác nhau để phát hiện văn bản chính xác hơn trong các tình huống phức tạp.

Một ví dụ về phương pháp kết hợp là mô hình chú ý văn bản được đề xuất tại bài báo *Single shot text detector with regional attention (He P, Huang W, He T, Zhu Q, Qiao Y (2017)*. Mô hình này sử dụng mặt nạ văn bản tại mỗi điểm ảnh

để mã hóa thông tin cụ thể về văn bản. Điều này giúp loại bỏ hiện tượng nhiễu nền trong các đặc trưng convolutional và đồng thời sử dụng các đặc trưng multi-scale inception để mã hóa thông tin về văn bản cục bộ và ngữ cảnh. Mô hình hoạt động theo phương pháp từ thô đến tốt, mang lại kết quả phát hiện văn bản chính xác.

Một phương pháp khác là mạng RPN không dựa trên anchor (AF-RPN) được trình bày bởi *Zhong Z, Sun L, Huo Q (2018)* trong bài báo *An Anchor-Free Region proposal network for faster R-CNN based text detection approaches*. Phương pháp này cho phép tạo ra các đề xuất văn bản nghiêng chất lượng cao trực tiếp mà không cần thiết kế các anchor phức tạp. AF-RPN sử dụng ba mô-đun phát hiện gắn kết trên các mức pyramid khác nhau để phát hiện văn bản nhỏ, trung bình và lớn.

Các phương pháp kết hợp trong phát hiện văn bản trong cảnh nhằm tận dụng những ưu điểm của cả hai hướng tiếp cận và đảm bảo hiệu suất cao và độ chính xác trong việc phát hiện văn bản. Nhờ sự kết hợp đa dạng, các phương pháp này mang lại những kết quả ấn tượng và đóng góp quan trọng cho việc phát triển các ứng dụng thực tế liên quan đến xử lý và nhận dạng văn bản trong cảnh.

2.1.2 Một số thuật toán sử dụng

a. Dictguided

Các phương pháp dựa trên hướng tiếp cận segmentation-based yêu cầu lượng lớn dữ liệu tập trung vào kí tự, việc inference ở các phương pháp trên khá chậm. Thuật toán ABCnet giải quyết tốt các dữ liệu cong, xiên,.. tốc độ tính toán nhanh, giúp cho mô hình có thể chạy realtime.

Mô hình ABCnet có ý tưởng :

- Sử dụng đường cong Bezier để biểu diễn tham số dẫn đến việc chi phí tính toán ít hơn so với việc dự đoán bao đóng cho văn bản.
- Đề xuất thêm phương pháp lấy mẫu BezierAlign, căn chỉnh đối tượng chính xác giúp cho module recognition nhận dạng chữ tốt hơn.

Beizer Detection :

So với các mô hình sử dụng phương pháp phân đoạn ảnh, thì ABCnet sử dụng phương pháp hồi qui, giúp cho mô hình học ra các tham số dễ dàng vã đỡ phức tạp hơn. Hơn nữa đường con Beizer bậc ba có thể mô hình hóa được hầu hết các chữ có hình dạng cong, xiên.

Công thức đường cong Bezier :

$$c(t) = \sum_{i==0}^{n} b_i B_{i,n}(t), 0 <= t <= 1$$

Trong đó, n là hệ số góc, b_i biểu diễn điểm thứ i và $B_{i,n}(t)$ biểu diễn Bernstein polynomials, theo công thức sau:

$$B_{i,n}(t) = \binom{n}{i} t^i (1-t)^{n-i}, i = 0, ..., n$$

Trong đó (n i) là hệ số nhị thức.

Mô hình sử dụng 8 điểm để dự đoán box của một văn bản. Các phương pháp khác sử dụng 4 điểm để dự đoán tọa độ box của chữ, còn với ABCnet thêm 4 điểm vào 4 cạnh tạo thành 4 điểm điều khiển, giúp cho bbox linh hoạt có thể điều chỉnh theo chữ có hình dạng bất kì.

Để học tọa độ các điểm điều khiển, đầu tiên phải sinh ra đường cong Bezier, rồi sau đó sử dụng hồi quy tuyến tính để học điểm dữ liệu.

b. YOLOv7

YOLO (You Only Look Once) là một thuật toán phát hiện đối tượng phổ biến và được sử dụng rộng rãi trong lĩnh vực xử lý hình ảnh. Nổi tiếng với hiệu suất và tốc độ xử lý nhanh, YOLO có khả năng đồng thời dự đoán vị trí và nhãn của các đối tượng trong ảnh chỉ trong một lần chạy.

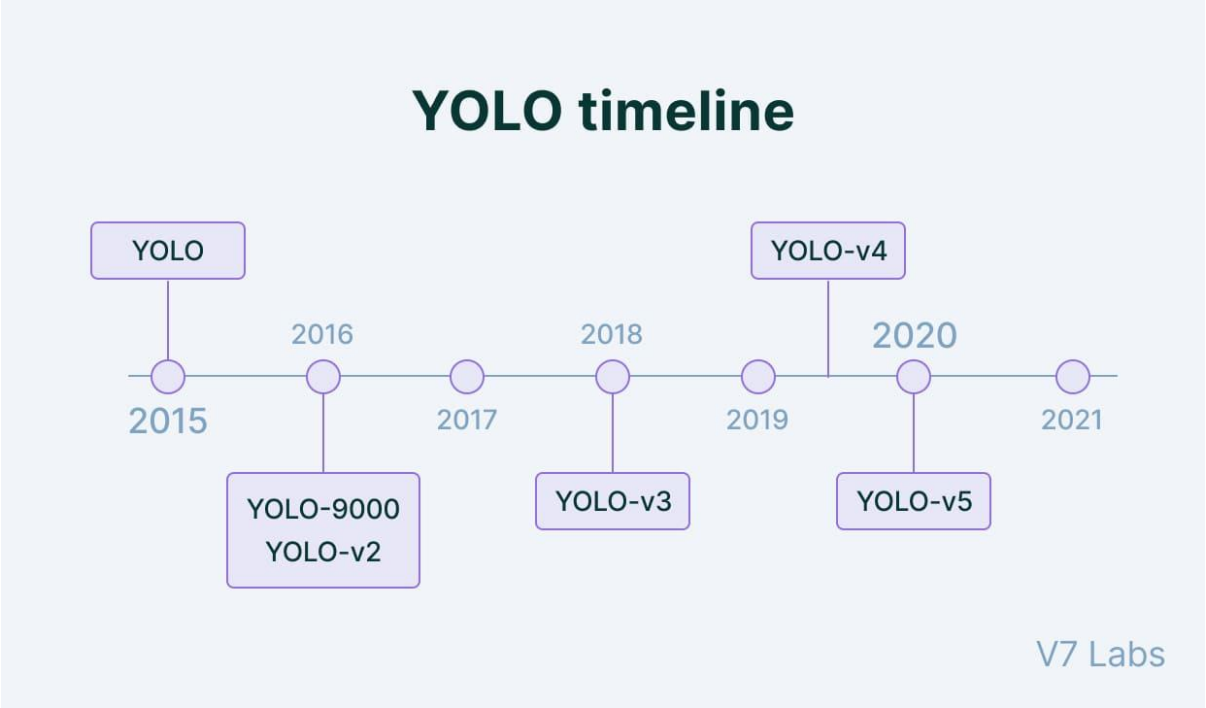
Với bài toán phát hiện văn bản (Text Detection), YOLO xử lý chỉ có một class duy nhất là "Text". Khác với các bài toán phát hiện đối tượng khác, YOLO tập trung vào việc xác định vị trí của văn bản trong ảnh và gán nhãn cho nó.

Ý tưởng

YOLO đề xuất sử dụng mạng thần kinh đầu cuối để đưa ra dự đoán về các hộp giới hạn (bounding box) và xác suất của đối tượng cùng một lúc. Nó khác với cách tiếp cận của các thuật toán phát hiện đối tượng trước đó, vốn sử dụng lại các trình

phân loại để thực hiện phát hiện. Theo một cách tiếp cận cơ bản khác để phát hiện đối tượng, YOLO đã đạt được kết quả tiên tiến, đánh bại các thuật toán phát hiện đối tượng thời gian thực khác với khoảng cách lớn. Trong khi các thuật toán như Faster RCNN hoạt động bằng cách phát hiện các khu vực quan tâm có thể có bằng cách sử dụng Region Proposal Network và sau đó thực hiện nhận dạng trên các khu vực đó một cách riêng biệt, thì YOLO thực hiện tất cả các dự đoán với sự trợ giúp của một lớp được kết nối đầy đủ duy nhất. Các phương pháp sử dụng Region Proposal Network thực hiện nhiều lần lặp cho cùng một hình ảnh, trong khi YOLO hoàn thành trong một lần duy nhất. Một số phiên bản mới của cùng một mô hình đã được giới thiệu kể từ lần phát hành đầu tiên của YOLO vào năm 2015. Mỗi phiên bản được xây dựng để cải tiến phiên bản tiền nhiệm. Dưới đây là mốc thời gian thể hiện sự phát triển của YOLO trong những năm gần đây.

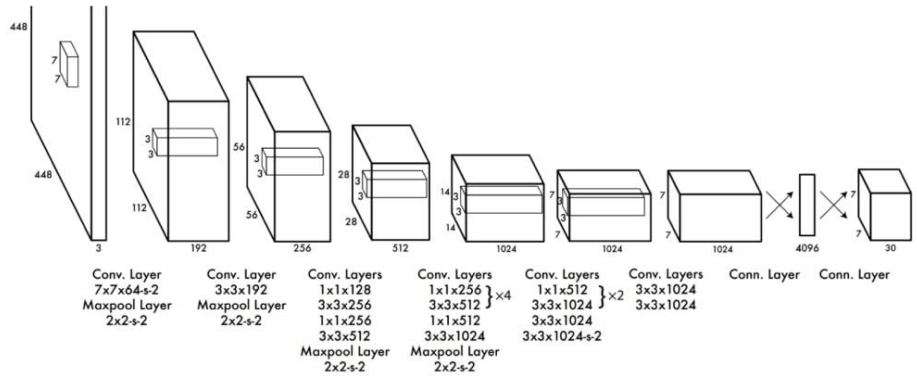
Một số phiên bản mới của cùng một mô hình đã được giới thiệu kể từ lần phát hành đầu tiên của YOLO vào năm 2015. Mỗi phiên bản được xây dựng để cải tiến phiên bản tiền nhiệm. Dưới đây là mốc thời gian thể hiện sự phát triển của YOLO trong những năm gần đây.



Kiến trúc và cách hoạt động

Thuật toán YOLO (You Only Look Once) sử dụng mạng neural tích chập sâu để phát hiện đối tượng trong ảnh. YOLO chia ảnh thành lưới S × S và mỗi ô lưới

dự đoán nhiều hộp giới hạn và điểm tin cậy cho các hộp đó. Để đảm bảo tính chính xác và độ tin cậy, YOLO chỉ định một bộ dự đoán hộp giới hạn dựa trên chỉ số IOU cao nhất với thực tế.



The Architecture. Our detection network has 24 convolutional layers followed by 2 fully connected layers. Alternating 1×1 convolutional layers reduce the features space from preceding layers. We pretrain the convolutional layers on the ImageNet classification task at half the resolution (224×224 input image) and then double the resolution for detection.

Một kỹ thuật quan trọng được sử dụng trong YOLO là NMS (non-maximum suppression), được áp dụng sau quá trình dự đoán. NMS giúp loại bỏ các hộp giới hạn trùng lắp hoặc không chính xác và chỉ đề xuất một hộp giới hạn duy nhất cho từng đối tượng trong ảnh. Điều này cải thiện độ chính xác và hiệu quả của quá trình phát hiện đối tượng.

Trong quá trình huấn luyện, YOLO sử dụng 20 lớp tích chập đầu tiên đã được đào tạo trước với ImageNet để cung cấp các thông tin trừu tượng cho việc phát hiện. Lớp kết nối đầy đủ cuối cùng của YOLO dự đoán xác suất lớp và tọa độ hộp giới hạn.

Điểm cải tiến của Yolov7

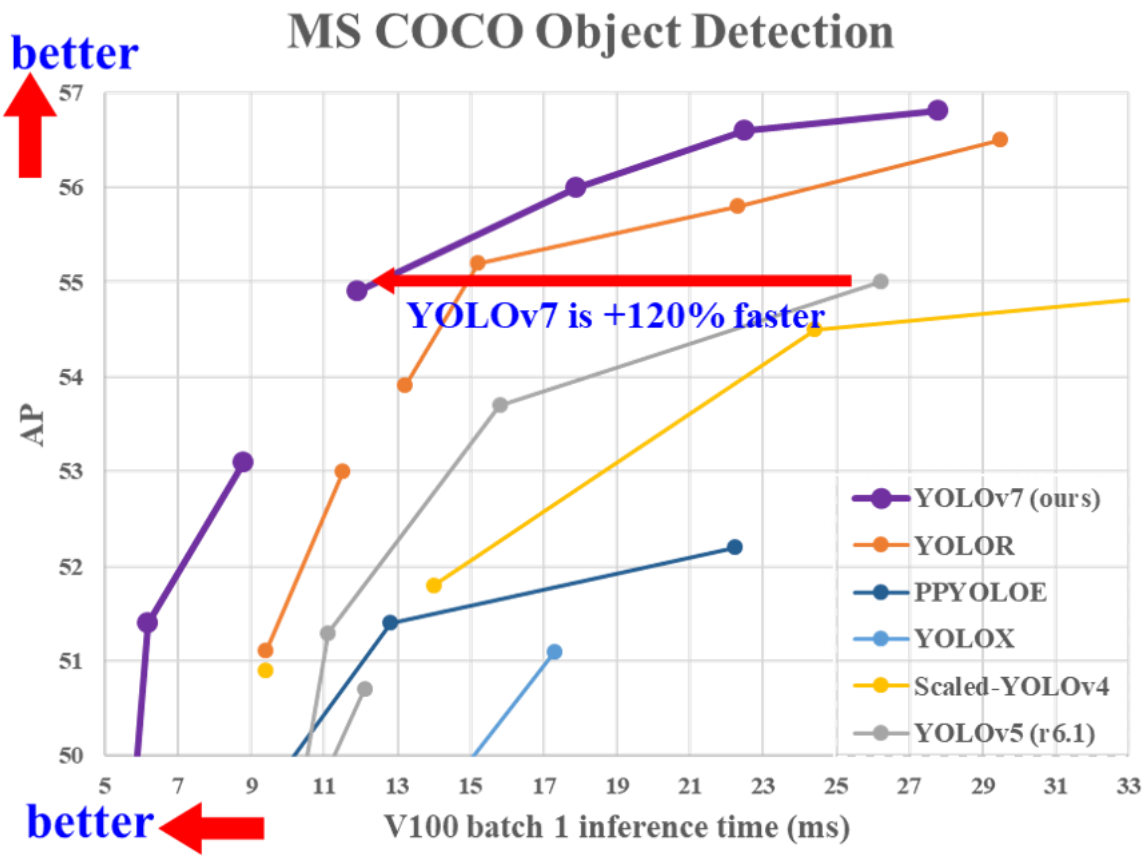
Các anchor box là một tập hợp các hộp được xác định trước với các tỷ lệ khung hình khác nhau được sử dụng để phát hiện các đối tượng có hình dạng khác nhau. YOLO v7 sử dụng chín anchor box, cho phép YOLO phát hiện phạm vi hình dạng và kích thước đối tượng rộng hơn so với các phiên bản trước, do đó giúp giảm số lượng xác định sai.

Một cải tiến quan trọng trong YOLOv7 là việc sử dụng một loss function mới gọi là “focal loss”. Các phiên bản trước của YOLO đã sử dụng cross-entropy loss function tiêu chuẩn, được biết là kém hiệu quả hơn trong việc phát hiện các đối tượng nhỏ. Focal loss giải quyết vấn đề này bằng cách giảm trọng số mất mát cho

các ví dụ được phân loại tốt và tập trung vào các ví dụ khó—các đối tượng khó phát hiện.

YOLOv7 cũng có độ phân giải cao hơn so với các phiên bản trước. Nó xử lý hình ảnh ở độ phân giải 608 x 608 pixel, cao hơn độ phân giải 416 x 416 được sử dụng trong YOLO v3. Độ phân giải cao hơn này cho phép YOLO v7 phát hiện các đối tượng nhỏ hơn và có độ chính xác tổng thể cao hơn.

Một trong những ưu điểm chính của YOLO v7 là tốc độ. Nó có thể xử lý hình ảnh với tốc độ 155 khung hình mỗi giây, nhanh hơn nhiều so với các thuật toán phát hiện đối tượng hiện đại khác. Ngay cả mô hình YOLO cơ bản ban đầu cũng có khả năng xử lý ở tốc độ tối đa 45 khung hình mỗi giây. Điều này làm cho nó phù hợp với các ứng dụng thời gian thực nhạy cảm như giám sát và ô tô tự lái, trong đó tốc độ xử lý cao hơn là rất quan trọng.



Nhược điểm

YOLO v7 là một thuật toán phát hiện đối tượng mạnh mẽ và hiệu quả, nhưng nó có một số hạn chế.

- YOLO v7, giống như nhiều thuật toán phát hiện đối tượng, gặp khó khăn trong việc phát hiện các đối tượng nhỏ. Nó có thể không phát hiện chính xác các đối tượng trong các cảnh đông đúc hoặc khi các đối tượng ở xa máy ảnh.
- YOLO v7 cũng không hoàn hảo trong việc phát hiện các đối tượng ở các tỷ lệ khác nhau. Điều này có thể gây khó khăn cho việc phát hiện các đối tượng rất lớn hoặc rất nhỏ so với các đối tượng khác trong cảnh.
- YOLO v7 có thể nhạy cảm với những thay đổi về ánh sáng hoặc các điều kiện môi trường khác, vì vậy có thể bất tiện khi sử dụng trong các ứng dụng thực, nơi điều kiện ánh sáng có thể thay đổi.
- YOLO v7 có thể đòi hỏi nhiều tính toán, điều này gây khó khăn khi chạy trong thời gian thực trên các thiết bị hạn chế về tài nguyên như điện thoại thông minh hoặc các thiết bị biên khác.

2.2 Text Recognition

2.2.1 Các phương pháp tiếp cận

a. Phương pháp phân loại theo ký tự

Các văn bản được cắt thành các ký tự, việc nhận dạng các ký tự được coi trở thành bài toán phân loại ký tự. Phân loại các ký tự cắt ra thành về các lớp ký tự được khai báo sẵn, tương tự như bài toán phân loại chữ cái viết tay. Sau đó các ký tự sẽ được kết nối với nhau tạo thành một từ, một câu hoàn chỉnh.

b. Phương pháp phân loại từ

Cũng tương tự như phương pháp phân loại ký tự văn bản được cắt thành các từ và được coi trở thành bài toán phân loại từ. Với sự phát triển của CNN việc phân loại các từ được sử dụng rộng rãi. Tuy nhiên nó phải dựa vào bộ từ điển định nghĩa trước và không thể nhận ra các vấn đề out of vocabulary, đối với các từ quá dài độ biến dnajg hình ảnh đầu vào lớn gây ảnh hướng đến tốc độ nhận dạng.

c. Phương pháp phân loại theo chuỗi

Hiệu suất của các phương pháp phân loại theo ký tự, phân loại theo từ phụ thuộc rất nhiều vào độ chính xác của bước text detection (segmentation base). Để giải quyết các vấn đề trên các nhà khoa học máy tính đã đưa ra phương pháp nhận dạng theo chuỗi. Có 2 hướng chính là:

- *CNN + họ các thuộc toán RNN (LSTM, GRU) + CTC*
- *CNN + kiến trúc seq2seq*

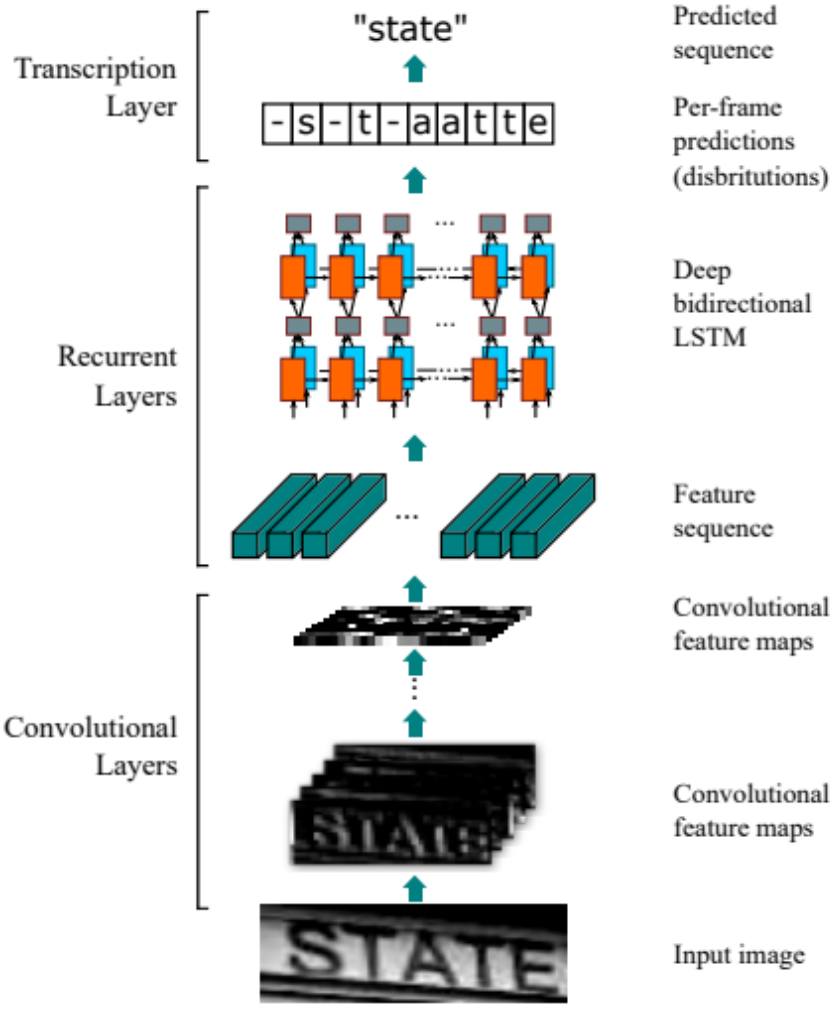
Hiện nay có thêm cách tiếp cận mới là *CNN + Transformer*.

Để thực hiện thì đầu tiên các ảnh chứa chữ sẽ được trích xuất đặc trưng và sau đó cho vào các module tiếp theo để nhận dạng. Các mô hình thuộc kiểu sử dụng CTC sẽ sử dụng CTC loss làm hàm mục tiêu và CTC layer để decoder. Còn các mô hình thuộc kiểu seq2seq và transformer sẽ sử dụng cross entropy loss làm hàm mục tiêu.

2.2.2 Chi tiết các thuật toán phân loại theo chuỗi

a. RCNN:

CRNN là kiến trúc mạng neural có sự kết hợp giữa mạng neural tích chập (CNN) và mạng neural hồi qui (RNN). Với bài toán nhận dạng văn bản trong ảnh chúng ta sử dụng mạng neural tích chập để tiến hành trích xuất đặc trưng của ảnh, tạo thành feature map, với feature map chúng ta sẽ thực hiện cắt thành các feature sequence trở thành input cho mạng neural hồi qui. Ở mạng hồi qui chúng ta sử dụng hàm mục tiêu là CTC loss để huấn luyện mô hình, và CTC layer để thực hiện decode từ output của mạng RNN thành chuỗi text.



Tuy nhiên việc sử dụng CTC loss và CTC layer có hạn chế. Số lượng kí tự tối đa có thể dự đoán bằng với width*height của feature map do đó chúng ta phải cẩn thận điều chỉnh kiến trúc mô hình để tạo ra feature map phù hợp với số lượng từ chúng ta mong muốn,trong bài toán text recognition số lượng text trong một ảnh không được cố định nên việc CTC layer và CTC loss là không hợp lí để giải quyết bài toán.

b. Kiến trúc seqseq

Kiến trúc seq2seq có 2 thành phần chính : bộ mã hóa (encoder), bộ giải mã (decoder). Với kiến trúc seq2seq chúng ta không gặp phải hạn chế của CTC về số lượng tối đa chữ cái đầu ra. Số lượng từ và chữ cái được mã hóa tùy ý. Cũng tương tự như CRNN, để giải quyết bài toán text recognition với kiến trúc seq2seq, đầu tiên chúng ta cần thực hiện trích xuất đặc trưng của ảnh đầu vào bằng một mạng CNN => tạo ra 1 feature map. sau đó sử dụng 1 mạng RNN để làm bộ mã hóa các đặc trưng => tạo thành bộ vector ngữ cảnh và một mạng RNN khác để thực hiện giải mã sinh ra chuỗi kí tự từ vector ngữ cảnh.

Kiến trúc seq2seq cũng có nhược điểm là chưa thể hiện được mối quan hệ giữa output của encoder và input đầu vào của decoder.

c. Seg2seg + attention

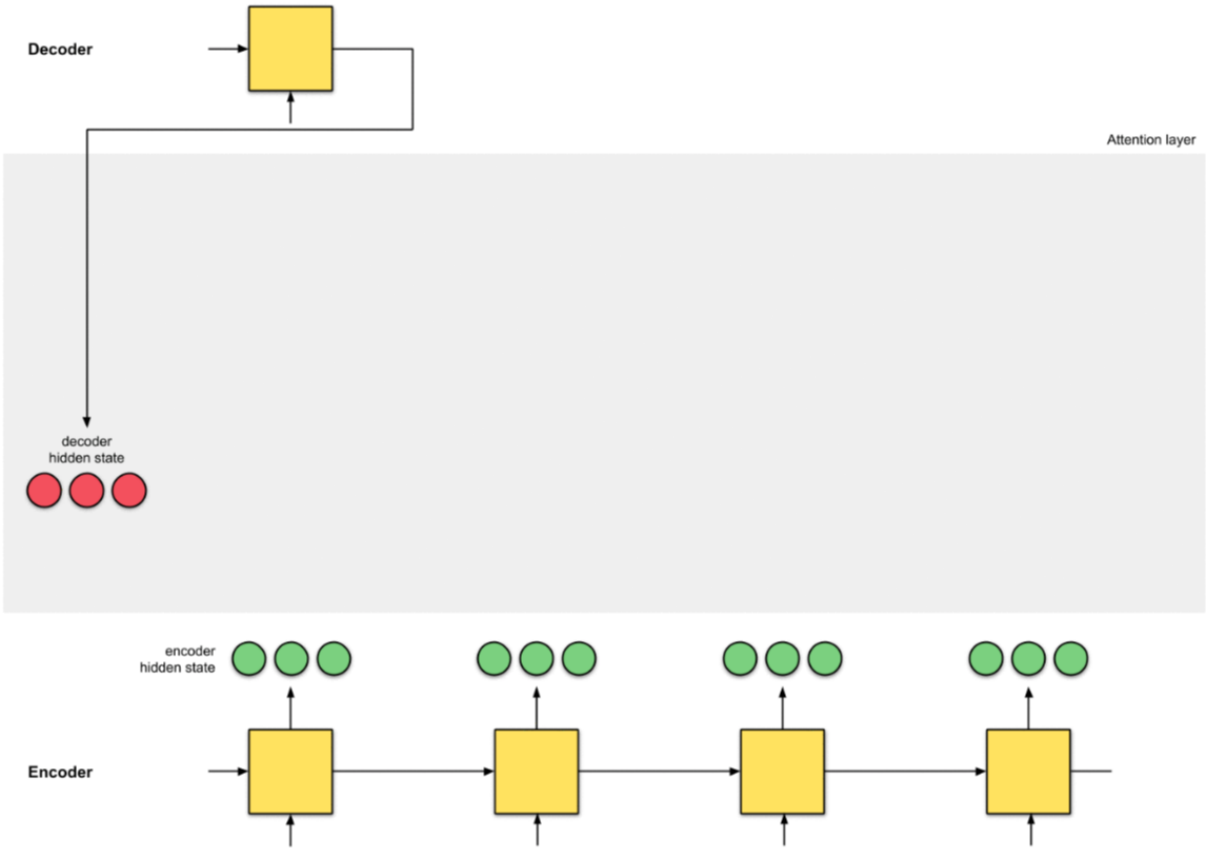
Kiến trúc seq2seq + attention cũng tương tự như cấu trúc seq2seq nhưng cải tiến thêm cơ chế attention để tạo nên mối quan hệ giữa output của encoder và input của decoder => các input của decoder sẽ tập trung vào các cặp từ có liên quan nhất với nó trong output của encoder. Giảm một chút chi phí tính toán và thể hiện được sự liên kết ngữ nghĩa cao hơn so với kiến trúc seq2seq. Vậy tại sao cơ chế attention làm được như vậy?

Cơ chế attention:

Ví dụ cách làm việc của cơ chế attention

Bước 1 : Chúng ta đã có output của Encoder hidden state :

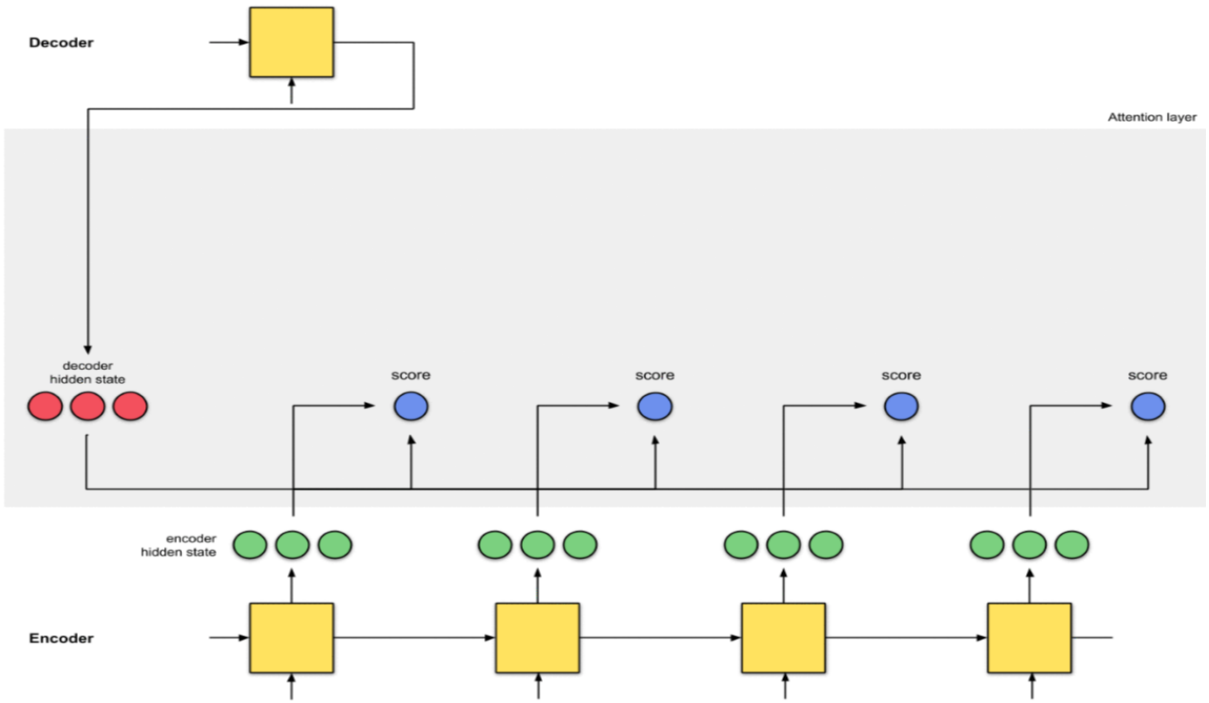
Có 4 Encoder hidden state màu xanh và decoder hidden state đầu tiên màu đỏ.



Bước 2 – Tính Alignment score

Alignment score là điểm giữa Decoder Hidden State, tính điểm này chúng ta sử dụng dot product.

Ví dụ :

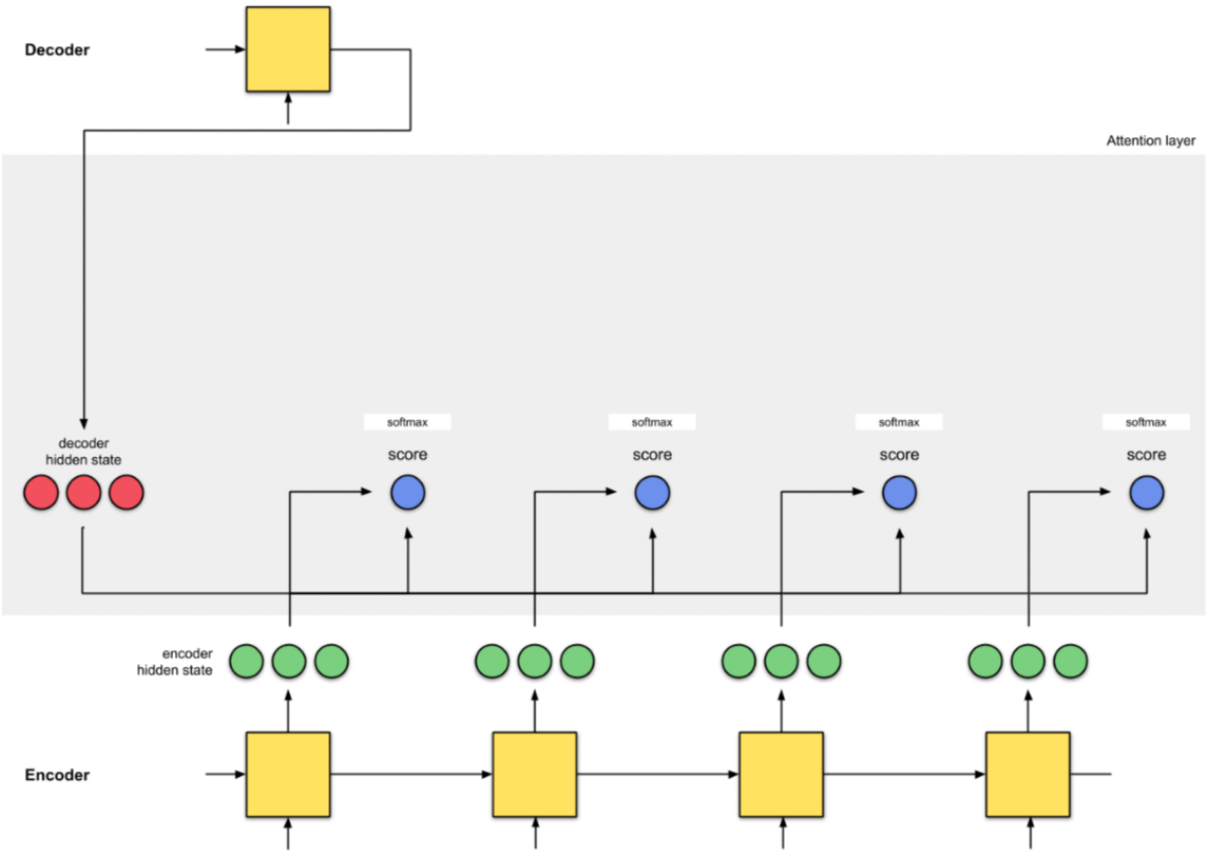


```
decoder_hidden = [10, 5, 10]

encoder_hidden    alignment score
--------------------
     [0, 1, 1]     15 (= 10×0 + 5×1 + 10×1, the dot product)
     [5, 0, 1]     60
     [1, 1, 0]     15
     [0, 5, 1]     35
```

Như ở trên alignment score cao nhất là 60 tại Timestep thứ 2 của Encoder [5, 0, 1] => cho thế output tiếp theo của decoder sẽ hiểu ảnh hưởng cao nhất của hidden state này.

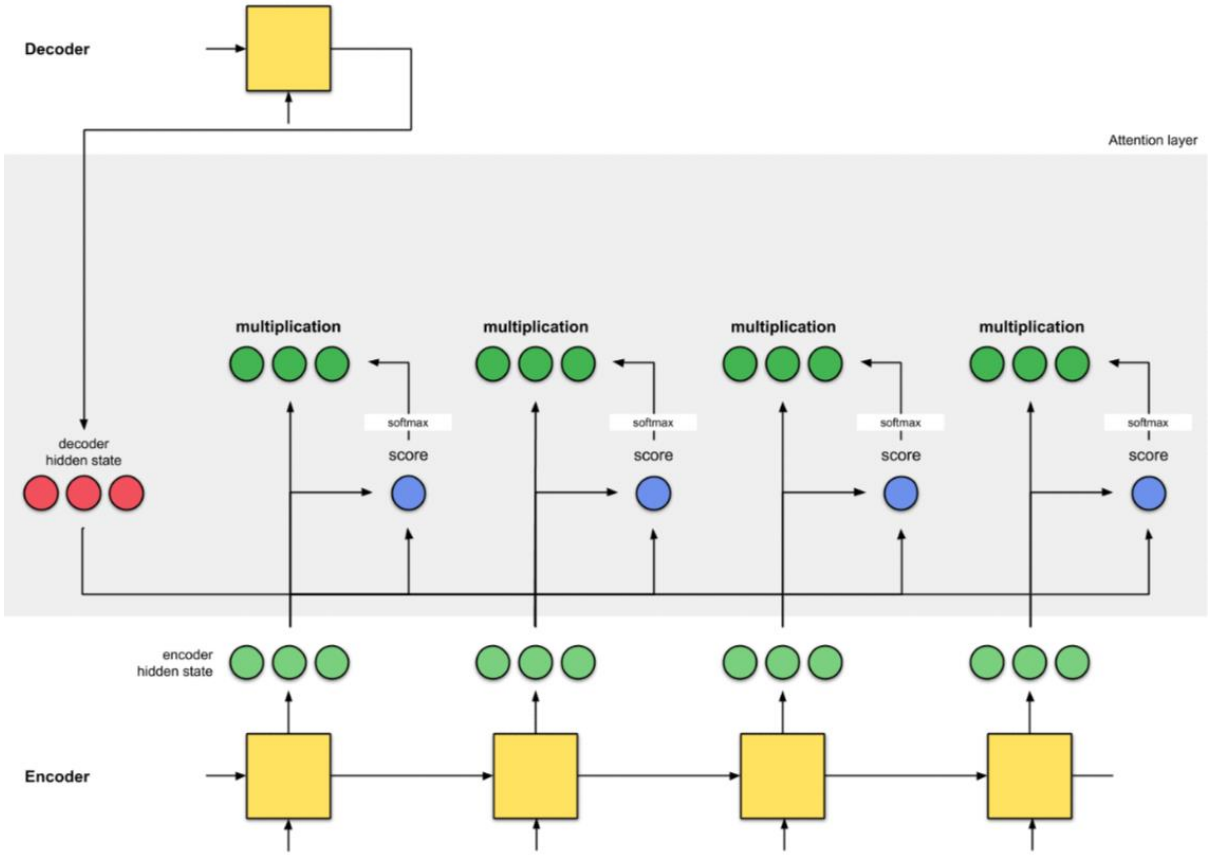
Bước 3. Cho Alignment score qua Softmax Function




```
encoder_hidden    alignment score    softmaxed
-----------------------------
     [0, 1, 1]     15                0
     [5, 0, 1]     60                1
     [1, 1, 0]     15                0
     [0, 5, 1]     35                0
```

Bước 4: Tính context vector (vector ngữ cảnh) của mỗi timestep

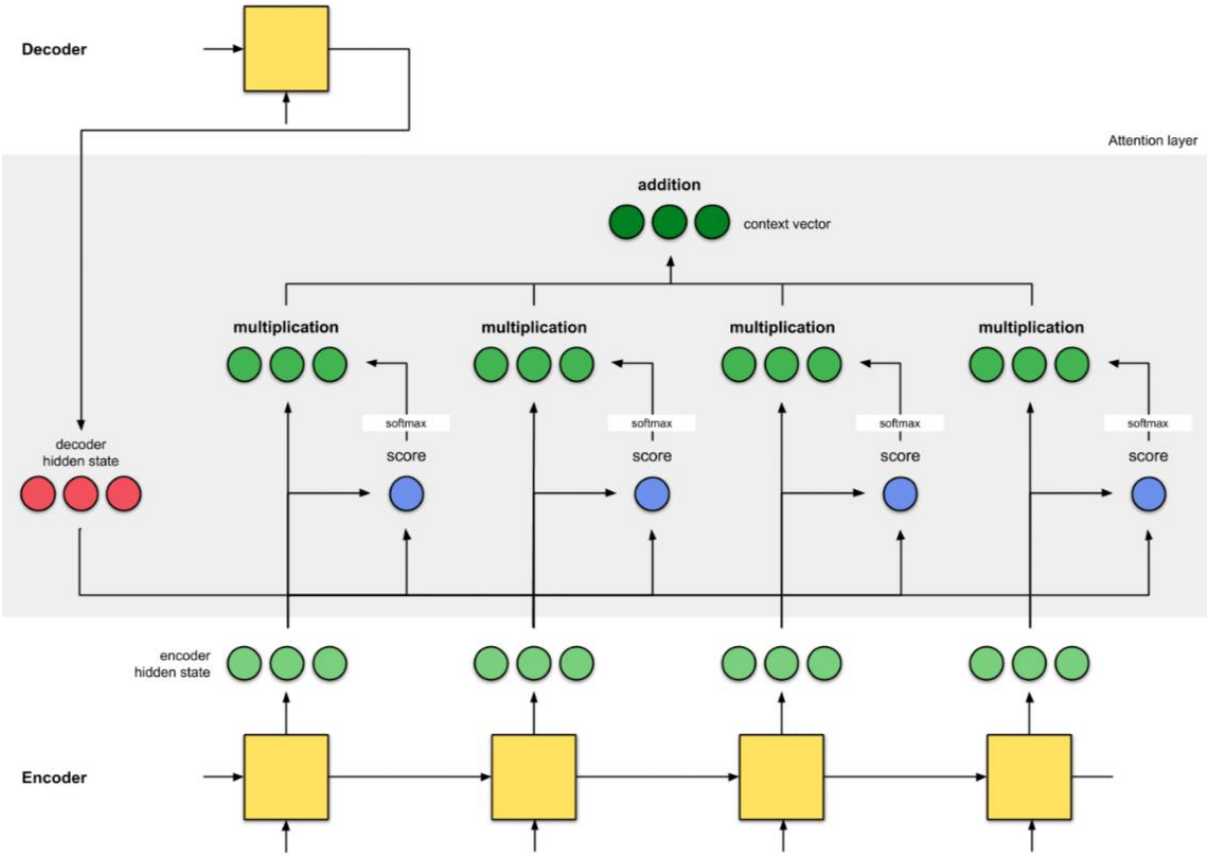
Ở bước này chúng ta nhận score đã qua softmax với từng encoder sẽ cho ra được context vector



encoder	alignment score	soft-maxed	context vector
[0, 1, 1]	15	0	[0, 0, 0]
[5, 0, 1]	60	1	[5, 0, 1]
[1, 1, 0]	15	0	[0, 0, 0]
[0, 5, 1]	35	0	[0, 0, 0]

Bước 5 Tính tổng của các context vector

Việc này chỉ việc cộng các ma trận context vector lại với nhau

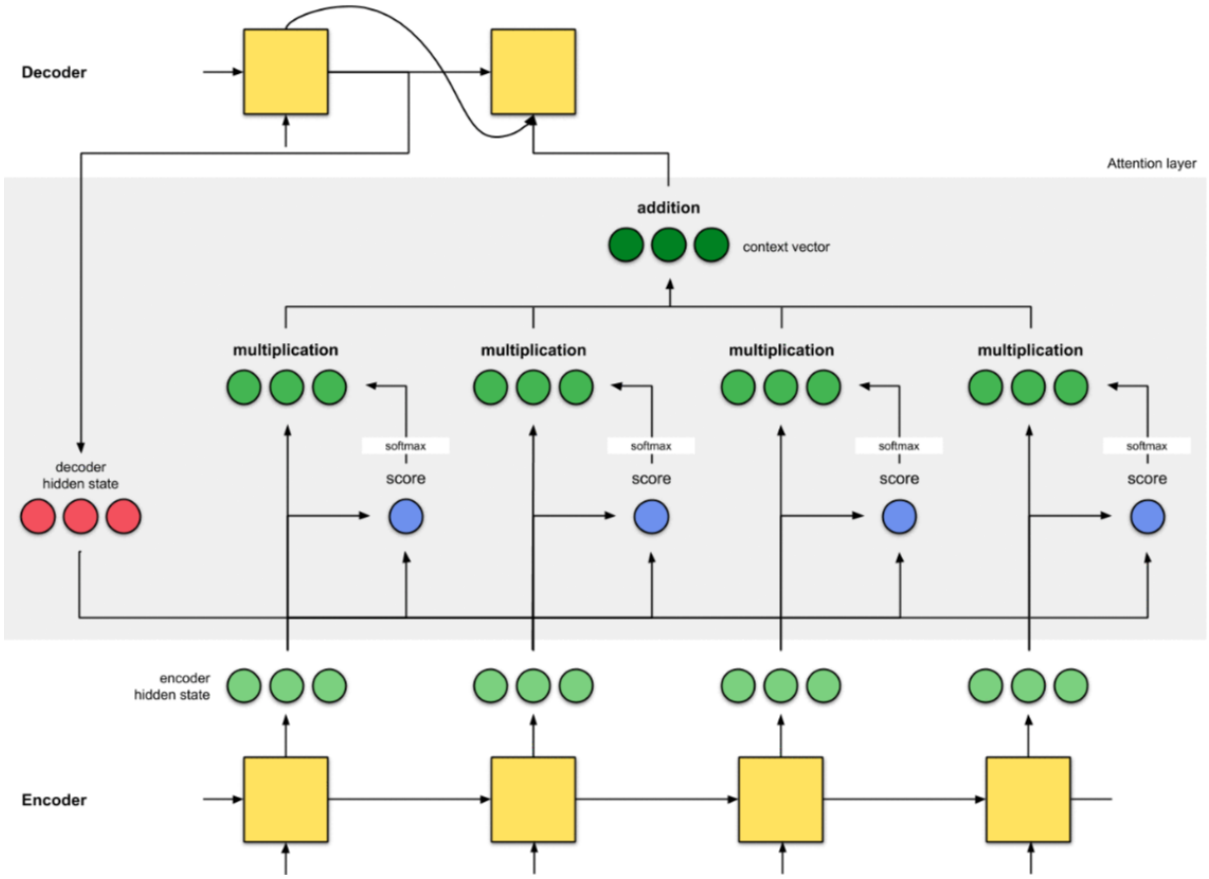


encoder	alignment score	soft-maxed	context vector
[0, 1, 1]	15	0	[0, 0, 0]
[5, 0, 1]	60	1	[5, 0, 1]
[1, 1, 0]	15	0	[0, 0, 0]
[0, 5, 1]	35	0	[0, 0, 0]

context = [0+5+0+0, 0+0+0+0, 0+1+0+0] = [5, 0, 1]

Bước 6 : sử dụng context vector cho decoder

Đến đây chúng ta đã có được context vector đầy đủ cho toàn bộ input sequence. Chúng ta sẽ đưa nó vào decoder để sử dụng tạo ra input mới.



Qua các bước thực hiện và ví dụ tính toán ta có thể thấy được cách hoạt động của cơ chế attention đem lại sự liên hệ ngữ nghĩa giữa input sequence và target sequence.

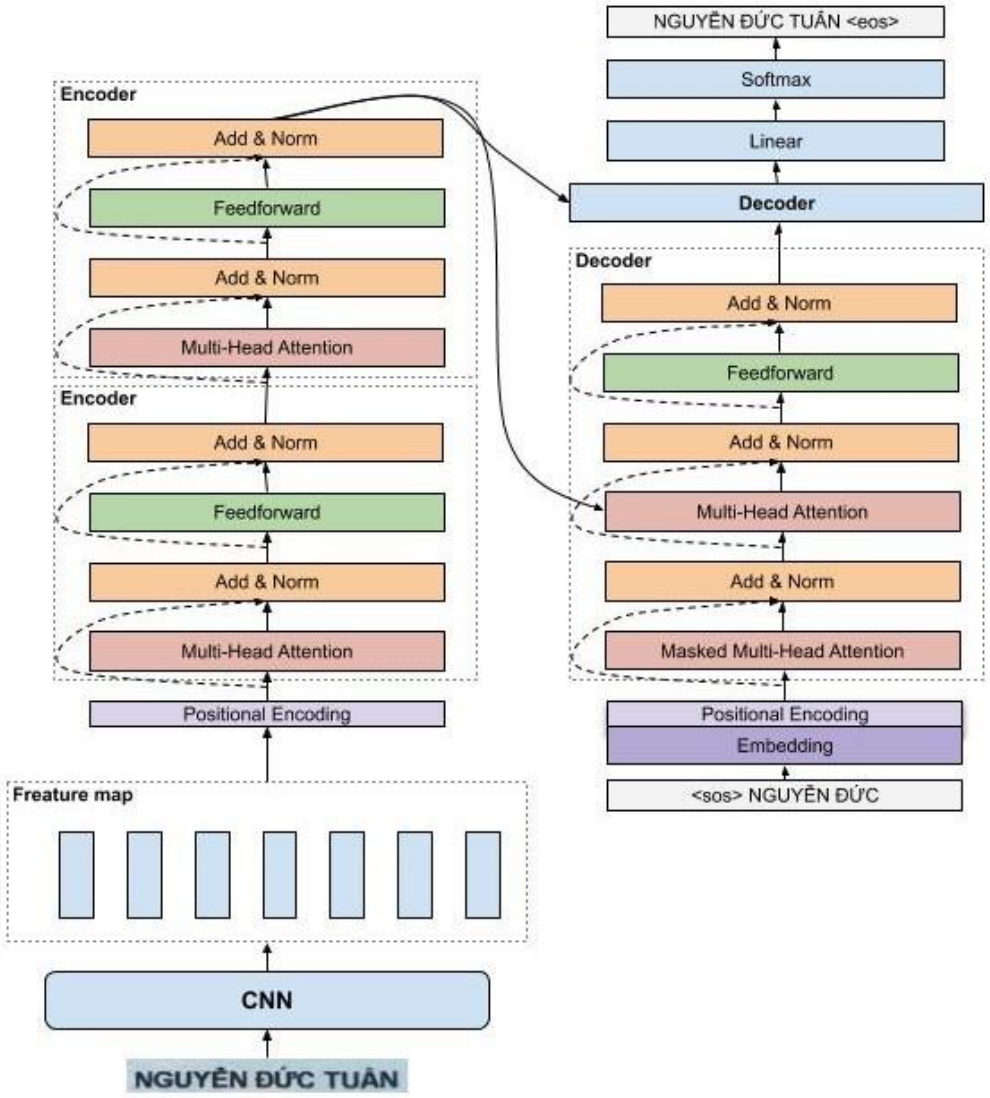
Tuy nhiên cơ chế attention thể hiện ngữ nghĩa của 2 câu, chưa thể hiện được sự liên kết và ngữ nghĩa của các từ có trong câu. Cơ chế self-attention sẽ giải quyết vấn đề này, và cơ chế self-attention thể hiện rõ sức mạnh trong kiến trúc Transformer

d. Kiến trúc transformer

Kiến trúc Transformer cũng dựa trên cấu trúc mã hóa (encoder) – giải mã (decoder) nhưng không sự dụng các mạng neural hồi qui, mà dựa vào cơ chế self attention để thực hiện xử lý các chuỗi. Kiến trúc Transformer có nhiều ưu điểm hơn so với kiến trúc seq2seq truyền thống :

- Có thể sử lý song song được nhiều tính toán không cần phải input theo tuần tự từng timestep như các mô hình mạng neural hồi quy khác, giúp giảm thời gian huấn luyện và tăng hiệu suất.
- Có thể xử lý được các chuỗi dài và phức tạp , không bị giới hạn bởi chiều dài của vector mã hóa hay bộ nhớ của mạng neural hồi qui.

Để áp dụng mô hình Transformer cho bài toán text recognition, chúng ta cũng làm tương tự như các mô hình khác là trước tiên trích xuất đặc trưng ảnh đầu vào và đem feature map trích xuất được chia thành các feature sequence và sau đó cho vào bộ encoder của mô hình transformer.



Như đã nói ở trên Transformer không thực hiện input tuần tự feature sequence theo từng time step mà input vào cùng 1 lúc, vậy thì câu hỏi đặt ra là làm thế nào transformer có thể giữ được tính thứ tự của các feature sequence ?

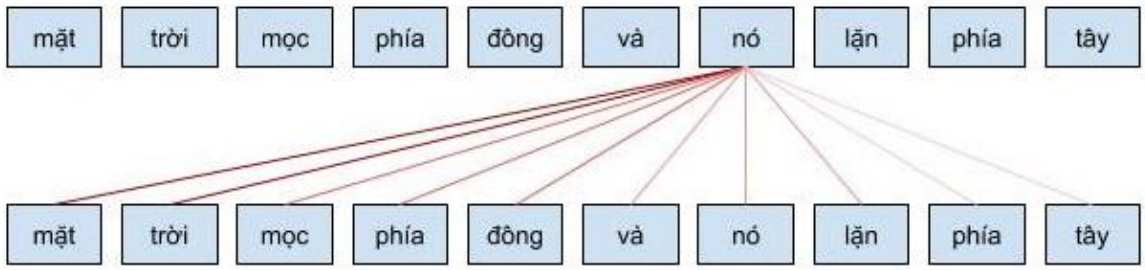
Để giải quyết vấn đề này tác giả đã thêm vào đó cơ chế Position Encoding dùng để đưa thông tin vị trí của từng feature sequence vào mô hình Transformer.

Chúng ta thấy trong khối encoder và decoder không sử dụng các mạng neural hồi qui mà sử dụng khối Multi-Head Attention, vậy khối này hoạt động như thế nào?

Trước khi nói về khối Multi-head Attention, chúng ta sẽ tìm hiểu về cơ chế self attention, cơ chế chính tạo nên khối Multi-head Attention.

Self Attention :

Self Attention là cơ chế cho phép mô hình mã hóa một từ có thể sử dụng thông tin của những từ liên quan tới nó trong câu.



Trên ảnh ta thấy cơ chế attention sẽ tìm ra các từ liên quan tới từ “nó” trong câu. Việc này sẽ đóng góp vào giúp cho mô hình có tính ngữ nghĩa cao hơn.

Chúng ta có thể hình dung cơ chế self attention giống như cơ chế tìm kiêm : tìm kiếm các từ trong câu từ nào giống với từ “nó” nhất.

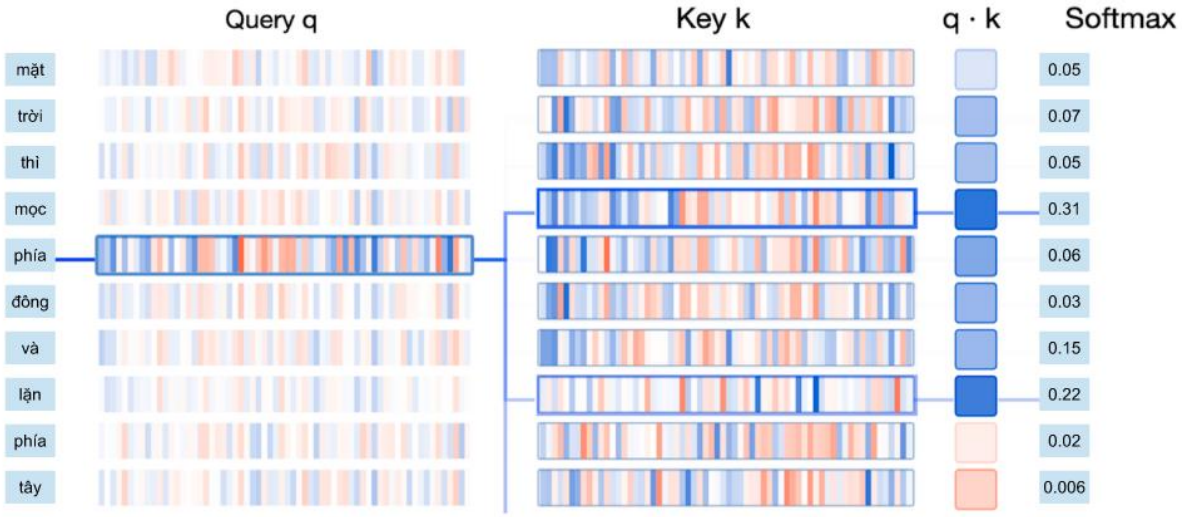
Giả sử chúng ta có 3 vector : query, key, value đại diện cho từng từ.

Thì có thể hiểu :

- vector query : vector chứa thông tin của từ tìm kiếm. Giống như câu querry của google search

- vector key : vector dùng để biểu diễn thông tin các từ được so sánh với từ tìm kiếm trên, giống như các trang web mà google sẽ so sánh với từ khóa mà bạn đưa vào để tìm kiếm

- vector value : vector dùng để biểu diễn nội dung thông tin, ý nghĩa của các từ, có thể nghĩ nó giống như nội dung trang web sẽ hiển thị ra khi bạn tìm kiếm được.



Với các ý tưởng trên, sau đây là minh họa cách hoạt động của cơ chế self-attention

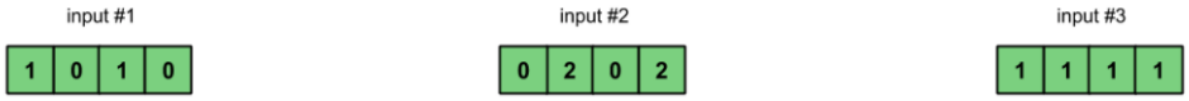
Như vậy ta có thể hình dung hóa cơ chế self-attention là sự kết hợp giữa query và key-value để tạp ra output :

⇨ Công thức như sau Attention(Q,K,V) = softmax($\frac{Q*K^T}{\sqrt{d_k}}$) x V

Trong đó $\frac{1}{\sqrt{d_k}}$ là hệ số tỷ lệ, d_k là số chiều của key.

Bài toán ví dụ :

Giả sử ta có 3 input (3 từ trong input sequence), mỗi input vector có 4 chiều :




```
Input 1: [1, 0, 1, 0]
Input 2: [0, 2, 0, 2]
Input 3: [1, 1, 1, 1]
```

Bước 1 : Khởi tạo ma trận trọng số cho Key, Query, Value

Ma trận trọng số cảu Key :

```
[[0, 0, 1],
 [1, 1, 0],
 [0, 1, 0],
 [1, 1, 0]]
```

Ma trận trọng số của Query :

```
[[1, 0, 1],
 [1, 0, 0],
 [0, 0, 1],
 [0, 1, 1]]
```

Ma trận trọng số của Value :

```
[[0, 2, 0],
 [0, 3, 0],
 [1, 0, 3],
 [1, 1, 0]]
```

Bước 2. Tính Key, Query, Value của từng từ

Chúng ta chỉ cần nhân dot product của input và các ma trận của nó

Key `

```
                [0, 0, 1]
[1, 0, 1, 0]    [1, 1, 0]   [0, 1, 1]
[0, 2, 0, 2] x  [0, 1, 0] = [4, 4, 0]
[1, 1, 1, 1]    [1, 1, 0]   [2, 3, 1]
```

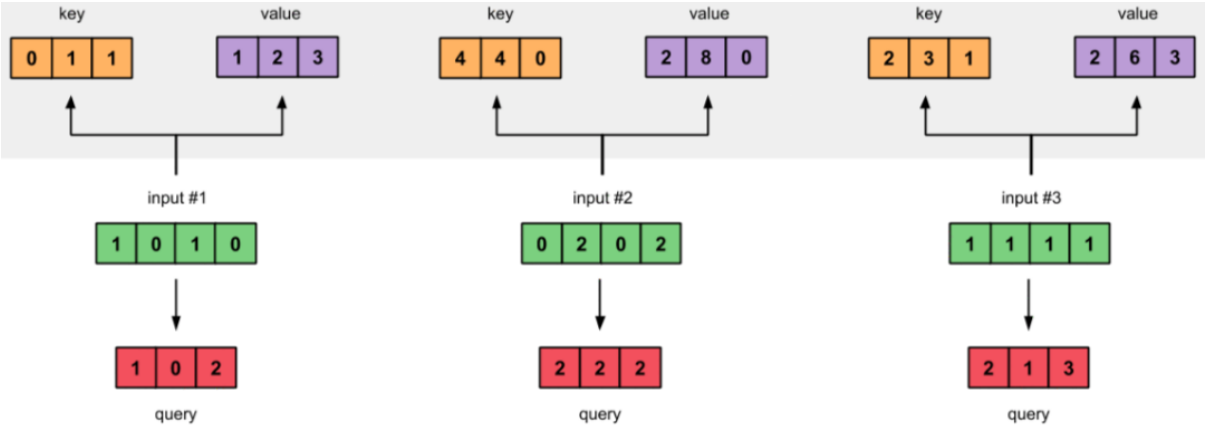
Query

```
                [1, 0, 1]
[1, 0, 1, 0]    [1, 0, 0]   [1, 0, 2]
[0, 2, 0, 2] x  [0, 0, 1] = [2, 2, 2]
[1, 1, 1, 1]    [0, 1, 1]   [2, 1, 3]
```

Value

```
                [0, 2, 0]
[1, 0, 1, 0]    [0, 3, 0]   [1, 2, 3]
[0, 2, 0, 2] x  [1, 0, 3] = [2, 8, 0]
[1, 1, 1, 1]    [1, 1, 0]   [2, 6, 3]
```

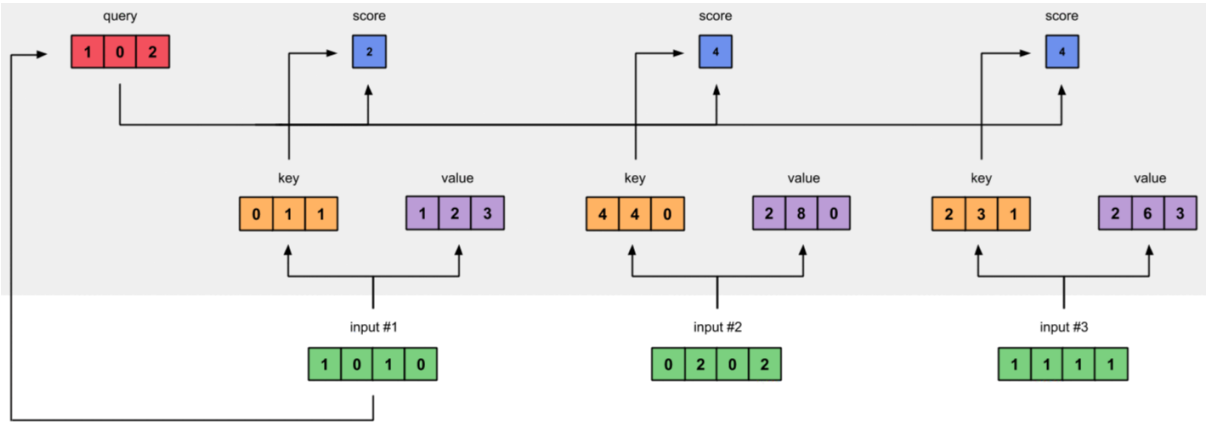
⇨ Ta được các ma trận Query, Key, Value của từng từ input



Bước 3 : Tính Attention Scores

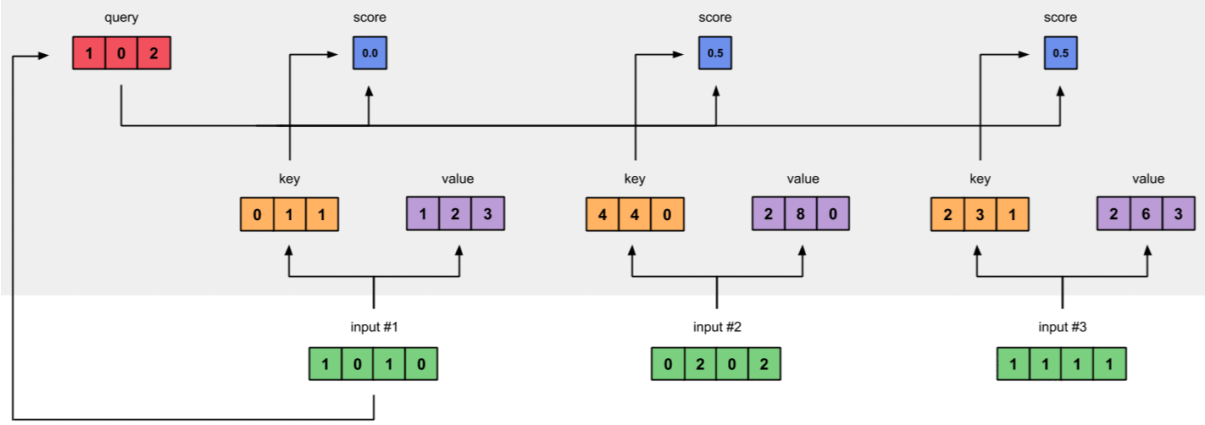
Nhân dot product của ma trận Query với các Key

```
                [0, 4, 2]
[1, 0, 2] x     [1, 4, 3] = [2, 4, 4]
                [1, 0, 1]
```



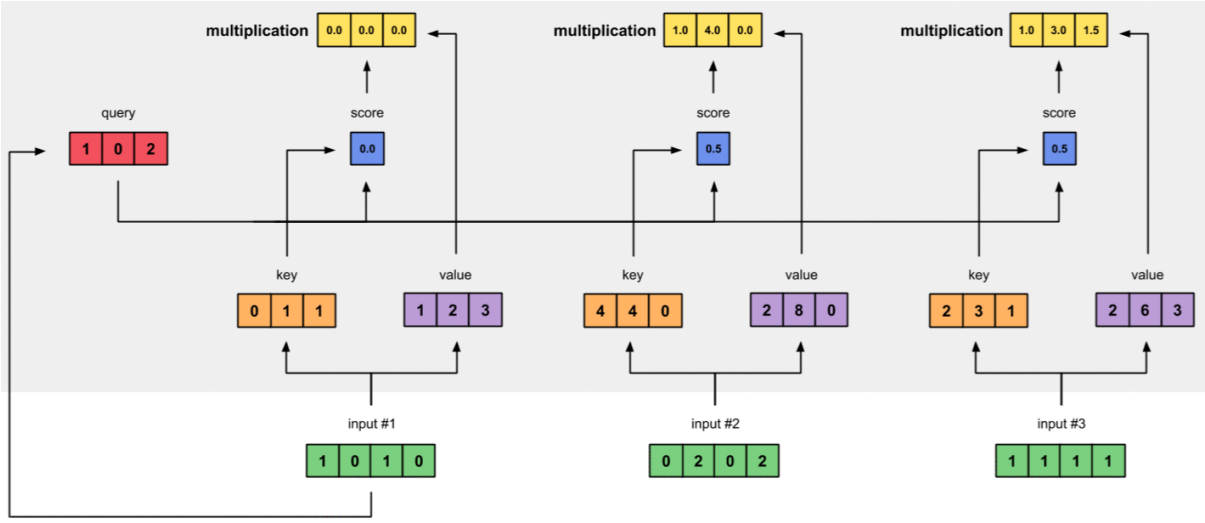
Bước 4 : TÍnh softmax

```
softmax([2, 4, 4]) = [0.0, 0.5, 0.5]
```



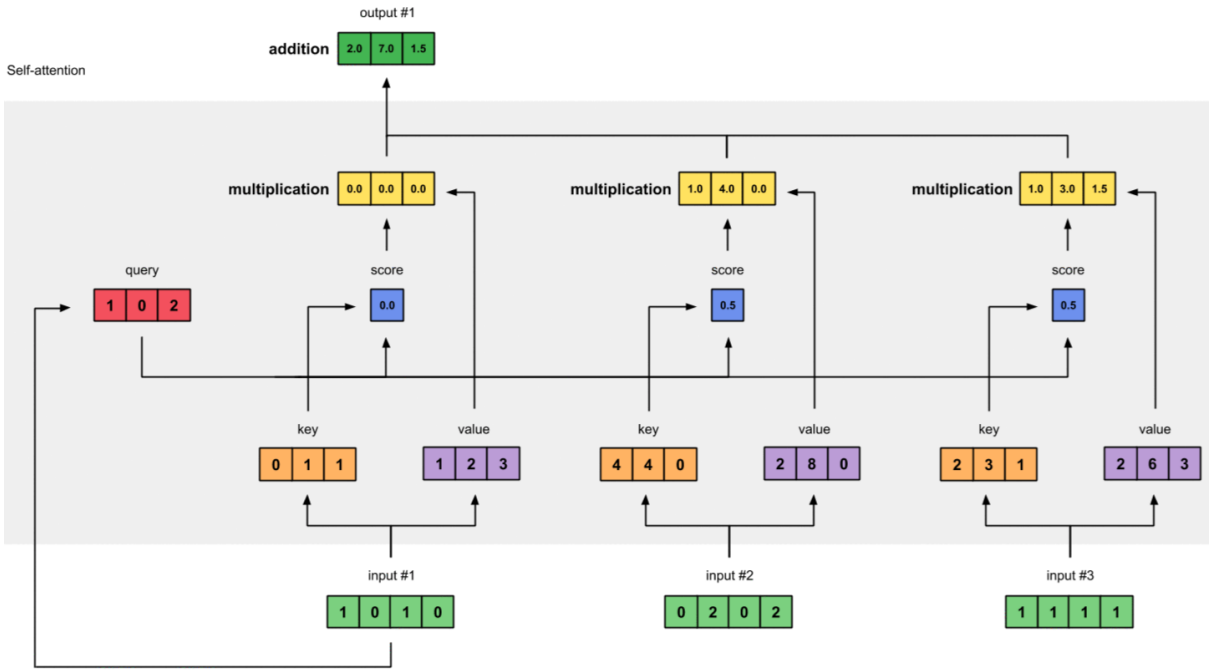
Bước 5. Tính tích của softmax($\frac{Q*K^T}{\sqrt{d_k}}$) x V

```
1: 0.0 * [1, 2, 3] = [0.0, 0.0, 0.0]
2: 0.5 * [2, 8, 0] = [1.0, 4.0, 0.0]
3: 0.5 * [2, 6, 3] = [1.0, 3.0, 1.5]
```

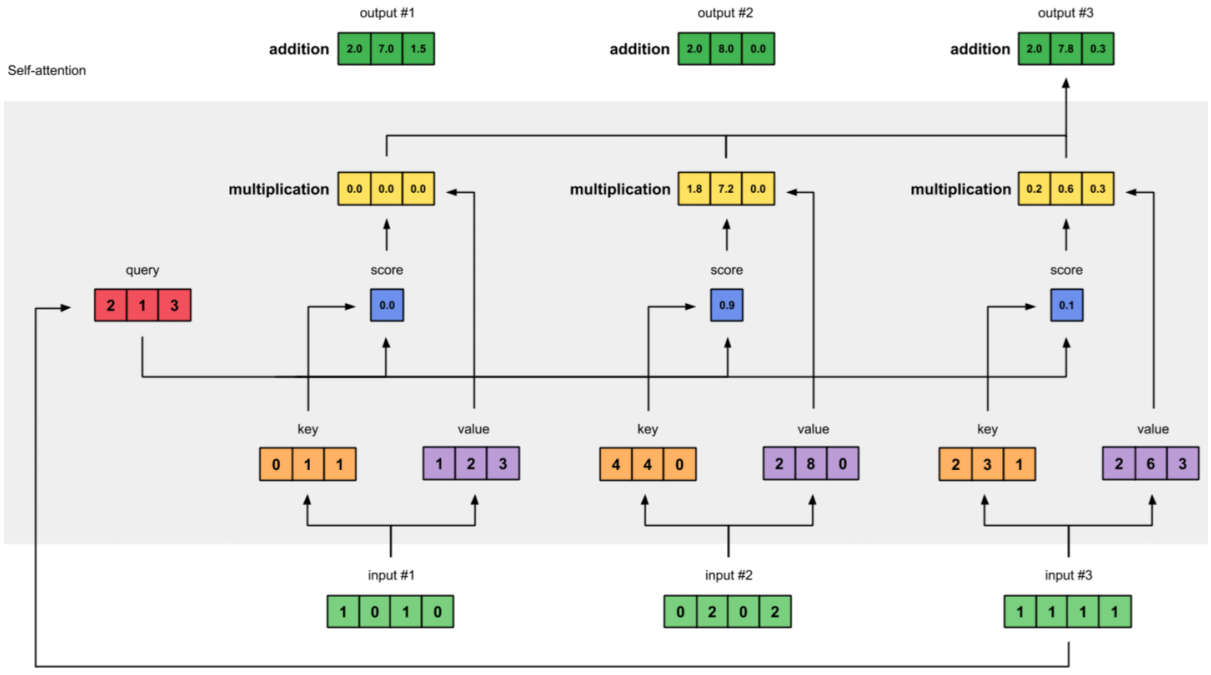


Bước 6 : Tính tổng giá trị của mutilplication ở trên chúng ta sẽ tìm được vector trọng số thể hiện mỗi quan hệ của từ đang xét với các từ còn lại trong câu

```
  [0.0, 0.0, 0.0]
+ [1.0, 4.0, 0.0]
+ [1.0, 3.0, 1.5]
-----------------
= [2.0, 7.0, 1.5]
```

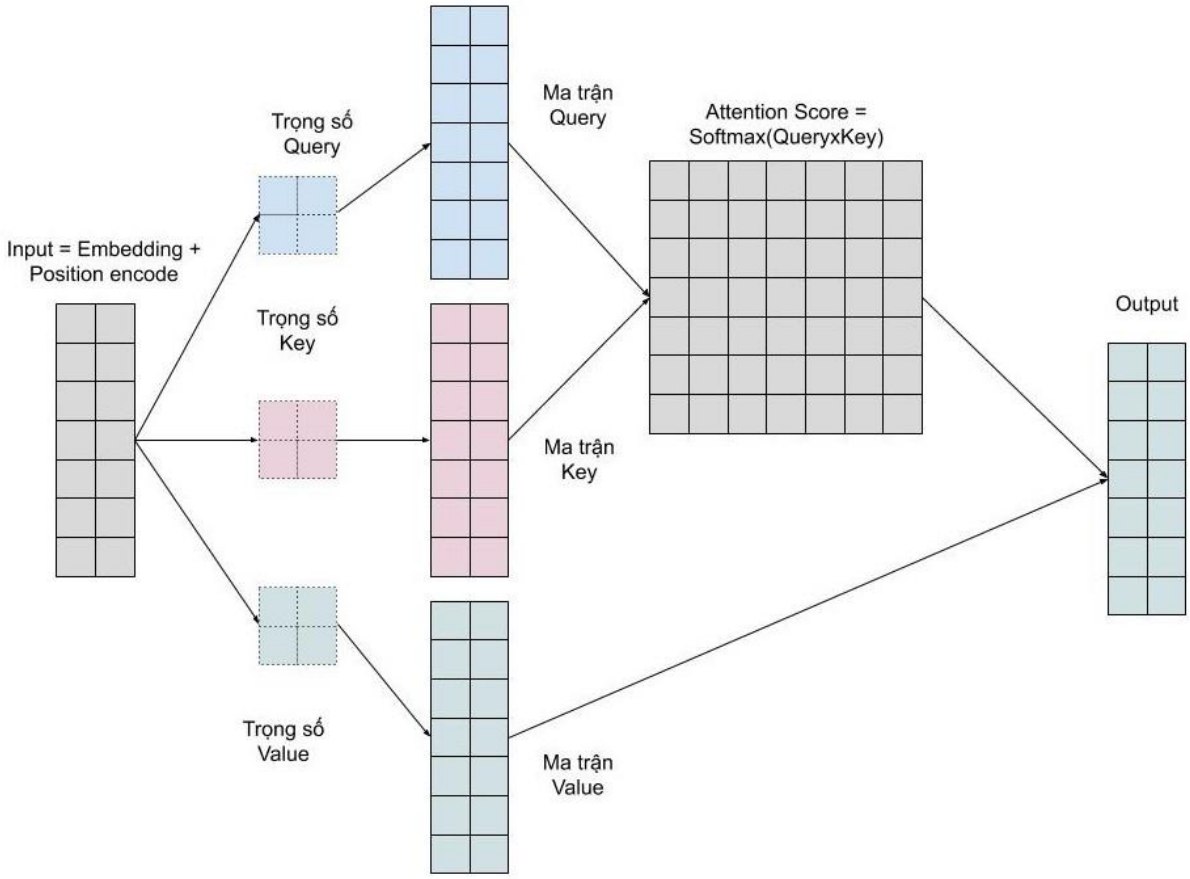


Bước 7 : Lặp lại bước 3 – 6 cho các từ input khác



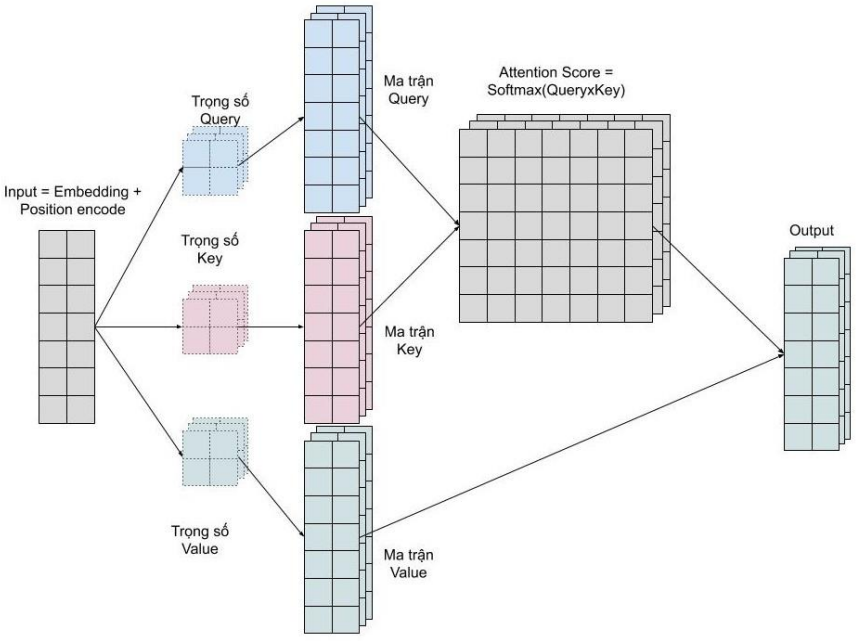
Như vậy sau 7 bước ta thấy được cách mà cơ chế self attention tìm sự tương quan của từ xét với các từ trong câu và cả chính nó có trong câu.

Đây là tóm gọn của cơ chế self attention :

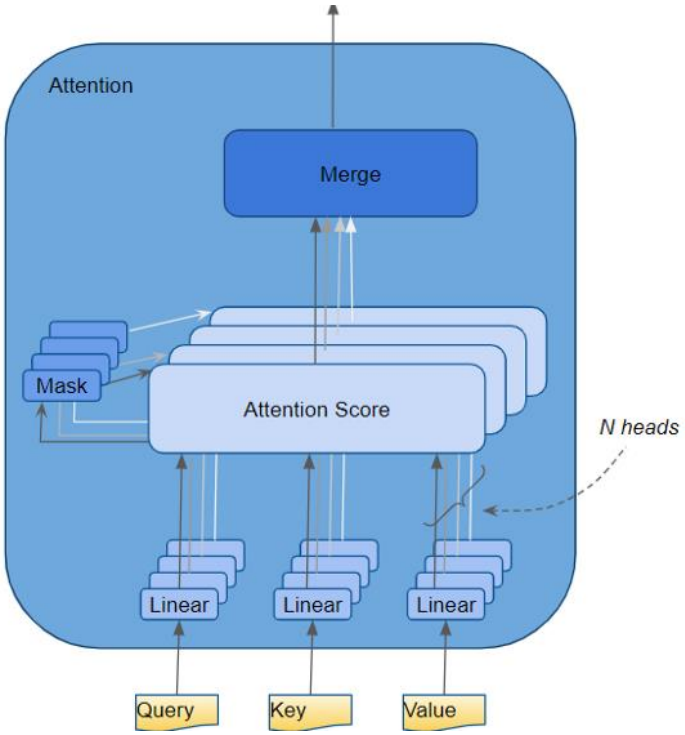


Multi-head attention

Nói đơn giản là chúng ta xếp chồng các lớp self attention lên nhau tạo nên một khối Multi-head attention ở bộ mã hóa. Chúng ta chỉ cần concat các output của từng self-attention lại với nhau thì sẽ tạo nên output của khối Multi-head attention.



Ở bộ giải mã ta thấy có thêm layer Mask, layer này có tác dùng che giấu đi phần thông tin của token phía sau, chỉ cho phép decoder sử dụng thông tin của token hiện tại và trước đó để tạo output.



2.2.3 Phương pháp sử dụng

Chúng em sử dụng mô hình vietocr cho bài toán text recognition, với mô hình vietocr sử dụng mô hình vgg để làm feature extractor, và sử dụng kiến trúc transformer cho việc học và nhận dạng chữ.

CHƯƠNG 3. THỰC NGHIỆM

3.1 Kết quả thực nghiệm

Kết quả đánh giá detection				
Method	Precision	Recall	H-mean	mAP
Dictguided	0.9214	0.4017	0.66155	0.7115
YOLOv7	0.8929	0.5128	0.70285	0.8108

Bảng kết quả so sánh các phương pháp detection

Kết quả đánh giá recognition		
Method	CER	WER
Vietocr	0.2107	0.4348

Bảng kết quả so sánh các phương pháp recognition

3.2 Nhận xét

- Yolo có độ chính xác cao hơn so với phương pháp Dictguided, tuy nhiên khi quan sát kết quả detect, yolo thường detect thiếu dấu dẫn đến sai về mặt ngữ nghĩa.
- Dictguided là phương pháp end to end , tuy nhiên kết quả huấn luyện và đánh giá với reg rất tệ, chỉ áp dụng được phương pháp detect, đạt kết quả khá cao trong bộ dữ liệu.
- Vietocr là phương pháp đạt độ chính xác rất cao trong tiếng Việt, tuy nhiên phương pháp này xử lý chưa tốt các trường hợp mờ và bị nhiễu, nhận diện thiếu dấu.

CHƯƠNG 4. CHƯƠNG TRÌNH DEMO

Source Code: https://github.com/cauhamau/CS338-Nhandang

4.1 Công cụ gán nhãn dữ liệu tự động

Công cụ gán nhãn dữ liệu là một ứng dụng được phát triển nhằm hỗ trợ quá trình tự động gán nhãn dữ liệu cho bài toán Scene Text từ các model có sẵn. Ứng dụng này giúp tiết kiệm thời gian và chi phí khi xây dựng bộ dữ liệu mới bằng cách tự động nhận diện và gán nhãn, sau đó chỉ cần điều chỉnh lại kết quả để đạt được độ chính xác cao hơn.

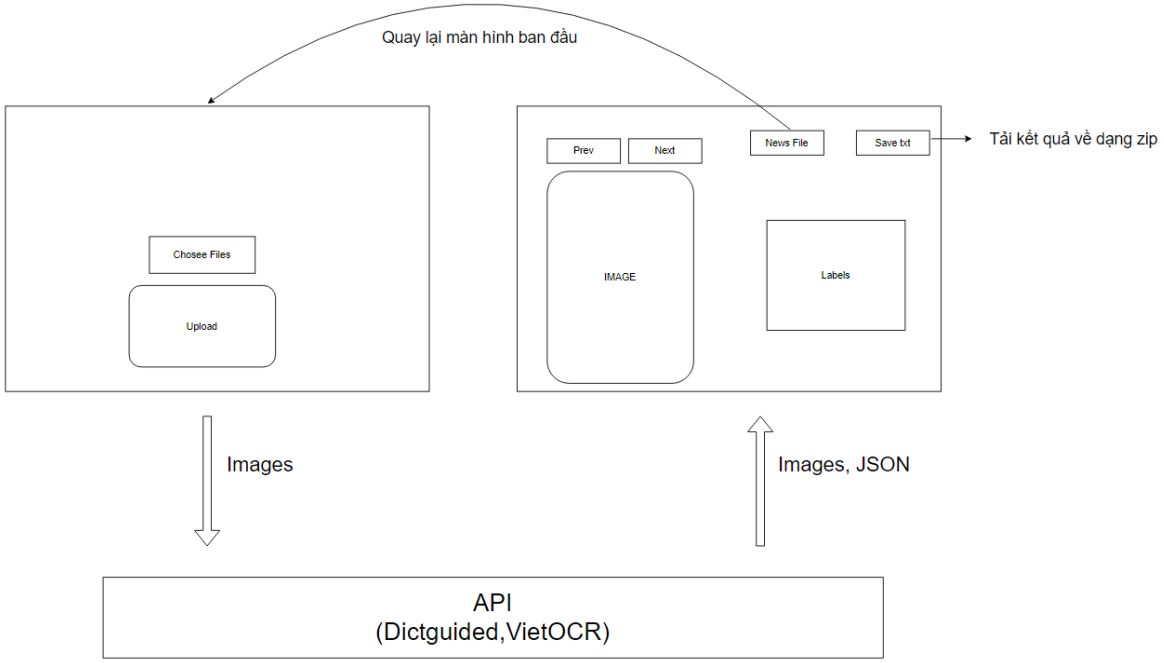
Công cụ này sử dụng Flask, JavaScript và Docker để dễ dàng sử dụng. Flask là một framework phát triển ứng dụng web sử dụng ngôn ngữ Python, được sử dụng để xây dựng backend của ứng dụng. Nó giúp xử lý các yêu cầu từ người dùng, tạo ra các trang HTML và kết nối với cơ sở dữ liệu.

JavaScript được sử dụng để tạo ra các tính năng tương tác và xử lý dữ liệu phía client. Nó cho phép người dùng tương tác với các thành phần trên giao diện người dùng, xử

lý sự kiện, điều chỉnh kết quả cho chính xác hơn và gửi yêu cầu tới máy chủ thông qua các API.

Để triển khai ứng dụng một cách dễ dàng và hạn chế lỗi môi trường, công cụ được đóng gói bằng Docker. Docker là một nền tảng mã nguồn mở cho việc đóng gói ứng dụng và các thành phần của nó vào các container độc lập. Điều này giúp đơn giản hóa quá trình triển khai và đảm bảo tính nhất quán giữa môi trường phát triển và triển khai.

4.2 Các chức năng chính



Hình 3: Cách hoạt động của công cụ

Công cụ Gán nhãn dữ liệu cung cấp các chức năng quan trọng để hỗ trợ quá trình gán nhãn dữ liệu tự động cho bài toán Scene Text. Dưới đây là mô tả về các chức năng chính của công cụ:

Màn hình 1: Chọn và Upload Files

- Người dùng có thể chọn nhiều file ảnh từ thiết bị của mình bằng cách nhấn nút "Chosee Files".
- Sau khi chọn các file ảnh, người dùng có thể nhấn nút "Upload" để gửi các file ảnh lên server.
- Các file ảnh được gửi đến server để tiến hành nhận dạng thông qua 2 model.

Màn hình 2: Xem và Gán nhãn dữ liệu

- Khi quá trình xử lý hoàn tất, công cụ sẽ hiển thị các kết quả được trả về từ 2 model trên giao diện. (Ảnh và bounding box ở bên trái, nội dung ở bên phải).
- Người dùng có thể thêm, xóa, sửa bounding box và nội dung.
- Người dùng có thể sử dụng các nút "Prev" và "Next" để chuyển đổi giữa các ảnh và xem kết quả tương ứng của từng ảnh.
- Nút "New Files" cho phép người dùng quay lại Màn hình 1 để chọn và upload các file ảnh mới. Đồng thời, dữ liệu cũ trên server sẽ được xóa.
- Nút "Save txt" cho phép người dùng tải về kết quả gán nhãn dưới dạng file zip. Các kết quả này bao gồm thông tin về vị trí và nội dung của tất cả ảnh.

CHƯƠNG 5. TỔNG KẾT

Qua các nghiên cứu và tìm hiểu nhóm đã thực hiện một số nội dung như sau:

1. Tìm hiểu tổng quan về bài toán các phương pháp về phát hiện và nhận dạng văn bản trong ảnh.
2. Đánh giá một số phương pháp tiên tiến hiện nay trên tập dữ liệu VinText.
3. Xây dựng demo dựa trên các kết quả đã đạt được.

Tên	Công việc	Mức độ hoàn thành
THƯƠNG	Huấn luyện và đánh giá các mô hình, làm slide, hỗ trợ xây dựng môi trường, viết báo cáo.	100%
TRỌNG	Xây dựng ứng dụng, tìm hiểu các phương pháp giải quyết bài toán, làm slide, viết báo cáo.	100%
HẬU	Tìm hiểu tổng quát bài toán, huấn luyện mô hình, làm slide, viết báo cáo.	100%
HÀO	Xây dựng ứng dụng, làm slide, viết báo cáo.	100%

Bảng phân công

CHƯƠNG 6. TÀI LIỆU TRÍCH DẪN

[1] Liu, Y., Chen, H., Shen, C., He, T., Jin, L., & Wang, L. (2020). ABCNet: Real-Time Scene Text Spotting With Adaptive Bezier-Curve Network. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 9806-9815.

[2] Nguyen, N.L., Nguyen, T., Tran, V., Tran, M., Ngo, T.D., Nguyen, T.H., & Hoai, M. (2021). Dictionary-guided Scene Text Recognition. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 7379-7388.

[3] Wang, C., Bochkovskiy, A., & Liao, H.M. (2022). YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *ArXiv, abs/2207.02696*.

[4] Tìm hiểu mô hình Transformer - Ngươi Không Phải Là Anh Hùng, Ngươi Là Quái Vật Nhiều Đầu. (Nguồn: Quoc Pham)

[5] Self-Attention và Multi-head Sefl-Attention trong kiến trúc Transformer (Nguồn: SuNT)

[6] Qin, Y., & Zhang, Z. (2020). *Summary of Scene Text Detection and Recognition. 2020 15th IEEE Conference on Industrial Electronics and Applications (ICIEA).*

[7] Ye, Q., & Doermann, D. (2015). Text Detection and Recognition in Imagery: A Survey. IEEE Transactions on Pattern Analysis and Machine Intelligence, 37(7), 1480–1500.