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

ĐỒ ÁN MÔN HỌC
MÁY HỌC

Chủ đề: SỐ HÓA TỦ SÁCH

Giảng viên: TS. Lê Đình Duy

Ths. Phạm Nguyễn Trường An

Lớp: CS114.M11

Sinh viên thực hiện:		
STT	Họ tên	MSSV
1	Nguyễn Ngọc An	19521182
2	Trần Duy quang	19522102
3	Nguyễn Khắc Thái	20511888

TP. HỒ CHÍ MINH – 1/2022

Phụ Lục

1. GIẢI TRÌNH CHỈNH SỬA VẤN ĐÁP

Các điểm đã được cập nhật và làm mới sau vấn đáp:

1.1. Dữ liệu

- Bổ sung 2 file gồm 1 file csv và 1 file excel chứa thông tin trên bìa sách của khoảng 300 ảnh bìa sách: ***III. Nội dung - 1. Mô tả dữ liệu***

1.2. Phương pháp lọc theo ngưỡng

- Bổ sung phần trình bày các bước của phương pháp lọc theo ngưỡng: ***III. Nội dung - 5. Phương pháp lọc theo ngưỡng***

1.3. Các phương pháp đánh giá

- Đánh giá các mô hình VietOCR

+ Cập nhật khái niệm các điểm đánh giá: ***III. Nội dung - 1. Sơ lược về VietOCR - c. Đánh giá mô hình***

- Đánh giá phương pháp lọc theo ngưỡng:

+ Bổ sung cách đánh giá và khái niệm các điểm đánh: ***III. Nội dung - 5. Phương pháp lọc theo ngưỡng - b. Đánh giá***

1.4. Thực nghiệm

- Bổ sung các nhận xét về kết quả thực nghiệm sau khi train-test 2 mô hình của VietOCR: ***IV. Thực nghiệm - 1. Các mô hình trong VietOCR - b. TransformOCR và AttentionOCR (sau khi train)***
- Bổ sung các nhận xét về kết quả của phương pháp lọc theo ngưỡng: ***IV. Thực nghiệm - 2. Phương pháp lọc theo ngưỡng***

2. GIỚI THIỆU

2.1. Giới thiệu bài toán

- Bài toán: Số hóa tủ sách
- Input: Một tấm ảnh chụp đầy đủ trang bìa của một cuốn sách. Bìa sách phải ở trung tâm ảnh, chiếm phần lớn diện tích ảnh. Bìa sách nằm theo chiều dọc, được căn chỉnh, không nghiêng, không nằm ngang trong ảnh
- Output: Thông tin trên cuốn sách đó (Gồm: Tên tác giả, tên sách, nhà xuất bản)

2.2. Ngữ cảnh ứng dụng

- Sử dụng để số hóa tủ sách ở nhà như số hóa tủ sách thành một danh sách để lưu thông tin các cuốn sách (gồm Tên tác giả, tên sách, nhà xuất bản)

2.3. Mô hình đánh giá và dự kiến đạt được

- Mô hình được huấn luyện là Text recognition, được đánh giá trên: Accuracy full sequence và Accuracy per character
- Dự kiến đạt được: Mô hình sẽ cho 2 điểm (dựa trên cách đánh giá mô hình trên) đạt được từ 80-90%

3. NỘI DUNG

3.1. Mô tả dữ liệu

- Bộ dữ liệu gốc gồm: 600 ảnh bìa sách được chụp lại và 400 ảnh bìa lấy từ Internet
- Bộ dữ liệu cho train các mô hình củ VietOCR gồm: 7000 dòng text đã gán nhãn được cắt ra từ các bìa sách đã thu thập. Trong đó:

+ Training set gồm: 5600 dòng
+ Test set gồm: 1400 dòng

- Thao tác xử lý dữ liệu:

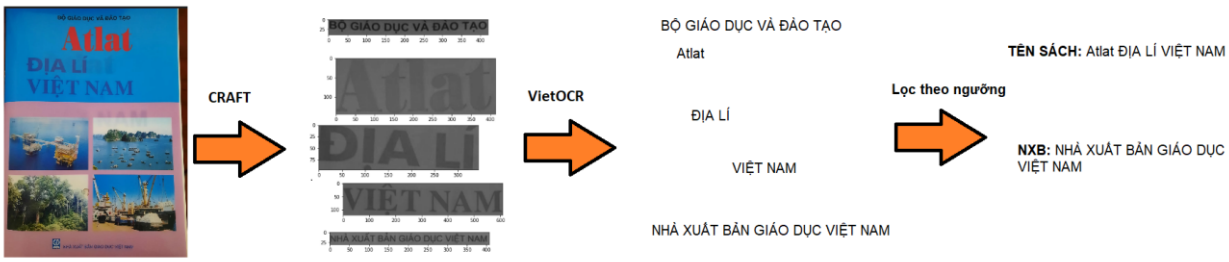
+ Sử dụng pre-train model của CRAFT để lấy các ảnh chứa văn bản trên mỗi ảnh bìa sách. Chuyển các ảnh về dạng grayscale
+ Gán nhãn cho mỗi ảnh đó để tạo thành dataset phù hợp cho việc training model Vietocr

- Bộ dữ liệu cho test cả mô hình về lấy các thông tin trên sách (Tên tác giả, tên sách, nhà xuất bản) gồm file excel (nhóm tự gán nhãn) chứa thông tin của gần 300 ảnh bìa sách

3.2. Tổng quát mô hình

- Đầu vào của mô hình là một tấm ảnh chụp hình bìa sách. Bìa sách phải ở trung tâm ảnh, chiếm phần lớn diện tích và nằm theo chiều dọc của tấm

  Đầu ra của mô hình gồm 3 loại thông tin trên bìa sách: Tên tác giả, tên sách và Nhà xuất bản



Pipeline số hóa tủ sách.

- Cấu trúc mô hình gồm:

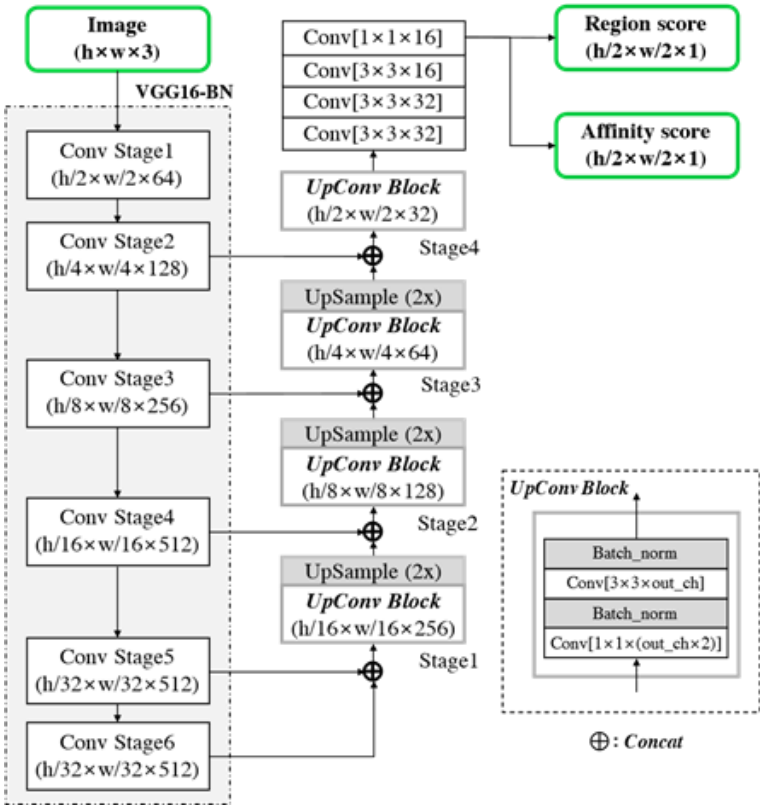
+ Pre-train model CRAFT: Cắt ảnh đầu vào là ảnh bìa sách thành tập các ảnh chứa các vùng văn bản được phát hiện. Sau đó mỗi ảnh sẽ chuyển thành ảnh xám
+ Mô hình trong VietOCR: Với mỗi ảnh trong tập ảnh cắt trên, đưa qua mô hình Text recognition trong VietOCR để dự đoán văn bản trong mỗi ảnh
+ Phương pháp lọc theo ngưỡng: Với mỗi ảnh trong tập ảnh cắt trên, lấy kích thước theo chiều cao của mỗi ảnh (số dòng pixel của mỗi ảnh) và văn bản được dự đoán trong ảnh đó. Khi đó, các ảnh cắt có vị trí nằm phía trên cùng trong ảnh bìa sách và kích thước nhỏ hơn ngưỡng sẽ được cho là chứa Tên tác giả; phía dưới cùng nhỏ hơn ngưỡng sẽ được cho là chứa Nhà xuất bản. Còn các ảnh còn lại được

cho là chứa Tên sách. Sau khi đã phân loại ảnh cắt chứa thông tin gì (tên tác giả, tên sách hoặc nhà xuất bản) thì văn bản dự đoán trên ảnh đó sẽ ứng với thông tin đó.

3.3. Sơ lược về CRAFT

- CRAFT (hay *Character Region Awareness For Text detection*) là một thuật toán tìm dò chữ trong văn bản được đề xuất năm 2019, bằng cách phát hiện các vùng kí tự và các liên kết các vùng kí tự đó với nhau.
- CRAFT sử dụng kiến trúc mạng hoàn toàn phức tạp dựa trên VGG-16 với việc chuẩn hóa hàng loạt được áp dụng làm xương sống. Đầu ra cuối cùng có 2 kênh biểu diễn 2 score map: Region score map (gồm các Gaussian kernel tại mỗi ký tự) và Affine score map (gồm các Gaussian kernel tại mỗi khoảng cách giữa 2 ký tự kề nhau). Region score map được sử dụng để bản địa hóa các ký tự riêng lẻ trong hình ảnh. Trong khi đó, Affine score map được sử dụng để nhóm mỗi ký tự thành một từ hoặc đoạn văn bản duy nhất.

- Hình trên là một visualization của kết quả craft trên các văn bản hình dạng khác nhau. Cho thấy tính linh hoạt cao của phương pháp được đề xuất trên các trường hợp phức tạp, chẳng hạn như các văn bản dài, cong hoặc có hình dạng tùy ý.
- Kiến trúc mạng được minh họa sơ đồ sau:



- ***Huấn luyện mô hình:***

- Quy trình đào tạo bao gồm hai bước: đầu tiên chúng tôi sử dụng bộ dữ liệu Synth Text để đào tạo mạng cho các lần lặp lại 50k, sau đó mỗi bộ dữ liệu chuẩn được thông qua để tinh chỉnh mô hình sử dụng trình tối ưu hóa ADAM trong tất cả các quy trình đào tạo
- Model đã đc train trên bộ dataset có chữ tiếng Việt nhưng ít. Nên detect chữ tiếng Việt có dấu không được tốt

3.4. Sơ lược về VietOCR

VietOCR bao gồm 2 mô hình nhận diễn chữ viết tiếng Việt là Attention OCR và Transformer OCR, được phát triển bởi Phạm Bá Cường Quốc

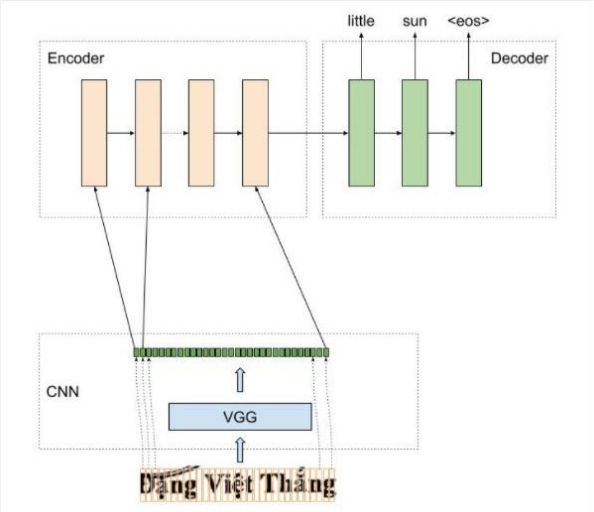
3.4.1. Attention OCR

- Attention OCR là mô hình sử dụng kiến trúc attention seq2seq - một kiến trúc khá nổi tiếng được sử dụng trong các bài toán NLP và cả OCR
- Attention OCR là sự kết hợp giữa mô hình CNN và Attention seq2seq. Trong đó:

+ Mô hình CNN là một mô hình Neural network gồm nhiều layer, mỗi layer gồm các convolutional kernel. CNN được sử dụng để rút trích các đặc trưng trên ảnh. Ảnh đầu vào sau khi qua các convolutional layer đã được huấn luyện từ trước, mô hình sẽ cho ra Feature map (một ma trận 2 chiều). Ở đây, ảnh qua mô hình

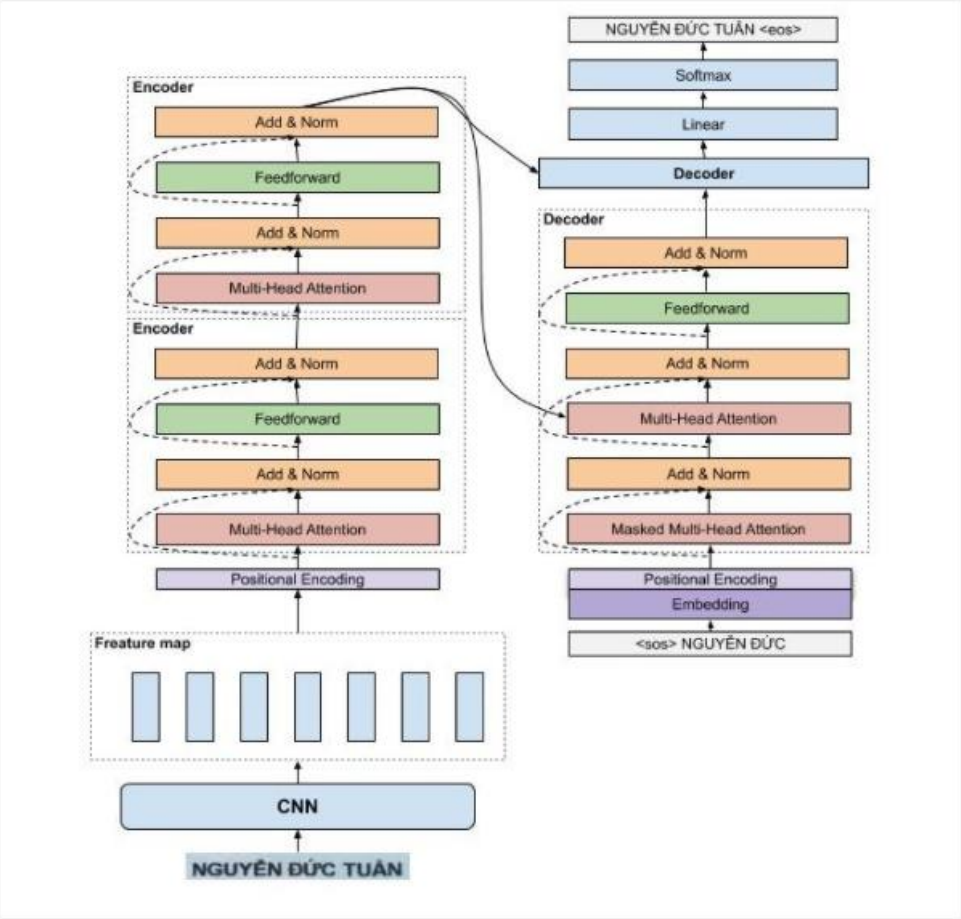
CNN sẽ cho ra một vector đặc trưng, và vector này lại là đầu vào của mô hình Attention seq2seq tiếp theo

+ Attention seq2seq là một mô hình seq2seq (cụ thể là mô hình LSTM) có kết hợp kiến trúc Attention. Mô hình này thường được sử dụng trong các bài toán nhận dạng chữ tiếng Việt vì khả năng "ghi nhớ" của nó, nguyên nhân vì mỗi từ trong một câu đều được mã hóa và lưu trong 1 vector (encoder), và vector đó luôn được cập nhật lần lượt từng từ trong câu. Chính vì khả năng "ghi nhớ" đó, đầu ra của mô hình này (decoder) gồm câu có các chữ cái mà ngữ nghĩa của nó sẽ đúng hơn



3.4.2. Transformer OCR

- Transformer OCR sử dụng kiến trúc transformer - đây là kiến trúc đã đạt được nhiều tiến bộ vượt bậc cho cộng đồng NLP
- Tương tự như Attention OCR, sau khi ảnh qua mô hình CNN sẽ được rút trích đặc trưng, đầu ra là một vector đặc trưng. Vector này sẽ qua một kiến trúc Encoder-Decoder lớn như bên dưới



- Điểm khác biệt lớn nhất giữa Transformer OCR và Attention OCR ở chỗ:

+ Transformer OCR gồm nhiều tầng Encoder. Và vector đặc trưng được đưa vào mô hình này phải thông qua Positional Encoding để nắm được vị trí của các đặc trưng trong vector, và các đặc trưng này sẽ được xử lý song song với nhau

Còn với Attention OCR, chỉ có một hệ thống Encoder-Decoder. Mỗi đặc trưng trong vector đặc trưng sẽ được đưa lần lượt vào Encoder để encode ra vector, từ đó mới decode vector đó để ra câu văn bản tiếng Việt

3.4.3. Đánh giá theo mô hình

- Ở đây, mô hình sử dụng 2 loại Accuracy score để đánh giá là Accuracy full sequence và Accuracy per character

+ Accuracy full sequence là được tính bằng số từ đúng trong chuỗi dự đoán chia cho tổng số từ trong chuỗi đúng đã biết trước tính từ trái sang phải.
Ví dụ:
- Ta có chuỗi dự đoán là:"Nguyễn Văn A" và chuỗi đúng của nó là "Nguyễn Văn C" thì điểm Accuracy full sequence=⅔

- Ta có chuỗi dự đoán là: ”Nguyễn Văn C Lý” và chuỗi đúng của nó là “Nguyễn Văn C” thì điểm Accuracy full sequence=1

+ Accuracy per character là được tính bằng trung bình điểm dự đoán của cái ký tự trong từ của chuỗi tính từ trái sang phải

Ví dụ:

- Ta có chuỗi dự đoán là: “Nguyễn Văn**x** S” và chuỗi đúng là “Nguyễn Văn S” thì điểm dự đoán của cái từ lần lượt là: 1,1,1 nên điểm Accuracy per character=(1+1+1)/3=1

- Ta có chuỗi dự đoán là: “Nguyễn Văx S” và chuỗi đúng là “Nguyễn Văn S” thì điểm dự đoán của cái từ lần lượt là: 1,2/3,1 nên điểm Accuracy per character=(1+2/3+1)/3=8/9

- Ta có chuỗi dự đoán là: “Nguyễn Văn Sx Thị ” và chuỗi đúng là “Nguyễn Văn S” thì điểm dự đoán của cái từ lần lượt là: 1,1,1 nên điểm Accuracy per character=(1+1+1)/3=1

3.5. Phương pháp lọc theo ngưỡng

3.5.1. Phương pháp

- Với mỗi ảnh bìa sách, sau khi sử dụng mô hình CRAFT cắt thành các ảnh chứa vùng văn bản được phát hiện. Từ mỗi ảnh cắt, lấy kích thước theo chiều cao của ảnh. Sau đó lưu các kích thước này thành một danh sách (ở đây ký hiệu là ***heightArr***). Thứ tự các kích thước ảnh cắt trong ***heightArr*** theo thứ tự vị trí các ảnh cắt ra nằm từ trên xuống dưới trên ảnh bìa sách

- Chuyển mỗi giá trị trong ***heightArr*** thành các giá trị nằm trong miền [0, 1] bằng cách sau:

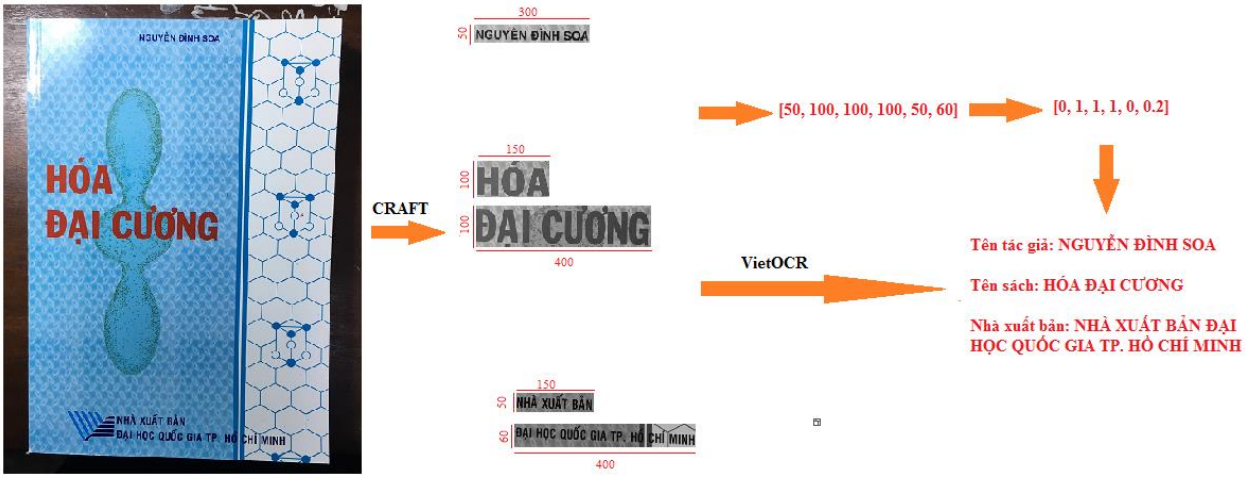
min, max = min(heightArr), max(heightArr)

for i = 0 : length(heightArr):

heightArr[i] = (heightArr[i] - min) / (max - min)

[50, 100, 100, 100, 50, 60] → [0, 1, 1, 1, 0, 0.2]

- Lọc theo ngưỡng: Chọn giá trị ngưỡng bằng 0.3. Duyệt tuyến tính từ phần tử cuối cùng đến phần tử đầu của ***heightArr***.
+ Nếu: heightArr[i] ≥ 0.3 hoặc abs(heightArr[i] - heightArr[i-1]) ≥ 0.1 thì các văn bản trong ảnh cắt ở các vị trí theo thứ thứ đó trên ảnh bìa sẽ là Nhà xuất bản
+ Tiếp tục lọc từ heightArr[i] vừa xét ở trên. Nếu: heightArr[i] < 0.3 thì các văn bản trong ảnh cắt ở các vị trí tiếp theo đó trên ảnh bìa sẽ là Tên sách
+ Các văn bản ở các ảnh cắt ở các vị trí còn lại trên ảnh bìa sẽ là Tên tác giả



3.5.2. Đánh giá

- Để đánh giá phương pháp này, nhóm đã tự gán phân các thông tin trên mỗi bìa sách thành Tên tác giả, tên sách, nhà xuất bản. Thông tin của gần 300 ảnh sách sẽ được lưu dưới dạng file excel. Với mỗi ảnh bìa sách, ta sẽ so sánh mức độ tương đồng giữa từng cặp chuỗi thông tin dự đoán (lưu dưới dạng file csv) và thông tin thực tế (lưu dưới dạng file excel) của từng loại thông tin. Đánh giá chung độ chính xác của mô hình bằng *f1-score*. Với:
- *True positive*: Số lần 2 chuỗi thông tin dự đoán và thông tin thực tế (thông tin gồm: Tên tác giả, Tên sách và Nhà xuất bản) của từng ảnh bìa sách được cho là giống nhau (dựa trên ***Ratio-score*** và giá trị ngưỡng ***T*** sẽ được giải thích bên dưới)
- *True negative*: Số lần chuỗi thông tin thực tế và chuỗi thông tin dự đoán đều là Null (Không có ký tự nào)
+ *False positive*: Số lần chuỗi thông tin thực tế không phải Null nhưng chuỗi dự đoán là Null hoặc Số lần 2 chuỗi thông tin dự đoán và thông tin thực tế được cho là không giống nhau*False positive*: Số lần chuỗi thông tin thực tế không phải Null nhưng chuỗi dự đoán là Null hoặc Số lần 2 chuỗi thông tin dự đoán và thông tin thực tế được cho là không giống nhau

+ *False negative*: Số lần chuỗi thông tin thực tế là Null và chuỗi thông tin dự đoán không phải Null
+ $PrecisionScore = \frac{True\ positive}{True\ positive + False\ positive}$
+ $RecallScore = \frac{True\ positive}{True\ positive + False\ negative}$
+ $f1\text{-}score = \frac{2 * PrecisionScore * RecallScore}{PrecisionScore + RecallScore}$

- Mã giả:

```
for BookInfo in ['Tên tác giả', 'Tên sách', 'Nhà xuất bản']:
    for index in range(len(ListBook)):
        PredictedInfo_IsNull = PredictedInfo[index][BookInfo].IsNull()
        ActualInfo_IsNull = ActualInfo[index][BookInfo].IsNull()
        if (ActualInfo_IsNull == True & PredictedInfo_IsNull == True):
            TrueNegative += 1
        elif (ActualInfo_IsNull == True & PredictedInfo_IsNull != True):
            FalseNegative += 1
        elif ((ActualInfo_IsNull != True & PredictedInfo_IsNull == True):
            FalsePossitive += 1
        else:
            if (RatioScore(PredictedInfo[index], ActualInfo[index]) ≥
Threshold):
                TruePossitive += 1
            else:
                FalsePossitive += 1

PrecisionScore = TruePossitive / (TruePossitive + FalsePossitive)
RecallScore = TruePossitive / (TruePossitive + FalseNegative)
f1Score = (2*Precision*Recall) / (Precision + Recall)

# BookInfo: Loại thông tin (Tên tác giả, Tên sách hay Nhà xuất bản)
# len(ListBook): Số lượng ảnh bìa sách
```

PredictedInfo[***index***][***BookInfo***]: Loại thông tin BookInfo tại ảnh bìa sách thứ

index

- Đánh giá 2 chuỗi thông tin A và B giống nhau dựa trên ***Ratio-score*** và giá trị ngưỡng **T**:

- Bước 1: Sử dụng ***Ratio-score*** của *Fuzzy wuzzy package* để đánh giá độ chính xác giữa từng thông tin dự đoán với thông tin thực tế (thông tin gồm: Tên tác giả, Tên sách và Nhà xuất bản) của từng ảnh bìa sách. Trong đó, ***Ratio-score*** được đánh giá dựa trên *Levenshtein distance*:

+ *Levenshtein distance* là số phép biến đổi ít nhất từ 1 chuỗi sang 1 chuỗi mới. Các phép biến đổi gồm: Thêm 1 ký tự, xóa 1 ký tự và thay thế 1 ký tự.

Ví dụ: "cat" → "hat" (chỉ cần 1 phép biến đổi là thay thế ký tự 'c' thành ký tự 'h')
"cat" → "at" (chỉ cần 1 phép biến đổi là xóa ký tự 'c')
"cat" → "hate" (cần 2 phép biến đổi gồm thay thế ký tự 'c' thành ký tự 'h' và thêm ký tự 'e' vào cuối chuỗi "cat")

+ ***Ratio-score*** được tính bằng cách: Sau khi đưa ra Levenshtein distance giữa 2 chuỗi A và B gồm $x_{Insert}, x_{Delete}, x_{Replace}$ lần lượt là số thao tác thêm 1 ký tự, xóa 1 ký tự và thay thế 1 ký tự thành 1 ký tự khác. Tac có: $ratio = 1 - \frac{2*x_{Replace} + 1*(x_{Insert} + x_{Delete})}{len(A) + len(B)}$ và $0 \leq ratio \leq 1$

Ví dụ: A = "abcd" và B = "abedf". Để biến đổi A thành B, sử dụng Levenshtein distance, ta chỉ cần 1 phép biến đổi ký tự 'c' thành ký tự 'e' và 1 phép thêm ký tự 'f'.

Khi đó: $x_{replace} = 1$; $x_{Insert} = 1$; $x_{Delete} = 0$ và len(A) = 4, len(B) = 5

$$ratio = 1 - \frac{2*x_{Replace} + 1*(x_{Insert} + x_{Delete})}{len(A) + len(B)} = 1 - \frac{2*1 + 1*(1+0)}{4+5} = 0.(6)$$

- Bước 2: Chọn giá trị ngưỡng **T**. Nếu ***ratio-score*** giữa 2 chuỗi so sánh A và B lớn hơn ngưỡng **T**, trả về *True* (hay 2 chuỗi này tương đối giống nhau); nhỏ hơn ngưỡng **T**, trả về *False* (hay 2 chuỗi này không giống nhau)

4. THỰC NGHIỆM

4.1. Các mô hình trong VietOCR cho Text recognition

4.1.1. Các parameters

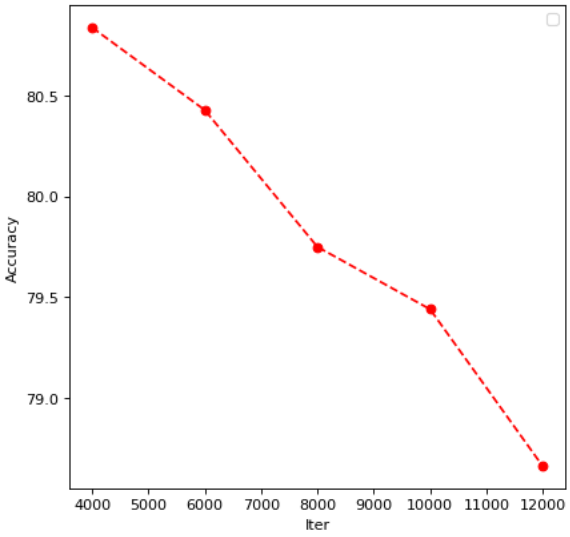
- Ở đây, nhóm em huấn luyện 2 mô hình nhận diện văn bản tiếng Việt gồm: Transformer OCR và Attention OCR. Cả 2 mô hình này gồm có 8 parameters được truyền vào:

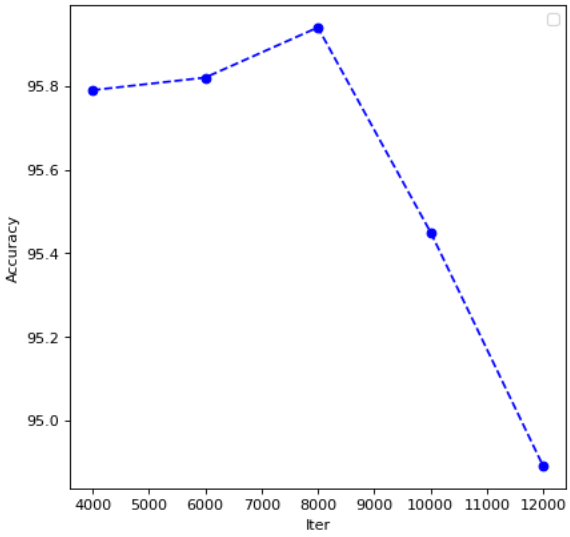
+ data_root: Thư mục lưu tất cả các ảnh trong tập dataset
+ train_annotation: Đường dẫn đến file train
+ valid_annotation: Đường dẫn đến file test
+ print_every: Sau n iters, train loss được in ra
+ valid_every: Sau n iters, validation loss được in ra
+ iters: Số vòng lặp để train model

+ export: Đường dẫn thư mục lưu trọng số của mô hình
+ metrics: Số lượng examples lưu trong tập validation để mô hình tính validation loss trong khi train

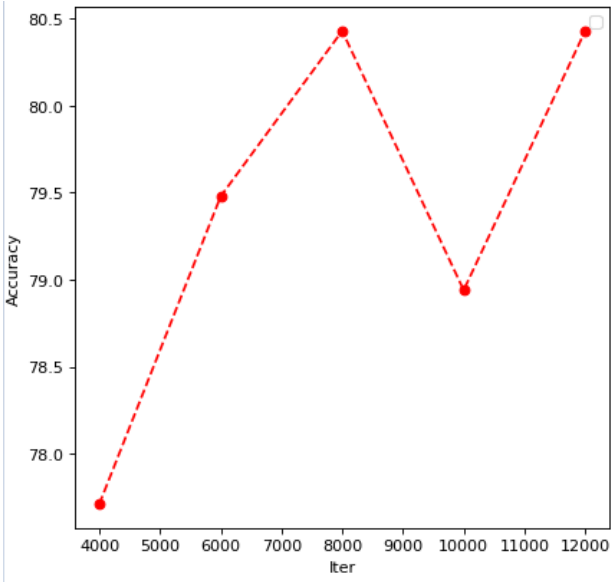
4.1.2. Transformer OCR và Attention OCR (sau khi training)

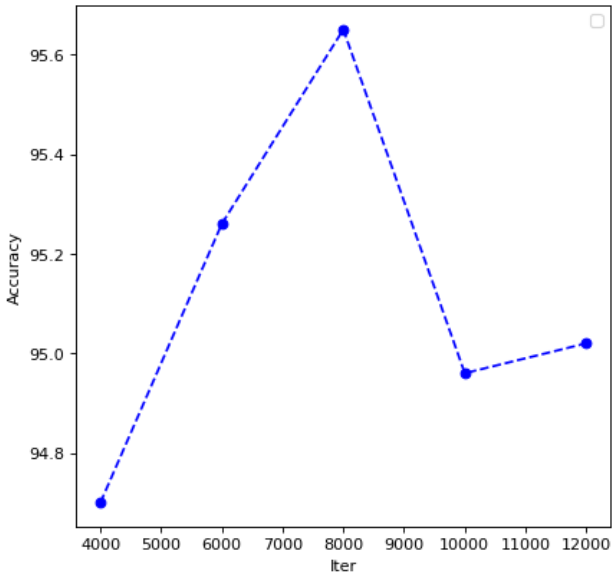
- Kết quả huấn luyện mô hình Transformer OCR trên nhiều vòng lặp khác nhau:





- Kết quả huấn luyện mô hình Transformer OCR trên nhiều vòng lặp khác nhau:





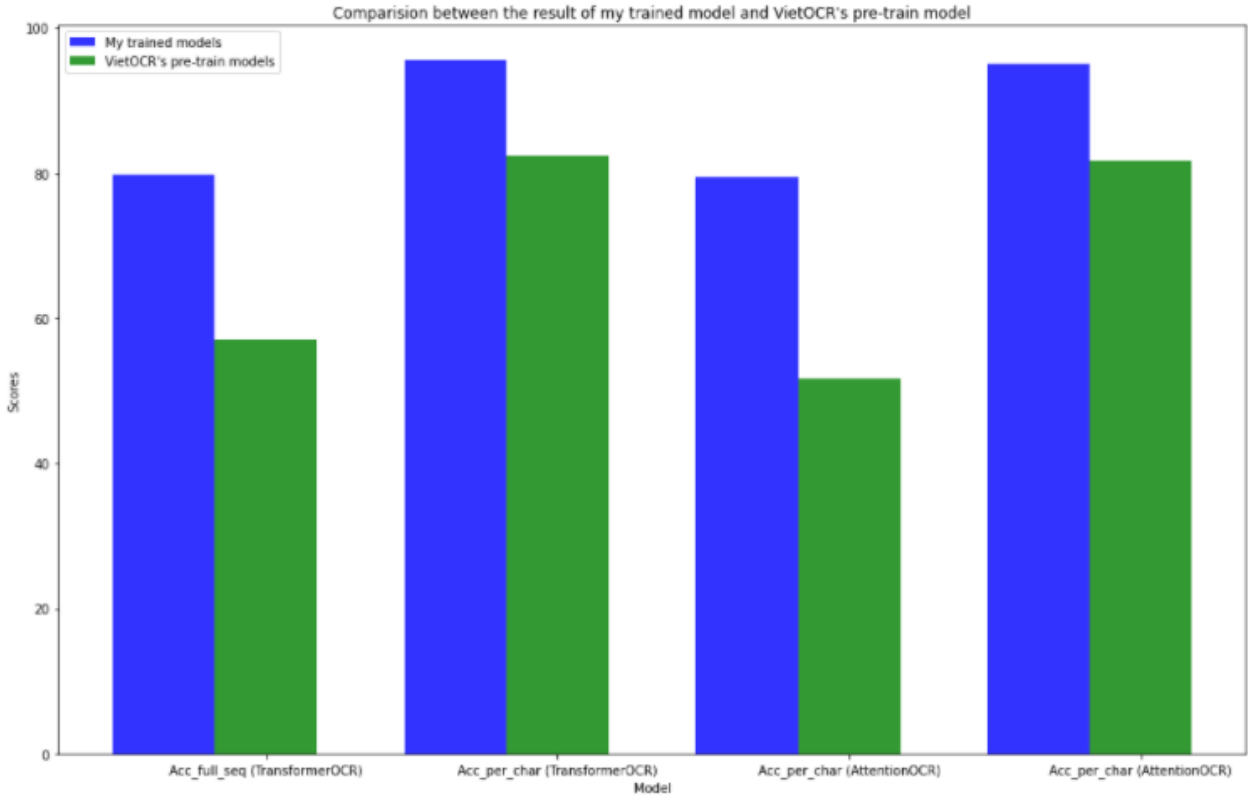
- ***Nhận xét:***

Khi sử dụng TransformerOCR, số vòng lặp trong quá trình training tăng thì Accuracy full sequence và Accuracy per character không tăng đáng kể, thậm chí còn giảm. Mặc dù mô hình này đã được train với bộ dữ liệu văn bản tiếng Việt viết tay và được tác giả đánh giá cao. Tuy nhiên, ta thấy có vẻ mô hình này lại không phù hợp lắm với bộ dữ liệu hiện tại

Ngược lại với TransformerOCR, AttentionOCR lại cho kết quả tốt hơn. Đúng như nhận xét của tác giả về 2 mô hình này (tham khảo tại đây), mặc dù ban đầu mô hình TransformerOCR cho kết quả tốt hơn, nhưng độ chính không có sự cải thiện đáng kể so với AttentionOCR. Ngoài ra, theo em, tốc độ tăng accuracy của AttentionOCR tốt hơn TransformerOCR có thể do với kiến trúc Attention kết hợp với mô hình seq2seq (LSTM) và quá trình encode từng đặc trưng của vector đặc trưng, điều này khiến mô hình AttentionOCR hoạt động tốt hơn với bộ dataset của nhóm em, cụ thể là với mỗi ảnh cắt, văn bản trong ảnh được crop có nhiều nhiễu hơn và ngắn hơn so với bộ dataset tác giả đã huấn luyện

- **So sánh kết quả giữa các mô hình Text recognition của nhóm đã train và các mô hình có sẵn của VietOCR:**

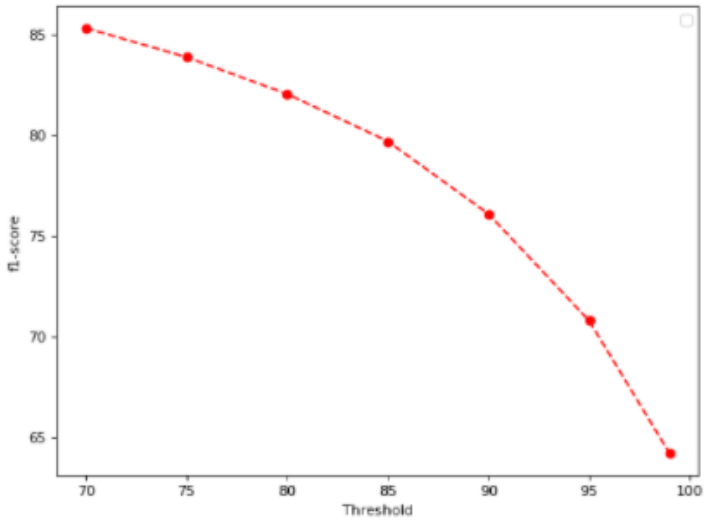
- Gồm 4 mô hình: 2 mô hình nhóm đã train và 2 mô hình cùng loại có sẵn của VIetOCR (Transformer OCR và Attention OCR)
- Khi đánh giá trên cùng một bộ Test set, thu được kết quả như sau:



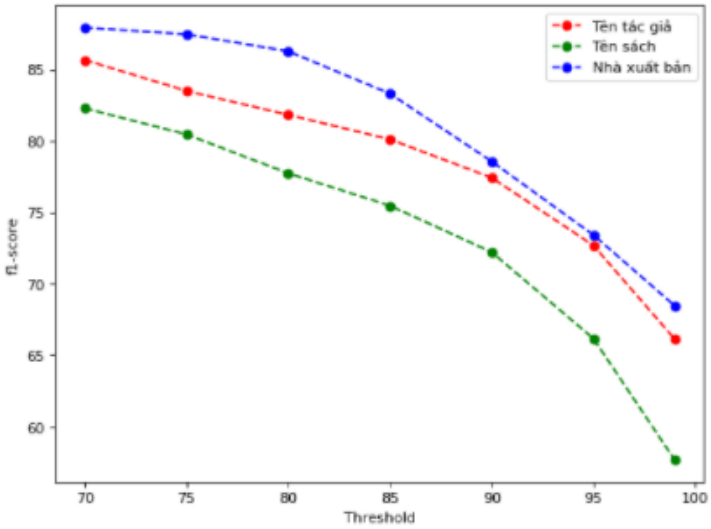
- ***Nhận xét:*** Có sự chênh lệch về cả Accuracy full sequence và Accuracy per character giữa các mô hình nhóm đã train và các mô hình có sẵn của VietOCR. Theo em, nguyên nhân do bộ dữ liệu mà các mô hình trong VietOCR đã được train trước tuy nhiều nhưng lại không có nhiễu (Ảnh có nền trắng, rất ít nhiễu; màu chữ đậm, không bị mờ, dễ phân biệt được chữ với nền ảnh), còn bộ dữ liệu chụp từ ảnh bìa sách thì lại rất nhiều nhiễu (bìa sách có rất nhiều họa tiết trang trí xung quanh nên đó là thông tin gây nhiễu; hay ảnh chụp bìa sách không được tốt như thiếu sáng, sáng chói, mờ,...). Do đó, mặc dù các mô hình có sẵn của VietOCR có Accuracy full sequence và Accuracy per character cao trên bộ Test set của VietOCR nhưng trên bộ Test set là ảnh bìa sách thì lại không hiệu quả
- ***Kết luận chung:*** TransformerOCR có thể hoạt động tốt hơn AttentionOCR với các bộ dataset ít nhiễu, văn bản rõ ràng. Do đó Attention OCR phù hợp với bộ dataset của nhóm em hơn

4.2. Phương pháp lọc theo ngưỡng

- Kết quả đánh giá độ chính xác của phương pháp lọc theo ngưỡng này với ***f1-score*** dựa trên ***Ratio-score*** với các mốc Threshold **T** khác nhau



- Kết quả đánh giá trên từng thông tin Tên tác giả, Tên sách và NXB:



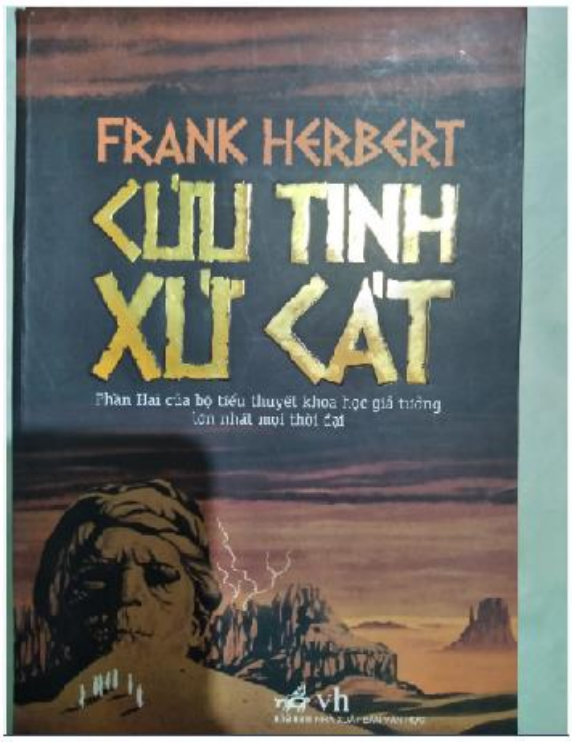
- ***Nhận xét:*** Dĩ nhiên việc càng tăng giá trị Threshold **T** thì số chuỗi thông tin dự đoán được cho là giống với chuỗi thông tin thực tế càng giảm nên các trường hợp True positive càng giảm và tổng các trường hợp False positive và False negative giảm, do đó ***f1-score*** càng giảm (có thể chứng minh từ công thức tính ***f1-score***). Tuy nhiên ở biểu đồ dưới ta thấy:

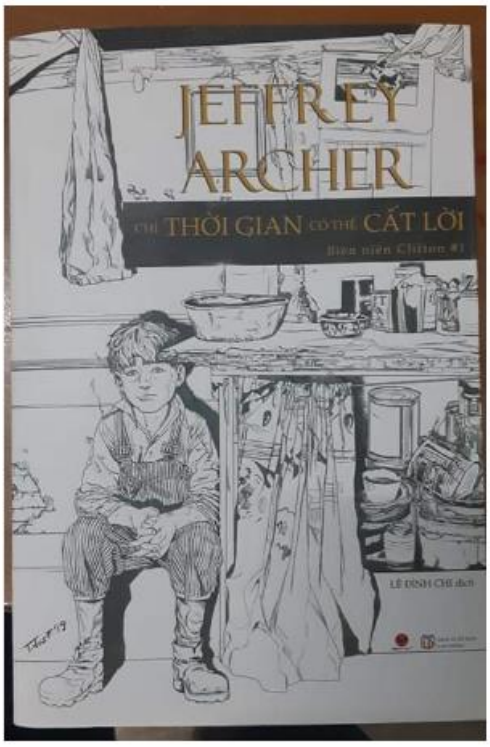
+ Thông tin NXB ban đầu có giá trị ***f1-score*** khá cao ở các giá trị **T** thấp nhưng tốc độ giảm ***f1-score*** lại rất nhanh. Nguyên nhân vì trong tập Test set, có một số ảnh được chụp là các sách giáo khoa có bao bìa, trên bao bìa đó có chữ và mô hình *CRAFT* phát hiện được và kích thước vùng văn bản đó cũng gần với kích thước vùng chứa thông tin NXB (ảnh 1). Ngoài ra, 1 số sách bên dưới thông tin NXB còn có thông tin thêm không liên quan đến thông tin NXB (ảnh 2)

Ảnh 1

Ảnh 2

+ Thông tin Tên tác giả có ***f1-score*** trung bình và tốc độ giảm trung bình cũng không quá nhanh. Tuy nhiên, sau mốc **T** = 90, tốc độ giảm khá nhanh. Nguyên nhân do trong Test set có 1 số ảnh bìa sách có vùng văn bản chứa Tên tác giả có kích thước khá lớn nên dùng phương pháp lọc theo ngưỡng không đem lại hiệu quả





+ Thông tin Tên sách cho ***f1-score*** thấp nhân do các thông tin Tên tác giả và NXB bị lấy sai (lấy dư hoặc thiếu) sẽ gây ảnh hưởng đến thông tin Tên sách. Ngoài ra, phương pháp lọc theo ngưỡng là lọc tuyến tính nên có với 1 cuốn sách, không phải thông tin gì không thuộc 2 loại thông tin trên đều được xếp vào Tên sách. Do đó việc lấy thông tin Tên sách thường sẽ dư hoặc đôi khi thiếu, dẫn tới ***Ratio-score*** thấp.

5. TỔNG KẾT

5.1. Nhận xét mô hình

- Mô hình của nhóm em có thể chạy tốt với các ảnh bìa sách khác nhau với độ lỗi ở mức chấp nhận được (Lỗi nhận diện văn bản; lỗi lấy tên tác giả, tên sách). Tuy nhiên, mô hình còn có các điểm yếu cần khắc phục như:

+ Thời gian cho quá trình Text detection cần phải được tối ưu hơn.
+ Phương pháp lấy tên tác giả, tên sách, nhà xuất bản cần phải chính xác hơn.

PHỤ LỤC PHÂN CÔNG NHIỆM VỤ

Ngày	Công việc	Thành viên
25/12 - 27/12	Tìm và chọn đề tài. Xây dựng sơ lược cấu trúc mô hình	Cả nhóm
27/12 - 2/1	Thu thập dataset	Cả nhóm
	Tìm hiểu các mô hình cho Text detection và Text recognition đang có hiện nay	Nguyễn Khắc Thái
3/1 - 8/1	Tìm hiểu cấu trúc và quy trình hoạt động của CRAFT cho Text detection	Trần Duy Quang
	Tìm hiểu về source code của CRAFT trong paper và các source code pre-made khác từ mô hình CRAFT trong paper	Nguyễn Ngọc An
9/1	Họp nhóm và trao đổi kiến thức	Cả nhóm
10/1 - 14/1	Thực hiện gán nhãn cho các ảnh trong tập dataset tự thu thập được của nhóm	Cả nhóm
14/1 - 16/1	Chuyển hóa dataset thành folder có format để mô hình có thể dùng để train	Nguyễn Khắc Thái Nguyễn Ngọc An
	Tìm hiểu cách lọc thông tin trên bìa sách để lấy tên tác giả, tên sách và nhà sản xuất của mỗi ảnh bìa sách	Trần Duy Quang
17/1 - 18/1	Train 2 mô hình Transformer OCR và Attention OCR của VietOCR trên dataset tự thu thập	Cả nhóm
19/1 - 20/1	Hoàn thành nội dung báo cáo	Cả nhóm
22/1 - 30/1	Chỉnh sửa, bổ sung và hoàn thiện báo cáo	Cả nhóm