



ĐỒ ÁN CUỐI KỲ
CS532.M21.KHCL
GVHD: ThS. Đỗ Văn Tiến

TRÍCH XUẤT THÔNG TIN BÌA SÁCH

Ngày 21 tháng 6 năm 2022

Nhóm thực hiện:
Lương Phạm Bảo 19521242
Nguyễn Gia Thông 19520993
Phạm Ngọc Dương 19521412

MỤC LỤC

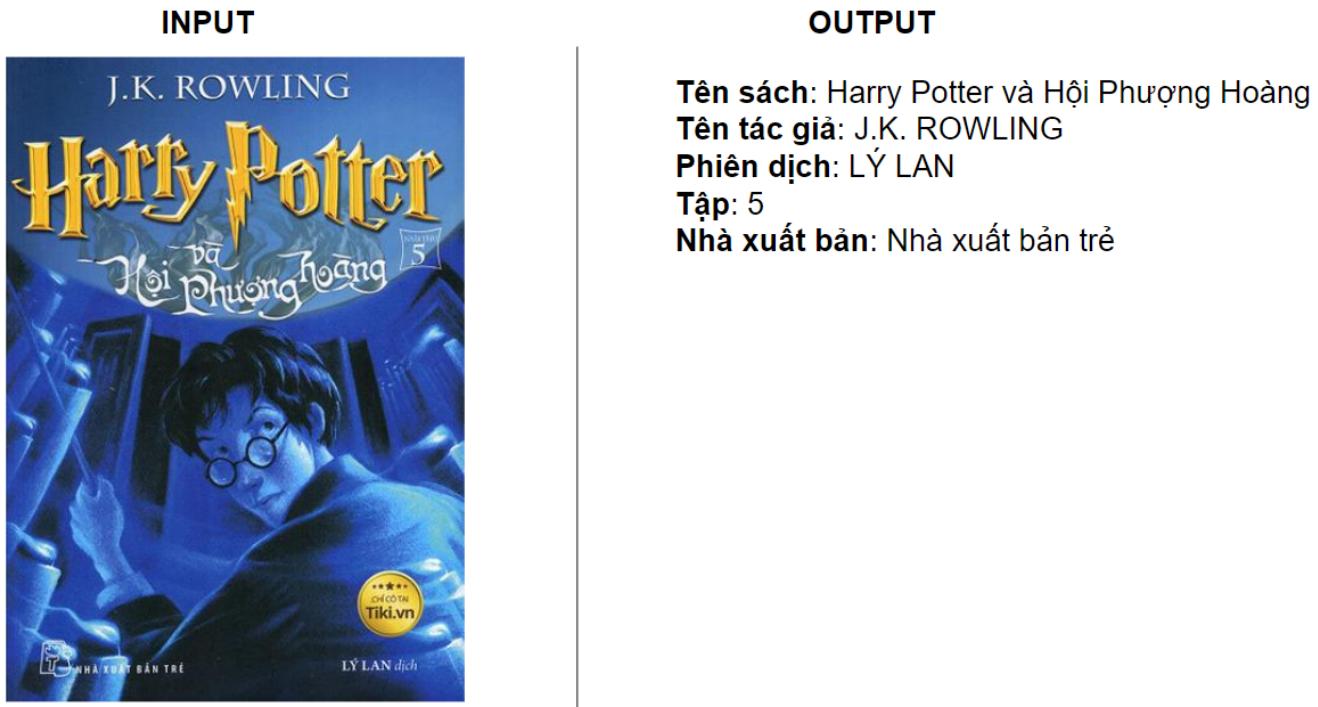
1 Giới thiệu	2
1.1 Dataflow	2
1.2 Phương pháp đề xuất	2
1.3 Thách thức	3
2 Giới thiệu bộ dữ liệu	3
3 Giới thiệu các Model	4
3.1 Giới thiệu về Model YOLO cho object detection	4
3.1.1 Cấu trúc mạng của YOLO	4
3.1.2 Cách YOLO hoạt động	4
3.1.3 Loss Function	5
3.2 Các Model sử dụng cho Text Localization	6
3.2.1 ABCNet	6
3.2.2 CRAFT Text Detector	6
3.3 Text Recognition-VietOCR	7
3.3.1 Giới thiệu mô hình VietOCR	7
3.3.2 Kiến trúc Network	7
4 Kết quả thực nghiệm	9
4.1 Datasets	9
4.1.1 Object detection	9
4.1.2 Text localization	9
4.1.3 VietOCR	10
5 Đánh giá	10
5.1 Intersection over Union (IoU)	10
5.2 True/False Positive/Negative	10
5.3 Precision	11
5.4 Recall	11
5.5 Average Precision (AP)	11
5.6 Mean Average Precision (mAP)	11
5.7 Accuracy sequence và character	12
5.8 Đánh giá chung	12
6 Thiết kế hệ thống	12
6.1 Kiến trúc hệ thống	13
6.1.1 Giao diện hệ thống	13
6.1.2 Screenflow	14
6.2 Demo	14

1 Giới thiệu

Ngày nay theo sự phát triển CNTT nói chung và AI nói riêng, việc chúng ta chuyển đổi, số hóa thông tin đã trở thành nhu cầu cấp bách, thiết yếu nhất đối với xã hội ngày nay. Trên thực tế có rất nhiều ứng dụng số hóa thông tin như : nhận diện biển số xe, hỗ trợ đọc văn bản cho người khiếm thị, tự động rút trích thông tin các giấy tờ tùy thân như chứng minh thư, CCCD... Với việc các thư viện lớn của các viện nghiên cứu ,trường đại học ngày càng có nhiều tài liệu hơn, việc trích xuất thông tin từ các tài liệu, thông tin từ bìa sách trở thành yêu cầu cấp thiết để phục vụ cho việc quản lý, chuyển đổi lên các thư viện sách điện tử....

1.1 Dataflow

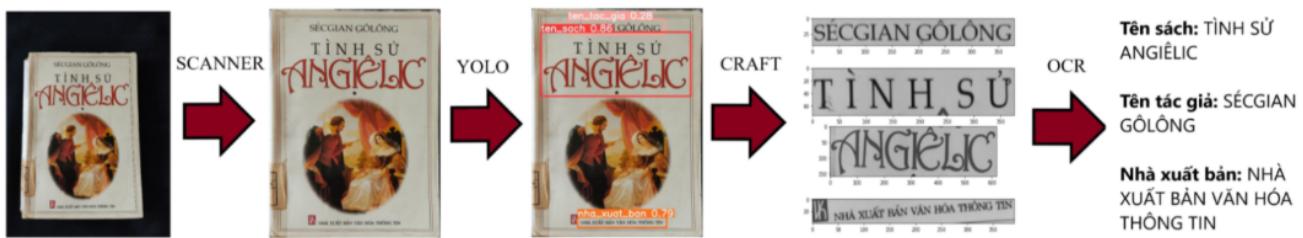
- Đầu vào: Ảnh bìa sách.
- Đầu ra: Các thông tin của bìa sách: Tên sách, Tên tác giả, phiên dịch, Tập, Nhà xuất bản.



Hình 1: Mô tả bài toán

1.2 Phương pháp đề xuất

Để giải quyết bài toán Số hóa tủ sách, ta cần phải giải quyết các bài toán nhỏ hơn sau đây:



Hình 2: Kiến trúc tổng quan hệ thống

- Preprocessing:** Ở bước này là giai đoạn tiền xử lý nhằm loại bỏ các vùng không thuộc sách nhằm giảm thiểu các yếu tố nhiễu để tăng hiệu suất huấn luyện
- Object detection:** Ở bước này, ứng với mỗi trường thông tin ta có thể xem như là một đối tượng cần nhận diện và áp dụng một số phương pháp tiêu biểu cho bài toán Object detection như Yolov4, Yolov5,...
- Text detection:** Ở bước này sau khi detect được bounding box chứa các trường thông tin, ta cần nhận biết các text nằm trong các bbox trên, vì thế ta sẽ sử dụng các mô hình phát hiện text như CRAFT, ABCNet, DB net,....

- **Text recognition:** Sau khi crop các vùng chứa text thành các ảnh chứa text, các bức ảnh trên sẽ được đưa qua mô hình nhận diện chữ như VietOCR, SRN, DeepText,...

1.3 Thách thức

- Một trong những thách thức đối với bài toán này là các dạng Art-text, trên thực tế các nhà xuất bản luôn tìm cách để bìa sách họ phát hành trong đẹp mắt và cuốn hút người đọc. Chính vì thế đa số các bìa sách hiện nay đều có các dạng chữ được cách điệu và điều đó sẽ ảnh hưởng đến mô hình nhận diện chữ.
- Vị trí của các trường thông tin không cố định. Không có một chuẩn mực nào cho việc đặt vị trí của các trường thông tin. Các nhà thiết kế bìa sách tùy vào mục đích mà đặt các thông tin ở vị trí khác nhau gây khó khăn cho việc xác định vị trí của chúng.
- Ngoài ra, các bìa sách còn chứa các thông tin gây nhiễu, những thông tin không thật sự cần thiết.

2 Giới thiệu bộ dữ liệu

Do bài toán gồm nhiều bước và ngoài trừ giai đoạn thu thập dữ liệu bìa sác thô là không yêu cầu phải label dữ liệu vì vậy, nhóm sẽ chuẩn dữ liệu cho 3 phần: dữ liệu tự thu thập để kiểm thử phần mềm, dữ liệu dùng để huấn luyện Object detection, dữ liệu dùng để huấn luyện mô hình Text recognition



Tác giả: Cécile Jugla và Jack Guichard

Lời: Nguyễn Trần Thiên Lộc

Người dịch: Thanh Hà

Tranh: Nguyễn Công Hoan - Biên soạn: Hiếu Minh

NHÀ XUẤT BẢN KIM ĐÔNG

NHÀ XUẤT BẢN KIM ĐÔNG
CÔNG TY TNHH
NGUYỄN SONG TÂM QUYỀN
Tạ Phương Hà dịch

DRAGON BALL
7 VIÊN NGỌC RỒNG

câu chuyện
lãng mạn

Những kẽ
trong bóng tối



3 Giới thiệu các Model

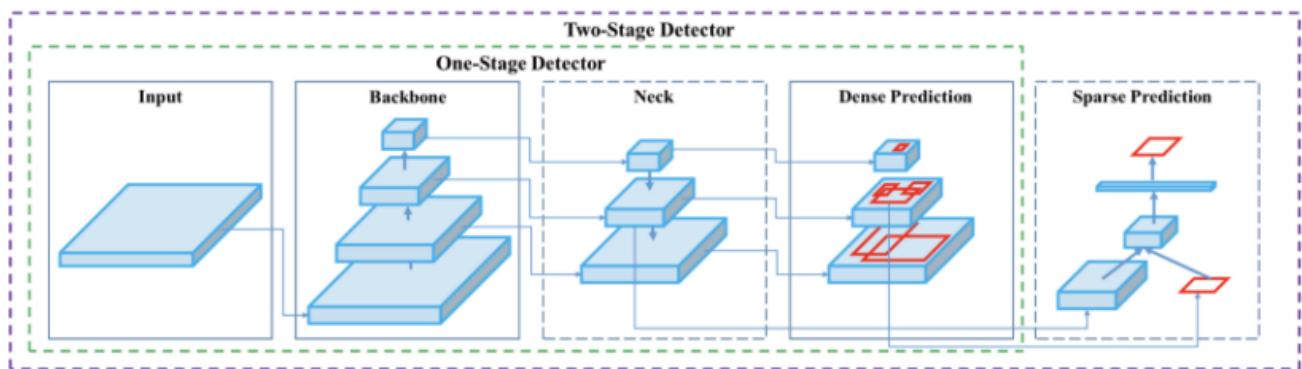
- Object detection: sử dụng model YOLOv5 cho bài toán Object Detection, YOLO được xem là phương pháp đầu tiên xử lý dữ liệu theo thời gian thực và đạt đến độ chính xác cao.
- Text localization:
- Text recognition: sử dụng model VietOCR cho bài toán text recognition chữ tiếng Việt.

3.1 Giới thiệu về Model YOLO cho object detection

YOLO là một mô hình mạng CNN cho việc phát hiện, nhận dạng, phân loại đối tượng. YOLO được tạo ra từ việc kết hợp giữa các convolutional layers và connected layers. Trong đó các convolutional layers sẽ trích xuất ra các feature của ảnh, còn full-connected layers sẽ dự đoán ra xác suất đó và tọa độ của đối tượng

3.1.1 Cấu trúc mạng của YOLO

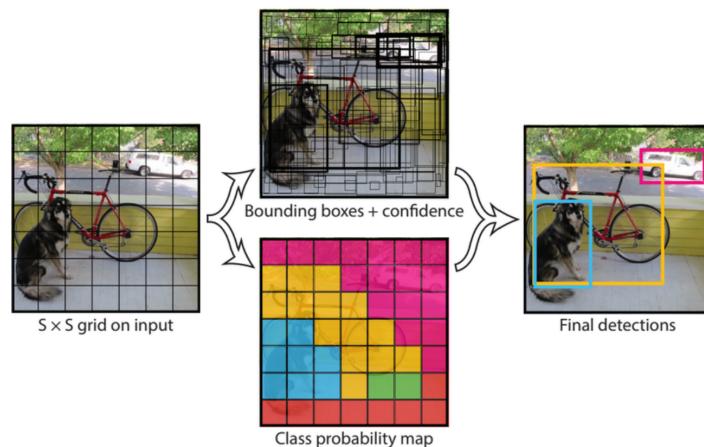
Kiến trúc mạng của YOLO gồm 4 phần:



Hình 3: Mô hình You Only Look Once

- Mạng Backbone: Trong YOLOv5, mạng CSP được dùng để trích xuất đặc trưng từ ảnh đầu vào. Sự thay đổi đã giúp tăng tốc thời gian xử lý đáng kể đối với các cấu trúc mạng nhiều tham số.
- Mạng Neck: Mạng Neck hoạt động dựa trên khái niệm pyramid (FPN) giúp cải thiện độ chính xác của mô hình đối với cái vật có kích thước nhỏ.
- Mạng Head: Nhiệm vụ chính của mạng này là phần nhận diện vật thể. Đầu ra của mạng sẽ là xác suất của lớp, độ tự tin của box và 4 kích thước chính của bounding box.

3.1.2 Cách YOLO hoạt động



Ý tưởng chính của YOLOv1 là chia ảnh thành một lưới các ô (grid cell) với kích thước SxS (mặc định là 7x7). Với mỗi grid cell, mô hình sẽ đưa ra dự đoán cho B bounding box. Ứng với mỗi box trong B bounding box này sẽ là 5 tham số x, y, w, h, confidence,

lần lượt là tọa độ tâm (x, y), chiều rộng, chiều cao và độ tự tin của dự đoán. Với grid cell trong lưới SxS kia, mô hình cũng dự đoán xác suất rơi vào mỗi class. Độ tự tin của dự đoán ứng với mỗi bounding box được định nghĩa là $p(Object)IoU$ trong đó $p(Object)$ là xác suất có vật trong cell và IoU là intersection over union của vùng dự đoán và ground truth. Xác suất rơi vào mỗi class cho một grid cell được ký hiệu $p(Class|Object)$. Các giá trị xác suất cho C class sẽ tạo ra C output cho mỗi grid cell. Lưu ý là B bounding box của cùng một grid cell sẽ chia sẻ chung một tập các dự đoán về class của vật, đồng nghĩa với việc tất cả các bounding box trong cùng một grid cell sẽ chỉ có chung một class.

Vậy tổng số output của mô hình sẽ là $SS(5B + C)$.

3.1.3 Loss Function

Hàm lỗi trong YOLO được tính trên việc dự đoán và nhãn mô hình để tính. Cụ thể hơn nó là tổng độ lỗi của 3 thành phần con sau:

Độ lỗi của việc dự đoán loại nhãn của object - Classification loss, hàm lỗi này chỉ tính trên những ô vuông có xuất hiện object, còn những ô vuông khác ta không quan tâm. Classification loss được tính bằng công thức sau:

$$L_{classification} = \sum_{i=0}^{S^2} \mathbb{I}_i^{obj} \sum_{c \in class} (p_i(c) - \hat{p}_i(c))^2$$

Trong đó:

\mathbb{I}_i^{obj} : bằng 1 nếu ô vuông đang xét có object ngược lại bằng 0

$\hat{p}_i(c)$: là xác xuất có điều của lớp c tại ô vuông tương ứng mà mô hình dự đoán

Độ lỗi của dự đoán tọa độ tâm, chiều dài, rộng của boundary box (x, y, w, h) (Localization loss) là hàm lỗi dùng để tính giá trị lỗi cho boundary box được dự đoán bao gồm tọa độ tâm, chiều rộng, chiều cao của so với vị trí thực tế từ dữ liệu huấn luyện của mô hình. Lưu ý rằng chúng ta không nên tính giá trị hàm lỗi này trực tiếp từ kích thước ảnh thực tế mà cần phải chuẩn hóa về [0, 1] so với tâm của bounding box. Việc chuẩn hóa này kích thước này giúp cho mô hình dự đoán nhanh hơn và chính xác hơn so với để giá trị mặc định của ảnh. Giá trị hàm Localization loss được tính trên tổng giá trị lỗi dự đoán tọa độ tâm (x, y) và (w, h) của predicted bounding box với ground-truth bounding box. Tại mỗi ô có chứa object, ta chọn 1 boundary box có IOU (Intersection over union) tốt nhất, rồi sau đó tính độ lỗi theo các boundary box này. Giá trị hàm lỗi dự đoán tọa độ tâm (x, y) của predicted bounding box và (,) là tọa độ tâm của truth bounding box được tính như sau:

$$\lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{I}_{ij}^{obj} (\sqrt{w_i} - \sqrt{\hat{w}_i})^2 + (\sqrt{h_i} - \sqrt{\hat{h}_i})^2$$

Giá trị hàm lỗi dự đoán (w, h) của predicted bounding box so với truth bounding box được tính như sau :

$$\lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{I}_{ij}^{obj} (\sqrt{w_i} - \sqrt{\hat{w}_i})^2 + (\sqrt{h_i} - \sqrt{\hat{h}_i})^2$$

Độ lỗi của việc dự đoán bounding box đó chứa object so với nhãn thực tế tại ô vuông đó - Confidence loss Là độ lỗi giữa dự đoán boundary box đó chứa object so với nhãn thực tế tại ô vuông đó. Độ lỗi này tính trên cả những ô vuông chứa object và không chứa object.

$$L_{confidence} = \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{I}_{ij}^{obj} (C_i - \hat{C}_i)^2 + \lambda_{noobject} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{I}_{ij}^{noobj} (C_i - \hat{C}_i)^2$$

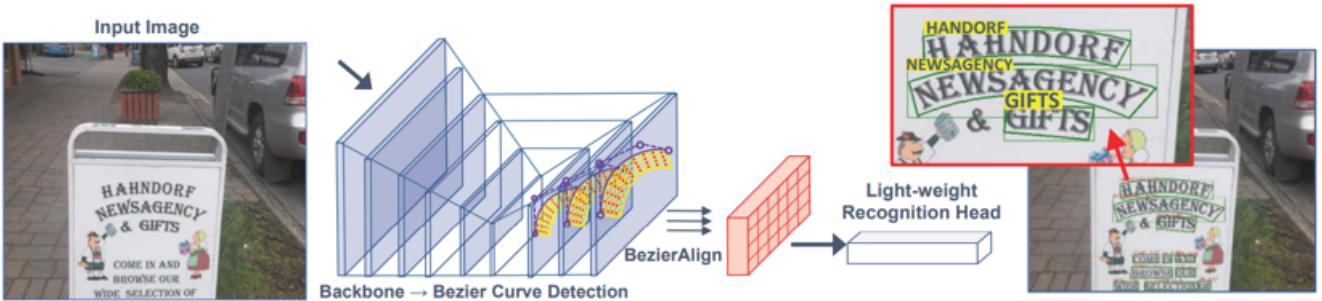
Total loss Tổng lại chúng ta có hàm lỗi là tổng của 3 hàm lỗi trên:

$$L_{total} = L_{classification} + L_{localization} + L_{confidence}$$

Models	MAP@0.5	MAP@0.5:0.95	Inference Time
YOLOv4	0.927	0.678	0.3
YOLOv5	0.908	0.653	0.25

3.2 Các Model sử dụng cho Text Localization

3.2.1 ABCNet



Hình 4

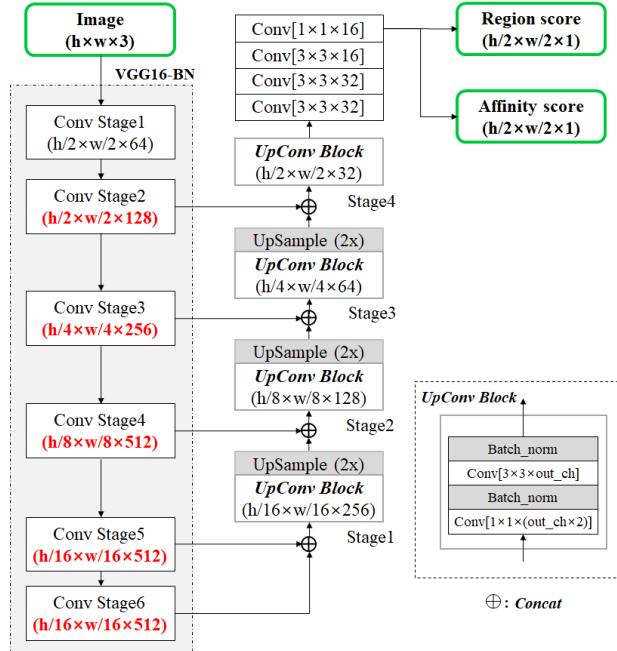
ABCNet là một framework End-to-end dùng để phát hiện văn bản có hình dạng tùy ý (xiên, cong, hoặc có dạng gợn sóng,...). Mô hình sử dụng single-shot, anchor-free CNN làm detection framework. Việc loại bỏ các anchor boxes giúp đơn giản hoá và giảm thời gian đáng kể trong việc thực hiện detect → phù hợp với bài toán Real-time Scene Text spotting. Dưới đây là pipeline của mô hình, trong đó Bezier Curve Detection dùng để detect scene text ở phần Detection Head và Bezier-Align dùng để chuẩn hoá các text cong về dạng thẳng để hỗ trợ cho việc nhận dạng nội dung text đó ở phần Recognition Head. Còn ở phần recognition sẽ sử dụng một mô hình rất hiệu quả cho các bài toán OCR là mô hình CRNN+CTC loss.

3.2.2 CRAFT Text Detector

Nhóm sử dụng model deep-learning có sẵn trên pypi/craft-text-detector 0.4.2: CRAFT: Character-Region Awareness For Text detection để thực hiện locate các text trên bìa sách. Đây là một PyTorch dùng cho craft text detection, nó detect được khá hiệu quả bằng cách tìm ra phân vùng của từng từ chữ cái và mối quan hệ giữa các chữ cái đó. Nó tạo ra hộp chữ nhật chứa các đoạn text dựa vào mối quan hệ giữa các chữ nó tách ra được. Nhóm sử dụng đoạn code có sẵn trên pypi, chỉ điều chỉnh một số tham số để thực hiện craft ảnh bìa sách. Ứng dụng này bao gồm:

- Input: ảnh cần nhận diện chữ
- Output: toạ độ các bbox chứa chữ từ đó để có thể crop các chữ ra để đưa vào model nhận diện chữ

Kiến trúc network: ứng dụng này hoạt động với mục đích chính là locate chính xác từng ký tự trong ảnh. Nhóm tác giả đã train một deep-learning neural network để predict ra vị trí của ký tự và mối quan hệ của chúng với nhau. Họ train model bằng một mạng tích chập đầy đủ được minh họa như sau:



Hình 5: Cấu trúc mạng sử dụng trong model.

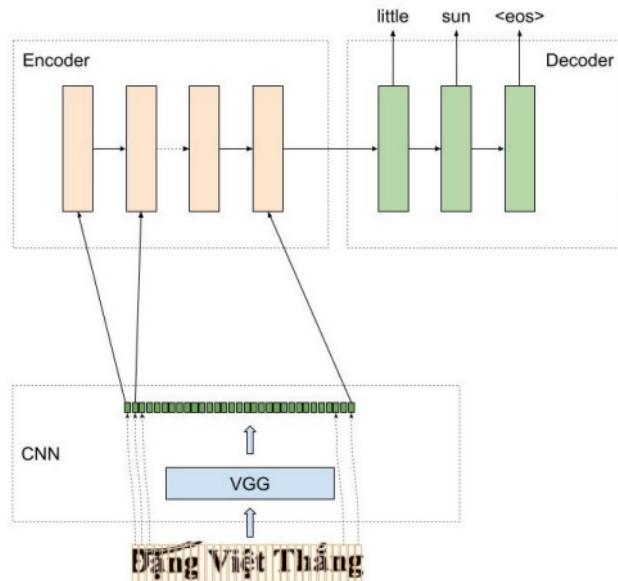
3.3 Text Recognition-VietOCR

3.3.1 Giới thiệu mô hình VietOCR

Thư viện này kết hợp CNN cùng hai mô hình khá nổi tiếng trong việc xử lý ngôn ngữ tự nhiên (cũng như về mặt hình ảnh) là: Transformer và Attention của seq2seq. Đây đều là những mô hình nổi tiếng, hiệu quả, đã được khắc phục nhiều hạn chế của các mô hình trước đó. Đặc biệt là Transformer (mới xuất hiện gần đây), khắc phục được tốc độ train của model sử dụng RNN cũng như về độ chính xác. Tuy nhiên Transformer lại predict khá chậm (cụ thể là so với Attention). Đặc biệt là Transformer (mới xuất hiện gần đây), khắc phục được tốc độ train của model sử dụng RNN cũng như về độ chính xác. Tuy nhiên Transformer lại predict khá chậm (cụ thể là so với Attention).

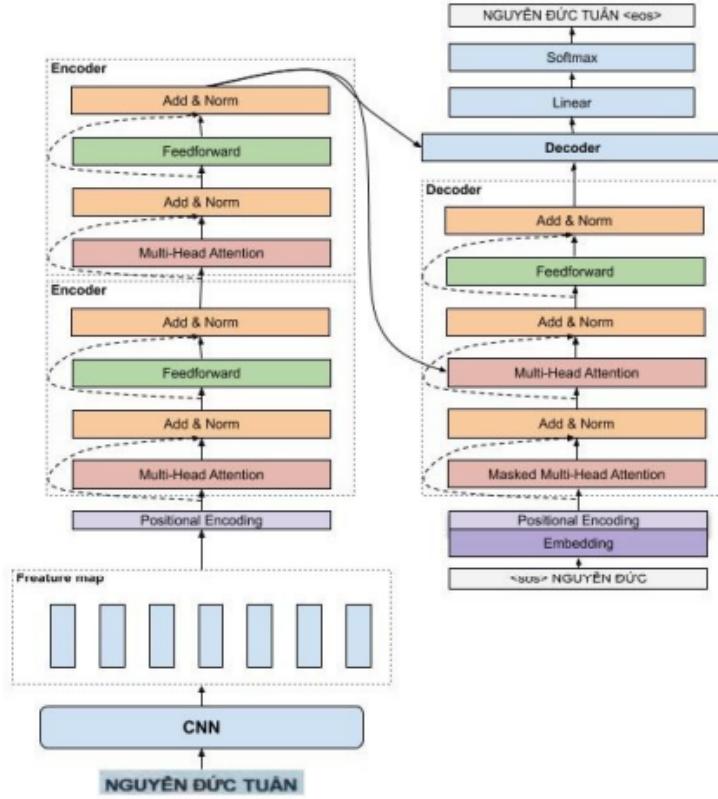
3.3.2 Kiến trúc Network

AttentionOCR



Hình 6: Mô hình dùng CNN để trích xuất đặc trưng sau đó đi qua seq2seq sử dụng cơ chế attention

TransformerOCR



Hình 7: Mô hình sử dụng CNN để trích xuất đặc trưng sau đó đi qua transformer

Đầu tiên nhóm không sử dụng model pretrain vì khi thử nó vô cùng không chính xác, gần như độ chính xác rất thấp. Nhóm chọn model **Transformer_OCR** do nó train nhanh hơn và có độ chính xác cao hơn nhiều so với **Attention_OCR**, điểm bất lợi duy nhất so với mô hình kia chính là thời gian predict chậm hơn như đã đề cập ở trên.

- **Model Zoo:** Mô hình này được huấn luyện trên tập dữ liệu gồm 10m ảnh, bao gồm nhiều loại ảnh khác nhau như ảnh tự phát sinh, chữ viết tay, các văn bản scan thực tế. Pretrain model được cung cấp sẵn. Model này có vẻ thích hợp với các tài liệu scan, đánh máy trên giấy,... Mô hình được train bằng 2 phương pháp attention và cả transformer với độ chính xác cùng thời gian predict như sau:

Backbone	Config	Precision full sequence	Time
VGG19-bn - Transformer	vgg_transformer	0.93	60ms @ 2080
VGG19-bn - Seq2Seq	vgg_seq2seq	0.88	10ms @ 2080

Ta có thể thấy độ chính xác của transformer cao hơn nhưng thời gian predict lại lâu hơn.

4 Kết quả thực nghiệm

4.1 Datasets

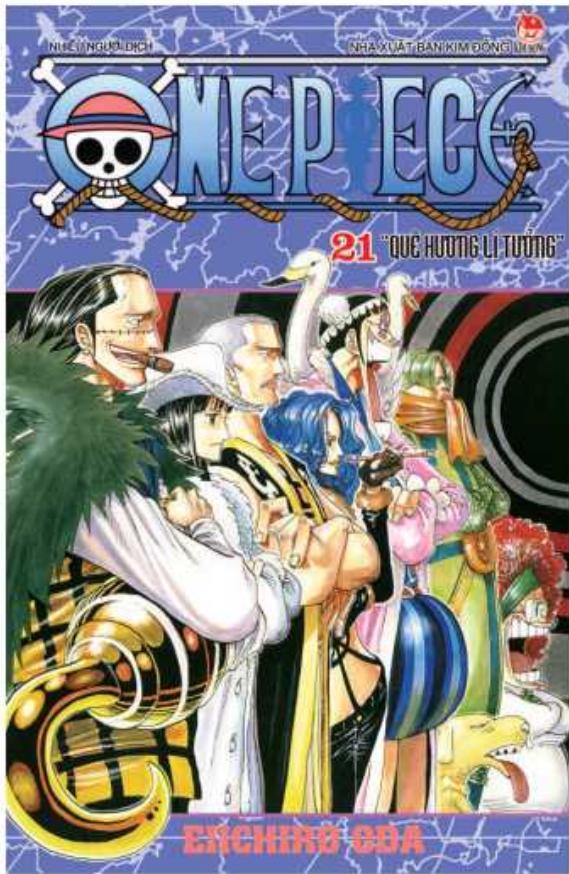
4.1.1 Object detection



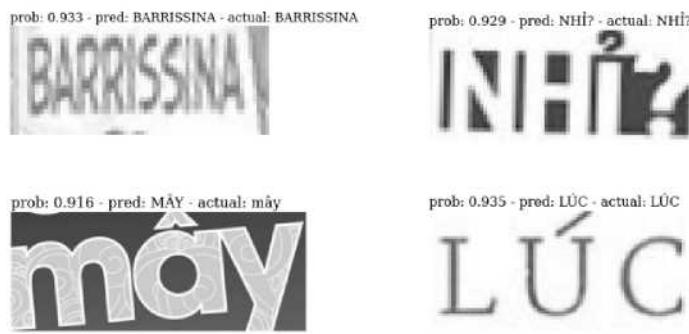
Hình 8

4.1.2 Text localization





4.1.3 VietOCR



Hình 9

5 Đánh giá

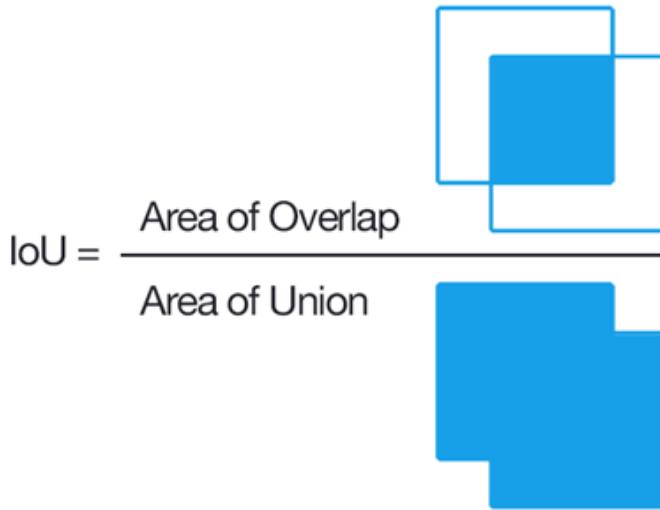
5.1 Intersection over Union (IoU)

Intersection over Union (IoU) là tỷ lệ diện tích giữa phần giao và phần hợp của bounding box dự đoán và bounding box thực tế.

Với Area of Overlap là phần diện tích mà hai bounding box dự đoán và bounding box thực tế giao nhau và Area of Union là phần diện tích mà cả hai bounding box bao phủ trên ảnh như hình 10.

5.2 True/False Positive/Negative

Kết quả của IoU là những giá trị trong khoảng (0,1) mỗi dự đoán sẽ có một giá trị IoU riêng. Để xác định liệu đó là dự đoán sai hay dự đoán đúng, chúng ta dựa vào một ngưỡng (threshold) cho trước (có thể là 0.5, 0.75, 0.95 tùy vào bài toán), nếu IoU



Hình 10: Hình biểu diễn cho độ đo Intersection over Union (IoU)

lớn hơn hoặc bằng ngưỡng thì đó là dự đoán đúng, còn lại là dự đoán sai. Dựa vào những khái niệm trên chúng ta định nghĩa True/false positive/negative như sau:

- True Positive (TP): các bounding box dự đoán với IoU lớn hơn hoặc bằng 1 giá trị threshold (thường là 0.5).
- False Positive (FP): các bounding box dự đoán với IoU nhỏ hơn threshold.
- False Negative (FN): mô hình không bắt được đối tượng trong ảnh (ứng với ground truth tương ứng).
- True Negative (TN): Đây là thông số ít được quan tâm đến. Có thể hiểu là những phần của ảnh không chứa đối tượng và thực tế thì đúng là như vậy.

5.3 Precision

Precision là thang đo độ chính xác của dự đoán, được định nghĩa là tỉ lệ số điểm Positive mà mô hình dự đoán đúng trên tổng số điểm mà mô hình dự đoán là Positive.

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} = \frac{\text{TP}}{\text{all detections}} \quad (1)$$

5.4 Recall

Recall là thang đo độ nhạy của khả năng tìm thấy các dự đoán đúng, được định nghĩa là tỉ lệ số điểm Positive mô hình dự đoán đúng trên tổng số điểm thật sự là Positive (hay tổng số điểm được gán nhãn là Positive ban đầu).

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} = \frac{\text{TP}}{\text{all ground truth}} \quad (2)$$

5.5 Average Precision (AP)

Average Precision (AP) là một độ đo dùng để xấp xỉ phần diện tích phía dưới precision-recall curve. AP được tính bằng tích của precision ở mức k và sự chênh lệch của recall ở hai mức recall thứ k và $k + 1$:

$$\text{AP} = \sum_{k=0}^{k=n-1} [\text{Recall}(k) - \text{Recall}(k+1)] \times \text{Precision}(k) \quad (3)$$

Với n là số lượng mức threshold, recall(n) bằng 0 và precision(n) = 1.

5.6 Mean Average Precision (mAP)

Mean average precision (mAP) là một độ đo dùng để tính AP trung bình trên nhiều lớp khác nhau và nhiều mức threshold khác nhau:

$$\text{mAP} = \frac{1}{n} \sum_{k=1}^{k=n} \text{AP}_k \quad (4)$$

Với n là số lớp, AP_k là AP của lớp thứ k

5.7 Accuracy sequence và character

Ở bài toán Text Recognition, chúng ta sẽ đánh giá dựa trên hai độ đo là độ chính xác theo từng kí tự và độ chính xác cho từng chuỗi

5.8 Đánh giá chung

Công thức đánh giá bài toán: Sử dụng thư viện fuzzywuzzy để so sánh khoảng cách giữa 2 chuỗi (fuzzywuzzy.fuzz.ratio()), với 3 tiêu chí đặt ra:

- Tương đồng 100%
- Tương đồng từ 95% trở lên
- Tương đồng từ 90% trở lên

Đánh giá dựa trên f1 score:

- Những thuộc tính thực tế có mang giá trị, dự đoán ra kết quả đúng => TP (True Positive)
- Những thuộc tính thực tế không có, dự đoán cũng ra không có => TN (True Negative)
- Những thuộc tính thực tế không có nhưng dự đoán ra có => FN (False Negative)
- Những thuộc tính thực tế có mang giá trị nhưng dự đoán ra không có hoặc dự đoán ra kết quả sai => FP (False Positive)

Phân chia đánh giá: Tập dữ liệu 400 ảnh được chụp thực tế được nêu ở trên chưa được dùng qua để training YOLO hay VietOCR, ta chia thành 3 tập con:

- Easy: Vị trí thuộc tính và font chữ có thể chấp nhận được.
- Medium: Vị trí thuộc tính khó nhận dạng hay font chữ khó nhận dạng.
- Hard: Cả vị trí thuộc tính và font chữ khó nhận dạng.

Kết quả:

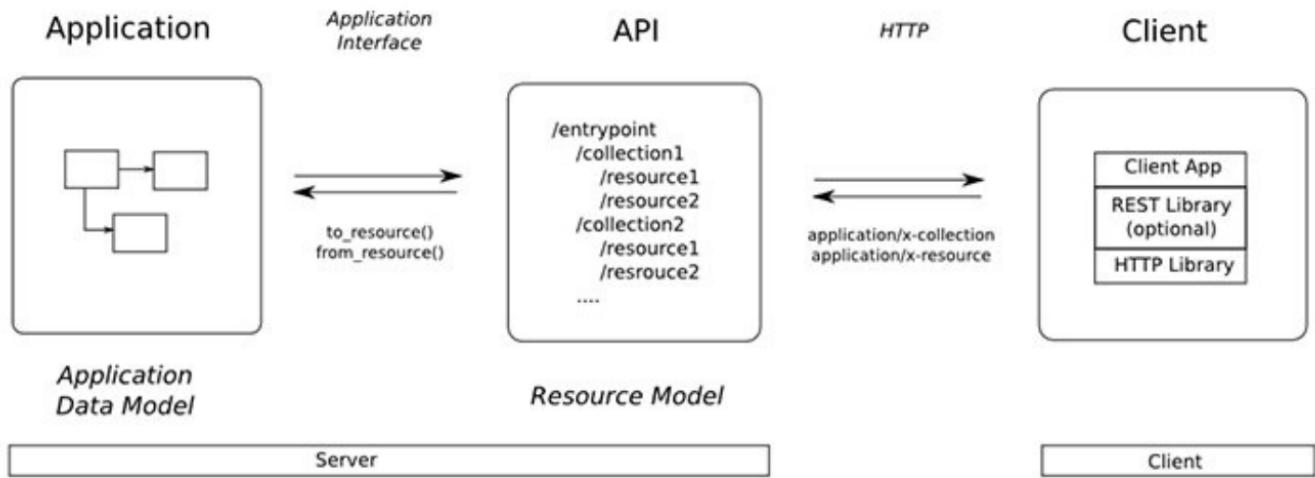
- Easy:
 - 100%: 0.80355
 - >= 95%: 0.86644
 - >= 90%: 0.88694
- Medium:
 - 100%: 0.67382
 - >= 95%: 0.70687
 - >= 90%: 0.73651
- Hard:
 - 100%: 0.57132
 - >= 95%: 0.60857
 - >= 90%: 0.6948

6 Thiết kế hệ thống

Trong đồ án này, nhóm chúng em xây dựng một API bằng Flask web framework cho phía server.

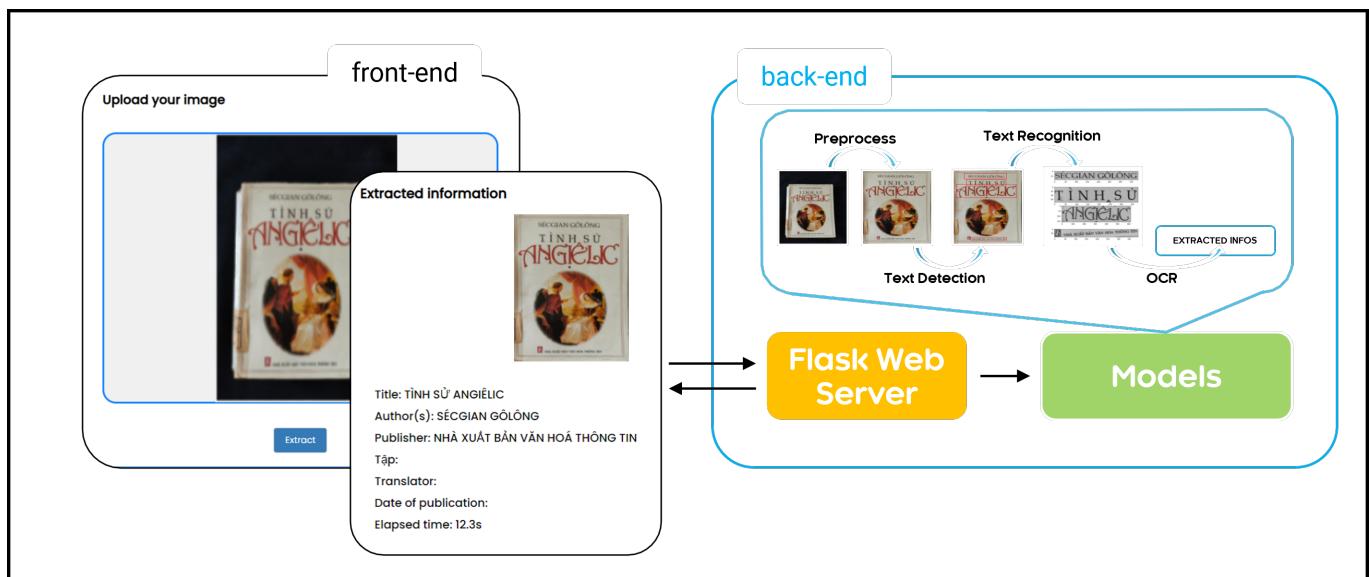
Một số yêu cầu cơ bản:

- Hệ thống có thể upload file từ người dùng(định dạng file ảnh jpg/png/jpeg)
- Sau khi upload file thì khi muốn upload tiếp thì giao diện sẽ cho phép upload tiếp (upload->extract->re upload)



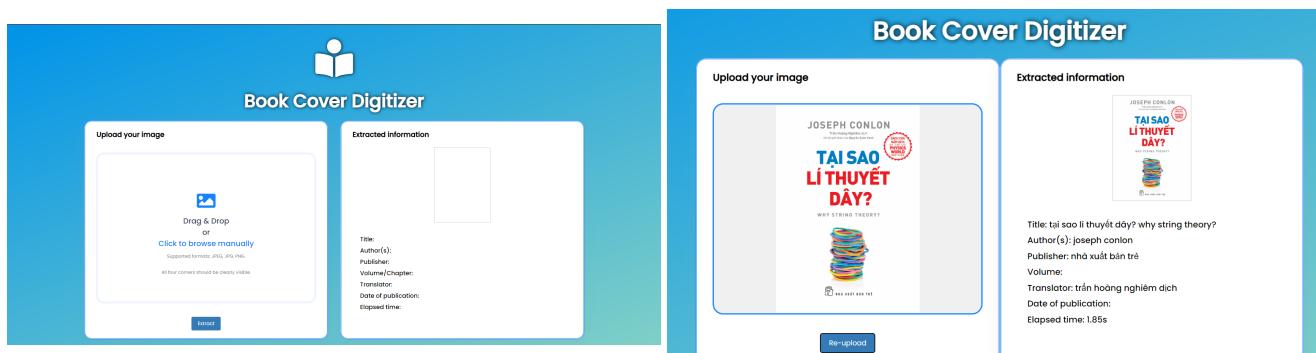
Hình 11: RESTful API

6.1 Kiến trúc hệ thống

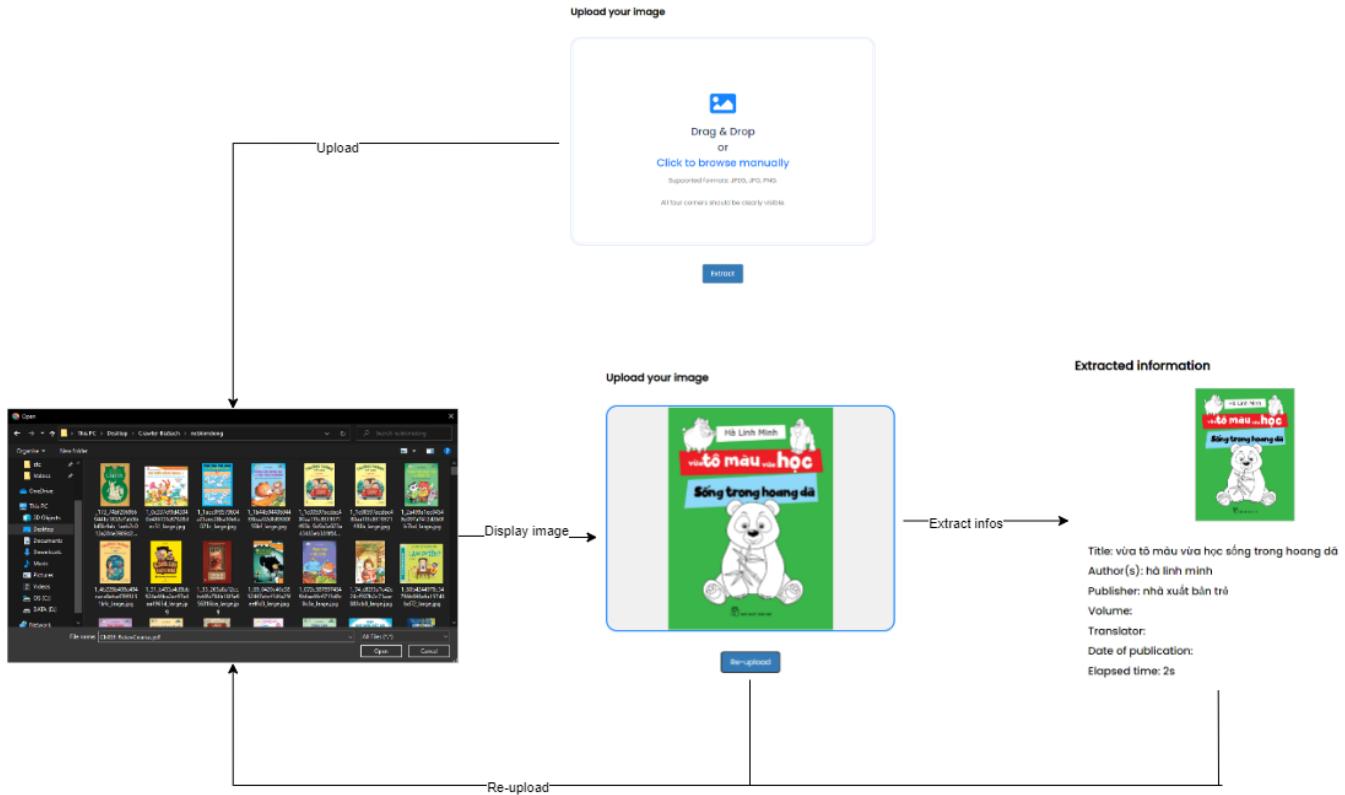


Hình 12: Tổng quan kiến trúc hệ thống

6.1.1 Giao diện hệ thống

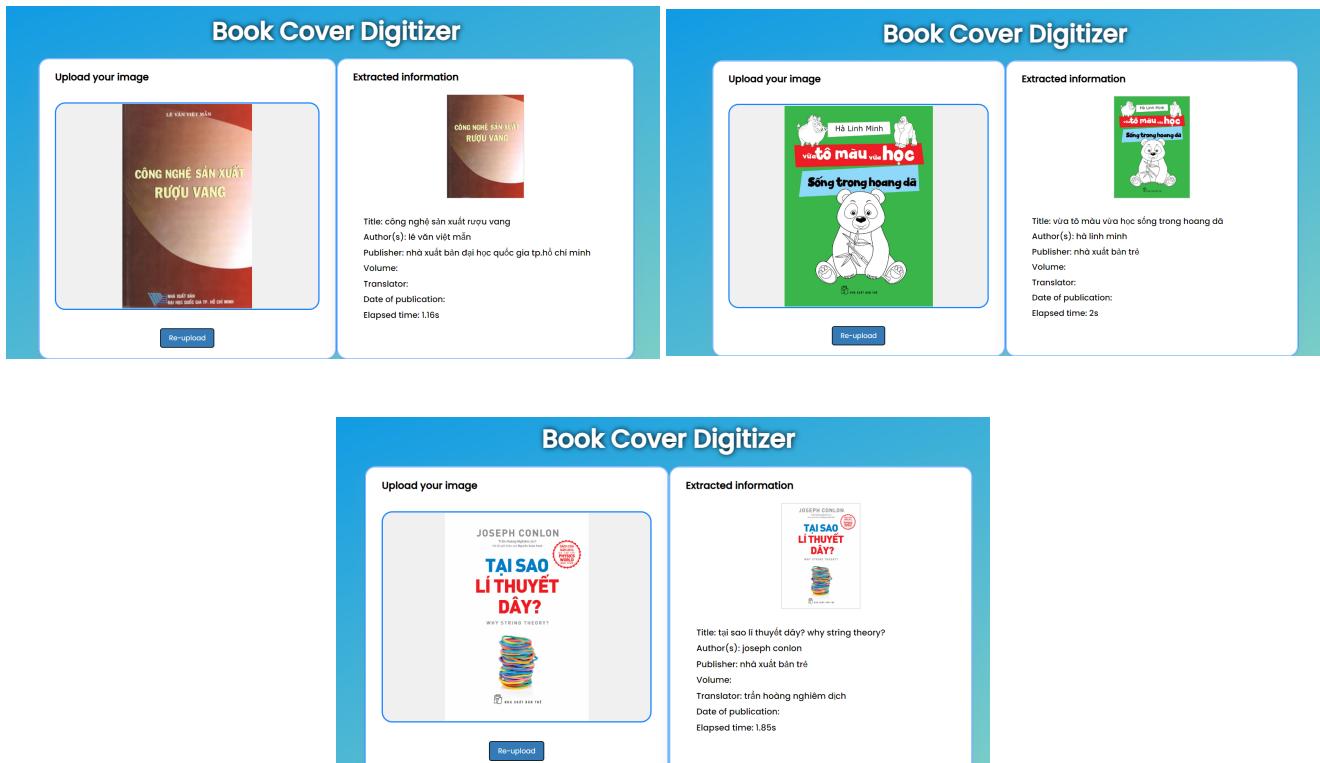


6.1.2 Screenflow



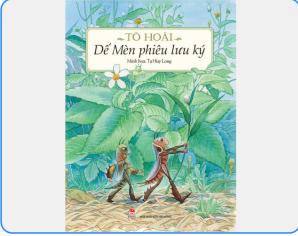
Hình 13: Screenflow

6.2 Demo



Hình 14: Kết quả predict đúng

Upload your image



Re-upload

Extracted information



Title: tô hoai dế mèn phiêu lưu ký
 Author(s):
 Publisher: nhà xuất bản kim đồng
 Volume:
 Translator:
 Date of publication:
 Elapsed time: 1.45s

Upload your image



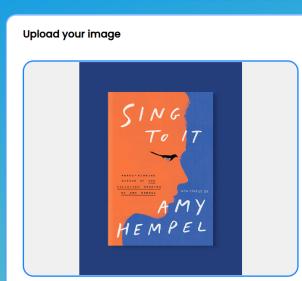
Re-upload

Extracted information



Title: lâm
 Author(s): vũ trọng phụng
 Publisher: nhà xuất bản
 Volume:
 Translator:
 Date of publication:
 Elapsed time: 1.16s

Upload your image



Re-upload

Extracted information



Title: cinc g to it amy hempel
 Author(s): new story by
 Publisher:
 Volume:
 Translator: hempel
 Date of publication:
 Elapsed time: 1.75s

Hình 15: Kết quả predict chưa tốt

TÀI LIỆU THAM KHẢO

- [1] Medium. 2022. Can a neural network train other networks?
<https://towardsdatascience.com/can-a-neural-network-train-other-networks-cf371be516c6>.
- [2] Subramani, N., Matton, A., Greaves, M. and Lam, A., 2022. A Survey of Deep Learning Approaches for OCR and Document Understanding. arXiv.org.
<https://arxiv.org/abs/2011.13534>
- [3] Scene text detection and recognition: a survey.
<https://link.springer.com/article/10.1007/s11042-022-12693-7>
- [4] vietocr. Github
<https://github.com/pbcquoc/vietocr>
- [5] vnese-id-extractor. Github
<https://github.com/ntvuonggg/vnese-id-extractor>
- [6] MC-OCR Challenge 2021: An end-to-end recognition framework for Vietnamese Receipts
<https://ieeexplore.ieee.org/document/9642121>
- [7] You Only Look Once: Unified, Real-Time Object Detection
<https://arxiv.org/abs/1506.02640>
- [8] Character Region Awareness for Text Detection
<https://arxiv.org/abs/1904.01941>