BỘ CÔNG THƯƠNG
TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI

ĐA, KLTN ĐẠI HỌC
NGÀNH CÔNG NGHỆ THÔNG TIN

NGHIÊN CỨU TÌM HIỂU VỚI MÔ HÌNH PICK VÀ ỨNG DỤNG TRONG VIỆC TRÍCH XUẤT THÔNG TIN HÓA ĐƠN

CBHD: Th.S. Nguyễn Thị Hương Lan
Sinh viên: Dương Ngọc Anh
Mã số sinh viên: 2020600274

Hà Nội – Năm 2024

LỜI CẢM ƠN

Để hoàn thành đề tài "Đồ án tốt nghiệp" này, em xin gửi lời cảm ơn chân thành đến quý thầy cô khoa Công Nghệ Thông Tin, trường Đại học Công Nghiệp Hà Nội đã giúp em có những kiến thức cực kì bổ ích trong vòng 4 năm vừa qua, giúp em có được nền tảng kiến thức vững chắc để em có thể thực hiện được đồ án.

Ngoài ra, em cũng gửi lời cảm ơn tới cô Nguyễn Thị Hương Lan, cảm ơn cô vì tất cả sự nhiệt tình, sự hết mình trong việc hỗ trợ các kiến thức liên quan đã giúp em hoàn thành đề tài đồ án tốt nghiệp này.

Em xin gửi lời cảm ơn đến Ban Giám Hiệu Trường Đại học Công nghiệp Hà Nội, các Ban, Ngành đã hỗ trợ hết mức tạo điều kiện tốt nhất để em có thể đăng ký đề tài đồ án tốt nghiệp.

Em xin chân thành cảm ơn!

Sinh viên

Dương Ngọc Anh

MỤC LỤC

DANH MỤC CÁC THUẬT NGỮ, KÝ HIỆU VÀ CHỮ VIẾT TẮT

STT	Từ viết tắt	Đầy đủ	Ý nghĩa
1	NLP	Natural Language Processing	Xử lý ngôn ngữ tự nhiên
2	SOTA	State-Of-the-Art	Hiện đại nhất
3	KIE	Key information extraction	Trích xuất thông tin quan trọng
4	BiLSTM	Bidirectional Long short-term memory	Kiến trúc mạng LSTM 2 chiều
5	NER	Name Entity Recognition	Nhận diện thực thể trong câu
6	MLP	Multi-layer perception	Mạng nơron truyền thẳng nhiều lớp
7	RNN	Recurrent Neural Network	Mạng neural hồi quy
8	BERT	Bidirectional Encoder Representations from Transformers	Mô hình biểu diễn từ theo 2 chiều ứng dụng Transformers
9	AUC	Area under curve	Diện tích dưới đường cong ROC
10	AP	Average Precision	Độ chính xác trung bình
11	AUPC	Area Under Precision-Recall Curve	Diện tích dưới đường cong precision-recall
12	ROC	Receiver Operating Characteristic	Đồ thị biểu diễn hiệu suất của một mô hình phân loại

			nhị phân trên dữ liệu thử nghiệm

DANH MỤC HÌNH ẢNH

DANH MỤC BẢNG BIỂU

MỞ ĐẦU

Những năm gần đây, AI - Artificial Intelligence (Trí Tuệ Nhân Tạo), và cụ thể hơn là Machine Learning (Máy Học) nổi lên như một minh chứng của cuộc cách mạng công nghiệp lần thứ tư (1 - động cơ hơi nước, 2 - năng lượng điện, 3 - công nghệ thông tin). AI hiện diện trong mọi lĩnh vực của đời sống con người, từ kinh tế, giáo dục, y khoa cho đến những công việc nhà, giải trí hay thậm chí là trong quân sự. Những ứng dụng nổi bật trong việc phát triển AI đến từ nhiều lĩnh vực để giải quyết nhiều vấn đề khác nhau. Nhưng những đột phá phần nhiều đến từ Deep Learning (học sâu) - một mảng nhỏ đang mở rộng dần đến từng loại công việc, từ đơn giản đến phức tạp. Deep Learning đã giúp máy tính thực thi những việc tưởng chừng như không thể vào 15 năm trước: phân loại cả ngàn vật thể khác nhau trong các bức ảnh, tự tạo chú thích cho ảnh, bắt chước giọng nói và chữ viết của con người, giao tiếp với con người, hay thậm chí cả sáng tác văn, phim, ảnh, âm nhạc.

Hiện nay, các mô hình SOTA Deep Learning trong Computer Vision đã đã thành công lớn trong các bài toán về Optical Character Recognition (OCR) bao gồm cả Text Detection và Text Recognition. Nhưng đối với bài toán trích xuất từ văn bản, là một bài toán phụ của OCR, có rất nhiều tình huống khác nhau trong thực tế. Bài toán trích xuất đúng là một thử thách bởi vì trong thực tế thì các văn bản trên thực tế có cấu trúc và thành phần khác nhau, các thành phần ngữ nghĩa không được trực quan và có sự nhập nhằng giữa các trường thông tin trong văn bản. Để có thể trích xuất được những thông tin cần thiết trong văn bản chúng ta cần đòi hỏi một mô hình có thể hiểu được ngữ nghĩa, cấu trúc của chúng trong văn bản và cuối cùng là phân loại chúng.

Nhắc đến trích xuất thông tin là gì chắc nhiều người đang chưa có câu trả lời, vì lâu nay đối với bài toán Optical Character Recognition (OCR) thường thì chúng ta sẽ quan tâm đến các bài toán như Text Detection và Text Recognition. Trong bài toán Text Detection hiện nay có một số mô hình khá nổi tiếng và tốt trên multi language texts như Character-Region Awareness For

Text detection (CRAFT), Difference Binarization (DB). Trong bài toán Text Recogniztion thì có Deep-Text-Recognition-Benchmark hay trong tiếng Việt hiện nay đang đạt kết quả khá tốt với ngôn ngữ tiếng Việt như VietOCR.

Chính vì những lý do đó, em chọn đề tài "Nghiên cứu tìm hiểu với mô hình PICK và ứng dụng trong việc trích xuất thông tin hoá đơn" nhằm tìm hiểu phương pháp để có thể trích xuất thông tin và có thể áp dụng cho bài toán trích xuất thông tin trên hóa đơn. Đồ án thiết kế gồm 4 chương:

- Chương 1: Tổng quan về đề tài
- Chương 2: Cơ sở lý thuyết
- Chương 3: Tìm hiểu về mô hình PICK
- Chương 4: Ứng dụng mô hình PICK vào bài toán để trích xuất thông tin hóa đơn.

CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI

1.1. Tên đề tài

Nghiên cứu tìm hiểu với mô hình PICK và ứng dụng trong việc trích xuất thông tin hoá đơn.

1.2. Lý do chọn đề tài

Hóa đơn là một loại giấy tờ thiết yếu được sử dụng trong các giao dịch tài chính, nhằm mục đích cung cấp hồ sơ về hàng hóa hoặc dịch vụ được trao đổi cùng với các chi phí liên quan. Chính vì thế, việc nhập liệu và số hóa các hóa đơn có thể giúp doanh nghiệp theo dõi phân tích và ra quyết định, chẳng hạn như xác định xu hướng chi phí, giám sát hoạt động của nhà cung cấp và tối ưu hóa quy trình mua – bán. Tuy nhiên, nhập liệu bằng phương pháp thủ công không chỉ gây lãng phí về nguồn lực nhân sự của doanh nghiệp mà còn dễ gây ra lỗi, điều này có thể dẫn đến những sai lầm tốn kém và sự chậm trễ trong hoạt động kinh doanh. Tự động hóa quy trình trích xuất thông tin chính từ hóa đơn có thể cải thiện đáng kể hiệu quả của hoạt động kinh doanh và giảm thiểu những rủi ro không đáng có.

Với sự phát triển ngày càng mạnh mẽ của các mô hình máy học, việc tự động trích xuất thông tin chính từ hóa đơn đạt được hiệu năng và độ ổn định cực kì ấn tượng. Trích xuất thông tin tự động bao gồm quá trình xác định vị trí và nội dung của các phần khác nhau trong văn bản, sau đó phân loại văn bản thành các trường thông tin được định nghĩa từ trước như số hóa đơn, ngày, tổng số tiền và chi tiết nhà cung cấp.

Nghiên cứu các mô hình xử lý trích xuất thông tin trong hóa đơn hiện nay là một đề tài hấp dẫn, có tính cấp thiết và có giá trị thực tiễn cao. Nhận thức được sự quan trọng và cũng như để có thể áp dụng được những kiến thức đã được học và tìm hiểu, em xin được áp dụng những kiến thức đã được học và em chọn đề tài đồ án "**Nghiên cứu tìm hiểu với mô hình PICK và ứng dụng trong việc trích xuất thông tin hoá đơn**", trong đó tập trung nghiên cứu sử dụng mô hình PICK để trích xuất thông tin.

1.3. Mục tiêu của đề tài

Đề tài**: Nghiên cứu tìm hiểu với mô hình PICK và ứng dụng trong việc trích xuất thông tin hoá đơn** đáp ứng được những mục tiêu:

Nghiên cứu mô hình PICK trong nhiệm vụ trích xuất thông tin.

Nắm được các kiến thức cơ bản về Python, mô hình PICK và các thư viện cần thiết để huấn luyện mô hình, … cùng các kiến thức, công cụ liên quan.

Ứng dụng mô hình PICK trong trích xuất thông tin hóa đơn với bộ dữ liệu đầu vào là các hóa đơn được cung cấp bởi cuộc thi Mobile-Captured Image Document Recognition for Vietnamese Receipts (MC-OCR) – Legacy.

1.4. Đối tượng và phạm vi

Đối tượng nghiên cứu: Mô hình PICK trong nhiệm vụ trích xuất thông tin hóa đơn.

Phạm vi nghiên cứu: Dựa trên bộ dữ liệu đầu vào, mô hình sẽ đưa ra thông tin tương ứng với hình ảnh dựa trên các trường thông tin được định nghĩa từ trước như số hóa đơn, ngày, tổng số tiền và chi tiết nhà cung cấp.

1.5. Kết quả dự kiến đạt được

Nghiên cứu một cách tổng quan về mô hình Pick.

Sử dụng các công cụ phần mềm có sẵn để tiền xử lý các trường hợp ảnh. Từ đó, có thể huấn luyện mô hình một cách chính xác.

1.6. Tổng kết chương 1

Trong chương này em đã trình bày những hướng cơ bản trong nghiên cứu để tìm hiểu về đề tài. Em tìm hiểu các nội dung về lý do chọn đề tài, mục tiêu của đề tài, đối tượng và phạm vi nghiên cứu từ đó tìm ra được hướng để tìm hiểu và hoàn thiện đề tài.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1. Lịch sử phát triển

- **Dấu mốc và lịch sử phát triển**:
  - Những năm 1950-1960: Các nghiên cứu sơ bộ về xử lý ảnh và trí tuệ nhân tạo bắt đầu, nhưng công nghệ và tài nguyên tính toán còn hạn chế.
  - Những năm 1970-1980: Bắt đầu xuất hiện các phương pháp xử lý ảnh cơ bản như làm mịn, phân đoạn và nhận dạng đối tượng.
  - Những năm 1990: Sự phát triển của mạng nơ-ron nhân tạo và các thuật toán học sâu đã làm cho việc xử lý ảnh trở nên hiệu quả hơn.
  - Những năm 2000 đến nay: Các công nghệ xử lý ảnh kỹ thuật số phổ biến, với sự phát triển đáng kể của các phương pháp học sâu và học máy trong việc xử lý và phân tích hình ảnh.
- **Các dấu mốc cụ thể:**
  - 1979: Paul Viola và Michael Jones phát minh ra phương pháp phát hiện khuôn mặt có thể áp dụng thực tế.
  - 1980s: Các phương pháp xử lý hình ảnh truyền thống như Convolutional Neural Networks (CNNs) bắt đầu xuất hiện.
  - 1998: Việc ra mắt thuật toán SIFT (Scale-Invariant Feature Transform) giúp phân tích và nhận dạng đặc trưng của hình ảnh.
  - 2012: AlexNet, một mạng nơ-ron học sâu, giành chiến thắng trong cuộc thi ImageNet, đánh dấu bước ngoặt quan trọng trong việc sử dụng học sâu cho xử lý ảnh.
  - 2014: Facebook công bố một hệ thống nhận diện khuôn mặt có thể nhận diện mọi người trong hình ảnh với độ chính xác cao.
  - 2019: RAISR (Rapid and Accurate Image Super-Resolution), một phương pháp sử dụng học máy để nâng cao độ phân giải của hình ảnh, được Google giới thiệu.

2.2. Tổng quan về xử lý ảnh

2.2.1. Khái niệm ảnh

Ảnh là thông tin về vật thể hay quang cảnh được chiếu sáng mà con người quan sát được và cảm nhận được bằng mắt và hệ thống thần kinh thị giác. Ảnh được tạo thành từ ba yếu tố:

- Vật thể, không gian quan sát được chiếu sáng.
- Nguồn sáng.
- Cảm nhận (mắt).

Ảnh trong tự nhiên (ảnh liên tục) là những tín hiệu liên tục về không gian và giá trị độ sáng. Tín hiệu thuộc loại tín hiệu đa chiều: tọa độ (x, y, z), độ sáng (λ), thời gian (t).

Ảnh được lưu trữ trong máy tính như một mảng 2 chiều chứa giá trị số. Các số tương ứng với các thông tin khác nhau như màu sắc hay cường độ mức xám, độ chói, thành phần màu… Để có thể lưu trữ và biểu diễn ảnh trên máy tính (ảnh số) con người phải tiến hành biến đổi các tín hiệu liên tục đó thành một số hữu hạn các tín hiệu rời rạc thông qua quá trình lượng tử hóa và lấy mẫu thành phần giá trị độ sáng. Ảnh trong không gian 2 chiều được dịnh nghĩa là một hàm 2 biến S (x, y), với S là giá trị độ sáng tại tọa độ (x, y).

Với ảnh liên tục S (x, y): Miền xác định (x, y) liên tục, miền giá trị S liên tục.

Với ảnh số S (m, n): Là ảnh liên tục được số hóa, miền xác định (m, n) rời rạc, miền giá trị S rời rạc.

Ảnh số: Tín hiệu số 2D.

2.2.2. Điểm ảnh

Điểm ảnh được xem là dấu hiệu hay cường độ sáng tại một tọa độ trong khong gian của đối tượng. Ảnh được xem như một tập hợp các điểm ảnh được tổ chức dưới dạng ma trận các điểm ảnh gồm M dòng và N cột. Giao giữa dòng và cột là điểm ảnh (pixel). Mỗi điểm ảnh gồm hai thông số: tọa độ (x,y) và giá trị màu/mức xám.

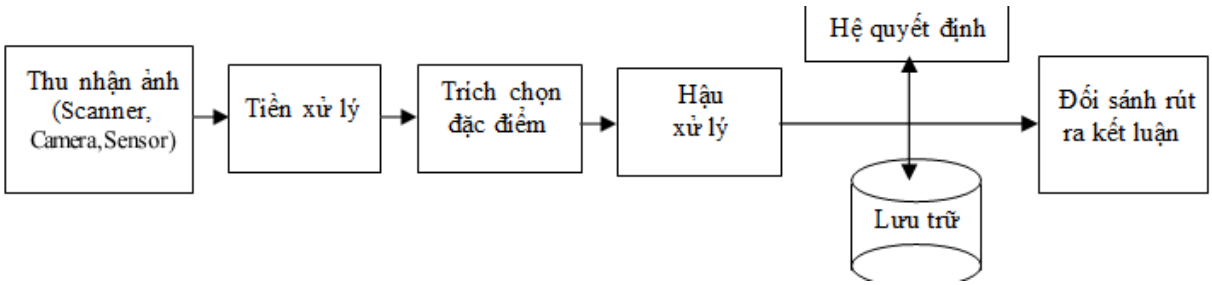
Màu (color): Là giá trị được tổ hợp từ ba thành phần cơ bản: Đỏ (Red), Xanh lam (Blue), Xanh lục (Green). Trong không gian màu RGB, một màu được tổ hợp từ ba thành phần R, G, B theo công thức color = R + G*2^8 + B*2^{16}. Nếu ba thành phần này không đồng thời bằng nhau thì giá trị tổ hợp trên sẽ có sắc thái màu, khi đó ta gọi là ảnh màu.

Mức xám (Gray level): Màu xám là màu có ba thành phần R = G = B. Ảnh xám hay ảnh đa cấp xám là ảnh chỉ chứa toàn màu xám.

Quá trình xử lý ảnh gồm một dãy các thao tác trên ảnh đầu vào nhằm cho ra kết quả mong muốn. Đầu ra của quá trình xử lý ảnh có thể là một ảnh "tốt hơn" hoặc một kết luận.

Hình 2.1 Xử lý ảnh

Xử lý ảnh là một tiến trình gồm nhiều công đoạn nhỏ, các công đoạn đó bao gồm:



Hình 2.2 Các giai đoạn của quá trình xử lý ảnh.

- Thu nhận ảnh: Việc thu nhận có thể thực hiện qua các thiết bị như máy ảnh, ảnh chụp từ về tinh qua các bộ cảm biến (Sensors), qua máy quét (Scanners).
- Tiền xử lý: Nhằm nâng cao chất lượng ảnh đầu vào để làm nổi bật một số đặc điểm của ảnh đầu vào hay làm cho ảnh giống nhất với trạng thái

gốc của nó. Có nhiều công cụ khác nhau để thực hiện tùy thuộc vào trạng thái của ảnh đầu vào như:

- Xóa nhiễu: Loại bỏ các đối tượng dư thừa trong ảnh (có thể đo chất lượng của thiết bị thu nhận, do nguồn sáng).
- Nắn chỉnh hình học: Khắc phục các biến dạng do các thiết bị điện tử và quang học gây nên, có thể khắc phục bằng các phép chiếu.
- Chỉnh mức xám: Khắc phục tính không đồng đều của mức xám, thường dùng để tăng hoặc giảm số mức xám của ảnh.

- Trích chọn đặc trưng: Nhằm tới hiểu ảnh. Có thể sử dụng các công cụ như: Dò biên để xác định biên, phân vùng, làm mảnh để trích xương, …
- Hậu xử lý: Nhằm hiệu chỉnh lại đặc điểm của những đặc trưng đã trích được từ bước bên trên sao cho bước thực hiện tiếp theo được thuận tiện và nhanh chóng nhưng vẫn không làm ảnh hưởng đến kết quả.
- Tùy mục đích của ứng dụng mà chuyển sang giai đoạn khác là lưu trữ, nhận dạng, phân lớp để rút ra kết luận…

2.2.4. Các vấn đề cơ bản của xử lý ảnh

a. Khử nhiễu

Có 2 loại nhiễu cơ bản trong quá trình thu nhận ảnh:

Nhiễu hệ thống: Là nhiễu có quy luật có thể khử bằng các phép biến đổi trên miền tần số.

Nhiễu ngẫu nhiên: Là dạng nhiễu xuất hiện ngẫu nhiên và có thể theo một phân bố nào đó. Chúng có thể được loại bỏ bằng các phép lọc.

Hình 2.3 Ảnh nhiễu và sau khi lọc nhiễu.

b. Chỉnh mức xám

Chỉnh số mức xám nhằm khắc phục tính không đồng đều mức xám do hệ thống gây ra. Thông thường có 2 hướng tiếp cận:

Giảm số mức xám: Thực hiện bằng cách nhón các mức xám gần nhau thành một bó. Trường hợp chỉ có 2 mức xám thì chính là chuyển về ảnh đen trắng. Ứng dụng là in ảnh màu ra máy in đen trắng.

Tăng mức xám: Thực hiện nội suy ra các mức xám trung gian bằng kỹ thuật nội suy. Kỹ thuật này nhằm tăng cường độ mịn cho ảnh.

c. Phân tích ảnh

Là khâu quan trọng trong quá trình xử lý ảnh để tiến tới hiểu ảnh. Trong phân tích ảnh, việc trích chọn đặc điểm là một bước quan trọng. Các đặc điểm của đối tượng được trích chọn tùy theo mục đích nhận dạng trong quá trình xử lý ảnh. Có thể nêu ra một số đặc điểm của ảnh sau:

- Đặc điểm không gian: Phân bố mức xám, phân bố xác suất, biên độ điểm uốn, …
- Đặc điểm biến đổi: Các đặc điểm loại này được trích chọn bằng việc thực hiện lọc vùng (zonal filtering). Các bộ vùng được gọi là mặt nạ đặc điểm (feature mask) thường là các khe hẹp với hình dạng khác nhau (chữ nhật, tam giác, cung tròn …).

- Đặc điểm biên và đường biên: Đặc trưng cho đường biên của đối tượng rất hữu ích trong việc trích chọn các thuộc tính bất biến được dùng khi nhận dạng đối tượng. Các đặc điểm này có thể được trích chọn nhờ toán tử gradient, toán tử la bàn, toán tử laplace, …

Việc trích chọn hiệu quả các đặc điểm giúp cho việc nhận dạng các đối tượng ảnh chính xác, với tốc độ tính toán cao và dung lượng lưu trữ nhỏ.

d. Nhận dạng

Nhận dạng tự động (automatic recognition), mô tả đối tượng, phân loại và phân nhóm các mẫu là những vấn đề quan trọng trong thị giác máy, được ứng dụng trong nhiều ngành khoa học khác nhau. Khi biết một mẫu nào đó, để nhận dạng hoặc phân loại mẫu đó có thể sử dụng các phương pháp sau:

- Phân loại có mẫu (phân lớp): Trong đó mẫu đầu vào được định danh như một thành phần của một lớp đã xác định.
- Phân loại không có mẫu (phân cụm): Trong đó các mẫu được gán vào các lớp khác nhau dựa trên một tiêu chuẩn đồng dạng nào đó. Các lớp này cho đến thời điểm phân loại vẫn chưa biết hay chưa được định danh.

Hệ thống nhận dạng tự động bao gồm ba khâu tương ứng với ba giai đoạn chủ yếu sau:

- Thu nhận dữ liệu và tiền xử lý.
- Biểu diễn dữ liệu.
- Nhận dạng, ra quyết định.

Bốn cách tiếp cận khác nhau trong lý thuyết nhận dạng là:

- Đối sánh mẫu dựa trên các đặc trưng được chính chọn.
- Phân loại thống kê.
- Đối sánh cấu trúc.
- Phan loại dựa trên mạng nơ-ron nhân tạo.

Trong các ứng dụng, không thể chỉ dùng cho một cách tiếp cận đơn lẻ để phân loại “tối ưu”. Thay vào đó, ta cần sử dụng cùng một lúc nhiều phương

pháp và cách tiếp cận khác nhau. Các phương thức phân loại tổ hợp hay được sử dụng khi nhận dạng và nay đã có những kết quả có triển vọng dựa trên thiết kế các hệ thống lai (hybrid system) bao gồm nhiều mô hình kết hợp.

Việc giải quyết bài toán nhận dạng trong những ứng dụng mới này sinh trong cuộc sống không chỉ tạo ra những thách thức về mặt giải thuật, mà còn đặt ra những yêu cầu về tốc độ tính toán. Đặc điểm chung của tất cả các ứng dụng đó là những đặc điểm đặc trưng cần thiết thường là nhiều, phải được trích chọn dựa trên các thủ tục phân tích dữ liệu.

e. Nén ảnh

Nhằm giảm thiểu không gian lưu trữ. Thường được tiến hành theo cả 2 khuynh hướng là nén có bảo toàn và không bảo toàn thông tin. Nén không bảo toàn thì thường có khả năng nén cao hơn nhưng khả năng phục hồi thì kém hơn. Trên cơ sở 2 khuynh hướng, có 4 cách tiếp cận cơ bản trong nén ảnh:

- Nén ảnh thống kê: Kỹ thuật nén này dựa vào việc thống kê tần xuất xuất hiện của giá trị các điểm ảnh, trên cơ sở đó mà chiến lược mã hóa thích hợp.
- Nén ảnh không gian: Kỹ thuật này dựa vào vị trí không gian của các điểm ảnh để tiến hành mã hóa. Cụ thể là lợi dụng sự giống nhau của các điểm ảnh trong các vùng gần nhau để tiến hành mã hóa.
- Nén ảnh sử dụng phép biến đổi: Đây là kỹ thuật tiếp cận theo hướng nén không bảo toàn dựa trên các phép biến đổi trên miền tần số.
- Nén ảnh Fractal: Sử dụng tính chất Fractal của các đối tượng ảnh, thể hiện sự lặp lại của các chj tiết. Kỹ thuật nén sẽ tính toán để chỉ cần lưu trữ phần gốc ảnh và quy luật sinh ra ảnh theo nguyên lý Fractal.

2.2.5. Các phép xử lý ảnh cơ bản

Có nhiều phép xử lý ảnh được sử dụng trong thị giác máy tính để trích xuất thông tin và phân tích hình ảnh. Dưới đây là một số phép xử lý ảnh phổ biến:

Chuyển đổi không gian màu: Chuyển đổi không gian màu giúp chúng ta chuyển đổi hình ảnh từ không gian màu này sang không gian màu khác, ví dụ như chuyển đổi từ RGB sang Grayscale. Điều này giúp chúng ta giảm kích thước dữ liệu và tăng tốc độ xử lý.

Làm mờ (Blurring): Làm mờ là một phép xử lý ảnh được sử dụng để giảm nhiễu và loại bỏ chi tiết không cần thiết trong hình ảnh. Các phương pháp làm mờ phổ biến bao gồm làm mờ Gaussian và làm mờ trung bình.

Phát hiện biên: Phát hiện biên là một phép xử lý ảnh được sử dụng để tìm ra các đường biên trong hình ảnh. Các phương pháp phát hiện biên phổ biến bao gồm phương pháp Sobel và phương pháp Canny.

Phân đoạn (Segmentation): Phân đoạn là một phép xử lý ảnh được sử dụng để phân tách hình ảnh thành các phần khác nhau để phân tích và hiểu hình ảnh. Các phương pháp phân đoạn phổ biến bao gồm phương pháp phân đoạn ngưỡng và phân đoạn dựa trên mô hình.

Trích xuất đặc trưng (Feature extraction): Trích xuất đặc trưng là một phép xử lý ảnh được sử dụng để trích xuất các đặc trưng quan trọng trong hình ảnh. Các phương pháp trích xuất đặc trưng phổ biến bao gồm phương pháp HOG (Histogram of Oriented Gradients) và phương pháp SIFT (Scale-Invariant Feature Transform).

Phân loại (Classification): Phân loại là một phép xử lý ảnh được sử dụng để phân loại hình ảnh vào các lớp khác nhau dựa trên các đặc trưng được trích xuất. Các phương pháp phân loại phổ biến bao gồm phương pháp SVM (Support Vector Machine) và phương pháp mạng thần kinh nhân tạo.

Các phép xử lý ảnh trên đây chỉ là một số ví dụ phổ biến. Khi xử lý hình ảnh, chúng ta có thể sử dụng nhiều phương pháp và kết hợp chúng để đạt được kết quả tốt nhất cho một bài toán cụ thể.

2.2.6. Ứng dụng của xử lý ảnh

Bảng 2.1 Lĩnh vực ứng dụng và tác dụng của xử lý ảnh

Các lĩnh vực ứng dụng	Tác dụng
- Thông tin ảnh, truyền thông ảnh. - Xử lý ảnh vệ tinh, viễn thám. - Thiên văn, nghiên cứu không gian vũ trụ.	- Cải thiện thông tin hình ảnh cho sự nhận biết của con người. - Cải thiện và phục hồi ảnh. - Dữ liệu vào và ra đều là ảnh.
- Địa chất thăm dò. - Người máy, tự động hóa. - Máy thông minh, thị giác máy nhân tạo. - Sinh học, y học. - Vật lý, hóa học. - Giám sát kiểm soát quân sự. - Xử lý ảnh phục vụ cuộc sống	- Trích thông tin từ ảnh cho sự phân tích sâu hơn. - Hiểu ảnh và nhận dạng ảnh. - Đầu vào là ảnh, đầu ra không phải là ảnh mà là biểu diễn nội dung của ảnh như mô tả, giải thích, phân loại.

2.3. Tổng quan về thị giác máy tính

2.3.1. Khái niệm

Thị giác máy tính (Computer Vision) là lĩnh vực nghiên cứu và ứng dụng của trí tuệ nhân tạo. Lĩnh vực này liên quan đến việc xử lý và phân tích hình ảnh và video để có thể hiểu và tạo ra thông tin dựa trên những gì được nhìn thấy. Việc phát triển lĩnh vực này có bối cảnh từ việc sao chép các khả năng thị giác của con người bởi sự nhận diện và hiểu biết một hình ảnh mang tính điện tử. Sự nhận diện hình ảnh có thể xem là việc giải quyết vấn đề của các biểu tượng thông tin từ dữ liệu hình ảnh qua cách dùng các mô hình được xây dựng với sự giúp đỡ của các ngành lý thuyết học, thống kê, vật lý và hình học. Thị giác máy tính cũng được mô tả là sự tổng thể của một dải rộng các quá trình tự động và tích hợp và thể hiện cho các nhận thức thị giác.

Thị giác máy tính là một môn học khoa học liên quan đến lý thuyết đằng sau các hệ thống nhân tạo có trích xuất các thông tin từ các hình ảnh. Dữ liệu hình ảnh có thể nhiều dạng, chẳng hạn như dạng chuỗi video, các cảnh từ camera, hay dữ liệu đa chiều từ máy quét y học. Đây còn là một môn học kỹ thuật, trong đó tìm kiếm và áp dụng các mô hình và các lỹ thuyết cho việc xây dựng các hệ thống thị giác máy tính.

Các lĩnh vực con của thị giác máy tính bao gồm tái cấu trúc cảnh, dò tìm sự kiện, theo dõi video, nhận diện bố cục đối tượng, đánh giá chuyển động, phục hồi ảnh…

Các kỹ thuật phổ biến trong thị giác máy tính bao gồm xử lý ảnh, trích xuất đặc trưng, phân loại và nhận dạng. Các phương pháp khác nhau được sử dụng để giải quyết các vấn đề khác nhau, bao gồm các kỹ thuật dựa trên mô hình, học máy, mạng nơ-ron nhân tạo, và nhiều kỹ thuật khác.

2.3.2. Các ứng dụng

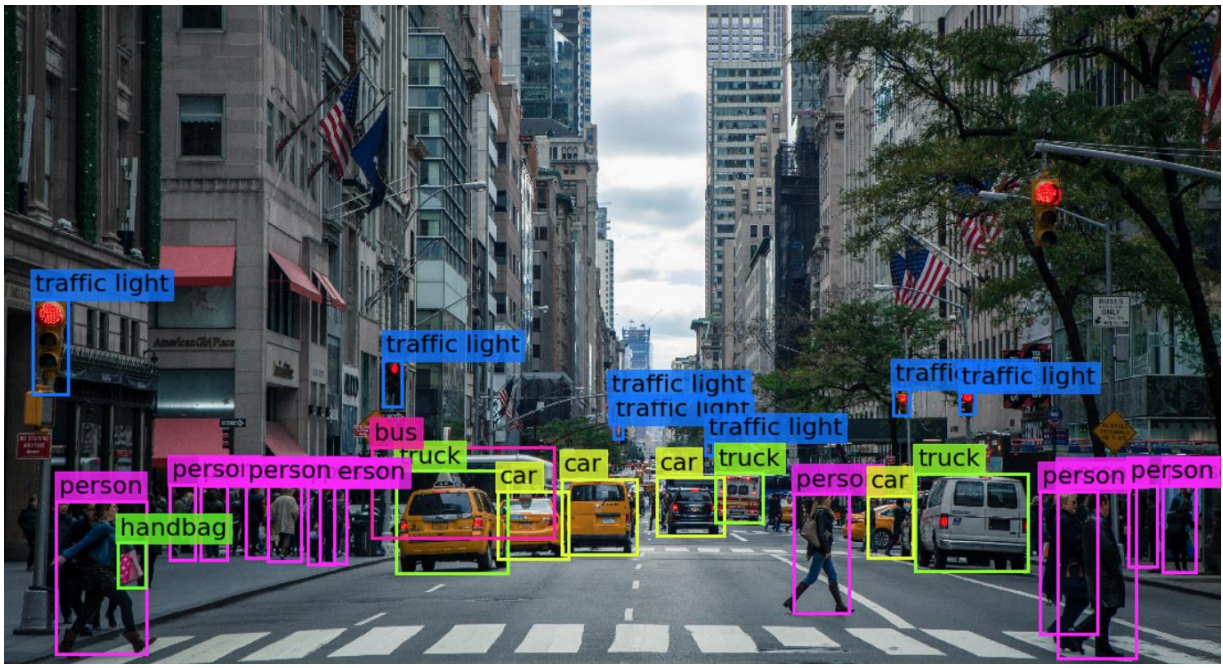
Ứng dụng của thị giác máy tính rất phong phú và đa dạng, bao gồm nhận dạng khuôn mặt, phát hiện đối tượng, phân tích hình ảnh y sinh, phân tích ảnh vệ tinh, tạo ra thế giới ảo, và nhiều ứng dụng khác. Các ứng dụng này được sử dụng trong nhiều lĩnh vực khác nhau, bao gồm y tế, an ninh, giao thông, sản xuất, giải trí và nhiều lĩnh vực khác.

Một số ứng dụng tiêu biểu của thị giác máy tính bao gồm:

- Nhận dạng khuôn mặt: Sử dụng để xác định một người trong hình ảnh hoặc video.
- Phát hiện đối tượng: Sử dụng để phát hiện các tối tượng trong ảnh hoặc video, bao gồm phát hiện xe cộ, người, động vật, vật thể và nhiều thứ khác.
- Xử lý ảnh y sinh: Sử dụng để phân tích hình ảnh y tế, bao gồm phát hiện ung thư, xác định bệnh và giúp phẫu thuật viên trong quá trình phẫu thuật.

- Phân tích ảnh vệ tinh: Sử dụng để phân tích hình ảnh vệ tinh để giám sát môi trường, dự báo thời tiết, dự đoán đường đi của bão, phát hiện cháy rừng và đánh giá tài nguyên đất đai.
- Tạo ra thế giới ảo: Sử dụng để tạo ra thế giới ảo trong trò chơi điện tử hoặc phim ảnh.
- Nhận diện chữ số viết tay: Ứng dụng trong việc chuyển đổi hình ảnh chữ, số sang văn bản…

Tổng quan về thị giác máy tính là rất đa dạng và phức tạp. Khi sử dụng các kỹ thuật và công cụ thích hợp, thị giác máy tính có thể giúp chúng ta giải quyết nhiều vấn đề phức tạp trong nhiều lĩnh vực khác nhau.



Hình 2.4 Thị giác máy tính.

2.3.3. Các lĩnh vực liên quan

Các lĩnh vực liên quan của trí tuệ nhân tạo giải quyết các vấn đề như lên kế hoạch tự động hay các suy tính cho các hệ thống robot để dò đường ở một môi trường nào đó. Sự hiểu biết chi tiết của các môi trường này được yêu cầu để dò đường thông qua chúng. Thông tin về môi trường có thể được cung cấp bởi một hệ thống thị giác máy tính, hoạt động như các cảm biến và cung cấp thông tin mức độ cao về môi trường và robot.

Trí tuệ nhân tạo và thị giác máy tính chia sẻ các chủ đề như nhận dạng mẫu và các kỹ thuật học. Kết quả là thị giác máy tính đôi khi được xem là một phần của lĩnh vực trí tuệ nhân tạo hay lĩnh vực khoa học máy tính nói chung.

Thị giác máy tính theo một cách nào đó là sự đảo ngược của đồ họa máy tính. Trong khi đồ họa máy tính sản sinh hình ảnh từ mô hình 3D, thì thị giác máy tính lại thường sản sinh ra các mô hình 3D từ dữ liệu hình ảnh. Có một khuynh hướng kết hợp 2 môn học này, ví dụ như khám phá trong tăng cường thực tế.

2.4. Bài toán Text Detection – Xác định văn bản

- Bài toán phát hiện chữ trong ảnh là bài toán xác định vị trị vùng có chữ trong ảnh đã trở nên phổ biến cả về mặt ứng dụng và nghiên cứu vì khả năng ứng dụng của nó. Đến hiện tại, bài toán phát hiện chữ trong ảnh thường được tiếp cận theo các hướng:
  1. Mô hình phát hiện chữ dựa trên cơ chế hồi quy (Regression based method)
     - Mô hình này giống như các mô hình dùng để phát hiện vật thể trong ảnh (object detector) và vật thể ở đây là chữ. Tuy nhiên do chữ có kích thước và hình dáng đa dạng như chữ cong, chữ ngiêng. Do đó các mô hình phát hiện chữ dựa trên mô hình hồi quy như SSD, YOLO, Faster RCNN ... đều gặp hạn chế trong biểu diễn tất cả các trường hợp có thể gặp trong thực tế.
  2. Mô hình phát hiện chữ dựa trên cơ chế phân đoạn (Segmentation based method)
     - Mô hình này hoạt động dựa trên phương pháp phân đoạn ý nghĩa trong hình ảnh (segmentation), dự đoán các pixel ở cùng một đối tượng rồi sử dụng các thuật toán hậu xử lý để lấy ra vị trí các đối tượng vì vậy có thể xử lý tốt đối với độ đa dạng về hình dạng cũng như kích thước của chữ. Có rất nhiều các

phương pháp đã được nghiên cứu như phương pháp dự đoán trên mức kí tự CRAFT hay dự đoán trên mức từ DB, Multi-scale FCN và PixelLink. Việc sử dụng trên mức kí tự sẽ không phải quan tâm nhiều tới vấn đề các từ sẽ được chia tách như thế nào vì việc chia tách dựa trên nhiều tiêu chí như ngữ nghĩa, khoảng cách, màu sắc, ... nhưng lại gặp nhược điểm là việc chuẩn bị dữ liệu cho mức kí tự khá khó khăn.

3. Mô hình phát hiện chữ end-to-end
    - Phương pháp end-to-end kết hợp cả hai mô hình phát hiện và nhận dạng chữ do đó có thể tăng cường độ chính xác của mô hình phát hiện chữ thông qua kết quả của mô hình nhận dạng chữ. Có một số mô hình đã được nghiên cứu như FOST, ...

2.5. Bài toán Text Classification - Phân loại Văn bản

- Bài toán phân loại văn bản là nhiệm vụ gán một hoặc nhiều nhãn cho một đoạn văn bản dựa trên nội dung của nó. Đây là một trong những ứng dụng phổ biến của xử lý ngôn ngữ tự nhiên (NLP), có thể áp dụng trong nhiều lĩnh vực như phân loại email (spam và không spam), phân loại tin tức (thể thao, kinh tế, giải trí), phân loại đánh giá sản phẩm (tích cực, tiêu cực), v.v... Có nhiều phương pháp và mô hình được sử dụng trong phân loại văn bản, trong đó một số phổ biến bao gồm:

1. Naive Bayes:
    - Mô tả: Dựa trên định lý Bayes và giả định rằng các đặc trưng độc lập với nhau.
    - Ưu điểm: Đơn giản, nhanh chóng và hiệu quả với các bài toán phân loại văn bản cơ bản.
    - Nhược điểm: Giả định độc lập giữa các từ có thể không hợp lý trong nhiều trường hợp.
2. Logistic Regression:

- Mô tả: Sử dụng hồi quy logistic để dự đoán xác suất một văn bản thuộc về một nhãn nào đó.
- Ưu điểm: Hiệu quả, dễ triển khai và giải thích.
- Nhược điểm: Hiệu suất có thể kém nếu dữ liệu không tuyến tính.

3. Support Vector Machines (SVM):
   - Mô tả: Tìm một siêu phẳng để phân tách các lớp văn bản với biên lớn nhất.
   - Ưu điểm: Hiệu quả với dữ liệu không tuyến tính và không đồng nhất.
   - Nhược điểm: Thời gian huấn luyện có thể lâu với dữ liệu lớn.
4. Recurrent Neural Networks (RNN):
   - Mô tả: Sử dụng mạng nơ-ron hồi tiếp để xử lý dữ liệu tuần tự.
   - Ưu điểm: Xử lý tốt dữ liệu tuần tự và có ngữ cảnh.
   - Nhược điểm: Dễ bị vấn đề gradient biến mất hoặc bùng nổ.
5. Convolutional Neural Networks (CNN):
   - Mô tả: Sử dụng mạng nơ-ron tích chập để trích xuất đặc trưng từ văn bản.
   - Ưu điểm: Tốt trong việc trích xuất đặc trưng cục bộ.
   - Nhược điểm: Không nắm bắt được ngữ cảnh dài hạn.
6. Transformer-based Models (như BERT):
   - Mô tả: Sử dụng cơ chế self-attention để mô hình hóa mối quan hệ giữa các từ trong câu.
   - Ưu điểm: Hiệu suất cao, xử lý tốt ngữ cảnh dài và quan hệ từ.
   - Nhược điểm: Cần nhiều tài nguyên tính toán và dữ liệu huấn luyện.

2.6. Tổng kết chương 2

Trong chương này, em đã trình bày tổng quan về xử lý ảnh cũng như hai bài toán Text detection, Text classification và các mô hình liên quan đến hai bài toán. Từ đó, ta có thể chọn lựa được phương pháp hoặc mô hình phù hợp trong việc tiền xử lý ảnh.

CHƯƠNG 3: TÌM HIỂU VỀ MÔ HÌNH PICK

3.1. Khái niệm

PICK (Processing Key Information Extraction from Documents using Improved Graph Learning-Convolutional Networks) là một framework được thiết kế cho KIE với khả năng xử lý các bố cục tài liệu phức tạp. Mô hình đạt được điều này bằng cách kết hợp học đồ thị với các hoạt động convolution đồ thị. Sự kết hợp này mang lại một biểu diễn ngữ nghĩa toàn diện hơn, tích hợp cả các đặc điểm văn bản và hình ảnh cũng như chi tiết bố cục toàn cảnh, mà không gây ra sự mơ hồ.

3.2. Tại sao lại cần PICK?

Hầu hết các hệ thống KIE đơn giản chỉ coi các nhiệm vụ trích xuất như các vấn đề về gắn thẻ chuỗi và được thực hiện bằng cấu trúc Nhận dạng Thực thể Đặt Tên (NER), xử lý văn bản thuần túy như một chuỗi tuyến tính dẫn đến việc bỏ qua hầu hết thông tin quan trọng hình ảnh và không tuần tự (ví dụ như văn bản, vị trí, bố cục và hình ảnh) của tài liệu cho KIE.

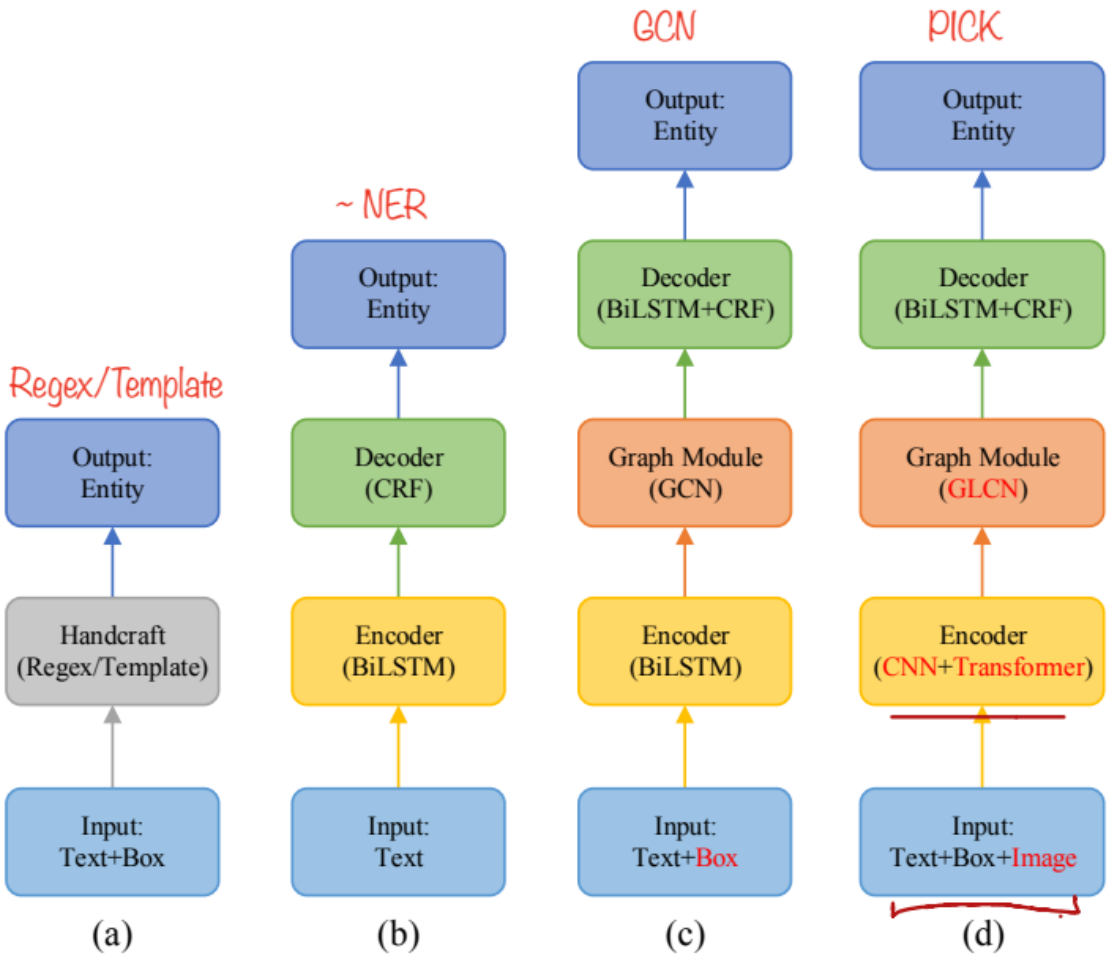
Gần đây, một số nghiên cứu trong nhiệm vụ KIE đã cố gắng tận dụng đầy đủ các đặc điểm chưa được khai thác trong các tài liệu phức tạp, đã đề xuất phương pháp LayoutLM, lấy cảm hứng từ BERT.

Hoặc phương pháp tiếp cận khác: xác định trước một đồ thị để kết hợp thông tin văn bản và hình ảnh bằng cách sử dụng hoạt động convolution đồ thị.

Vì lẽ đó, mô hình Pick ra đời nhằm tổng hợp lại các yếu tố đặc điểm của tài liệu (bao gồm văn bản, vị trí, bố cục và hình ảnh) để mang lại một biểu diễn ngữ nghĩa toàn diện hơn, điều này rất quan trọng để trích xuất thông tin.

3.3. Phương pháp tiếp cận

Hiện nay, có rất nhiều mô hình với các hướng tiếp cận khác nhau trong bài toán này. Hướng tiếp cận đơn giản nhất đó là sử dụng Text Classification để phân loại ra những thông tin nào thuộc lớp nào, cách giải quyết này có thể đơn giản và tốt trên những dạng văn bản có sự phân biệt rõ rệt giữa các trường thông tin, và đặc biệt là cấu trúc văn bản đó đơn giản.



Hình 3.1 Các loại cấu trúc và phương thức cho bài toán trích xuất thông tin

Trên ảnh là 4 hướng tiếp cận với bài toán này:

- Phương thức 1a) sử dụng đầu vào là text và box chứa text đó và dùng các công thức, mẫu chuẩn để giải quyết bài toán, cuối cùng là đưa ra dự đoán cho thực thể đó.
- Phương thức 1b) sử dụng đầu vào là text và dùng một mô hình Encoder gồm các layer BiLSTM và Decoder sử dụng layer CRF để dự đoán thực thể đó. (bài toán Text Classification)
- Phương thức 1c) sử dụng đầu vào là text và box chứa text đó đi qua một mô hình Encoder gồm các layer BiLSTM sau đó đưa qua một mô hình Graph gọi là GCN cuối cùng đưa qua mạng Decoder bao gồm cả BiLSTM và CRF để dự đoán ra thực thể đó

- Phương thức 1d) phức tạp hơn các phương thức còn lại. Nhận đầu vào là text, box chứa text và cả hình ảnh. Đầu tiên cho qua một mô hình Encoder gồm (CNN+ Transformer) sau đó cho qua mô hình Graph gọi là GLCN rồi cuối cùng đưa vào mô hình Decoder bao gồm (BiLSTM+CRF) để đưa ra dự đoán cho thực thể đó.

Phương pháp 1d) cũng chính là phương pháp mô hình PICK nhắm tới.

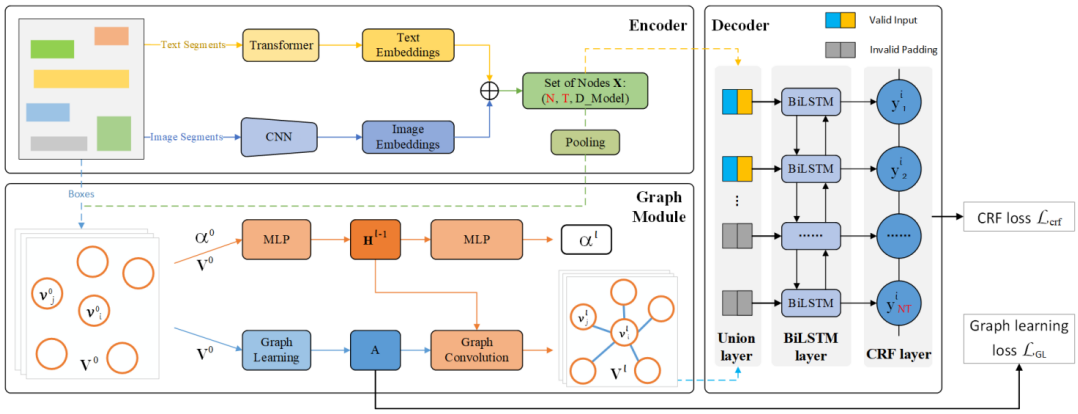
3.3.1. Nhiệm vụ phía sau (Downstream task)

Là những nhiệm vụ học hỏi được giám sát được cải thiện dựa trên những mô hình được huấn luyện trước.

Ví dụ: Chúng ta sử dụng lại các biểu diễn từ học được từ những mô hình được huấn luyện trước trên bộ hóa đơn lớn vào một nhiệm vụ huấn luyện trên bộ hóa đơn có kích thước nhỏ hơn.

3.4. Phương pháp

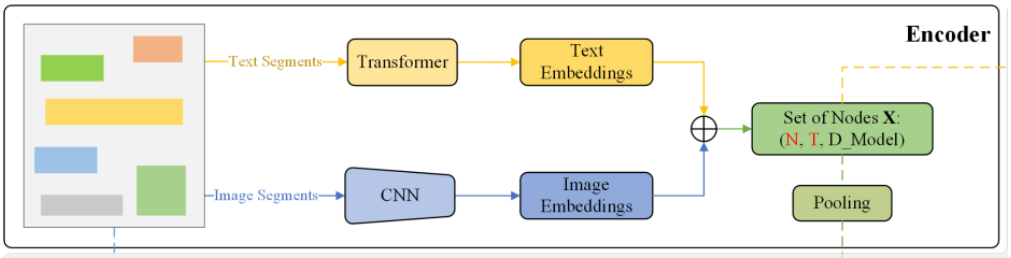
Mô hình sử dụng có 3 mô-đun chính: Encoder, Graph Module, Decoder.



Hình 3.2 Sơ đồ kiến trúc của mô hình PICK.

Encoder

- Sử dụng mô hình Transformer để trích xuất đặc trưng từ văn bản và sử dụng một mạng CNN để trích xuất đặc trưng từ ảnh. Sau đó kết hợp text embbedings và image embbedings lại thành vector biểu diễn X thể hiện khả năng biểu diễn text và image chứa text đó. X được đưa xem là đâu vào của Graph mô-đun.



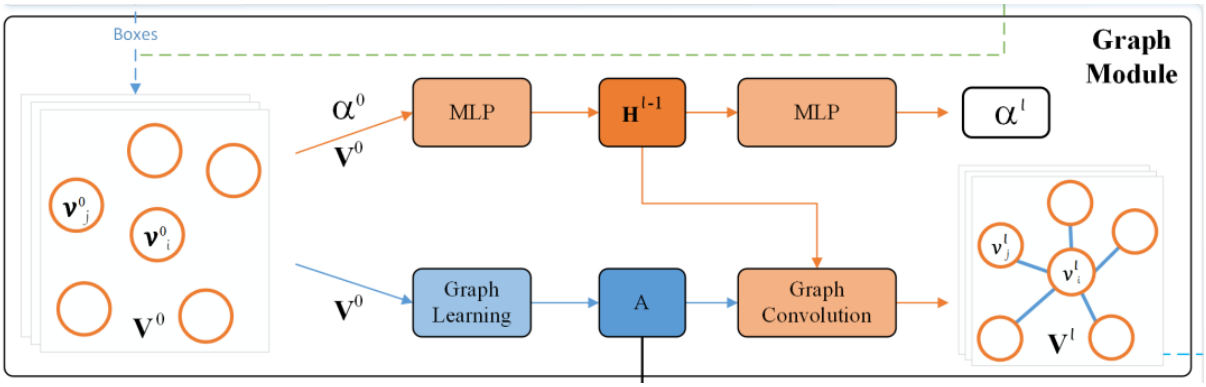
Hình 3.3 Encoder

Như hình vẽ trên, Encoder gồm 2 luồng xử lí. Luồng đầu tiên, sử dụng các vùng text để đưa qua mô hình Transformer giúp việc trích xuất dữ liệu từ dạng text. Mặt khác, luồng thứ 2 sử dụng một mạng CNN để trích xuất đặc trưng từ các vùng ảnh. Sau đó, kết hợp chúng lại bằng phép toán cộng ma trận được một ma trận **X** có $\mathbf{X} \in R^{NxTxd_{model}}$. Trong đó, N là số lượng vùng văn bản hoặc câu, T là độ dài của văn bản hoặc câu đó và D_model là số chiều ẩn của mô hình.

$\mathbf{X} \in R^{NxTxd_{model}}$ được làm để biểu diễn một tập hợp các nodes của Graph và *X*X được sử dụng làm input $\mathbf{X_0}$ của **Graph Module** bằng cách qua một layer pooling và $\mathbf{X_0} \in R^{Nxd_{model}}$.

Graph Module

- Sử dụng một mạng GCN để làm giàu khả năng biểu diễn giữa các node với nhau. Việc các thông tin cần trích xuất có vị trí và nội dung khác nhau, nó không cục bộ và không theo thứ tự nên việc sử dụng Graph giúp mô hình có thể học được khả năng biểu diễn mối tương quan giữa chúng về khoảng cách và vị trí trong văn bản.



Hình 3.4 Graph module

Ở mô-đun này, mô hình sử dụng một mạng Graph neural network để mô hình hóa bối cảnh toàn cục và các thông tin có cấu trúc không tuần tự để xác định trước loại cạnh và ma trận kề của đồ thị. Định nghĩa cạnh là việc các nodes/text segments kết nối với nhau theo chiều ngang hay chiều dọc. Ma trận kề được xác định dựa trên 4 loại cạnh sau: "Trái sang Phải", "Phải sang Trái", "Trên xuống Dưới", "Dưới lên Trên". Nhưng phương pháp này không thể sử dụng đầy đủ tất cả các nút của đồ thị và khai thác các nút được kết nối tiềm ẩn cách xa nhau trong văn bản. Mặc dù sử dụng mô hình kết nối đầy đủ nhưng thao tác này dẫn đến tổng hợp thông tin nút dư thừa và vô ích của biểu đồ. Bởi vì thế, Paper đã kết hợp phát triển mạng Graph learning-convolutional dựa trên cấu trúc mô hình Graph để học một soft adjacent matrix thay vì hard adjacent matrix. Gồm 2 phần chính:

1. **Graph learning**:

   Nhận đầu vào là một vector $\mathbf{V} = [v_1, \ldots, v_N]^T \in R^{Nxd_{model}}$, trong đó $v_i \in R^{d_{model}}$ là node thứ i trong đồ thị và khởi tạo giá trị **V** bằng với $\mathbf{X}_0$, Graph module sinh ra một ma trận kề A biểu diễn một quan hệ giữa 2 node đầu tiên cho qua Graph learning và ma trận H được trích xuất cho mỗi node v_i sử dụng một mạng multi-layer perception (MLP), nhận đầu vào là **V** và vector α embbeding từ mối quan hệ tương ứng giữa các nodes. Sau đó đưa ma trận *H* qua mạng Graph Convolutional và biểu diễn chúng thành **V'.** Về toán học, để học

một soft adjacent matrix A sử dụng một layer neural hoạt động như sau:

$$\begin{cases} A_i = softmax(e_i), \\ e_{ij} = Leak\ Relu(w_i^T|v_i - v_j|), \end{cases} \quad i = 1, \dots, N,\ j = 1, \dots, N,$$

Trong đó, $w_i \in R^{d_{model}}$ là learnable weight vector. Hàm softmax được tính toán trên mỗi hàng của ma trận A.

Hàm loss $£_{GL}$ được định nghĩa như sau:

$$£_{GL} = \frac{1}{N^2} \sum_{i,j=1}^{N} \exp\left(A_{ij} + \eta\left\|v_i - v_j\right\|_2^2\right) + \gamma\|A\|_F^2$$

Với biểu thức đầu tiên, mục tiêu là với các node v_i và v_j càng xa nhau thì giá trị weight của A_ij càng nhỏ. Tương tự, các node gần nhau có thể có trọng số mạnh mẽ hơn. Việc này có thể ngăn Graph convolutional học các node nhiễu. Sau đó tính trung bình lại do số lượng các node là không cố định trong từng văn bản.

2. **Graph convolutional**:

Graph convolutional network có nhiệm vụ biểu diễn thông tin và bố cục của các node. Biểu diễn graph convolutional theo node-edge node

$(v_i, \propto_{ij}, v_j)$. Khởi tạo ban đầu nhận đầu vào là $\boldsymbol{V}^0 = \boldsymbol{X}_0 \in R^{Nxd_{model}}$ và khởi tạo $\propto_0$ theo công thức sau:

$$\propto_{ij}^0 = W_\propto^0 \left[x_{ij}, y_{ij}, \frac{w_i}{h_i}, \frac{h_j}{h_i}, \frac{w_j}{h_i}, \frac{T_j}{T_i}\right]^T$$

Trong đó, W là trọng số học của mô hình, x_{ij} và y_{ij} là khoảng cách chiều ngang và chiều dọc của node i đến node j. Các thông số w_i, h_i, w_j, h_j, là chiều rộng và chiều cao của các node i và j.

Sau đó, tính toán ra đặc trưng ẩn h_{ij}^{l} giữa node v_i và v_j từ đồ thị sử dụng bộ ba node-edge-node $(v_i, \propto_{ij}, v_j)$, h_{ij}^{l} được tính theo công thức sau:

$$h_{ij}^{l} = \propto \left(W_{v_i h}^{l} v_i^{l} + W_{v_j h}^{l} v_j^{l} + \propto_{ij}^{l} + b^{l} \right)$$

Cuối cùng, v_i^{l+1} tổng hợp thông tin từ các đặc trưng ẩn h_{ij}^{l} và soft adjacent matrix **A** sử dụng graph convolutional để cập nhật các node. Ta có công thức:

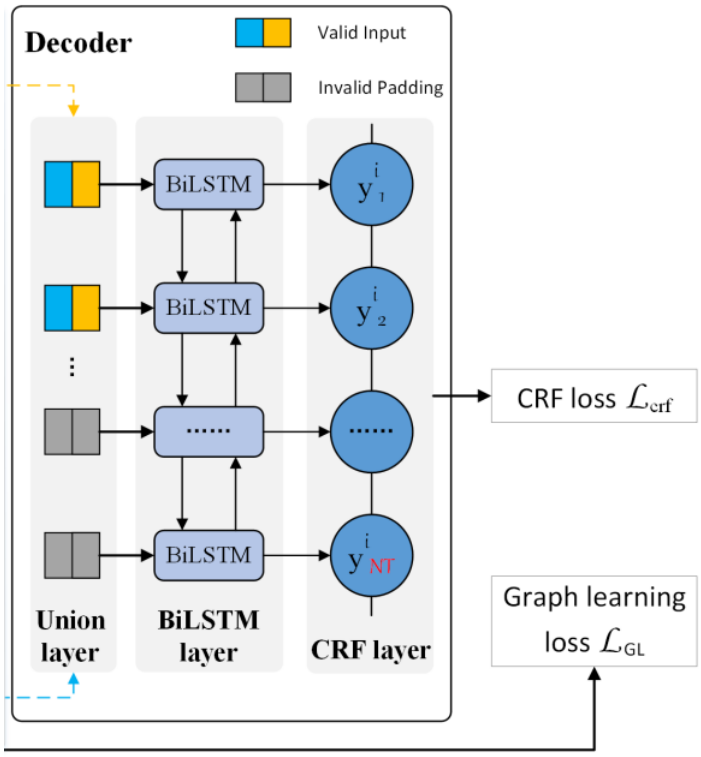
$$v_i^{l+1} = \propto (A_i h_i^{l} W^{l})$$

$\propto_{ij}^{l+1}$ được cập nhật theo công thức:

$$\propto_{ij}^{l+1} = \propto (\propto (W_{\propto}^{l} h_{ij}^{l})$$

Decoder

- Nhận đầu vào là sự kết hợp giữa đầu ra của **Encoder** và **Graph Module**. Cho đi qua một mạng BiLSTM layer và CRF layer để phân loại chúng. Cuối cùng mô hình sử dụng 2 hàm loss để tối ưu đồng thời chúng, đó là loss của Graph learning và CRF loss. Ở mô-đun này mô hình PICK sử dụng BiLSTM layer + CRF layer.



Hình 3.5 Decoder

3.5. Khái niệm liên quan

- **LayoutLM** là một mô hình học sâu dựa trên ý tưởng của BERT (mô hình ngôn ngữ sử dụng kiến trúc Transformers), được tiền huấn luyện trên lượng lớn dữ liệu văn bản và nhờ đó có tính tổng quát cao trên cả các đặc trưng về hình ảnh và ngữ nghĩa của nhiều loại tài liệu khác nhau.

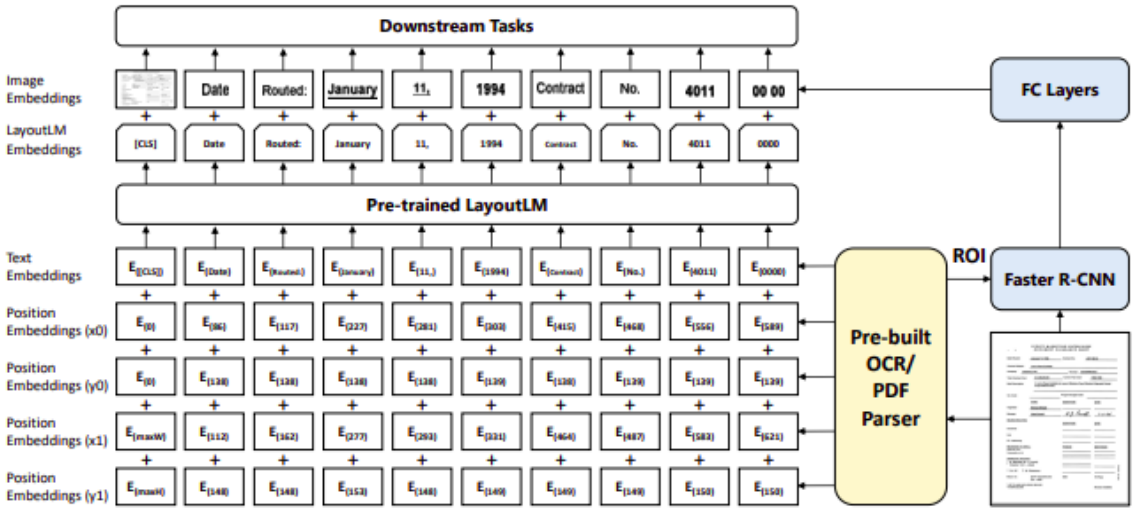


Figure 2: An example of LayoutLM, where 2-D layout and image embeddings are integrated into the original BERT architecture. The LayoutLM embeddings and image embeddings from Faster R-CNN work together for downstream tasks.

Hình 3.6 Kiến trúc LayoutLM

- **Mô hình Bert**: BERT là một model biểu diễn ngôn ngữ (Language Model - LM) được google giới thiệu vào năm 2018. BERT là viết tắt của cụm từ Bidirectional Encoder Representation from Transformer có nghĩa là mô hình biểu diễn từ theo 2 chiều ứng dụng kỹ thuật Transformer. BERT được thiết kế để huấn luyện trước các từ nhúng (pre-train word embedding).

  Điểm đặc biệt ở BERT đó là nó có thể điều hòa cân bằng bối cảnh theo cả 2 chiều trái và phải. Trước khi BERT ra đời thì các tác vụ như: phân loại cảm xúc văn bản (tốt hay xấu, tích cực hay tiêu cực), sinh văn bản, dịch máy, …. đều sử dụng kiến trúc RNN. Kiến trúc này có nhiều nhược điểm như train chậm, mất quan hệ giữa các từ xa nhau, ... BERT kế thừa từ Transformer đã giải quyết được các nhược điểm này.

- Mạng ***trí nhớ ngắn hạn định hướng dài hạn*** còn được viết tắt là LSTM làm một kiến trúc đặc biệt của RNN có khả năng học được sự phục thuộc trong dài hạn (*long-term dependencies*) được giới thiệu bởi Hochreiter & Schmidhuber (1997). Kiến trúc này đã được phổ biến và sử dụng rộng rãi cho tới ngày nay. LSTM đã tỏ ra khắc phục được rất nhiều những hạn chế của RNN trước đây về triệt tiêu đạo hàm. Tuy nhiên cấu trúc của chúng có phần phức tạp hơn mặc dù vẫn dữ được tư tưởng chính của RNN là sự sao chép các kiến trúc theo dạng chuỗi. Kiến trúc **BiLSTM** là kiến trúc mạng LSTM 2 chiều có khả năng đọc đầu vào theo chiều từ trái qua phải và từ phải qua trái.

3.6. Tổng kết chương 3

Trong chương này, em đã trình bày các nghiên cứu của em về tổng quan mô hình PICK và các khái niệm liên quan.

CHƯƠNG 4: ỨNG DỤNG MÔ HÌNH PICK VÀO TRÍCH XUẤT THÔNG TIN HÓA ĐƠN

4.1. Phát biểu bài toán

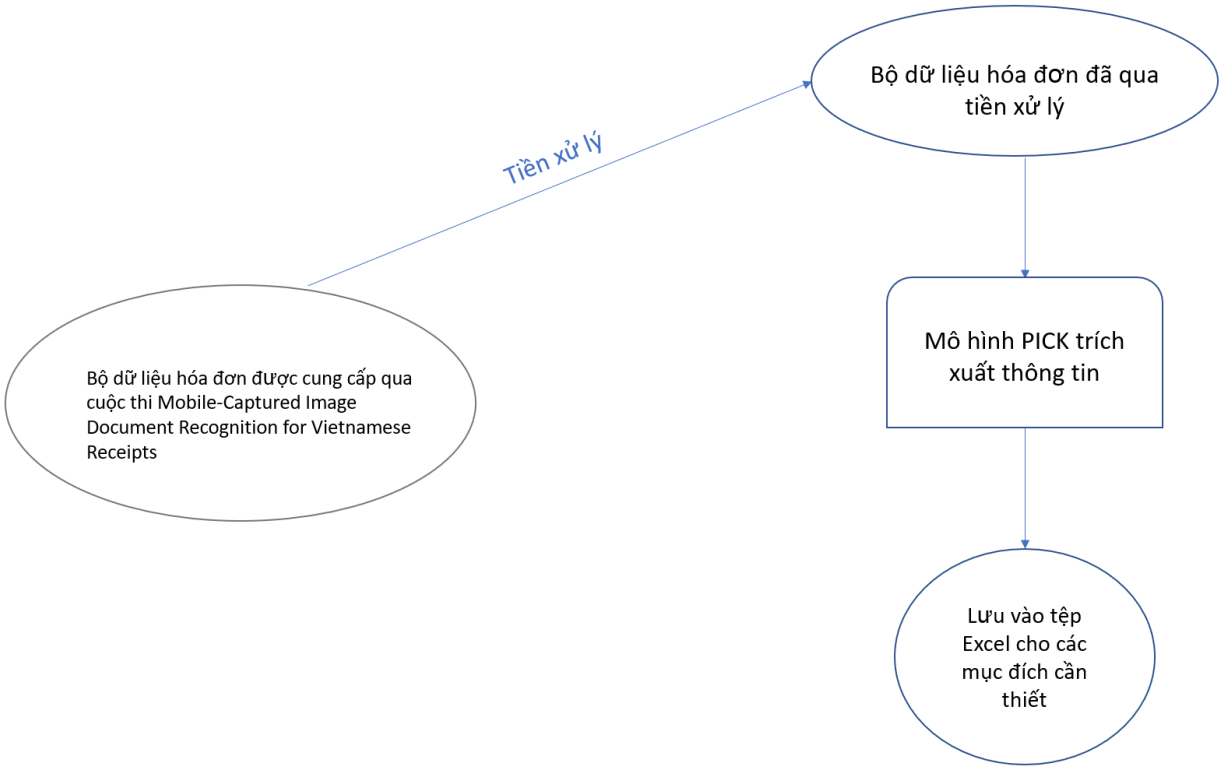
Các doanh nghiệp hiện nay đa phần vẫn nhận hóa đơn giấy, số ít còn lại sẽ là hóa đơn dưới dạng email, EDI - trao đổi dữ liệu điện tử. Nếu vẫn áp dụng các cách làm truyền thống như đọc hóa đơn và nhập liệu lên hệ thống thì rất dễ mắc vào tình trạng nhầm hóa đơn, bỏ sót, sai dữ liệu, ...

- Trích xuất thông tin từ văn bản là lĩnh vực đã và đang thu hút được sự quan tâm của cộng đồng các nhà nghiên cứu cũng như các nhà phát triển ứng dụng. Đầu tiên, hóa đơn có nhiều định dạng và bố cục khác nhau, khiến việc chuẩn hóa quy trình trích rút trên tất cả các hóa đơn trở nên khó khăn.
- Bố cục của các hóa đơn có thể khác nhau tùy thuộc vào nhà cung cấp, quốc gia hoặc ngành, khiến việc tạo một mô hình trích rút duy nhất hoạt động cho tất cả các hóa đơn trở nên khó khăn.
- Thứ hai, hóa đơn có thể dài, chứa nhiều trang thông tin, điều này có thể làm phức tạp thêm quá trình trích rút. Thông tin liên quan có thể được trải rộng trên nhiều trang hoặc nhiều phần của hóa đơn, khiến việc xác định và trích rút tất cả thông tin cần thiết trở nên khó khăn.
- Thứ ba, một số trường thông tin trong hóa đơn có thể gây nhầm lẫn ngay cả đối với con người. Chẳng hạn, hóa đơn thường chứa các trường thông tin có đặc trưng tương tự và dễ gây nhầm lẫn như “tên người bán” và “tên người mua” dẫn đến nhầm lẫn, sai sót trong quá trình trích rút.
- Cuối cùng, hóa đơn cũng có thể chứa lỗi hoặc thiếu thông tin, khiến việc trích rút chính xác tất cả thông tin cần thiết trở nên khó khăn. Ví dụ: hóa đơn có thể không có thông tin liên hệ của nhà cung cấp hoặc số tiền trên hóa đơn có thể không chính xác.

4.2. Dữ liệu, công cụ và môi trường thực nghiệm

4.2.1. Dữ liệu

Xét mô hình dữ liệu:



Hình 4.1 Mô hình dữ liệu

- **Xây dựng bộ dữ liệu**

1. Thực hiện đăng ký tài khoản cuộc thi và tải dữ liệu ảnh cuộc thi:
   - Chọn vị trí lưu và giải nén
   - Đánh giá chất lượng hóa đơn
2. Tiền xử lý dữ liệu

Thực hiện các bước tiền xử lý thật kỹ để tăng hiệu suất cho mô hình được chính xác nhất theo các bước:

- Dùng một số luật để lọc bớt ảnh sai thông tin hoặc gán nhãn sai.
- Xác định vùng văn bản (text – detector)
- Xoay lại hóa đơn nếu chưa đúng
- Phân loại văn bản (text – classifier)

4.2.2. Công cụ và môi trường thực nghiệm

❖ **Công cụ chính**

Phần mềm Anaconda:

Anaconda là một nền tảng phân phối các ngôn ngữ lập trình Python và R, phục vụ cho tính toán khoa học (Khoa học dữ liệu, machine learning, xử lý dữ liệu lớn, phân tích dự đoán…) nhằm mục đích đơn giản hóa việc quản lý và triển khai gói. Nền tảng này có trên cả Windows, MacOS và Linux.

Anaconda chứa tất cả các gói (packages) phổ biến nhất mà một nhà khoa học dữ liệu cần. Các packages trong Anaconda được quản lý bởi trình quản lý riêng của nền tảng này là conda. Ta thường dùng conda để tạo môi trường cô lập các dự án của mình, nhằm sử dụng các phiên bản Python khác nhau hoặc các phiên bản package khác nhau, cũng như dùng nó để cài đặt, gỡ cài đặt và cập nhật các package riêng trong từng dự án.

Ngôn ngữ lập trình Python:

Python được Guido van Rossum phát triển vào cuối những năm tám mươi và đầu những năm chín mươi tại Viện nghiên cứu quốc gia về toán học và khoa học máy tính ở Hà Lan.

Python là ngôn ngữ lập trình hướng đối tượng, cấp cao, mạnh mẽ, được tạo ra bởi Guido van Rossum. Nó dễ dàng để tìm hiểu và đang nổi lên như một trong những ngôn ngữ lập trình nhập môn tốt nhất cho người lần đầu tiếp xúc với ngôn ngữ lập trình. Python hoàn toàn tạo kiểu động và sử dụng cơ chế cấp phát bộ nhớ tự động. Python có cấu trúc dữ liệu cấp cao mạnh mẽ và cách tiếp cận đơn giản nhưng hiệu quả đối với lập trình hướng đối tượng. Cú pháp lệnh của Python là điểm cộng vô cùng lớn vì sự rõ ràng, dễ hiểu và cách gõ linh động làm cho nó nhanh chóng trở thành một ngôn ngữ lý tưởng để viết script và phát triển ứng dụng trong nhiều lĩnh vực, ở hầu hết các nền tảng.

Tính năng chính của Python:

• Ngôn ngữ lập trình đơn giản, dễ học

• Miễn phí, mã nguồn mở

• Khả năng di chuyển

• Khả năng mở rộng và có thể nhúng

• Ngôn ngữ thông dịch cấp cao

• Thư viện tiêu chuẩn lớn để giải quyết những nhiệm vụ phổ biến

• Hướng đối tượng

Thư viện máy học mã nguồn mở Pytorch: Trong số những framework hỗ trợ deeplearning thì pytorch là một trong những framework được ưa chuộng nhiều nhất (cùng với tensorflow và keras), có lượng người dùng đông đảo, cộng đồng lớn mạnh. Vào năm 2019 framework này đã vươn lên vị trí thứ 2 về số lượng người dùng trong những framework hỗ trợ deeplearning (chỉ sau tensorflow). Đây là package sử dụng các thư viện của CUDA và C/C++ hỗ trợ các tính toán trên GPU nhằm gia tăng tốc độ xử lý của mô hình. 2 mục tiêu chủ đạo của package này hướng tới là:

- Thay thế kiến trúc của numpy để tính toán được trên GPU.
- Deep learning platform cung cấp các xử lý tốc độ và linh hoạt.

Thư viện mã nguồn mở PaddleOCR: là một framework mã nguồn mở được phát triển bởi Baidu PaddlePaddle nhằm hỗ trợ việc nhận dạng và trích xuất thông tin từ hình ảnh. PaddleOCR ra đời hỗ trợ nhận dạng tiếng Anh, tiếng Trung, chữ số và hỗ trợ nhận dạng các văn bản dài. Hiện nay, PaddleOCR đã mở rộng thêm nhiều ngôn ngữ như Nhật, Hàn, Đức, … nhưng vẫn chưa có tiếng Việt. Văn bản tiếng Việt có dấu sẽ đọc ra không dấu, ví dụ như "Thị giác máy tính" sẽ nhận diện thành "Thi giac may tinh". Tuy nhiên, chúng ta vẫn có thể tự huấn luyện với bộ dữ liệu riêng để PaddleOCR có thể nhận dạng được tiếng Việt.

Thư viện mã nguồn mở VietOCR: Thư viện VietOCR được anh Phạm Bá Cường Quốc xây dựng với mục đích để giải quyết các bài toán liên quan đến OCR trong công nghiệp.

Các mô hình MobilenetV3 và VietOCR đã được huấn luyện trước.

❖ **Môi trường thực nghiệm:**

- Laptop: CPU: AMD Ryzen 5 4600H, Ram: 16.00 GB.
- Hệ điều hành Windows 11
- Công cụ lập trình: Python 3.8.19.

4.3. Tiền xử lý

- Đầu tiên, ta lọc những ảnh hóa đơn bị gán nhãn sai, hoặc thông tin bị nhầm lẫn, keywords_TOTAL_COST tương ứng với ['tổng tiền', 'cộng tiền hàng', 'tổng cộng', 'thanh toán', 'tại quầy'] và keywords_TIMESTAMP tương ứng với ['ngày', 'thời gian', 'giờ'], ta sẽ dùng 2 từ khóa này để áp dụng lọc hóa đơn.

Ví dụ như có ảnh có nhiều mốc thời gian thì ta sẽ lọc ảnh đó đi vì giả sử trường hợp ta cần ngày mua hàng mà trên ảnh hóa đơn lại gồm cả ngày nhập, ngày xuất thì sẽ dẫn đến sai sót trong quá trình huấn luyện.

4.3.1 Cài đặt các thư viện cần thiết

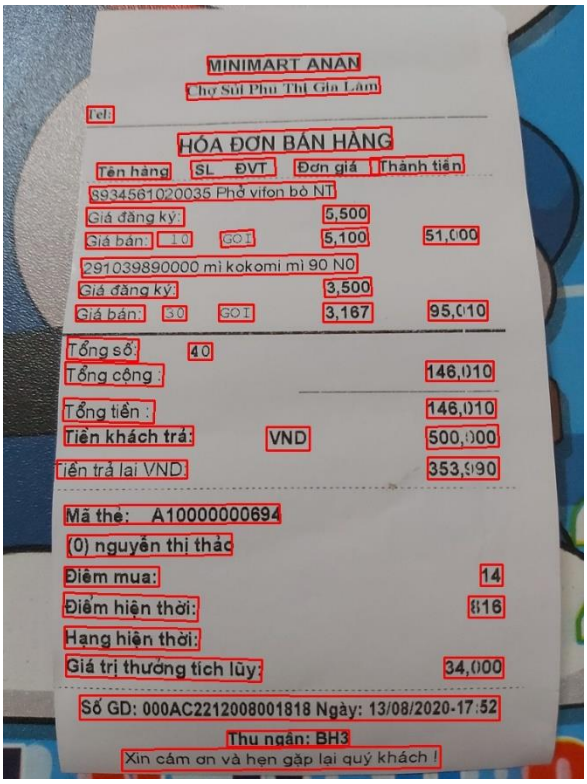
Các thư viện cần thiết: Python, PaddleOCR, Pytorch, CUDA, ….

```
# CUDA 11.8
conda install pytorch==2.2.2 torchvision==0.17.2 torchaudio==2.2.2 pytorch-cuda=11.8 -c pytorch -c nvidia
# cài paddleocr
!pip install paddleocr
# thư viện numpy
!pip install numpy
# thư viện matplotlib
!pip install matplotlib
...
```

Hình 4.2 Cài đặt thư viện

4.3.2 Text detector

- Bước này sẽ tìm vị trí của vùng chữ trên ảnh. Sử dụng pre-trained đã có sẵn từ PaddleOCR ta sẽ được thư mục gồm các tệp txt có thông tin các bounding box chứa text trên hóa đơn và ảnh trực quan sau khi xác định các boundingbox.

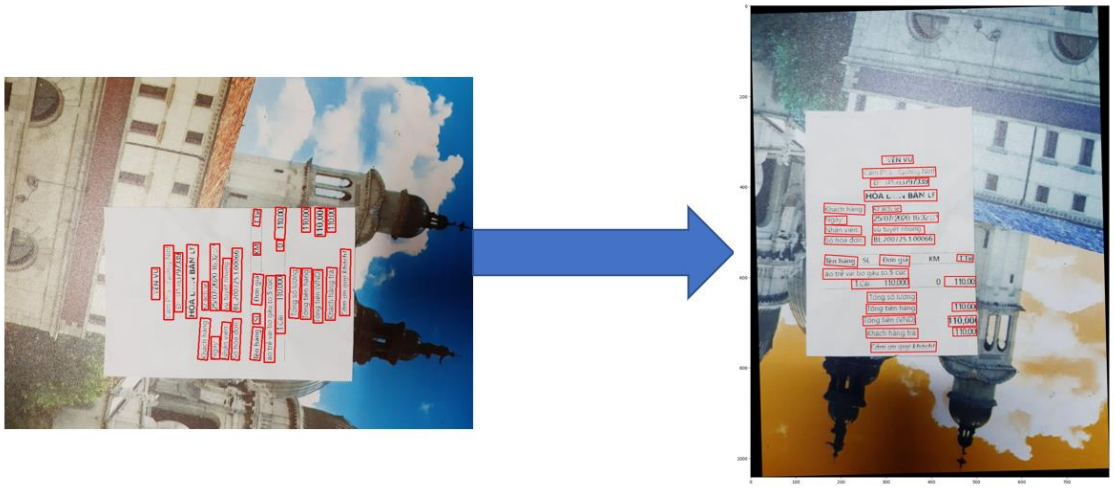


Hình 4.3 Ảnh trực quan sau quá trình tìm vị trí của vùng chữ

4.3.3 Rotation corrector và Text line rotation

- Bước này có nhiệm vụ xoay lại hóa đơn cho thẳng. Mô hình pretrained được sử dụng ở đây là Mobilenetv3, sau đó lọc các ảnh bị ngược hoặc xoay ngang trong tập train (sử dụng confidence của text classify để lọc, với threshold là 0.7). Cuối cùng khi xoay lại hóa đơn sẽ vẫn còn nhiều dòng chữ

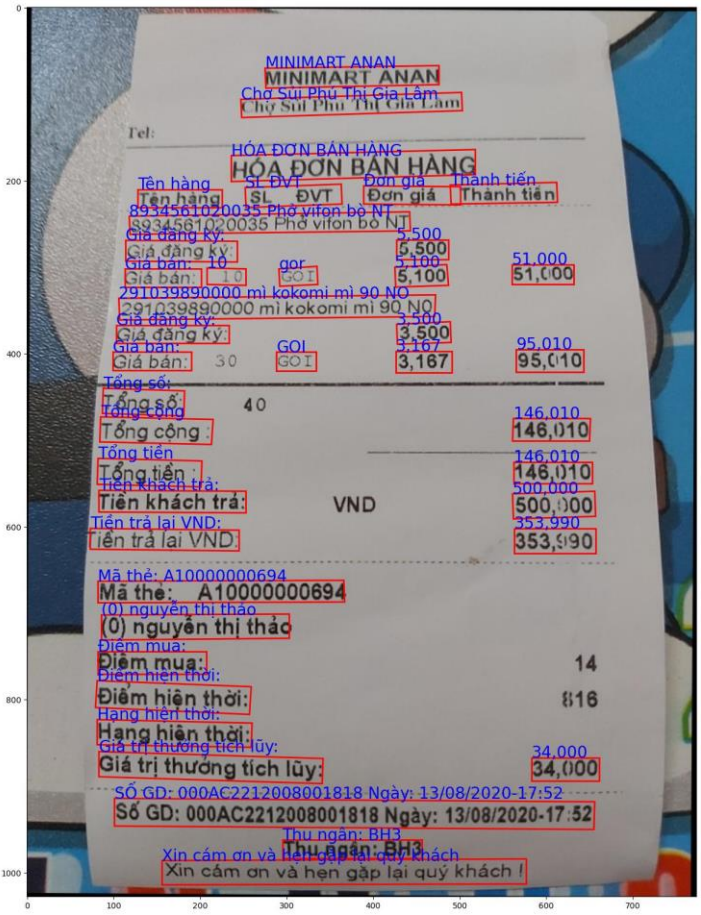
bị nghiêng, bước Text line rotation sẽ cắt vùng chữ đó và xoay lại cho thẳng để phần OCR được tốt hơn.



Hình 4.4 Ảnh sau khi xoay

4.3.4 Text classifier

- Đây chính là bước OCR đọc chữ từ vùng ảnh đã được xoay từ trên. Sử dụng pre-trained từ open soucre Vietocr ta được thư mục gồm các tệp txt chứa bounding box và text sau khi nhận diện.



Hình 4.5 Nhận dạng text với VietOCR

4.3.5 Sửa lại file csv để chuẩn bị huấn luyện mô hình

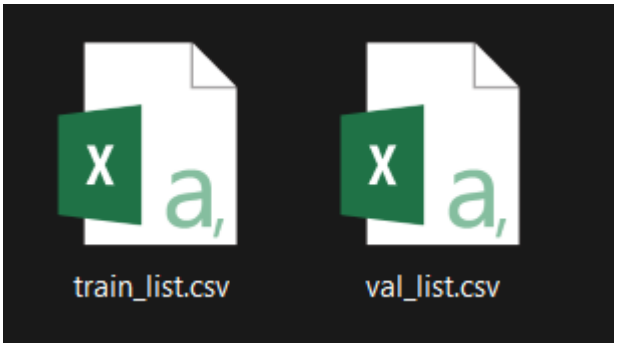
- File csv là file dữ liệu có sẵn đi cùng với ảnh hóa đơn được cung cấp. Dùng những tệp txt tại phần Rotation corrector để chỉnh lại thông số của bounding box sau khi đã xoay ảnh.

img_id	anno_poly	anno_text	anno_labe	anno_num	anno_image_quality
mcocr_pul	[{'categor	MINIMAR	SELLER|||	5	0.635309
mcocr_pul	[{'categor	VinComme	SELLER|||	7	0.774317
mcocr_pul	[{'categor	SIEU THI B	SELLER|||	5	0.664084
mcocr_pul	[{'categor	co.op mar	SELLER|||	8	0.715504
mcocr_pul	[{'categor	Tá»• 7, Kh	ADDRESS|	5	0.766884
mcocr_pul	[{'categor	MINIMAR	SELLER|||	5	0.829096
mcocr_pul	[{'categor	MINIMAR	SELLER|||	5	0.767563
mcocr_pul	[{'categor	Saigon Co.	SELLER|||	7	0.636108
mcocr_pul	[{'categor	VinComme	SELLER|||	4	0.701006
mcocr_pul	[{'categor	VinComme	SELLER|||	7	0.714937
mcocr_pul	[{'categor	PhÃºc Anh	SELLER|||	5	0.718378
mcocr_pul	[{'categor	TÃŠN Äá»	SELLER|||	8	0.45984
mcocr_pul	[{'categor	VinComme	SELLER|||	7	0.616947

Hình 4.6 File csv đã sửa

4.3.6 Chuẩn bị dữ liệu huấn luyện

Tệp dữ liệu được chia ra làm train_list và val_list với tỉ lệ 0.92 và 0.08



Hình 4.7 Dữ liệu huấn luyện

4.3.7 Huấn luyện mô hình

Model PICK lấy dữ liệu là ảnh hóa đơn và 2 file csv train_list và val_list để huấn luyện.

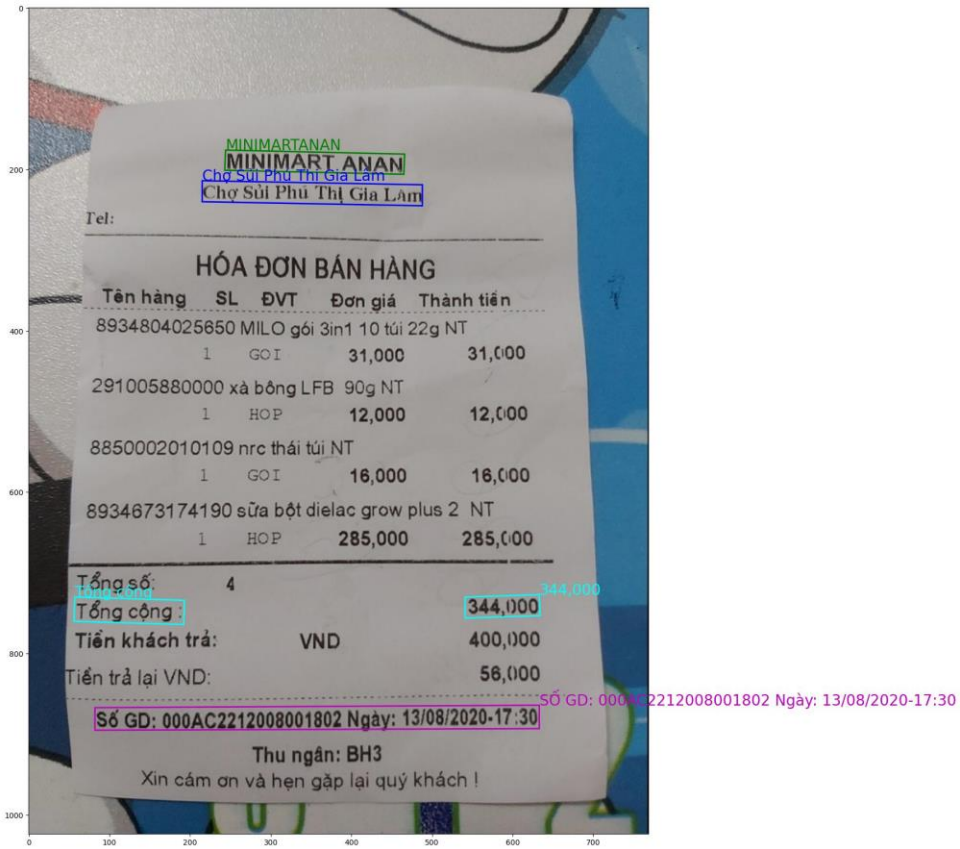
```
[ ] #!/bin/bash
    !python -m torch.distributed.launch --nnode=1 --node_rank=0 --nproc_per_node=1 \
       train.py -c config.json -d 0 --local_world_size 1
      # --resume /content/PICK-pytorch/saved/models/PICK_Default/test_0917_074722/model_best.pth ##uncomment for resume training
```

```
2020-10-03 05:37:10.929766: I tensorflow/stream_executor/platform/default/dso_loader.cc:48] Successfully opened dynamic library libcudart
[2020-10-03 05:37:13,574 - train - INFO] - Distributed GPU training model start...
[2020-10-03 05:37:13,574 - train - INFO] - [Process 372] Initializing process group with: {'MASTER_ADDR': '127.0.0.1', 'MASTER_PORT': '29
[2020-10-03 05:37:13,575 - train - INFO] - [Process 372] world_size = 1, rank = 0, backend=nccl
[2020-10-03 05:37:13,592 - train - INFO] - Dataloader instances created. Train datasets: 500 samples Validation datasets: 126 samples.
[2020-10-03 05:37:14,399 - train - INFO] - Model created, trainable parameters: 68567386.
[2020-10-03 05:37:14,400 - train - INFO] - Optimizer and lr_scheduler created.
[2020-10-03 05:37:14,400 - train - INFO] - Max_epochs: 100 Log_per_step: 10 Validation_per_step: 50.
[2020-10-03 05:37:14,400 - train - INFO] - Training start...
[2020-10-03 05:37:14,402 - trainer - INFO] - [Process 372] world_size = 1, rank = 0, n_gpu/process = 1, device_ids = [0]
/usr/local/lib/python3.6/dist-packages/torch/distributed/distributed_c10d.py:102: UserWarning: torch.distributed.reduce_op is deprecated,
  warnings.warn("torch.distributed.reduce_op is deprecated, please use "
[2020-10-03 05:37:42,210 - trainer - INFO] - Train Epoch:[1/100] Step:[10/250] Total Loss: 390.611877 GL_Loss: 0.370669 CRF_Loss: 390.241
[2020-10-03 05:38:01,990 - trainer - INFO] - Train Epoch:[1/100] Step:[20/250] Total Loss: 392.664154 GL_Loss: 0.252401 CRF_Loss: 392.411
[2020-10-03 05:38:20,633 - trainer - INFO] - Train Epoch:[1/100] Step:[30/250] Total Loss: 385.167877 GL_Loss: 0.112458 CRF_Loss: 385.055
[2020-10-03 05:38:39,237 - trainer - INFO] - Train Epoch:[1/100] Step:[40/250] Total Loss: 375.687164 GL_Loss: 0.147742 CRF_Loss: 375.539
[2020-10-03 05:38:59,310 - trainer - INFO] - Train Epoch:[1/100] Step:[50/250] Total Loss: 396.187317 GL_Loss: 0.213927 CRF_Loss: 395.973
[2020-10-03 05:39:34,332 - trainer - INFO] - [Step Validation] Epoch:[1/100] Step:[50/250]
+---------+-------+-------+-------+-------+
| name    |   mEP |   mER |   mEF |   mEA |
+=========+=======+=======+=======+=======+
| address |     0 |     0 |     0 |     0 |
+---------+-------+-------+-------+-------+
| company |     0 |     0 |     0 |     0 |
+---------+-------+-------+-------+-------+
| total   |     0 |     0 |     0 |     0 |
+---------+-------+-------+-------+-------+
| date    |     0 |     0 |     0 |     0 |
```

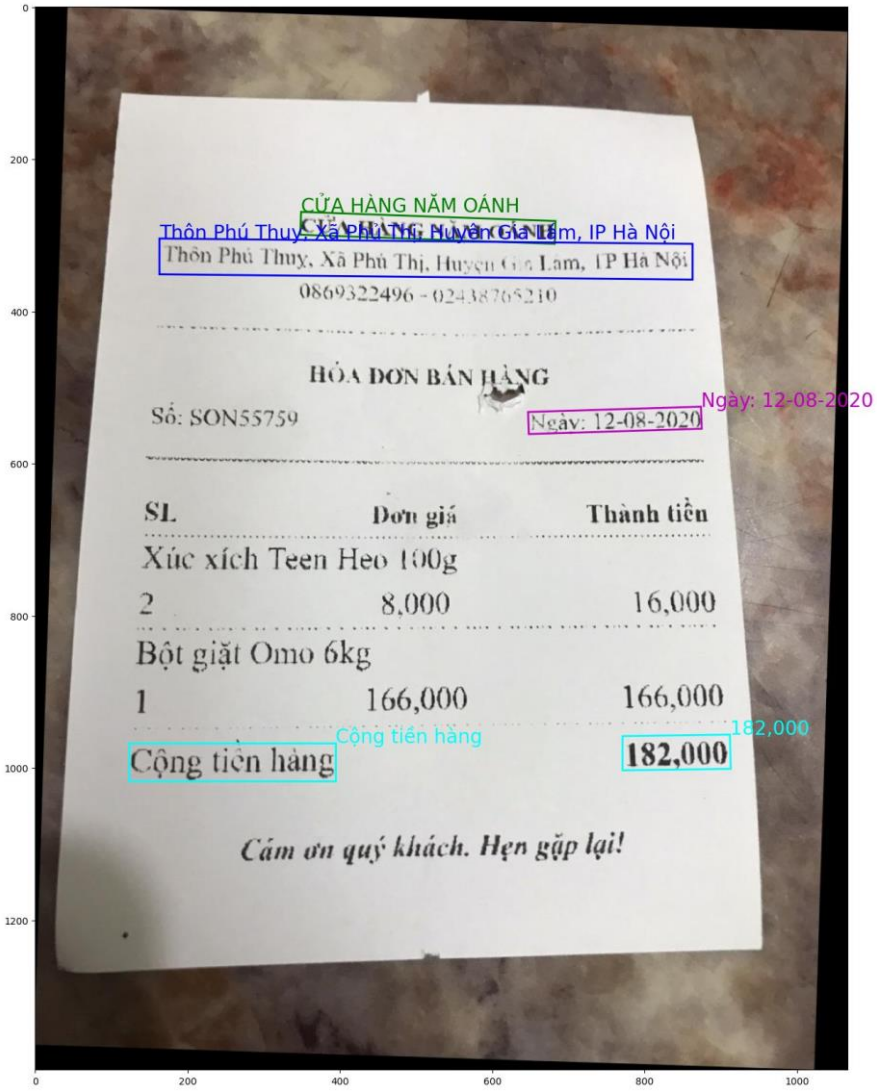
Hình 4.8 Train model PICK

4.3.8 Kết quả

- Mô hình đã nhận diện được thông tin cần trích xuất trên hóa đơn và xuất ra file excel với độ chính xác cao.



Hình 4.9 Ảnh kết quả (1)



Hình 4.10 Ảnh kết quả (2)

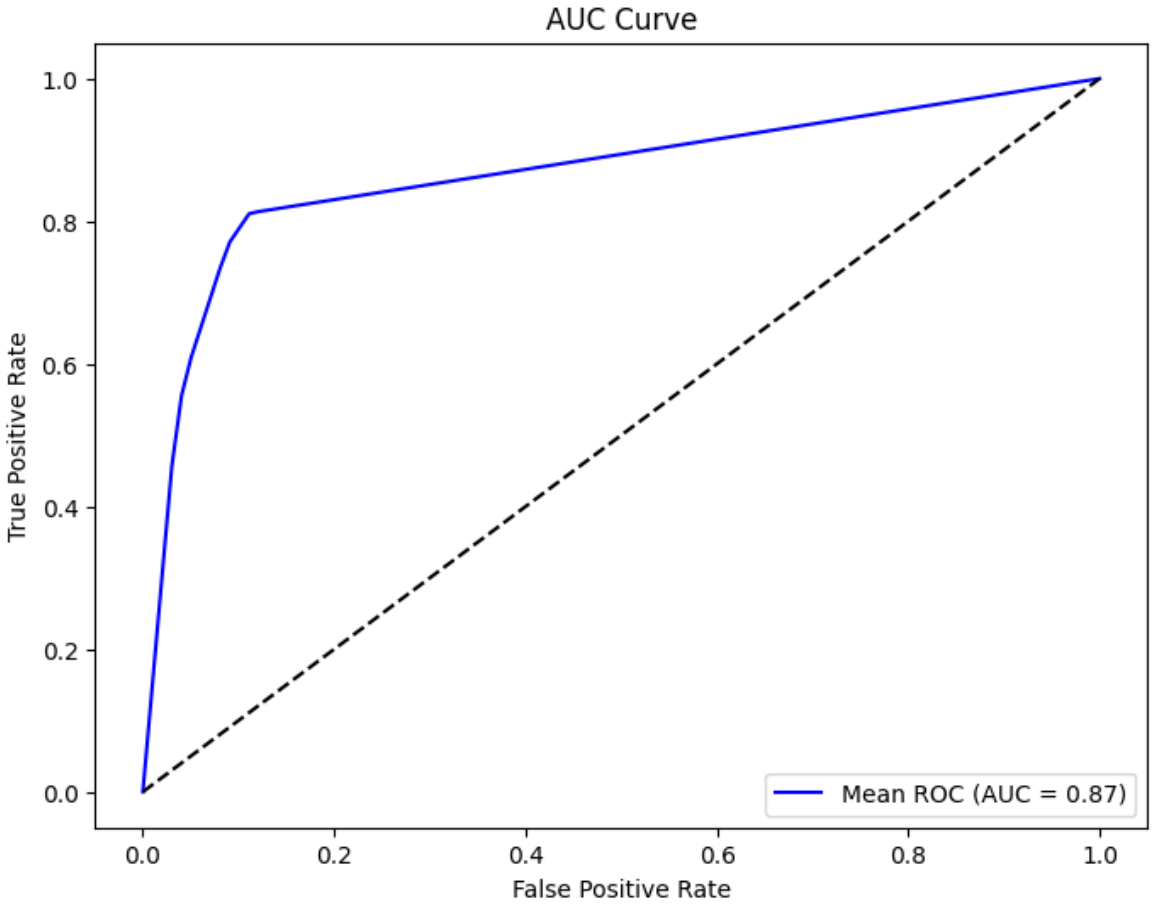
Ảnh	Nhà cung cấp	Địa chỉ	Thời gian	Tổng tiền
mcocr_private_145120hnosf	BỆNH VIỆN MẮT TW	PX191214000534 14/12/2019	Giờ in: 15:04:51	
mcocr_private_145120hnosf	LOOO	PX191214000534 14/12/2019	Giờ in: 15:04:51	
mcocr_private_145120hnuff	MINIMART ANAN	Chợ Sủi Phú Thị Gia Lâm		481000
mcocr_private_145120hsylc	UnCommerce	VIN4 ONH TỔ 8 Khu 38 Cầm Trung ; Tổ 8, Khu 38, P. Càm Trung ; TP. Cẩm Phả, T. Quảng Ninh	Ngày bán: 13/08/2020 11:28	113.400
mcocr_private_145120huazo	NHÀ SÁCH GD.TC CẨM PHẢ	Đo: 212 Đường Trần Phú-Cẩm Phả	Thời gian:10:43 17 15/08/2020	97.000
mcocr_private_145120ihcbk	Vincommerce	VIM4 QNH TỔ 7, KHU MINH TIẾN A ; Tổ 7, Khu Minh Tiến A ; P. Cẩm Bình, TP. Cẩm Phả, QNH	Thời gian:17:11 ; Ngày:14/08/2020	
mcocr_private_145120ilwww	VinCommerce	VIM4 ONH Dự án KDC lấn biển cọc s ; DA KHU DCLB CỌC 8, P. CẢM SƠN ; TP. Cẩm Phả, T. Quảng Ninh	Ngày bán: 12/08/2020 18:22	17.500
mcocr_private_145120imnbf	CỬA HÀNG NĂM OÁNH	Thôn Phú Thuy, Xã Phú Thị, Huyện Gia Lâm, TP Hà Nội	Ngày: 12-08-2020	72000

Hình 4.11 File excel sau khi trích xuất

4.3.9 Đánh giá mô hình

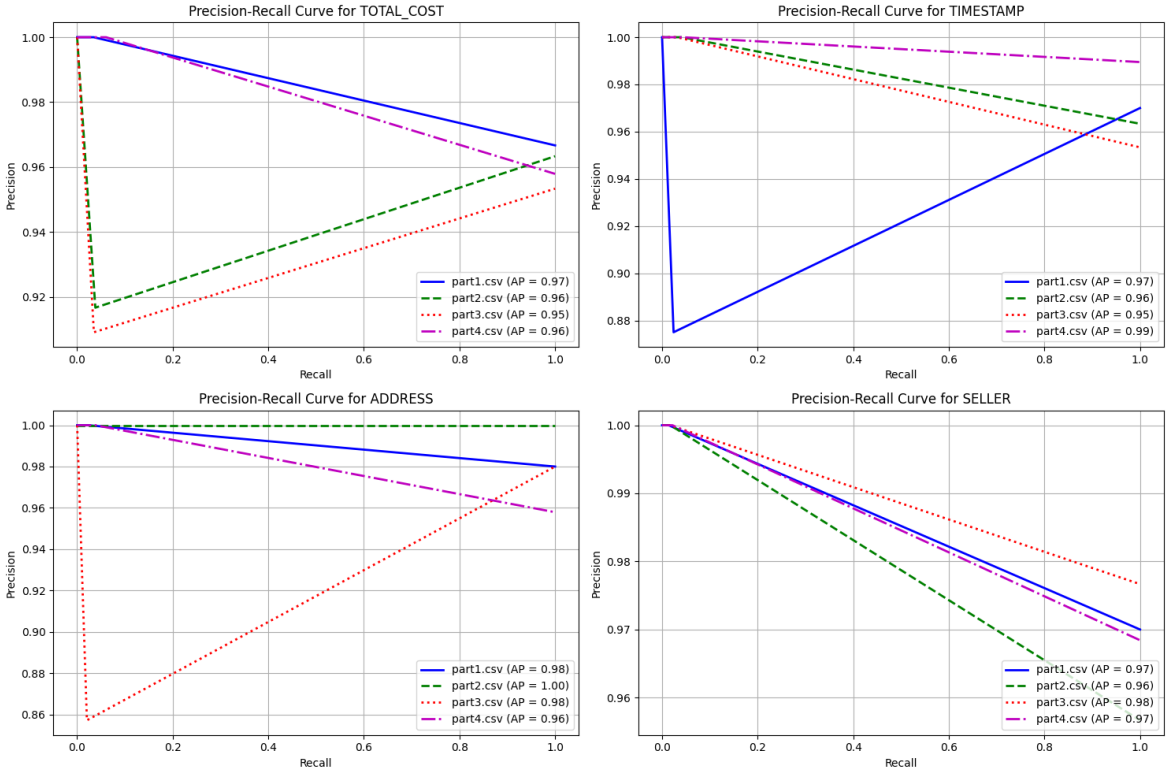
- Đánh giá trên tập huấn luyện và trên từng phần của bộ dữ liệu test:

Hình 4.12 Biểu đồ ROC và AUC



- Nhận xét:
  - Với AUC = 0.87, mô hình được đánh giá là khá tốt.
  - Mô hình có khả năng phân loại giữa các nhãn dương tính và âm tính một cách hiệu quả.

Hình 4.13 Biểu đồ AUPC



- Nhận xét:
  - Mô hình hoạt động rất hiệu quả với tất cả các nhãn (TOTAL_COST, TIMESTAMP, ADDRESS, SELLER). Các giá trị AP cao chỉ ra rằng mô hình có độ chính xác và độ nhạy cao trong việc nhận diện các nhãn này. Việc duy trì hiệu suất ổn định trên các phần dữ liệu khác nhau cũng cho thấy tính nhất quán và độ tin cậy của mô hình trong các tác vụ phân loại này.

Hình 4.14 F1 score trên tập dữ liệu validation

name	**mEP**	**mER**	**mEF**	**mEA**
exchange_rate_val	0	0	0	0
company_name_val	0.985507	0.957746	0.971429	0.957746
serial	1	1	1	1
website	0	0	0	0
amount_in_words	1	1	1	1
address_val	0.93	0.978947	0.953846	0.978947
buyer	0.916667	0.846154	0.88	0.846154
no_val	0.942857	1	0.970588	1
date	1	1	1	1
VAT_rate	0.964286	0.964286	0.964286	0.964286
tax_code_val	1	1	1	1
address	0.927536	0.969697	0.948148	0.969697
amount_in_words_val	0.933333	0.933333	0.933333	0.933333
no	0.972222	1	0.985915	1
company_name	0.944444	0.918919	0.931507	0.918919
seller	0.3	0.230769	0.26087	0.230769
tax_code	1	1	1	1
total_val	0.939394	0.96875	0.953846	0.96875
bank	0	0	0	0
VAT_rate_val	0.925926	0.961538	0.943396	0.961538
form_val	1	0.969697	0.984615	0.969697
account_no	0.25	0.153846	0.190476	0.153846
serial_val	0.970588	1	0.985075	1
grand_total_val	1	1	1	1
total	1	0.969697	0.984615	0.969697
exchange_rate	1	1	1	1
grand_total	1	1	1	1
VAT_amount_val	1	0.923077	0.96	0.923077
VAT_amount	0.962963	0.962963	0.962963	0.962963
form	1	1	1	1
Overall	0.951904	0.951904	0.951904	0.951904

*** Nhận xét chung:**

- Model PICK được huấn luyện đã đạt được mục tiêu đặt ra của bài toán là trích xuất được các trường địa chỉ, ngày bán, nhà cung cấp và tổng tiền.

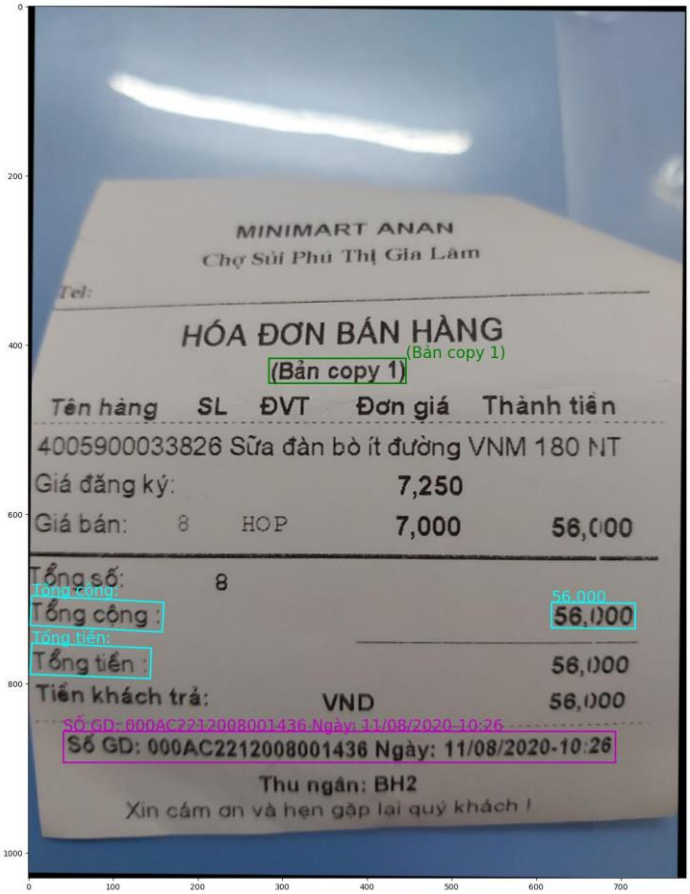
- Model đạt được độ chính xác khá cao, trên 95% meF (Hình 4.14).

- Vẫn còn nhiều ảnh hóa đơn chưa nhận diện chính xác được, các trường hợp như:

+ Ảnh nhỏ, mờ.

+ Có nhiều trường liên quan đến tổng tiền khiến cho mô hình khó xác định đâu là thông tin cần trích xuất.

+ Ngày bán bị cùng dòng với số giao dịch khiến cho mô hình nhận diện cả dòng là ngày bán.



Hình 4.15 Ảnh lỗi

4.4. Tổng kết chương 4

Trong chương này, em thực hiện huấn luyện model PICK để trích xuất thông tin hóa đơn. Em đã huấn luyện được mô hình trích xuất được các thông tin như địa chỉ, nhà cung cấp, ngày bán và tổng tiền. Mô hình sau khi huấn luyện đã đạt được độ chính xác rất tốt.

KẾT LUẬN

- **Kết quả đạt được**

Báo cáo đã trình bày được cơ bản nghiên cứu về mô hình PICK và ứng dụng mô hình được vào trích xuất thông tin trên hóa đơn; tìm hiểu, cài đặt thực nghiệm và đánh giá các mô hình và thư viện được sử dụng từ đó huấn luyện được mô hình giúp trích xuất thông tin hóa đơn ứng dụng mô hình PICK. Nội dung chi tiết các phần mà báo cáo đã thực hiện được:

  - Tìm hiểu về mô hình PICK và các khái niệm liên quan.
  - Ứng dụng mô hình PICK vào bài toán để trích xuất thông tin hóa đơn (các thông tin như địa chỉ, ngày bán, nhà cung cấp và tổng tiền).

- **Hướng nghiên cứu trong tương lai**
  - Cải thiện độ chính xác và tốc độ train model: Nghiên cứu để tăng tính nhất quán và đáng tin cậy trong việc trích xuất.
  - Train model chi tiết hơn: Nhiều trường được trích xuất hơn thay vì 4 trường địa chỉ, ngày bán, nhà cung cấp và tổng tiền như hiện tại.

TÀI LIỆU THAM KHẢO

[1] https://arxiv.org/abs/2004.07464

[2] https://github.com/wenwenyu/PICK-pytorch

[3] https://phamdinhkhanh.github.io/2019/08/10/PytorchTurtorial1.html

[4] https://phamdinhkhanh.github.io/2019/04/22/Ly_thuyet_ve_mang_LSTM.html

[5] https://phamdinhkhanh.github.io/2020/05/23/BERTModel.html

[6] https://viblo.asia/p/information-extraction-trong-ocr-la-gi-phuong-phap-nao-de-giai-quyet-bai-toan-yMnKMjzmZ7Pl

[7] https://github.com/PaddlePaddle/PaddleOCR

[8] https://github.com/pbcquoc/vietocr